



IEIT CNR Sezione di Bologna



Università
degli
Studi di
Bologna

Progetto finanziato dalla Regione Piemonte



TRIP

Transport Routing Information Platform



VOLUME I



Progetto TRIP



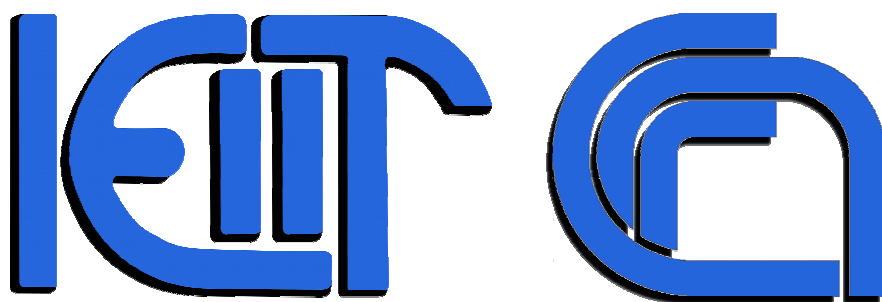
Transport Routing Information Platform

22 Ottobre 2007

31 Luglio 2009

Rapporto Scientifico

Progetto Finanziato dalla Regione Piemonte



Personale attivo nel progetto

Prof. Ing. Oreste Andrisano
Responsabile del Progetto

Prof. Ing. Davide Dardari
Dott. Lorenza Cavazza
Dott. Cristina De Castro
Dott. Alberto Carrara
Ing. Alberto Zanella
Ing. Gianni Pasolini
Ing. Andrea Conti
Ing. Rudi Paolo Paganelli
Ing. Matteo Mazzotti
Ing. Alessandro Bazzi
Ing. Barbara Masini
Ing. Paolo Toppan
Ing. Raffaele Soloperto
Ing. Thomas Pavani
Ing. Giacomo Leonardi
Ing. Angelo Dattilo
Ing. Laura Toni
Ing. Giovanni Chiurco
Ing. Andrea Toppan
Ing. Luca Piras



Personale attivo nel progetto

Ing. Sandro Cornetto

Ing. Niccolò Avico

Ing. Giuseppe Santo

Ing. Guido Occhetti

Ing. Fabio Perrone

Ing. Francesco Daprà

Ing. Roberto Donato

INDICE GENERALE

INDICE GENERALE	1
INDICE DELLE FIGURE	7
INDICE DELLE TABELLE	12
INTRODUZIONE.....	13
Generalità.....	13
Obiettivo Scientifico-Tecnico.....	16
Stato dell'Arte.....	16
Impatto Potenziale	17
Descrizione delle Risorse	18
Organizzazione	19
Disseminazione.....	19
Diffusione Dati	20
Questioni Etiche.....	20
PROGETTO TRIP: CONSIDERAZIONI GENERALI.....	21
Definizione del problema: obiettivi, contesto, requisiti.....	21
Scenario 0: accesso a internet con computer fisso.....	21
Scenario 1: terminale GSM (comunicazione via sms)	21
Scenario 2: terminale GSM/GPRS/UMTS/laptop con sistema Windows mobile o Symbian	22
Scenario 3: terminale con GPS	22
Confronto con altre piattaforme.....	22
Dai requisiti ai macroblocchi dell'architettura proposta	23
Flusso dati e dinamica del sistema.....	24
Architettura definitiva del sistema.....	25
Valutazione e scelta degli ambienti di sviluppo	27
Mappe e Sistema Informativo.....	27
Analisi particolareggiata dei moduli di sistema.....	30
Modulo Front-end.....	30
Interfaccia Utente-Sistema.....	31
Modulo GIS	31
Modulo Decisore del Percorso.....	33
Modulo Amministrativo	34
Modulo di Monitoraggio del Traffico.....	34
Modulo Real-Time Schedules	35
Modulo Real-Time Paths.....	36
VALUTAZIONE DEGLI SCENARI APPLICATIVI	38
Il settore degli ITS – Intelligent Transportation Systems.....	38
Sistemi di trasporto intelligente.	39
Il Quadro Nazionale.....	40
Il Quadro Internazionale.	40
Il Quadro Tecnologico.....	41
Sistemi ITS in Italia	42
Il mercato mondiale ITS	44
Offerta italiana ITS	45
Import-Export, un mercato ancora chiuso.	46
Gli investimenti in R&S.	48

La domanda italiana degli ITS	50
Le amministrazioni comunali.....	50
Gli ITS dal punto di vista delle aziende di trasporto pubblico.....	51
Conclusioni sulla situazione italiana.....	52
Il confronto con il mercato internazionale	53
Conclusioni sul mercato ITS.....	53
TECNOLOGIA A SUPPORTO DEL PROGETTO TRIP	55
Tecnologia wireless, un aiuto per incrementare i servizi ITS	55
Wi-Fi.....	56
Vantaggi e svantaggi del wi-fi	57
Wi-MAX	57
GSM.....	61
Servizi possibili con la rete GSM	62
Reti GSM	62
GPRS.....	63
Applicazioni del GPRS nel mondo reale	64
Connessione WAP e connessione web	65
DVB-H.....	65
UMTS.....	67
Caratteristiche tecniche	67
Terminali UMTS.....	68
Problemi e dibattiti aperti.....	68
Automatic Vehicle Location	69
BIBLIOGRAFIA	70
PIATTAFORMA DI COMUNICAZIONE MOBILE SHINE	71
Attività simulatori	71
STUDIO DELL'IMPATTO DEL SISTEMA SULL'INFRASTRUTTURA DI	
COMUNICAZIONE MOBILE	74
GSM.....	74
Breve introduzione al GSM	74
Canali logici nel GSM.....	76
Canali fisici nel GSM.....	76
L'accesso in uplink nel GSM.....	78
GPRS.....	81
Breve introduzione al GPRS	81
Canali logici nel GPRS	81
L'accesso in uplink nel GPRS	82
UMTS.....	84
Breve introduzione all'UMTS	84
I canali nell'UMTS	86
L'accesso in uplink nell'UMTS	87
Capacità di canale su rete di seconda e terza generazione.....	92
Bibliografia	94
MODULO GIS	95
Hardware.....	95
Architettura Software	95
GeoServer 1.7.5.....	97
OpenLayers	98
Postgresql	99

PostGIS	99
Shapefile	99
TeleAtlas Shapefile.....	100
INTEGRAZIONE SOFTWARE GIS.....	101
Configurazione Server	101
Importazione degli shapefiles	102
Sintassi	102
Opzioni	102
Configurazione GeoServer	104
Livello Punti di Interesse (PoI).....	111
Funzionalità GetFeatureInfo.....	113
Configurazione GeoWebCache	114
Database Geografico: PostgreSQL e PostGIS.....	118
Database Generico: MySQL.....	120
MODULO STIMA RITARDO	124
Algoritmo per la Stima del Ritardo	124
SCRIPTING PER AUTOMATIZZAZIONE SERVIZI.....	129
Linguaggio di Scripting	129
Perl	129
SCRIPT GESTIONE SOCKET	132
Tipologie di stringhe XML	132
Lettura Socket	132
Scrittura Socket.....	132
Monitoraggio Socket	133
Scenario tipico di utilizzo	133
Pagine di controllo dello script	133
SCRIPT GESTIONE SMS.....	135
Hardware:	135
Script.....	135
Fase di inizializzazione.....	136
Fase di lettura da SIM.....	136
Interpretazione SMS	136
Richiesta la portale	137
Creazione del messaggio	137
Fase di scrittura su SIM	137
Richiesta da SMS.....	138
Invio SMS dal portale.....	139
Pagina Demo SMS.....	140
SCRIPT REAL TIME	142
Primo Thread	143
Secondo Thread	144
Eventi significativi.....	147
Costruzione nuovo file richiesta	149
Pagine di gestione dello script	149
Scenario tipico	150
ALTRI SOTTOSISTEMI	152
Sottosistema calcolo percorsi	152
Sottosistema “simulazione real-time”.....	154
Sottosistema “amministrativo”	155

MANUALE DI UTILIZZO DEL PORTALE TRIP	156
Portale di Accesso	156
Portale di Accesso (versione Amministratori)	157
Portale di accesso (versione scalata)	158
Portale di Pianificazione Viaggio	159
Portale di Pianificazione Viaggio (versione scalata)	168
Calcolo del percorso	170
Portale di Registrazione Utente	174
Portale Area Utente	175
Portale di Informazione	178
Mappa Linee	180
Info Linee	181
Informazioni sull'orario effettivo di passaggio	182
PDF della Fermata	183
Portale Traffico	184
GPS tracking	185
Ritardo Mezzi	186
Ritardo Autovetture	187
Parcheggi	188
Cantieri	189
Webcam	190
Portale di Amministrazione	191
STUDI DI FATTIBILITA' PER GLI SCENARI DI POSSIBILE EVOLUZIONE ...	195
Informazioni via MMS	195
MMS	195
Integrazione nel sistema	196
Sviluppo Software	198
Terminali evoluti dotati di GPS	201
Sviluppo	201
J2ME	201
Librerie Abilitanti JAVA	204
Location e LocationProvider	206
ProximityListener	212
Possibile Scenario applicativo 1	214
Possibile scenario applicativo 2	215
Possibile scenario applicativo 3	216
Integrazione	216
APPENDICE	217
PORTALE DISEGNO PERCORSI	218
Descrizione e pianificazione del sistema	218
Requisiti e vincoli operativi	218
Pianificazione e scelte di progetto	219
Componenti architettureali	222
Progettazione del portale Linee Mezzi Pubblici	223
Model	223
Controller	224
View	225
Problematiche principali e relative soluzioni	228
Algoritmo di adattamento alle strade	228

Algoritmo di adattamento delle fermate	232
Gestione delle modifiche concorrenti	233
Principali scenari di utilizzo	235
Modifica dell'elenco degli operatori	235
Visualizzazione di una rotta.....	235
Inserimento di una nuova rotta	237
Modifica di una rotta preesistente.....	240
Ricerca di una rotta	241
MANUALE D'USO PORTALE DISEGNO PERCORSI.....	243
Descrizione e pianificazione del sistema	243
Requisiti e vincoli operativi.....	243
Componenti architetturali	244
Installazione e configurazione dei componenti	245
Apache Tomcat.....	245
Postgis.....	246
Caricamento Mappe TeleAtlas	247
Geoserver	249
GeoWebCache	256
Linee Mezzi Pubblici Web Application	257
Gestione della definizione di una rotta	257
Introduzione	257
Algoritmo di adattamento alle strade.....	257
Algoritmo di adattamento delle fermate	261
Gestione delle modifiche concorrenti	262
Principali scenari di utilizzo	264
Modifica dell'elenco degli operatori	264
Visualizzazione di una rotta.....	264
Inserimento di una nuova rotta	266
Modifica di una rotta preesistente.....	269
Ricerca di una rotta	270
CASO DI STUDIO PER LA LOCALIZZAZIONE REMOTA.....	272
Cenni sul Global Positioning System	272
Cenni storici.....	272
La costellazione	273
Funzionamento	273
Altri sistemi di posizionamento satellitare	276
I limiti del GPS	277
WGS84	277
NMEA 0183.....	278
Conversione di dati seriali per trasmissione su rete TCP/IP.....	281
HelloDevice LS100	281
Conversione RS232/Ethernet.....	289
Acquisizione dati da interfaccia GPS e conversione mediante LS100.....	290
Implementazione del software	291
Interfacciamento al database.....	297
Query sul database e mapping real-time.....	301
Lettura dati.....	301
Tracciamento delle coordinate.....	303
DG4ARCHEOS.....	304

Sintesi del progetto DG4ARCHEOS	304
Descrizione del Funzionamento di GPSRefiner	305
In breve	305
Costruttore.....	305
Metodo principale: refine	305
Creazione della griglia	305
Trasformazione delle distanze	306
Calcolo dei pesi.....	308
Libreria di gestione della comunicazione	312
Formato dei Messaggi	313
Connessione	313
Messaggio Ping	313
Messaggio NON-Ping	314
Messaggio da Server a Utente.....	315
Disconnessione.....	316
TESTBED	317
Architettura del test-bed.....	317
Prima fase.....	318
Portale Web.....	318
Portale Web Scalato	319
Accesso via SMS	319
Seconda fase.....	319
Registrazione di un percorso.....	319
Generazione evento ritardo	320
Ricezione SMS di avviso	320
Manuale del Simulatore WiMAX LLS per piattaforma SHINE.....	321
Introduzione	321
Assunzioni.....	321
Gestione dell'asse temporale	321
Prima di simulare	325
Relazioni con altri software	325
Lanciare il simulatore da riga di comando.....	325
Schema a blocchi del simulatore.....	326
Livello Fisico	330
Livello Fisico Trasmissivo.....	331
Modalità FUSC	334
Modalità PUSC	335
Modalità UL-PUSC.....	336
Apparati di Trsmissione Ricezione	337
Modello di Path-Loss e Shadowing	339
Gestione dell'Interferenza.....	342
Codifica di canale.....	344
Procedure di Aggancio.....	347
Simulazione con modello semi analitico	347
Livello Data Link	349
Radio Resource Management	354
File di Input e Configurazione	355
Esempi di simulazione	357
ALLEGATI.....	363

INDICE DELLE FIGURE

Fig. 1- Scenario di Rete	14
Fig. 2 - Architettura a macroblocchi del sistema TRIP	23
Fig. 3 - Flusso dati fra macroblocchi	25
Fig. 4 - Architettura del sistema TRIP suddivisa per blocchi di attività.....	26
Fig. 5 - Dati di TeleAtlas: strade, punti significativi, fiumi	28
Fig. 6 - Copertura delle mappe di TeleAtlas in Europa	28
Fig. 7 - Architettura completa del sistema TRIP	30
Fig. 8 - Architettura del modulo utente-mappe-utente	32
Fig. 9 - Gestione real-time dei ritardi	35
Fig. 10 - Servizio TRIP di aggiornamento dei percorsi su cellulari (MMS).	37
Fig. 11 - Il Mercato Mondiale dei sistemi ITS, valori attuali e previsioni future.....	44
Fig. 12 - Mercato ITS per modalità di trasporto	46
Fig. 13 - Paesi di destinazione delle esportazioni ITS.....	47
Fig. 14 - Paesi di provenienza delle importazioni ITS	48
Fig. 15 - Mercato italiano ITS: analisi attuale e previsioni future.....	49
Fig. 16 - Previsione di fatturato ITS per segmento di mercato.....	49
Fig. 17 - Investimenti delle Amministrazioni comunali per sistemi di gestione del traffico e della mobilità	50
Fig. 18 - Previsioni investimenti ITS per segmento di mercato	51
Fig. 19 - Investimenti ITS delle aziende di trasporto	52
Fig. 20 - Investimenti ITS delle aziende di trasporto divise per segmenti di mercato	52
Fig. 21 - Confronto sintetico delle principali tecnologie wireless.....	59
Fig. 22 - Elementi chiave della struttura di rete GSM.....	63
Fig. 23 - Demodulatore DVB-H	66
Fig. 24 - schema a blocchi di SHINE	72
Fig. 25 - risultati di simulazione	73
Fig. 26 - Architettura delle rete GSM.....	75
Fig. 27 - Tecnica FDD: trasmissioni simultanee in uplink e downlink in due bande diverse di frequenza	75
Fig. 28 - Canali logici nel GSM.....	76
Fig. 29 - Struttura accesso radio nel GSM.....	77
Fig. 30 - Multitrama associata ai canali TCH.....	77
Fig. 31 - Struttura dei burst radio nel GSM	78
Fig. 32 - Canali fisici nel GSM.....	78
Fig. 33 - Gli 8 bit dell'access burst.....	79
Fig. 34 - Multi-trama di segnalazione in uplink nel GSM.....	80
Fig. 35 - Canali logici nel GPRS che si aggiungono a quelli del GSM.....	82
Fig. 36 - Accesso in uplink sul RACH o PRACH nel GPRS	83
Fig. 37 - Schema di allocazione dei blocchi	84
Fig. 38 - Caratteristiche fondamentali del livello fisico per lo standard UMTS	85
Fig. 39 - I canali dell'UMTS	86
Fig. 40 - Mappatura dei canali nell'UMTS	86
Fig. 41 - Struttura del random access burst per l'accesso nell'UMTS	88
Fig. 42 - Ramping sul PRACH e trasmissione del messaggio	88
Fig. 43 - Procedura di accesso casuale in uplink	88
Fig. 44 - PRACH: formato dei dati nella parte di messaggio	89

Fig. 45 - Accesso casuale in uplink.....	90
Fig. 46- Struttura del PRACH.....	90
Fig. 47 - Periodico invio di informazioni dai veicoli alla rete cellulare	92
Fig. 48 - Numero di utenti per cella per singolo canale di segnalazione. Confronto tra rete GSM, GPRS ed UMTS	93
Fig. 49 - Numero di utenti in una città di 50 celle per singolo canale di segnalazione. Confronto tra rete GSM, GPRS ed UMTS	93
Fig. 50 - Sotto-sistema GIS.....	95
Fig. 51 - Esempio di mappa dinamica.....	97
Fig. 52 - Schema riassuntivo configurazione del Server.....	101
Fig. 53 - Da shapefile a formato sql.....	102
Fig. 54 - Esempio di tabelle contenute nel DB PostgreSQL.....	104
Fig. 55 - Richiesta di dati georeferenziati	105
Fig. 56 – Configurazione di GeoServer	106
Fig. 57 - Informazioni per la visualizzazione dei layer.....	107
Fig. 58 - GeoServer Style Editor.....	108
Fig. 59 - Gestione dei layers	110
Fig. 60 - Unione di diversi layer a formare una mappa completa.....	111
Fig. 61 - Esempio di mappa senza visualizzazione di Punti di Interesse.....	112
Fig. 62 - Esempio di mappa con visualizzazione di Punti di Interesse.....	113
Fig. 63 - Funzione GetFeatureInfo.....	113
Fig. 64 - Integrazione del sistema con GeoWebCache	114
Fig. 65 - Elenco di livelli all'interno del portale di amministrazione del tool GeoWebCache	115
Fig. 66 - Fase di caching	116
Fig. 67 - Tabelle nel DB PostgreSQL per la provincia di Alessandria.....	118
Fig. 68 - Tabelle relative a percorsi di mezzi e percorsi pedonali	119
Fig. 69 - Campi delle tabelle relative ai percorsi e linee.....	119
Fig. 70 - Tabelle del DB MySql.....	120
Fig. 71 - Tabella Corse.....	121
Fig. 72 - Tabella Evento Ritardo.....	121
Fig. 73 - Tabella Richieste	122
Fig. 74 - Tabella Session dashboard	123
Fig. 75 - Tabella Utente	123
Fig. 76 – Fermate indicate come percentuale di un percorso.....	125
Fig. 77 - Dato GPS che indica una posizione al 52,1% del percorso.....	126
Fig. 78 - Posizione GPS rispetto alle fermate di un percorso	126
Fig. 79 - Percentuale di percorso fatto all'interno della singola tratta	127
Fig. 80 - Screenshot della pagina del Portale TRIP per il controllo dello script di gestione della socket.....	134
Fig. 81 - Richiesta al sistema TRIP attraverso SMS.....	138
Fig. 82 - Risposta ad una richiesta via SMS	139
Fig. 83 - Invio SMS a seguito di operazioni sul Portale TRIP.....	140
Fig. 84 - Screenshot della pagina del Portale TRIP per uso demo dello script di gestione SMS.....	141
Fig. 85 - Schema di base del processo di notifica realtime.....	142
Fig. 86 - Campi che identificano l'oggetto Evento Ritardo	143
Fig. 87 - Schema a blocchi del primo thread dello script realtime	144

Fig. 88 - Attesa di conferma presa in carico dell'evento ritardo da parte del motore di calcolo	145
Fig. 89 - Analisi dei percorsi affetti dagli eventi ritardo già presi in carico.....	146
Fig. 90 - Ricalcolo del percorso e confronto con percorso suggerito in precedenza.....	147
Fig. 91 - Tabella evento significativo	148
Fig. 92 - Screenshot della pagina del Portale TRIP per il controllo dello script realtime ...	150
Fig. 93 - Scenario tipico per lo script realtime	151
Fig. 94 - Architettura generale della piattaforma TRIP	152
Fig. 95 - Homepage del portale TRIP.....	157
Fig. 96 - Homepage del portale TRIP dopo l'accesso come amministratore.....	157
Fig. 97 - Portale in versione scalata.....	158
Fig. 98 - Form di pianificazione del viaggio	159
Fig. 99 - Form di pianificazione del viaggio con preferenze avanzate.....	160
Fig. 100 - Scelta di provincia e comune	161
Fig. 101 - Introduzione di un campo di ricerca	162
Fig. 102 - Scelta di un indirizzo di partenza da una lista	163
Fig. 103 - Doppia possibilità per la scelta di una fermata di partenza.....	164
Fig. 104 - Scegliere una fermata di partenza: da mappa o da lista	165
Fig. 105 - Punto di interesse come punto di partenza.....	166
Fig. 106 - Partenza da punto GPS.....	167
Fig. 107 - Fermate in prossimità del punto di partenza	168
Fig. 108 - Pianificazione del viaggio da portale ridotto	169
Fig. 109 - Risultati di una ricerca	170
Fig. 110 - Visualizzazione grafica su mappa di un percorso.....	171
Fig. 111 - Prenotazione del servizio di avviso RealTime	172
Fig. 112 - Richiesta di informazione real-time accettata.....	173
Fig. 113 - Form di registrazione utente	174
Fig. 114 - Pagina di ingresso area utente	175
Fig. 115 - Storico utente	176
Fig. 116 - Pagina di ingresso al portale di informazione.....	178
Fig. 117 - Pagina di ingresso del portale di informazione su dispositivo portatile	179
Fig. 118 - Percorsi, fermate e orari dei mezzi pubblici	180
Fig. 119 - Schematiche dei percorsi dei mezzi pubblici con possibilità di visualizzare su mappa la posizione delle fermate	181
Fig. 120 - Esempio di risposta ad una richiesta di passaggi alla fermata	182
Fig. 121 - PDF con schematica percorso e orari di passaggio alla fermata scelta.....	183
Fig. 122 - Pagina iniziale del portale monitoraggio	184
Fig. 123 - Localizzazione dei mezzi in real-time	185
Fig. 124 - Visualizzazione su mappa della condizione dei mezzi pubblici.....	186
Fig. 125 - Visualizzazione su mappa della condizione del traffico stradale	187
Fig. 126 - Visualizzazione su mappa della posizione e condizione dei parcheggi.....	188
Fig. 127 - Visualizzazione su mappa della presenza di cantieri di lavoro stradale	189
Fig. 128 - Visualizzazione su mappa della presenza di webcam.....	190
Fig. 129 - Pagina iniziale del portale di amministrazione	191
Fig. 130 - Esempi di raccolta di dati statistici	192
Fig. 131 - Pagina del portale per la gestione della socket	193
Fig. 132 - Pagina del portale per la gestione dello script real-time	194
Fig. 133 - Schema di base della tecnologia MMS	195
Fig. 134 - Segnalazione realtime di un nuovo percorso attraverso MMS.....	196

Fig. 135 - Richiesta di informazioni attraverso SMS.....	197
Fig. 136 - Risposta ad una richiesta SMS via MMS.....	197
Fig. 137 - Evoluzione da SMS a MMS.....	198
Fig. 138 - Schema riassuntivo della piattaforma JAVA	202
Fig. 139 - Architettura J2ME	204
Fig. 140 - Sistemi di localizzazione radiomobile.....	207
Fig. 141 - Criteri di scelta del LocationProvider	208
Fig. 142 - Coordinate	209
Fig. 143 - Proximity listener	212
Fig. 144 - utilizzo del dato GPS per effettuare una richiesta	214
Fig. 145 - Avviso ricevuto in prossimità di una fermata di cambio mezzo	215
Fig. 146 - Visualizzazione della posizione dell'utente rispetto un punto di destinazione....	216
Fig. 147 - Componenti dell'architettura	222
Fig. 148 - Componenti del design patter MVC e relazioni.....	223
Fig. 149 - Package componenti il modello	224
Fig. 150 - Servlet dello strato controller, sincrone (giallo) e AJAX oriented (azzurro)	225
Fig. 151 - Modalità di navigazione per l'utente semplice	226
Fig. 152 - Modalità di navigazione per l'utente operatore	227
Fig. 153 - Strumenti di modifica e adattamento automatico.....	228
Fig. 154 - Percorso prima dell'adattamento	229
Fig. 155 - Percorso dopo l'adattamento.....	230
Fig. 156 - Algoritmo di adattamento con intervento manuale dell'operatore	231
Fig. 157 - Fermate non allineate	232
Fig. 158 - Fermate a seguito dell'allineamento	233
Fig. 159 - Avviso di modifica concorrente	234
Fig. 160 - Home Page del Portale	235
Fig. 161 - Elenco delle linee pubbliche presenti nel sistema	236
Fig. 162 - Visualizzazione di una rotta	237
Fig. 163 - Creazione di una nuova linea	238
Fig. 164 - Strumento di creazione linee e fermate	238
Fig. 165 - Elenco delle linee complessive presenti nel sistema	240
Fig. 166 - Modifica di una linea esistente	241
Fig. 167 - Maschera per la ricerca di linee.....	242
Fig. 168 - Risultato della ricerca	242
Fig. 169 - Componenti dell'architettura	244
Fig. 170 - Schermata iniziale dell'application server Tomcat.....	250
Fig. 171 - Tomcat Application Manager.....	251
Fig. 172 - Schermata iniziale del servizio Geoserver	252
Fig. 173 - Schermata iniziale delle configurazioni	252
Fig. 174 - Inserimento di un nuovo Feature Data Set	253
Fig. 175 - Aggiunta di un nuovo layer	254
Fig. 176 - Aggiunta di un layer WMS composito.....	255
Fig. 177 - Visualizzazione con OpenLayers	256
Fig. 178 - Strumenti di modifica e adattamento automatico.....	258
Fig. 179 - Percorso prima dell'adattamento	259
Fig. 180 - Percorso dopo l'adattamento.....	259
Fig. 181 - Algoritmo di adattamento con intervento manuale dell'operatore	260
Fig. 182 - Fermate non allineate	261
Fig. 183 - Fermate a seguito dell'allineamento	262

Fig. 184 - Avviso di modifica concorrente	263
Fig. 185 - Home Page del Portale	264
Fig. 186 - Elenco delle linee pubbliche presenti nel sistema.....	265
Fig. 187 - Visualizzazione di una rotta	266
Fig. 188 - Creazione di una nuova linea	267
Fig. 189 - Strumento di creazione linee e fermate.....	267
Fig. 190 - Elenco delle linee complessive presenti nel sistema.....	269
Fig. 191 - Modifica di una linea esistente.....	270
Fig. 192 - Maschera per la ricerca di linee	271
Fig. 193 - Risultato della ricerca.....	271
Fig. 194 - Costellazione di satelliti	272
Fig. 195 - Metodo della triangolazione.....	274
Fig. 196 - Convertitore LS100.....	282
Fig. 197 - Funzionalità LS100	282
Fig. 198 - Data/Console switch LS100.....	285
Fig. 199 - Parametri di accesso al convertitore LS100.....	285
Fig. 200 - Login LS100	286
Fig. 201 - Connessione della periferica seriale al convertitore LS100.....	286
Fig. 202 - Diagramma degli stati relativo alla modalità TCP client.....	288
Fig. 203 - get ip & get host.....	289
Fig. 204 - Connessione remota all'interfaccia GPS.....	289
Fig. 205 - Il server attende la chiamata del client.....	290
Fig. 206 - Ricezione dati GPS da client Telnet (HyperTerminal)	290
Fig. 207 - Banco di lavoro	291
Fig. 208 - Schema generale di una stream socket server-client.....	292
Fig. 209 - Schema di collegamento	294
Fig. 210 - Schema di connessione	294
Fig. 211 - Creazione database con phpMyAdmin	298
Fig. 212 - Screenshot database, tabella gps_data	299
Fig. 213 - GPS.php	303
Fig. 214 - Torino "Corso Regina Margherita"	303
Fig. 215 - Griglia con posizioni discrete	306
Fig. 216 - Architettura del testbed	318

INDICE DELLE TABELLE

Tab. 1 - Funzionalità a confronto.....	23
Tab. 2 - Ambienti software considerati.....	29
Tab. 3 - Investimenti ITS delle Aziende di trasporto pubblico.....	48
Tab. 4 - Mercato ITS mondiale per aree geografiche (Milioni di \$)	53
Tab. 5 - NMEA con prefisso “GPRMC”	279
Tab. 6 - NMEA con prefisso “GPGGA”	279
Tab. 7 - NMEA con prefisso “GPGSA”	280
Tab. 8 - NMEA con prefisso “GPGLL”	280
Tab. 9 - NMEA con prefisso “GPGSV”	281
Tab. 10 - Caratteristiche tecniche LS100.....	283
Tab. 11 - Gestione porte LS100	284

INTRODUZIONE

Generalità

Ogni giorno migliaia di veicoli attraversano le nostre città. I costi e l'impatto sulla salute pubblica del traffico pesante stanno diventando sempre più critici, così che l'inquinamento e la congestione dei trasporti diventano aspetti cruciali per il miglioramento della qualità della vita e dell'economia.

Lo sviluppo di un sistema di trasporto pubblico efficiente, che giochi il ruolo di un'alternativa reale all'uso dei mezzi privati, allevierebbe di molto l'impatto del traffico veicolare nella vita quotidiana. Si deve però osservare che l'efficienza percepita del trasporto pubblico non sta solo nella puntualità e nella frequenza, ma anche nella possibilità di trovare connessioni convenienti fra mezzi di trasporto eterogenei (treni, bus, ...), nell'avere accesso remoto e semplice ad informazioni di trasporto aggiornate (orari, ritardi, ...), nell'essere assistiti in remoto per la pianificazione del viaggio (trovare la soluzione migliore in termini di connessioni, prezzi, percorsi, ...) e così via.

Ne segue che il successo del trasporto pubblico verrebbe grandemente supportato dalla disponibilità di una varietà di servizi complementari, forniti dai mezzi di informazione e dalle tecnologie di comunicazione (ICT).

Questi servizi si intendono volti ad assistere l'utente finale durante tutte le fasi del viaggio, dalla pianificazione alla destinazione finale, ed a fornirgli ovunque ed in qualunque momento tutti i tipi di informazione aggiornata sul viaggio.

Allo stesso tempo, la disponibilità di informazioni aggiornate sullo stato effettivo della rete di trasporto pubblico, opportunamente convogliate verso un Centro di Controllo Centralizzato, sarebbe di grande aiuto agli Operatori del pubblico trasporto, sia nella fase di pianificazione della rete dei trasporti, sia nel monitoraggio del livello effettivo del servizio.

In questo contesto, l'adozione di tecnologie wireless è l'unica soluzione fattibile per fornire capacità di comunicazione sia agli utenti che ai mezzi di trasporto.

L'obiettivo di questo progetto è proprio lo sviluppo di una piattaforma multi-servizio e multi-tecnologia per il trasporto pubblico. Questa piattaforma, chiamata TRIP (Transport Routing Information Platform), vuole raccogliere, processare e distribuire tutte le informazioni relative al sistema di trasporto pubblico multimodale (bus, treni, ...) ed è quindi concepita come una rete di trasporto integrato per l'assistenza sia all'utente finale che agli Operatori dei trasporti di una data area.

Grazie alla piattaforma TRIP, l'utente finale è in grado di accedere, tramite una varietà di tecnologie fisse e wireless (GPRS, EDGE, UMTS, DVB-H, televisione digitale, Internet, ...) a tutta l'informazione sul sistema di trasporto multimodale (orari, ritardi effettivi, ...), così come a tool di pianificazione del viaggio mirati a fornire le migliori soluzioni in base alle sue preferenze (costi, percorsi, vincoli temporali, connessioni multimodali, ...).

Allo stesso tempo, la piattaforma TRIP permette il monitoraggio in tempo reale, basato su sistemi di comunicazione wireless, dello stato della rete integrata dei trasporti, fornendo così informazioni preziose agli utenti ed agli operatori.

Osserviamo infine che la piattaforma TRIP può anche fornire servizi di tipo context-aware, come ad esempio la trasmissione di informazione relativa a punti di interesse nelle vicinanze o mostre, che incrementano la conoscenza che gli utenti hanno del territorio e delle sue potenzialità.

Uno sviluppo futuro è lo studio della possibilità di estendere i benefici offerti da questa piattaforma ai non vedenti, attraverso tecniche di riconoscimento vocale.

In conclusione, la piattaforma TRIP può essere caratterizzata grazie a tre aggettivi:

- multi-modale, poiché considera l'intera rete dei trasporti (bus, treni, ...) come un sistema integrato;
- multi-servizio, poiché si rivolge sia agli utenti che agli operatori;
- multi-tecnologia, poiché supporta una varietà di sistemi di comunicazione per ampliare il più possibile il numero di clienti potenziali;

Nei servizi del tipo considerato è di fondamentale importanza l'infrastruttura di rete disponibile e la sua gestione. Oggigiorno nelle aree urbane e suburbane la presenza di una rete di telecomunicazione eterogenea (che integra cioè tecnologie di accesso diverse sia di tipo fisso che mobile) e pervasiva (cioè ben distribuita nel territorio) rende possibile lo scambio e la circolazione delle informazioni direttamente dai mezzi pubblici fino ad un centro di controllo preposto alla loro elaborazione, come illustrato in Fig. 1.

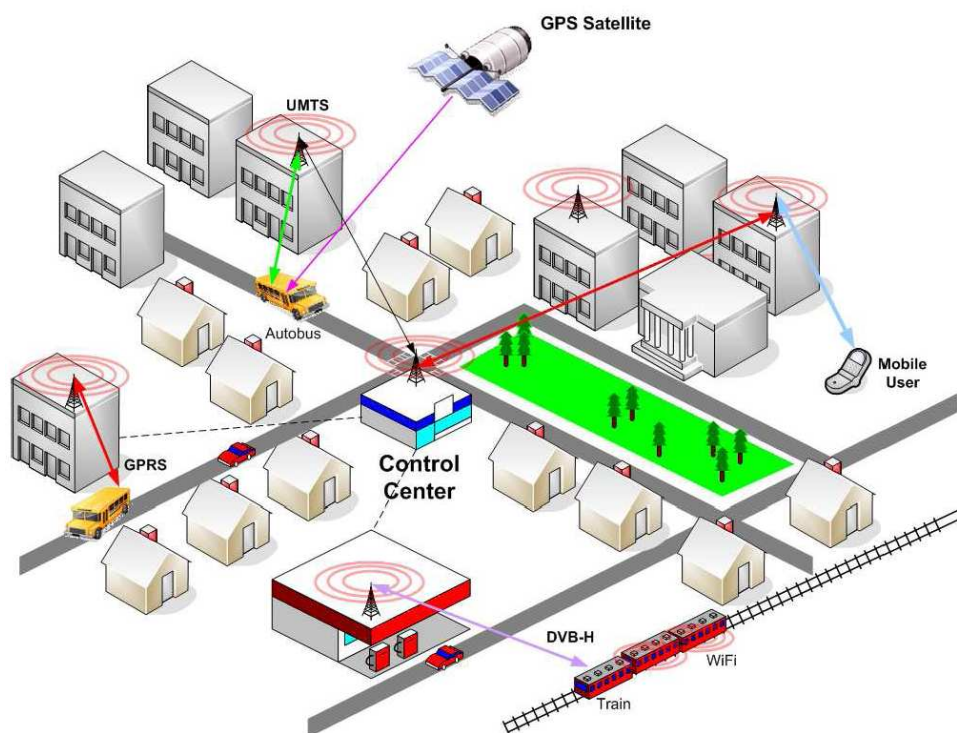


Fig. 1- Scenario di Rete

Mediante l'integrazione di diverse tecnologie (GPS, UMTS, WiFi, ..) è possibile sviluppare e implementare una serie di servizi innovativi sia per quel che riguarda la raccolta dei dati dai mezzi pubblici verso il centro di controllo, che nella direzione di collegamento opposta, cioè dal centro di controllo sia verso i mezzi stessi ma anche per esempio verso le fermate dotate di pannelli elettronici a messaggio variabile. I servizi possono andare dal tracciamento delle rotte dei mezzi pubblici in tempo reale, mappando cioè la posizione GPS di ogni mezzo su una mappa, alla trasmissione dei flussi video delle videocamere di sorveglianza a bordo dei mezzi, alla ricezione di messaggi video pubblicitari all'interno dei mezzi, all'invio alle fermate di avvisi di ritardo, ma anche connettività all'utente nelle fermate, nelle stazioni e anche a bordo dei mezzi stessi.

La possibilità di avere quindi delle vehicle-to-infrastructure communications technologies ci permette di creare innovazione sviluppando servizi che migliorano sia la circolazione dei

mezzi, il confort di viaggio dei passeggeri, la qualità del servizio pubblico offerto e la puntualità dei mezzi stessi. Ovviamente la rete di telecomunicazioni che rende possibile tutto questo deve essere una rete con una elevata copertura e un'ottima capacità: basti pensare alla complessità di fornire connettività ai convogli di una metropolitana sotterranea e alla mole di dati generato da un numero elevato di veicoli che circolano in un'area urbana geograficamente ristretta.

Inoltre un'altro aspetto importante da sottolineare è la necessità di provvedere ad una reale integrazione delle diverse reti di telecomunicazione che differiscono per esempio per gestore e tecnologia. Mentre per esempio esistono in commercio apparati AVM (Automatic vehicle monitoring) che già oggi integrano un ricevitore GPS per la localizzazione geografica e un modem UMTS per la trasmissione del dato, ben più complessi sono i sistemi di integrazione tra reti PMR (Professional Mobile Radio) o TETRA (TERrestrial Trunked RADIO), magari a disposizione delle regioni o delle aziende di trasporto stesse, con la rete IP di trasporto. E' necessaria dunque un'ottima competenza per progettare e gestire sistemi complessi, a volte anche molto diversi da loro, per poter abilitare i servizi innovativi. Altro aspetto importante da non trascurare è la necessità di configurare e ingegnerizzare accuratamente la rete e i sistemi di accesso alla rete stessa. Se non si effettua un dimensionamento opportuno della quantità dei dati da inviare, della modalità e soprattutto della frequenza di invio, si rischia di andare a penalizzare pesantemente l'efficienza della rete, offrendo un carico eccessivo rispetto a quello che potrebbe essere normalmente smaltito.

In questo ambito si sviluppano le ricerche svolte dal WiLab, per valutare quindi il carico della rete ed eventuali soluzioni per ottenere un compromesso di affidabilità e funzionalità.

L'infrastruttura di rete oltre ad essere utilizzata per trasferire i dati da e verso i mezzi pubblici, può venire efficacemente impiegata per fornire all'utente tutti i servizi di infomobilità che il sistema mette a disposizione. Infatti grazie ad una rete pervasiva di accesso wireless a larga banda eterogenea si può mettere a disposizione dell'utente un'elevata gamma di servizi dalla possibilità di richiedere informazioni sugli orari e sui passaggi alle fermate, a servizi evoluti che permettano all'utente di pianificare il proprio viaggio in real-time, di essere guidato alle fermate più vicine ed anche informato della presenza di traffico o ritardi sui mezzi per poter permettere un ricalcolo dinamico dei percorsi. Anche in questo caso le tipologie di terminali e la tecnologia a disposizione dell'utente sono molteplici da semplici terminali GSM, in grado solamente di inviare e ricevere SMS, a terminali evoluti con HSPA (High Speed Packet Access) e WiFi integrato, ma anche GPS in grado quindi di rilevare precisamente la posizione dell'utente.

In base quindi alla tecnologia a disposizione dell'utente è possibile pensare a servizi scalati/scalabili che prevedano un utilizzo sempre più massiccio della rete e delle informazioni da essa attingibili, via via che il terminale d'utente diventa evoluto.

E' possibile pensare per esempio, per terminali avanzati con connettività (UMTS, WiFi) e GPS un vero e proprio "accompagnamento mobile in scenario urbano" (keyword!!): l'utente viene letteralmente guidato verso la fermata più vicina, viene istruito sui mezzi pubblici da prendere e su tutte le eventuali variazioni che in real-time potrebbe essere necessario apportare al piano di viaggio. Quindi la possibilità di veicolare informazioni dai mezzi pubblici ai centri di elaborazione dati e viceversa, verso cioè gli utenti mobili, le fermate e i mezzi stessi, permette di portare veramente Internet sulla strada, un vero e proprio "Internet on the road", grazie ad una rete di telecomunicazione every-where, every-time, e all'integrazione di sistemi evoluti per l'erogazione dei servizi.

Obiettivo Scientifico-Tecnico

L'obiettivo tecnico del progetto è la realizzazione della piattaforma multi-servizio e multi-tecnologia TRIP, che ha un duplice intento:

- assistere l'utente del trasporto pubblico, fornendo informazioni sul sistema di trasporto multimodale e suggerendo la miglior soluzione di viaggio;
- raccogliere informazione sullo stato effettivo del sistema di trasporto multimodale, fornendo così agli operatori dei trasporti un tool prezioso per la pianificazione della rete dei trasporti ed il monitoraggio del livello dei servizi:

La realizzazione della piattaforma TRIP pone diverse sfide tecnico-scientifiche, che sono particolarmente correlate ai seguenti aspetti:

1. Integrazione su una stessa piattaforma di diverse tecnologie di comunicazione, sia fisse che wireless (GPRS, EDGE, UMTS, WiFi, DVB-H, televisione digitale, ...), per ampliare il più possibile il numero di utenti potenziali;
2. Fornitura di servizi scalabili; ciò richiede il monitoraggio delle proprietà delle connessioni ed il progetto di tecniche appropriate per modulare la quantità di dati da trasmettere in base ai dispositivi dell'utente. Bisogna considerare, per esempio, che la risoluzione di un display DVB-H è ben inferiore a quella di un televisore digitale domestico. Ne segue che per trasmettere, ad esempio, una mappa con il percorso migliore per una data destinazione, nel primo caso occorre inviare una minor quantità di dati. Ciò è anche in accordo con la minore banda disponibile. Nel caso di dispositivi GPRS, lo stesso contenuto informativo può essere fornito tramite servizi di messaggi (SMS o MMS);
3. La fornitura di servizi di tipo context-aware, che potrebbero essere correlati non solo al viaggio pianificato dall'utente, ma anche alla sua posizione corrente. Questo richiede l'adozione di appropriate strategie di localizzazione, che potrebbero essere basate, ad esempio, sull'infrastruttura di rete cellulare;
4. La possibile adozione di ambienti 3D per la visualizzazione dei percorsi;
5. La sicurezza delle comunicazioni. Ciò richiede l'adozione di adeguati livelli di protezione dell'informazione, qualunque sia la tecnologia di comunicazione adottata, per garantire la privacy dell'utente;
6. La realizzazione del Centro di Controllo Centralizzato che, per avere una visione aggiornata dello stato dell'intera rete dei trasporti, deve interfacciarsi con il Centro di Controllo di ogni singola rete di trasporti (rete dei bus, dei treni, ...)

Le competenze per affrontare queste sfide sono state dimostrate dal gruppo anche tramite la partecipazione ad importanti progetti nazionali come, ad esempio, il progetto "Virtual Immersive COMMunications - VICOM" ed il progetto "Multimedialità" (vedere www.wilab.org).

Stato dell'Arte

L'implementazione dell'ITS ("Intelligent Transportation Systems") in Europa ha raggiunto il punto critico di maturità.

Rispetto agli anni passati, lo stato dell'arte in Italia presenta alcuni miglioramenti, grazie a diversi progetti di ampio respiro, ma il gap rispetto ai paesi Europei più avanzati è ancora ampio. Ci sono diverse ragioni per questo ritardo, come:

- mancanza di un'industria del settore ed insufficienza di un framework legale ed organizzativo;
- mancanza di standard di codifica rilevanti;
- mancanza di dati primari (ad esempio dati geografici e mappe digitali);

A nostro avviso, questi problemi sono principalmente dovuti alla mancanza di metodi per la disseminazione dell'informazione ed di un opportuno schema organizzativo per l'ampliamento del mercato del trasporto pubblico e la sua evoluzione.

Il principale scopo del progetto è il miglioramento di questo stato dell'arte tramite la piattaforma TRIP, che, allo stesso tempo, utilizza le tecnologie delle comunicazioni a banda larga e le scienze dell'informazione per implementare applicazioni integrate per la gestione delle reti di trasporto, il supporto del viaggiatore e la fornitura di informazioni ai cittadini Piemontesi.

Uno degli obiettivi della piattaforma TRIP, che comprende l'uso di tecnologie avanzate quali, ad esempio, DVB-H, UMTS, televisione digitale, Web TV, è di incrementare lo stato dell'arte del pubblico trasporto sfruttando lo sviluppo tecnologico per il bene pubblico.

Impatto Potenziale

Il trasporto si pone come uno dei principali fattori per la determinazione della locazione dei siti ed è un'ancora di salvezza per la sopravvivenza economica in un mondo basato su dispositivi mobili.

Il progetto TRIP vuole, da una parte, rispondere alle esigenze dei cittadini e, dall'altra, alle richieste ed alle esigenze degli operatori del trasporto pubblico. Di conseguenza, la piattaforma TRIP agisce come:

- un collettore ed un diffusore di informazione: (i) raccoglie informazione sui servizi del trasporto pubblico offerti da più di un operatore sullo stesso territorio o aree adiacenti e (ii) rende disponibile l'informazione sui trasporti a chiunque, da dovunque ed in qualunque momento;
- una tecnologia abilitante per lo sviluppo di un sistema integrato di trasporti che utilizza l'approccio multi-modale per massimizzare il proprio potenziale in un'economia globale competitiva. Il trasporto inter-modale è definito come una sequenza di attività che coinvolgono trasporti, interconnessioni e terminali e richiedono l'intervento di una varietà di operatori parzialmente sovrapposti. Questo comporta un alto livello di competizione fra gli operatori, quindi il trasporto intermodale si dimostra consistentemente più economico che altre soluzioni, ed i suoi costi esterni significativamente più bassi, a conferma dell'alto potenziale dell'intermodalità per la sostenibilità del settore dei trasporti. Un efficiente sistema di trasporti è un fattore vitale per assicurare una buona qualità della vita e della salute e benessere socio-economico per la nostra società. Se il sistema di trasporto inter-modale è sicuro e garantisce lo spostamento efficiente di cittadini ed informazione, potrebbe in futuro essere applicato a beni e risorse e

sarà essenziale sia per il mantenimento che per la crescita della nostra economia. Infatti, un trasporto urbano efficiente può contribuire a:

- incrementare il numero di utenti del trasporto pubblico;
- migliorare la qualità, la competitività e l'efficienza del pubblico trasporto in Piemonte e, di conseguenza, in Italia ed in Europa;
- ridurre la congestione del traffico, particolarmente nelle aree urbane e metropolitane;
- controllare e ridurre il costo del così costituito trasporto pubblico intermodale;
- migliorare la qualità della vita dei cittadini, sia direttamente, attraverso il suo impatto sull'economia cittadina, sia indirettamente, attraverso il suo impatto sulle necessità quotidiane della gente;
- riduzione dei tempi morti grazie alla determinazione dei migliori percorsi possibili, ritmi e frequenze;
- rendere accessibile il pubblico trasporto a tutte le categorie di passeggeri (ad esempio informando quali bus hanno metodi di assistenza ai disabili);
- ridurre degli incidenti grazie alla riduzione del volume del traffico;
- gestire più efficientemente il traffico;
- facilitare il trasporto di merci;
- promuovere il progetto e la gestione di infrastrutture anche per ridurre i danni all'ambiente;
- portare alla collaborazione fra l'industria del trasporto e i provider di servizi per arrivare a soluzioni innovative;
- attrarre più investimenti in infrastrutture ed in tecnologia dell'informazione

Descrizione delle Risorse

In ottemperanza ai criteri di valutazione (art. 11 comma 4) a proposito delle capacità di gestione, organizzazione, marketing, tecniche e scientifiche dei tre soggetti che formano il team di ricerca (Proponente+Coproponente+Aggiuntivo), in qualità di PROPONENTE – CNR abbiamo definito il seguente team:

n. 1 Project Manager
n. 1 Assistente Project Manager
n. 7 Ricercatori strutturati
n. 6 Giovani ricercatori (meno di 32 anni)

COPROONENTE - TSF

n. 1 Senior representative
n. 3 Ricercatori strutturati
n. 2 Giovani ricercatori (meno di 32 anni)

AGGIUNTIVO - ALMAVIVA_CNR

n. 1 Senior representative – non finanziato

Organizzazione

Per la gestione e lo sviluppo delle attività di ricerca, abbiamo definito due uffici operazionali:

- i laboratori di ricerca dell'Istituto IEIIT del CNR (proponente) e
- il gruppo di sviluppo software di TSF (coproponente).

TSF, come da comunicazione inviata alla regione Piemonte in data 22.10.2007, è uscita dal progetto, sostituita da ALMAVIVA.

La società consortile ALMAVIVA CNR, che nella fase della redazione di proposta progettuale aveva il ruolo di "soggetto aggiuntivo" (art. 3 comma 4 punto b), è uscita dal progetto in seguito a scioglimento da parte della controllante ALMAVIVA.

Il progetto ha avuto durata 24 mesi con uno staff (art. 8 comma 5 punto a) che ha coperto più o meno il 50% delle spese generali (€ 2.000.000,00). Il 15% è stato di spese per HW e SW (art. 8 comma 5 punto b); il 15% per consulenze e prodotti semifiniti (art. 8 comma 5 punto c); non più di € 20.000,00 per oggetti di consumo (art. 8 comma 5 punto d); un massimo del 13% per trasferte temporanee (art. 8 comma 5 punto e) e il 9% per spese generali (art. 8 comma 5 punto f).

Disseminazione

Il risultato del progetto è stato lo sviluppo di una piattaforma che raccoglie, condivide e diffonde informazione agli utenti della rete di trasporto ed agli operatori e concorrenti tecnologici.

In particolare, la disseminazione dei risultati avviene tramite:

- un sito web;
- l'organizzazione di workshop indirizzati agli operatori del trasporto pubblico ed agli amministratori locali e regionali;
- documentazione del progetto;
- partecipazione a workshop industriali e conferenze scientifiche;
- contributo a database online;

È stata anche pianificata una valutazione socio-economica dell'impatto dei sistemi di trasporto integrati ed i risultati di tale attività verranno diffusi. Questo darà informazione su:

criteri di pianificazione per i sistemi di trasporto pubblico
architetture di sistema per i trasporti inter-modalità
indicatori accurati di traffico ed inquinamento

Diffusione Dati

Seguendo le indicazioni dell'Unione Europea, il CNR ha ricevuto dal partner "coproponente" una royalty equivalente al costo di mercato dei diritti intellettuali del progetto TRIP; il progetto è proprietà congiunta dei due partner.

I risultati che non sono compresi nei diritti intellettuali potranno essere distribuiti ad altri interessati. L'accordo legale con i partner prevede che tutti i diritti intellettuali derivanti da attività di ricerca e sviluppo ed il diritto di accesso ad essi siano assegnati ai partner in modo proporzionale in base ai mutui interessi, agli sforzi ed all'esposizione economica.

Questioni Etiche

Missione dell'attività è stata la ricerca industriale e scientifica e lo sviluppo di competenze tecniche nei campi dell'Infomobilità, dei sistemi di Trasporto Intelligenti e della Logistica.

Questo progetto deve essere visto come ricerca pianificata e sviluppo di rilevamenti critici mirati ad acquisire nuova conoscenza utile a creare nuovi prodotti, processi e servizi o a migliorare soluzioni esistenti.

PROGETTO TRIP: CONSIDERAZIONI GENERALI

Definizione del problema: obiettivi, contesto, requisiti

Come già discusso nell'introduzione, il progetto TRIP si colloca nell'ambito dei servizi di informazione di supporto all'utente mobile. Più in dettaglio, l'obiettivo del progetto è fornire all'utente che ne faccia richiesta attraverso un sistema di telecomunicazione, le informazioni necessarie per il raggiungimento di una destinazione nel minor tempo possibile, utilizzando i mezzi di trasporto pubblici (treni, autobus, ...).

Le principali modalità di funzionamento e caratteristiche del sistema sviluppato sono le seguenti:

l'utente notifica il punto di partenza, la destinazione ed eventuali vincoli sugli orari di partenza e di arrivo e riceve dal sistema indicazione del tragitto ottimale, ovvero della migliore combinazione treno/autobus/metropolitana;

l'informazione può essere inviata e ricevuta tramite dispositivi di comunicazione multi-tecnologia (telefono cellulare, palmare, laptop, connessione internet fissa);

i mezzi di trasporto previsti sono autobus, treni e metropolitana. Possono essere suggeriti tratti a piedi;

a fronte di particolari condizioni (es.: congestioni, ritardi dei mezzi pubblici, coincidenze perdute, ecc.), l'utente può chiedere di ricevere indicazione di un percorso alternativo, inteso come percorso migliore in base alla nuova situazione;

lo scenario più generale prevede la localizzazione dell'utente tramite GPS, ma – per ora – si suppone che l'utente invii inizialmente i propri dati e segua il percorso che gli è stato indicato.

si vuole gestire l'infrastruttura in modo mirato al miglioramento del servizio; in particolare, il sistema deve essere in grado di fornire indicazioni al gestore per funzionalità di tipo decisionale, anche non immediatamente applicabili, quali linee molto usate, situazioni di traffico che si ripetono, ecc.;

In sintesi, l'obiettivo è lo sviluppo di un sistema che, in tempo reale, possa monitorare particolari condizioni che influenzano il servizio richiesto, modificare di conseguenza le informazioni via via suggerite e notificarle all'utente. Gli scenari applicativi ed i servizi considerati o da tenere presente per eventuali estensioni sono i seguenti:

Scenario 0: accesso a internet con computer fisso

Attraverso un apposito portale web l'utente comunica il punto di partenza, quello di arrivo ed i vincoli temporali del viaggio; può richiedere e ricevere:

- informazioni aggiornate sul percorso
- indicazioni su possibili alternative
- comunicazioni sulle fermate previste
- informazioni sui servizi nelle vicinanze

Scenario 1: terminale GSM (comunicazione via sms)

L'utente comunica via **sms** il punto di partenza, quello di arrivo ed i vincoli temporali del viaggio; può richiedere e ricevere sms con:

- informazioni aggiornate sul percorso
- indicazioni su possibili alternative
- comunicazioni sulle fermate previste
- informazioni sui servizi nelle vicinanze

Scenario 2: terminale GSM/GPRS/UMTS/laptop con sistema Windows mobile o Symbian

In questo caso, c'è localizzazione parziale su base cella; L'utente può inviare e ricevere tutte le informazioni sopra elencate, **tramite sms o pagina web semplificata**.

Scenario 3: terminale con GPS

In questo caso c'è localizzazione precisa; l'utente comunica via sms i dati sul viaggio e può scegliere se ricevere informazioni aggiornate sul percorso tramite **servizi sms/web**.

Confronto con altre piattaforme

Il sistema TRIP si può raffrontare ad altri già disponibili in base a diversi parametri di fruibilità; in particolare, si prendono in considerazione la piattaforma 5T (www.5t.torino.it/5t/) del comune di Torino, la piattaforma GITA sviluppata da TSF e Google Transit (www.google.it/transit), che offre copertura anche sul comune di Torino.

I parametri di confronto sono:

- **tipo di ingresso** (punto di partenza e destinazione) che può fornire l'utente (solo stazioni o fermate oppure un generico indirizzo);
- possibilità di contemplare **tratte pedonali** nel percorso suggerito;
- **aggiornamento dinamico del percorso** migliore in base a condizioni di traffico variabili nel tempo. Alcuni sistemi mettono a disposizione dati sul monitoraggio del traffico, ma non li utilizzano per adattare dinamicamente la scelta dei percorsi;
- tipo di **tecnologie di accesso** a disposizione dell'utente;
- **flessibilità di sviluppo**, per esempio la possibilità di rappresentare altri mezzi di trasporto non ancora previsti;
- **multimodalità**, intesa come possibilità di includere nel percorso più mezzi di trasporto;
- **modalità di visualizzazione** del percorso

In Tab. 1 sono indicati i risultati del confronto e, come si può notare, la piattaforma TRIP presenta ottime caratteristiche di funzionalità, fruibilità e flessibilità.

	5T	GITA	Google Transit	TRIP
tipo input	qualunque punto (via e numero civico)	stazioni/fermate	qualunque punto (via e numero civico)	qualunque punto (via e numero civico)
tratte pedonali	si	no	no	si
aggiornamento dinamico percorsi	solo monitoraggio statico	no	no	si

tecnologie d'accesso	web, cellulare	web	web	web, cellulare
flessibilità di sviluppo	—	—	no	si
multimodalità (treni,...)	treni, bus	treni, bus	piedi, bus, auto	piedi, bus, treni, auto
visualizzazione	intero percorso	no (solo messaggi testuali)	intero percorso	intero percorso

Tab. 1 - Funzionalità a confronto

Dai requisiti ai macroblocchi dell'architettura proposta

Le specifiche hanno portato alla definizione dell'architettura di massima riportata in Fig. 2, costituita da sette macroblocchi:

1. modulo utente, con tutti i dispositivi previsti (**Front-end Module**);
2. moduli di comunicazione fra l'utente ed il sistema, adattabili e scalabili in base ai diversi dispositivi, per la raccolta della meta e di eventuali ulteriori dati (eventi) inviati dall'utente e per la ricezione dei percorsi (**User-System Interface**);
3. un modulo per la raccolta dati dai pubblici servizi (**Traffic Monitoring Module**);
4. un modulo di analisi della significatività dei ritardi e la ridefinizione degli orari (**Real-Time schedules Module**);
5. un modulo **Real-Time Paths** che raccolga gli eventi significativi, guidi la decisione del percorso e comunichi i percorsi aggiornati;
6. un modulo di **Path Decision** che suggerisca il primo percorso migliore e, in caso di intasamenti significativi o ritardi, ricalcoli dinamicamente il nuovo percorso migliore;
7. un **database** di supporto contenente tutti i dati necessari su mappe stradali, bus e treni, traffico, registrazione dati inviati dall'utente, percorsi suggeriti e loro modifica, ecc.;

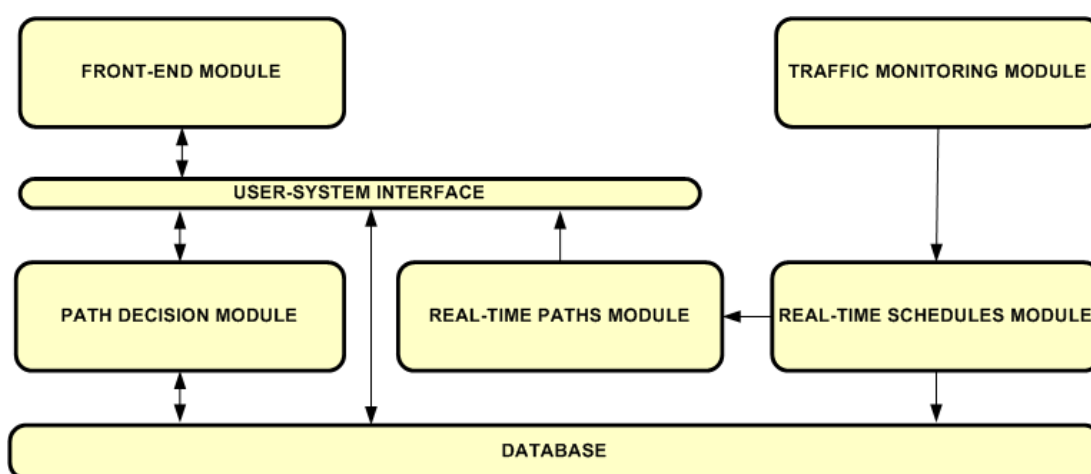


Fig. 2 - Architettura a macroblocchi del sistema TRIP

Quella di

Fig. 2 è solo una rappresentazione parziale dell'architettura finale adottata, che verrà presentata e discussa più avanti.

Flusso dati e dinamica del sistema

Per arrivare a definire la dinamica del sistema è necessario dettagliare le fonti informative ed il loro flusso.

In riferimento alla

Fig. 2, si può osservare che i principali dati di interesse appartengono principalmente a quattro classi: (i) informazioni di scambio con l'utente e percorsi suggeriti; (ii) mappe/dati georeferenziati; (iii) orari di mezzi pubblici; (iv) dati sulla viabilità.

Le prime vengono inviate al sistema informativo e rese ad esso comprensibili tramite il modulo di interfacciamento utente-sistema. Questo modulo ha la funzione di operare le necessarie conversioni di dati in formato compatibile con il sistema (in particolare il sistema informativo) e con i diversi dispositivi. Tali informazioni, infatti, non è detto siano immediatamente fruibili. Si realizza ciò tramite conversioni in formato XML.

Le informazioni sui dati geografici sono tratte da fonti preesistenti, come mappe stradali urbane, fino a mappe più ricche quali quelle utilizzate dai principali sistemi di navigazione.

Per quanto riguarda i mezzi pubblici, le fonti sono eterogenee, ad esempio orari comunali di autobus e regionali di treni. In entrambi i casi, non è detto che il formato d'origine sia immediatamente compatibile con quello del sistema informativo. In molti casi, però, i dati sono disponibili o possono essere convertiti in XML e, da qui, trasferiti al database utilizzato. Questo è stato uno dei criteri principali che hanno guidato la scelta del software per il database.

Le informazioni sulla viabilità si intendono riferite alle mappe, con tutti i dati di interesse per la dinamica del traffico urbano, ad esempio eventuali blocchi per lavori in corso, velocità effettive rilevate, ecc.

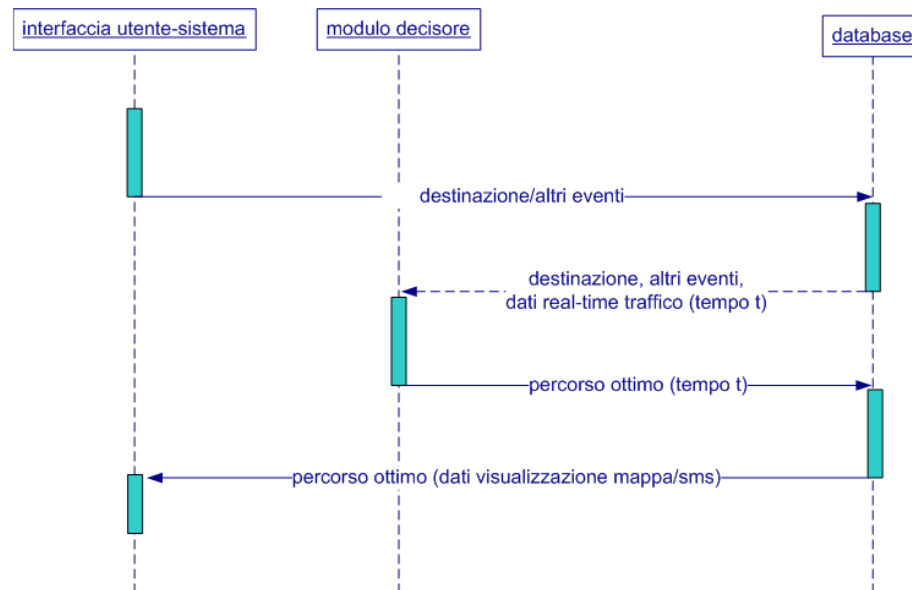
Il diagramma di sequenza UML di Fig. 3 rappresenta il flusso di dati, di richieste e di risposte fra i moduli del sistema. Da questa rappresentazione è escluso il blocco di fonti dati esterne, che interagisce solo con il database all'atto del caricamento dati.

L'utente, tramite il modulo di interfacciamento utente-sistema, invia il punto di partenza, la destinazione ed eventuali vincoli temporali sul viaggio, che vengono inoltrati al database.

I diversi moduli agiscono in parallelo. Dal database vengono reperiti dati sulle condizioni del traffico, inoltrati poi al modulo decisore del percorso. In base alla meta ed a questi dati, il modulo decisore calcola il percorso ottimo all'istante considerato e lo inoltra al database.

L'utilizzo congiunto di tutte queste informazioni permette di calcolare il percorso non solo in base ai tempi previsti per lunghezza delle strade ed orari attesi dei mezzi pubblici, ma anche alla effettiva viabilità rilevata.

In caso di variazioni significative, come ritardi che rendono impossibile arrivare in tempo per prendere una data coincidenza bus-treno, il blocco di instradamento notifica al modulo utente-sistema il percorso aggiornato. Una volta ricevuto il percorso ottimo, il sistema provvede a reperire dal database le informazioni per notificarlo all'utente nel formato opportuno, visuale o sms.

**Fig. 3 - Flusso dati fra macroblocchi**

Architettura definitiva del sistema

In Fig. 4 è riportata l'architettura completa e definitiva del sistema. I blocchi componenti sono:

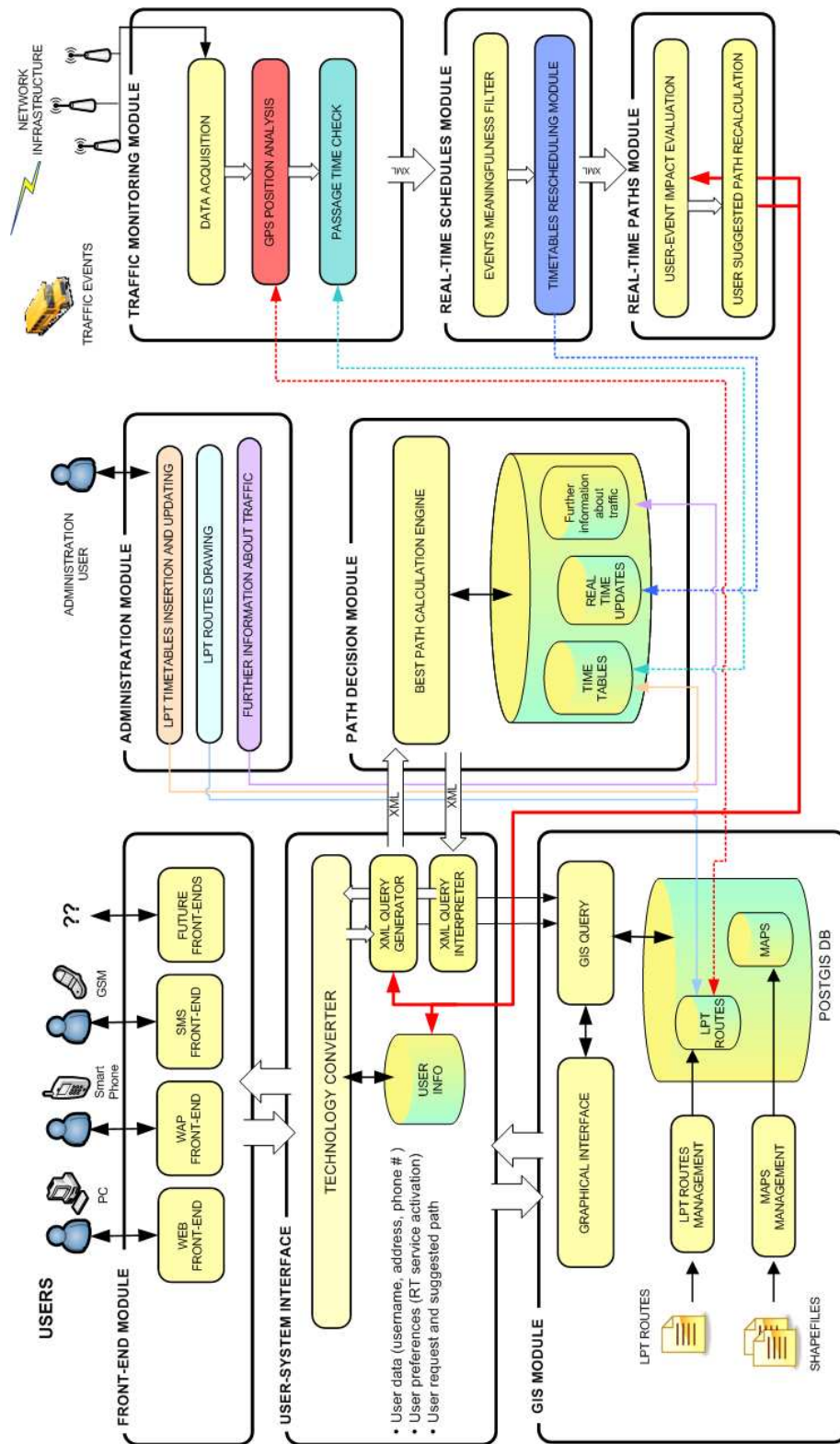


Fig. 4 - Architettura del sistema TRIP suddivisa per blocchi di attività

1. modulo front-end;
2. modulo user-system interface;
3. modulo GIS;

4. modulo di amministrazione;
5. modulo di path decision;
6. modulo di monitoraggio traffico;
7. modulo real-time gestione orari;
8. modulo real-time gestione percorsi;

Il sottosistema front-end si occupa dell'interfacciamento con l'utente in base alle diverse tecnologie, della raccolta e dell'invio di informazioni in formato comprensibile all'utente sul viaggio, sul traffico in tempo reale, anagrafica, ecc. Oltre ai diversi front-end (web, sms, wap), contiene un convertitore di tecnologia, un database per le informazioni utente e moduli per la raccolta dati.

Il sottosistema GIS riveste particolare importanza per questo tipo di applicazioni, poiché contiene database su mappe e percorsi e moduli di interfacciamento grafico che rendono questi tipi di dati fruibili all'utente in modo visuale ed immediato. Il sottosistema per il calcolo delle rotte contiene database con gli orari dei mezzi, informazioni aggiuntive sul traffico e orari aggiornati in tempo reale. Date queste informazioni e le specifiche dell'utente, restituisce il percorso ottimo in termini di tempo.

Il sottosistema amministrativo permette di registrare e gestire in modo semplice nuove informazioni su mezzi, orari e situazioni particolari sul traffico.

Il sottosistema di monitoraggio del traffico riceve degli eventi sulla viabilità e li invia a moduli

opportuni che ne filtrano la significatività e, di conseguenza, procedono o meno ad un nuovo scheduling degli orari.

Tutte le informazioni sul traffico in tempo reale e sull'aggiornamento dei percorsi utente sono gestiti e memorizzati nei due sottosistemi real-time.

Data l'eterogeneità delle fonti informative, i diversi sottosistemi colloquiano tramite XML.

In linea di massima, la dinamica del sistema è la seguente: l'utente invia le proprie richieste al sistema tramite il sottosistema front-end. Per la raccolta dei dati della query, l'interfaccia utente si appoggia al sottosistema GIS tramite il modulo di interfacciamento grafico. Questi dati vengono forniti al sottosistema calcolo delle rotte, che restituisce il percorso ottimo. Tale percorso viene memorizzato nel database percorsi utente del sottosistema real-time e visualizzato tramite il modulo di interfacciamento grafico delle rotte.

Il calcolo del percorso migliore è eseguito in base ad informazioni statiche e dinamiche sugli orari dei mezzi e sugli aggiornamenti real-time a fronte di ritardi, ecc. Queste ultime informazioni vengono rilevate dal sottosistema di monitoraggio del traffico, filtrate per significatività, registrate su database ed inoltrate al modulo di rescheduling degli orari, poi essere inoltrate al sottosistema di calcolo delle rotte.

Valutazione e scelta degli ambienti di sviluppo

Come discusso nel capitolo precedente, la Fig. 3 mostra l'architettura TRIP suddivisa per blocchi di attività da svolgere. In particolare, si considerino il sistema informativo e la comunicazione multimodale utente-sistema. Per entrambe, prima di procedere, sono state scelte piattaforme di sviluppo fra alcune possibili.

Mappe e Sistema Informativo

In un sistema che coinvolge informazioni georeferenziate, bisogna distinguere fra i dati ed il software che li gestisce. Il livello dati è costituito dalle cosiddette mappe geografiche,

rappresentate in modo più o meno ricco. Le mappe, però, non sono utilizzabili da sole: occorre un sistema di gestione di database, o di fruizione di dati di questo tipo, in grado di leggerle e metterle a disposizione.

Per quanto riguarda le mappe, la scelta è stata dall'inizio quasi univocamente orientata a quelle di

TeleAtlas, senz'altro le più avanzate per ricchezza rappresentativa, copertura (Figg. 5, 6, da <http://navigation/teleatlas.com>, ultimi dati disponibili al Maggio 2009) e diffusione, usate fra gli altri da TomTom, Nokia e Google.

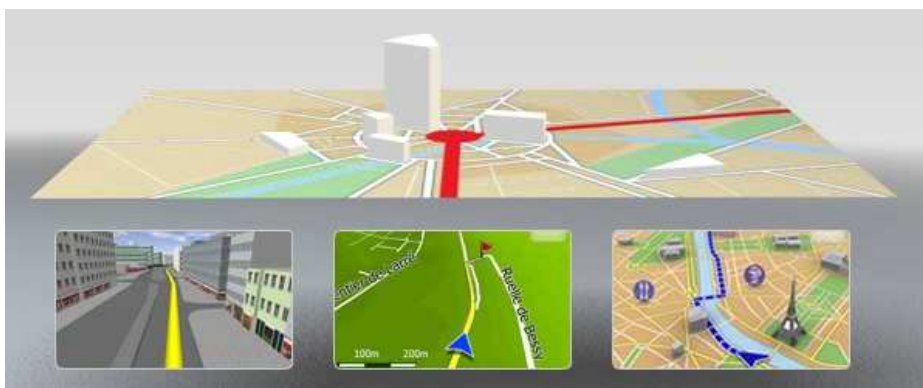


Fig. 5 - Dati di TeleAtlas: strade, punti significativi, fiumi

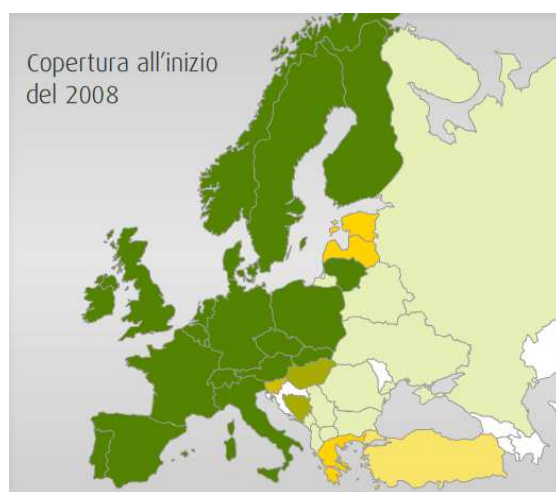


Fig. 6 - Copertura delle mappe di TeleAtlas in Europa

Oltre a detti parametri, però, e considerato l'elevato costo di questi prodotti (comprensibile, vista la difficoltà e l'enorme lavoro richiesti dalla raccolta di questo tipo di informazioni), si è pensato alla

possibilità di poter fruire delle mappe in modo efficiente, realizzando costi contenuti almeno sul fronte del software del sistema informativo.

Per quanto riguarda il DBMS (sistema di gestione di basi di dati), bisogna tenere conto di questi criteri:

- le informazioni del database TRIP si possono classificare non solo in base alla semantica, ma anche alla frequenza di aggiornamento: ci sono dati (a) statici, ossia non soggetti a variazioni frequenti, come le mappe e gli orari dei mezzi pubblici; ci sono poi (b) dati dinamici, ossia modificati di frequente, come l'andamento del

traffico, le richieste dei diversi utenti ed i percorsi suggeriti. Il DBMS deve quindi poter garantire l'utilizzo efficiente di dati di entrambi i tipi. parte dei dati derivano dalle mappe, preesistenti, quindi il sistema deve essere in grado di gestire dati georeferenziati;

Un DBMS relazionale può garantire efficienza per il primo requisito, ma non è così immediato che vada bene per la gestione delle mappe. Per questo motivo, come accennato sopra, si è deciso di analizzare più a fondo la fruibilità delle mappe e la possibilità di utilizzare con un buon sistema open source.

A priori questa soluzione poteva non sembrare così semplice: prodotti principe per l'utilizzo di dati georeferenziati sono infatti GIS (Sistemi Informativi Geografici), o - per meglio soddisfare entrambi i requisiti del database TRIP - sistemi object-relational come Oracle (proprietario e partner di TeleAtlas). Nella versione Enterprise 11g Spatial Locator, Oracle permette infatti la gestione integrata di dati spaziali, oltre ad essere – in base agli ultimi benchmark ufficiali disponibili – il DBMS più veloce al pari dell'open source object-relational Postgres con estensione PostGIS per il supporto di dati spaziali.

Un altro elemento da tenere in considerazione è che i partner di Torino hanno sviluppato il loro prodotto su piattaforma Microsoft SQL Server, proprietario, ed occorre dunque trovare il modo di allinearsi in modo efficiente.

Tutti questi fattori hanno più che mai focalizzato l'attenzione sulle mappe TeleAtlas, poiché i dati forniti sono accessibili in scrittura sia in schema che in estensione, e sono disponibili in diversi formati: shapefile, leggibile ai sistemi relazionali, formato Oracle, ecc.

Questo ha permesso di prendere in considerazione tutte le possibilità sopra citate per il software di gestione: Oracle, Postgres/PostGIS e Microsoft SQL Server, con i risultati illustrati in Tab. 2.

Si premette che, di entrambi i prodotti proprietari, sono indicati i prezzi education o per centri di ricerca e sviluppo. Di entrambi, inoltre, sono disponibili versioni gratuite: Oracle Light, ed altre per progetti di ricerca ma non di sviluppo, e MS Sql Server Express. Nessuno di questi prodotti, però, può supportare il carico di lavoro previsto o essere un giorno utilizzata a scopo commerciale.

	Oracle	Postgres/PostGIS	MS SQL Server
costo (1 licenza CPU)	circa 30.000 euro ¹	open source	circa 30.000 euro ²
robustezza/ affidabilità/velocità	ottimo	ottimo	buono
tool di sviluppo/altri fattori	Internet Application Server ³	ampia disponibilità di librerie on line	utilizzato dai partner di Torino

Tab. 2 - Ambienti software considerati

A fronte di queste informazioni, si è deciso di sviluppare il sistema informativo con Postgres/PostGIS, pearaltro efficacissimo per la gestione di dati geografici/georeferenziati di interesse per il progetto. Si utilizza anche un server MySql, anch'esso open source.

1 La versione light, gratuita, e quelle messe a disposizione per i progetti di ricerca, non sono sufficienti

2 La versione express, gratuita, è solo locale

3 A pagamento, a parte

Per quanto riguarda invece l'allineamento con i partner di Torino, un modulo di comunicazione di invio richieste e ricezione risposte realizza trasformazioni di formato grazie a XML e rende possibile l'utilizzo integrato dei due sistemi di Bologna e di Torino senza che questi debbano essere modificati per interoperare.

Analisi particolareggiata dei moduli di sistema

Riprendiamo ora (Fig. 7) l'architettura completa del sistema già illustrata in Fig. 4 e discutiamone i moduli componenti.

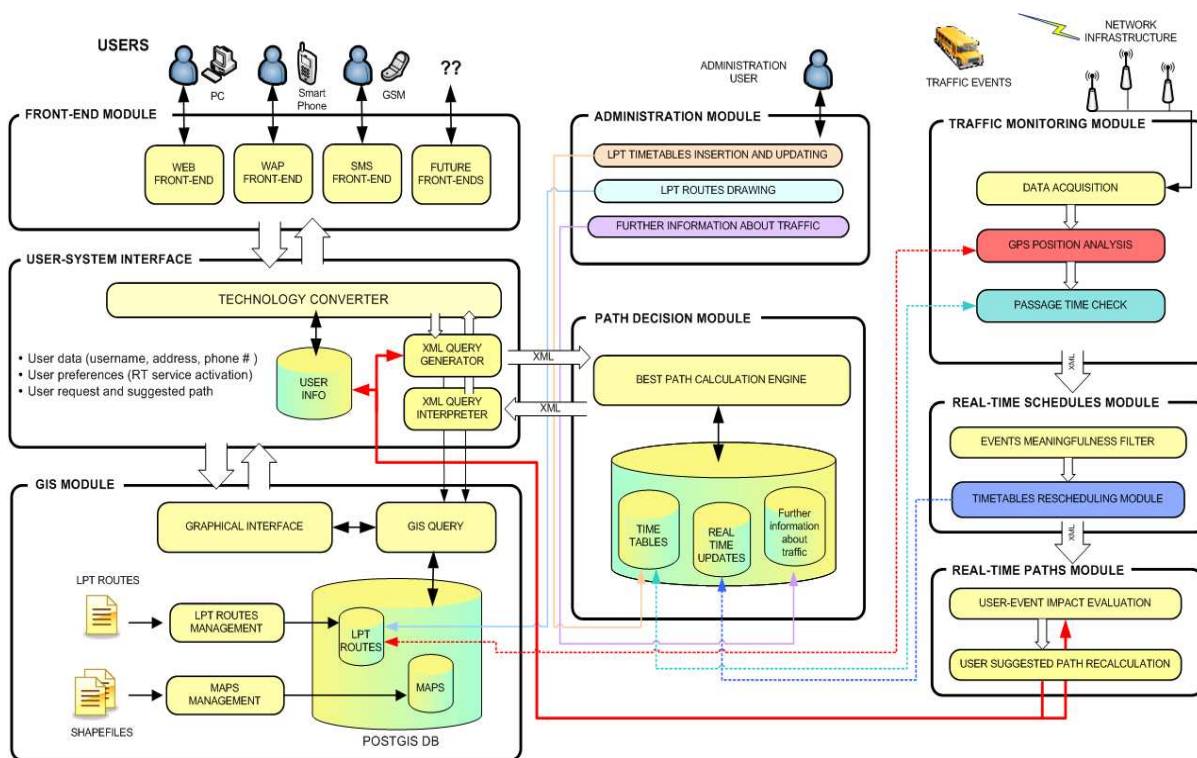


Fig. 7 - Architettura completa del sistema TRIP

Modulo Front-end

Il modulo front-end offre all'utente i punti di accesso al sistema. Attraverso tali punti di contatto utenti e sistema possono scambiarsi informazioni, tipicamente con modalità richiesta dell'utente (ad es. percorso migliore od orari di passaggio) e risposta del sistema. In taluni casi il sistema utilizza il front-end opportuno per fornire indicazioni all'utente non su base richiesta, ma sulla base di eventi specifici.

I front-end messi a disposizione attualmente dal sistema sono:

- portale web
- portale web per dispositivi mobili
- numero GSM per SMS

Utenti dotati di terminali capaci di gestire una connessione Internet possono accedere ad uno dei due portali web. Il punto di accesso ai portali è univoco: all'atto della connessione l'utente verrà reindirizzato automaticamente al portale adatto alle potenzialità del proprio terminale (PC desktop piuttosto che PDA).

La realizzazione di tale front-end ha visto l'utilizzo di tecnologie web dinamiche quali PHP, Ajax e JavaScript, che hanno permesso l'integrazione di controlli avanzati per la gestione grafica dei dati georeferenziati.

Gli utenti possono comunque accedere al sistema in maniera testuale, inviando un SMS opportunamente formattato al numero GSM del sistema TRIP. Questo front-end è stato realizzato attraverso la gestione di un modem GSM controllato da opportuni script in linguaggio Perl, che si occupano dell'inizializzazione e della gestione del modem e del rilevamento e lettura/scrittura di tutti gli SMS scambiati con l'utenza.

Interfaccia Utente-Sistema

Il modulo si occupa di gestire le informazioni scambiate con gli utenti. Nel contempo vengono salvati:

- dati identificativi utente
- tutte le richieste di ogni utente

Il modulo si occupa inoltre della conversione tecnologica, ossia di uniformare le richieste di informazioni effettuate attraverso i diversi front-end.

Ad esempio, nel caso web, i dati che definiscono la richiesta si ottengono dalla lettura del form compilato dall'utente su una pagina web; nel caso di comunicazione via SMS, tali dati vengono estrapolati dal testo del messaggio attraverso opportuni script software.

L'uniformazione delle richieste prevede anche una traduzione delle indicazioni fornite dall'utente (che può essere un indirizzo e numero civico di partenza come un punto di interesse) nella forma di un elemento partenza/arrivo che identifica un certo numero di fermate utili di mezzi pubblici.

Attraverso tale elementi, viene creato un file di interscambio XML opportunamente formattato che servirà come input per il Modulo Decisore del Percorso (descritto in seguito). Tale modulo fornirà in risposta il percorso migliore su mezzi pubblici che soddisfi la richiesta dell'utente.

Analogamente, le informazioni che devono essere inoltrate all'utente vengono convertite in un formato adeguato al front-end (es. pagina web grafica piuttosto che SMS di testo).

In questa fase, attraverso interrogazioni sui dati georeferenziati (si veda descrizione del blocco GIS), vengono anche fornite indicazioni, grafiche o testuali come da front end, su eventuali tratti a piedi da percorrere.

Modulo GIS

Il modulo GIS gioca un ruolo fondamentale in questo tipo di applicazioni, poichè gestisce database di mappe ed interfacce grafiche che rendono questi dati semplicemente fruibili all'utente in modo visuale.

Quando sono coinvolti dati georeferenziati, occorre fare una distinzione fra i dati stessi ed il software per la loro gestione.

Per quanto riguarda il livello dati, è composto da mappe digitali, rappresentate in formati di dettaglio differente. Come già accennato, le mappe usate nel progetto sono quelle di Teleatlas. Questa informazione è il centro di un'ampia gamma di applicazioni che guidano gli utenti nella ricerca dei luoghi e dei servizi richiesti.

Le mappe sono l'unico elemento proprietario di tutto il sistema TRIP. Per quanto riguarda il software di database, in presenza di dati georeferenziati, è stato scelto l'open source relazionale a oggetti PostgreSQL con estensione PostGIS. Viene utilizzato anche un server MySQL per i dati utente, gli orari ed i ritardi.

Il modulo GIS è rappresentato in Fig. 8 ed il suo scopo è far interagire l'utente ed il sistema tramite la visualizzazione di mappe dinamiche su un'interfaccia web. Per fare ciò, oltre alle mappe ed al server PostgreSQL/PostGIS, sono necessari altri strumenti.

Prima di tutto, le mappe devono essere a disposizione in un formato che il sistema GIS sappia comprendere; inoltre, è necessario un tool di visualizzazione interattiva delle mappe; infine, occorre un tool di visualizzazione web.

Gli strumenti scelti per fare questo sono rispettivamente il formato shapefile, GeoServer ed OpenLayers.

L'ESRI Shapefile o semplicemente shapefile è un format dati vettoriale geospaziale per sistemi informativi geografici.

È stato sviluppato ed è regolato da ESRI, come specifica aperta di interoperabilità fra ESRI ed altri prodotti software.

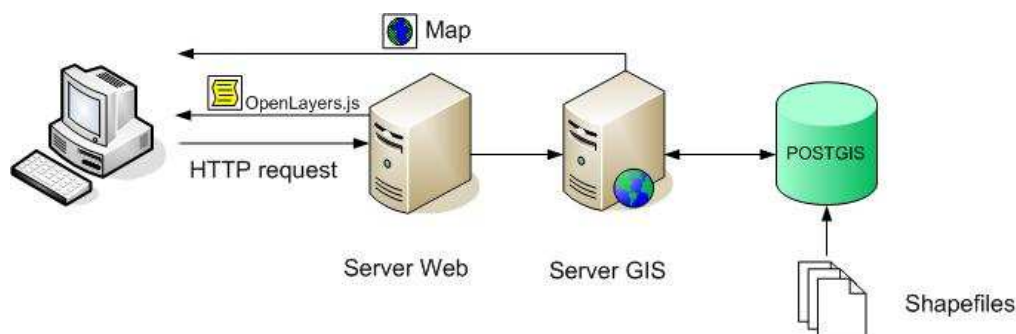


Fig. 8 - Architettura del modulo utente-mappe-utente

GeoServer è un server basato su Java che permette di visualizzare ed editare dati geospaziali. Utilizzando gli standard open stabiliti dall'Open Geospatial Consortium (OGC), GeoServer permette grande flessibilità nella creazione di mappe e condivisione di dati. Implementando lo standard Web Map Service (WMS), GeoServer può creare mappe in una varietà di formati di output.

È costruito su Geotools, un tool Java open source di tipo GIS. È anche conforme allo standard Web Feature Service (WFS), che permette la condivisione effettiva e l'editing di dati per generare mappe.

In più, GeoServer può connettersi con le tradizionali architetture GIS come ESRI ArcGIS e PostgreSQL/PostGIS.

OpenLayers, una libreria di mapping free, è integrata in GeoServer e rende semplice mettere mappe dinamiche sul web. Può mostrare dettagli e marcatori di mappe caricate da ogni sorgente. È una libreria JavaScript per mostrare dati di tipo mappa nella maggior parte dei browser moderni, senza dipendenze server-side.

Sviluppa e continua a migliorare una API JavaScript per la costruzione di ricche applicazioni geografiche basate sul web, simili a Google Maps, con un'importante differenza: è free software, sviluppato per e dalla comunità open source.

Come framework, OpenLayers è fatto per separare i tool per le mappe dalle mappe stesse, in modo che tutti i tool possano lavorare su tutte le sorgenti dati.

Fatte queste considerazioni, la Fig. 8 mostra come i dati base TeleAtlas in formato shapefile possano essere processati per fornire all'utente una mappa dinamica completamente navigabile.

Mappe così riccamente dettagliate possono essere presentate all'utente per diverse applicazioni, come:

- scelta di un punto di partenza o di una destinazione tramite click sulla mappa;
- Visualizzazione di percorsi e fermate di pubblici servizi;
- visualizzazione delle fermate più vicine ad un dato indirizzo;
- visualizzazione del percorso migliore suggerito;
- visualizzazione di punti di interesse, come ristoranti e musei, ecc.;

Ne segue che il database GIS deve immagazzinare sufficiente informazione da fornire mappe dettagliate e precise, visivamente nitide, navigabili e zoomabili.

Come mostrato in Fig. 8, la prima fase prevede che le mappe siano importate dal formato shapefile al formato PostGIS.

Ci sono tanti shapefile quanti sono i livelli rappresentati, come i fiumi, le foreste, gli edifici, le strade, le ferrovie, ecc.

Ogni shapefile viene trasformato in una tabella di database tramite uno specifico script. Questa operazione è fatta una tantum e off-line, all'atto della configurazione del sistema o all'aggiornamento dei dati.

A run-time, invece, per ogni utente connesso al web server che richiede una mappa, il server GIS agisce così:

- reperisce i dati geografici immagazzinati in format testuale nel database;
- applica uno specifico stile ad ogni geometria; questo viene effettuato grazie ad un file XML di rendering, come spessore, colori, ecc.;
- sovrappone i diversi strati per creare la mappa come oggetto geografico (jpg, png, png8);
- fornisce all'utente la libreria JavaScript OpenLayers.js, così che la mappa risulti dinamica, navigabile e zoomabile;

L'utente può anche interrogare i dati geografici del database senza coinvolgere il server GIS, ad esempio chiedendo la fermata più vicina ad un punto di partenza o di destinazione, un elenco di strade o alle caratteristiche delle applicazioni TRIP.

Tutte queste richieste vengono processate da pagine web e da linguaggi di scripting server-side per l'accesso a database via web, come PHP.

Modulo Decisore del Percorso

Questo modulo contiene due entità: una è un motore software che si occupa del calcolo del percorso migliore, l'altra è un database contenente gli orari dei mezzi pubblici.

Gli orari dei mezzi contenuti in questo DB vengono costantemente aggiornati per tener conto dei ritardi effettivi e presunti che il rilevamento delle posizioni dei mezzi forniscono. Ad occuparsi di questo aggiornamento è il modulo Real-time Schedules descritto in seguito.

Ulteriori informazioni sullo stato della viabilità possono essere fornite dal modulo di amministrazione nel caso siano note a priori (ad esempio manifestazioni o lavori in corso programmati).

In base a queste informazioni, il modulo restituisce il percorso migliore in base alle condizioni attuali.

Modulo Amministrativo

Il modulo di amministrazione fornisce una semplice e valida interfaccia per caricare, gestire e aggiornare i dati del sistema.

Grazie a questo modulo è possibile infatti:

- disegnare (o anche importare, in formato shapefile) la posizione delle fermate e i percorsi dei mezzi pubblici (LPT routes);
- importare gli orari di passaggio a tutte le fermate per ciascuna linea;
- modificare e aggiornare i dati sulle variazioni dei percorsi, in seguito ad eventi programmati (es. fiere, mercati, manifestazioni, ecc.);

Il modulo di gestione permette quindi di inserire dati geocodificati, grazie ad una semplice interfaccia web-grafica che si appoggia al modulo GIS e di gestire facilmente e in maniera flessibile anche tutte le possibili variazioni di percorso e di orario che possono interessare una linea.

Ovviamente, questo modulo, perfettamente integrato nell'architettura, condivide la struttura dei dati dell'intero sistema e accede ai diversi database per memorizzare le LPT routes, le timetables and altre informazioni.

E' da sottolineare, inoltre, che l'interfaccia di amministrazione può essere resa disponibile a tutti gli operatori di trasporto pubblico locale, in modo che ciascuno possa inserire i dati, le rotte e gli orari dei passaggi alle fermate della propria flotta.

E' stata anche implementata la possibilità di gestire diversi profili di amministratore, con credenziali e permessi differenziati, per permettere l'accesso a diverse figure professionali come system-manager, helpdesk, impiegati.

Modulo di Monitoraggio del Traffico

Questo modulo ha due compiti: prima di tutto, deve raccogliere informazione sulla posizione geografica di ogni veicolo (coordinate GPS); in second luogo, deve analizzare i dati del traffic e stimare di conseguenza gli intasamenti e gli arrivi in orario e quelli in ritardo alle fermate.

Nell'ambito del progetto TRIP, tutti i veicoli si intendono equipaggiati con un AVM (Automatic Vehicle Monitoring), capace di raccogliere le coordinate GPS ed inviare informazione come posizione ed identificativo ad un Centro di Controllo Dati, tramite diverse tecnologie wireless (GPRS, UMTS, WiFi, Tetra, etc.).

L'utilizzo di diverse tecnologie permette grande flessibilità alle compagnie di trasporto pubblico, che possono sviluppare soluzioni differenti per l'invio di informazioni, in base alle loro specifiche esigenze, quali contratti con compagnie di telecomunicazione, reti radio private, area geografica del servizio, ecc.

Occorre fare due osservazioni. Primo, il formato dell'informazione da mandare al Centro di Controllo deve essere condiviso; per esempio, ogni veicolo può dover notificare la targa, il numero di viaggio, le coordinate GPS ed un time temporale. Secondo, le dimensioni dei dati

da inviare, la modalità e la frequenza sono parametri da progettare in modo opportuno per evitare un sovraccarico di rete.

Il modulo di monitoraggio del traffico contiene un server che raccoglie e processa questi dati in real-time: le posizioni effettive e quelle previste di ogni veicolo vengono verificate e, di conseguenza, vengono calcolati i ritardi alle fermate successive.

Il calcolo del ritardo è eseguito in tre passi. In primo luogo, si deve valutare la distanza fra la posizione corrente del veicolo e la fermata successiva, accedendo a dati georeferenziati tramite il modulo GIS. In secondo luogo, occorre calcolare il tempo necessario a coprire tale distanza. Infine, occorre confrontare il tempo di passaggio calcolato e quello da orario, ottenendo così il ritardo.

Di più, raccogliere dati sulle condizioni della strada attraverso webcam o reti di sensori permette di fare previsioni sul breve periodo per tutti i veicoli che passano su strade congestionate, stimando il loro possibile ritardo.

Infine, a fini di supporto alle decisioni, tramite algoritmi ad hoc si analizzano le aree in cui si verificano molteplici ritardi, causati da più veicoli, in modo da determinare quali percorsi sono affetti da questi problemi.

Ogni ritardo è notificato ed inviato tramite un file XML opportunamente formattato.

Modulo Real-Time Schedules

Questo modulo si occupa di aggiornare le tabelle degli orari dei mezzi pubblici, in modo da tenere conto delle informazioni su ritardi e sulle condizioni del traffico rilevate o previste dal Modulo di Monitoraggio del Traffico.

Per non sovraccaricare il database degli orari con aggiornamenti continui, il modulo prevede di bufferizzare i dati ed effettuare aggiornamenti aggregati ogni minuto.

Il modulo si occupa di informare il modulo Real-Time Paths di tale aggiornamento, in modo che questo possa valutare l'impatto dei ritardi sui percorsi degli utenti confrontandoli con il DB degli orari opportunamente aggiornato.

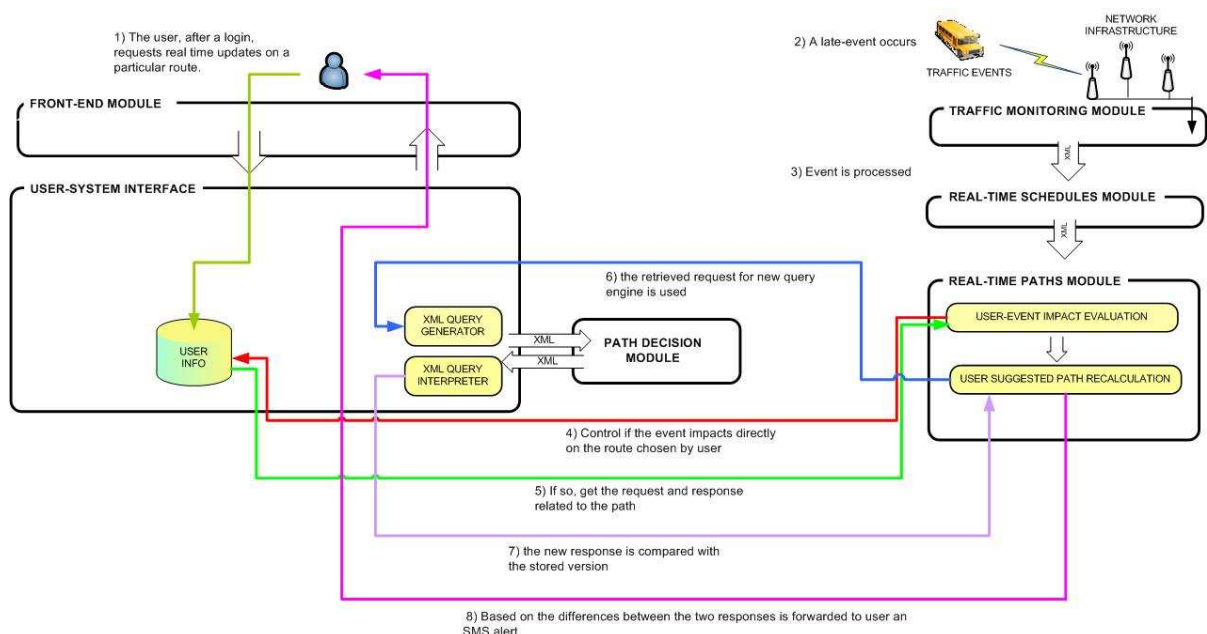


Fig. 9 - Gestione real-time dei ritardi

Modulo Real-Time Paths

Il sistema prevede quindi una infrastruttura in grado di monitorare la situazione reale dei mezzi pubblici. In caso di eventi che hanno impatto sulla viabilità dei mezzi, viene generato un messaggio di “evento ritardo” che riporta l’entità del ritardo e quali mezzi sono interessati.

Lo scenario tipico in cui si attivano le funzionalità di controllo degli eventi in real time prevede il coinvolgimento di tutte le strutture che compongono l’infrastruttura TRIP.

Per descrivere in che modo il sistema prenda in carico questi messaggi si fa riferimento allo schema di Fig. 9, che descrive in un certo numero di passi la sequenza di azioni eseguite per far fronte all’evento ritardo e fornire all’utente informazioni aggiornate in tempo reale.

1. Si prevede che almeno un utente del sistema registri un percorso su cui vuole avere informazioni aggiornate in real time.
2. Le posizioni di tutti i mezzi pubblici sono monitorate costantemente attraverso un modulo GPS in grado di inviare dati verso un Centro di Controllo.
3. Il centro di controllo aggrega i dati dai mezzi e attraverso l’analisi di tali dati è in grado di fornire informazioni su eventi di ritardo (effettivi e/o presunti).
4. La generazione di un evento di ritardo implica un aggiornamento degli orari dei mezzi per tener conto dell’effettiva posizione e velocità dei mezzi: dopo l’aggiornamento eventuali richieste terranno in considerazione tali ritardi.
5. Lo script di gestione real-time controllerà quindi se esistono dei percorsi registrati dagli utenti che sono affetti dai ritardi segnalati; in caso affermativo si procede a valutare se il percorso registrato è ancora il percorso ottimale.
6. Una nuova richiesta viene quindi inoltrata al motore di calcolo del percorso ottimale, eventualmente tenendo conto del fatto che l’utente può essere già in viaggio e quindi trovarsi in una posizione di partenza diversa o addirittura già a bordo di un mezzo.
7. La nuova soluzione fornita viene confrontata con quella già messa a disposizione dell’utente. Se la differenza tra le due è sostanziale, il sistema si occupa di informare l’utente.
8. Generalmente il sistema prevede di inviare all’utente un SMS contenente le informazioni sul nuovo percorso da seguire in modo da contrastare l’effetto dell’evento.

In base alle differenze tra i due percorsi viene il messaggio che viene inviato all’utente assume diverse strutture:

- CASO 1: i due percorsi sono totalmente differenti; questo significa che l’evento ritardo ha un impatto tale sul percorso che conviene utilizzare un’alternativa del tutto diversa. L’SMS sarà quindi del tipo: *Causa evento ritardo il percorso scelto subisce gravi ritardi, il percorso alternativo suggerito è....*
- CASO 2: i due percorsi sono simili, nel senso che utilizzano le stesse linee di mezzi pubblici ma a passaggi diversi (ad esempio il secondo percorso prevede sempre di utilizzare la linea 4, ma invece di salire sul bus delle 8:20 si deve prendere il bus alle 8:30) . In questo caso l’SMS sarà del tipo: *Il percorso scelto è interessato da un ritardo. Il percorso è ancora quello ottimale ma arriverai a destinazione con un ritardo di....*
- CASO 3: il percorso rimane invariato (o affetto da ritardi minimi); in questo caso non viene inoltrato nessun tipo di SMS all’utente.

Come evoluzione del sistema si può considerare di inviare al posto del semplice SMS, un MMS (Multimedia Message Service) che descrive le nuove condizioni di viaggio, eventualmente con una piccola mappa allegata (si veda Fig. 10).



Fig. 10 - Servizio TRIP di aggiornamento dei percorsi su cellulari (MMS).

VALUTAZIONE DEGLI SCENARI APPLICATIVI

Il progetto TRIP si prefigge di migliorare l'attuale sistema di trasporti torinese, in modo da renderlo più efficiente ed incentivare più persone all'utilizzo dei mezzi pubblici. Un efficace ed efficiente sistema di trasporti pubblici è un fattore di vitale importanza per assicurare ai cittadini un ambiente più vivibile e salutare, migliorando di conseguenza la qualità della vita delle persone.

Un efficiente sistema di trasporto pubblico può contribuire a:

- Aumentare il numero delle persone che si serviranno dei mezzi pubblici per muoversi all'interno delle città, evitando di conseguenza l'utilizzo di mezzi privati quali macchina o motocicli.
- Migliorare la qualità, la competitività e l'efficienza del trasporto pubblico in Piemonte e, di conseguenza, in Italia e Europa.
- Riduzione del traffico sulle strade urbane e interurbane. Una riduzione della congestione delle strade permette alle persone di muoversi più velocemente nelle strade cittadine.
- Miglioramento della qualità della vita dei cittadini, sia indirettamente attraverso il suo impatto sull'economia della città, sia direttamente attraverso il suo impatto sulle necessità quotidiane delle persone.
- Riduzione di perdite di tempo per i cittadini rendendo disponibile la migliore rotta possibile, gli orari, la frequenza e le diverse alternative.
- Rendere disponibili i trasporti pubblici a tutte le categorie di passeggeri, fornendo informazioni relative a quali mezzi dispongono della giusta assistenza per persone disabili.
- Riduzione del numero di incidenti sulla strada dal momento che un efficiente sistema di trasporti riduce il volume di traffico e di conseguenza il numero di veicoli sulle strade.
- Promozione di una progettazione e gestione di infrastrutture cercando di ridurre danni all'ambiente circostante.
- Permettere all'industria di trasporti di collaborare con i fornitori di servizi per sviluppare soluzioni innovative

Se un sistema intermodale di trasporto risulterà sicuro ed efficiente, garantendo un efficiente movimento di persone e informazioni, questo potrebbe essere applicato anche a merci e risorse, portando un enorme beneficio alla crescita della nostra economia.

Il settore degli ITS – Intelligent Transportation Systems

I trasporti rivestono un ruolo centrale per lo sviluppo economico e sociale di ogni Paese. Un sistema di trasporto efficiente consente di creare nuovi mercati e di potenziare quelli esistenti, e costituisce pertanto una leva essenziale per sostenere e favorire una crescita economica, creando occupazione e ricchezza. Un sistema poco efficiente riduce le possibilità di raggiungere nuovi mercati, allontana l'orizzonte degli scambi, comprime le capacità produttive, limitando le potenzialità di crescita economica e sociale.

Su questo versante, le società industrializzate stanno giocando una partita complessa e delicata. In tutte le nazioni economicamente avanzate negli ultimi anni si è assistito a un notevole incremento della domanda di mobilità sia di persone sia di merci. Tale crescita quantitativa è stata anche accompagnata da una profonda modifica strutturale della

domanda. In particolare, per i passeggeri, al tradizionale modello di mobilità a “spostamento singolo”, nel quale il viaggio aveva come origine e destinazione il domicilio, si è sostituito un approccio a “spostamenti multipli”, nel quale un viaggio tipo è costituito da un insieme concatenato di trasferimenti effettuati per compiere attività di varia natura. Questo incremento della domanda di mobilità che aveva segnato e addirittura anticipato la curva della crescita, rischia di saturare la capacità delle infrastrutture stradali, senza che le altre modalità riescano a porsi, a breve, come alternative realmente competitive alla strada. La conseguenza è un aumento esponenziale delle congestioni, con impatti negativi sull’ambiente, la qualità della vita, la sicurezza, che comportano costi enormi per la collettività.

La difficile questione del recupero della mobilità sta spingendo le Pubbliche Amministrazioni ad un salto di qualità delle politiche del settore, che consiste nel passaggio da un approccio improntato al controllo dei trasporti ad una vera e propria gestione dei trasporti, che distribuisca i flussi di traffico in modo equilibrato tra le varie modalità, ottimizzi l’utilizzo delle infrastrutture, permetta spostamenti più sicuri, veloci ed economici, ed incrementa i servizi per i cittadini in modo da invogliarli a utilizzare i trasporti pubblici.

Il settore ITS, appare come un settore nuovo, emergente e soprattutto di business. Ciò è dimostrato non solo dal trend crescente del numero di aziende che si occupano di tali attività e del personale dedicato a tale settore, ma soprattutto dalla crescita dei ricavi fatturati, in controtendenza rispetto all’andamento decrescente del mercato in genere.

Questo Capitolo si propone di descrivere il settore degli ITS, settore all’interno del quale si colloca il progetto TRIP, analizzandolo sia a livello nazionale che a livello internazionale.

Sistemi di trasporto intelligente.

L’Intelligent Transportation Systems (ITS) è un’iniziativa a livello mondiale tesa ad aggiungere l’Information and Communications Technology alle infrastrutture dei trasporti e dei veicoli. Punta a gestire fattori eterogenei come veicoli, carichi e strade al fine di aumentare la sicurezza riducendo l’usura dei veicoli, i tempi di trasporto ed i costi del carburante.

Attraverso l’integrazione dei diversi sistemi e servizi, l’impiego delle tecnologie avanzate, le potenzialità comunicative della telematica, ITS consentono di sviluppare nuove soluzioni per un trasporto dei passeggeri e delle merci più efficiente, più competitivo e soprattutto più sicuro.

L’ITS nasce dalla necessità di gestire i problemi causati dalla congestione del traffico attraverso una sinergia delle nuove tecniche informatiche per la simulazione, il controllo in tempo reale e le reti di comunicazione. Il traffico è in costante aumento in tutto il mondo, ed è il risultato di una crescente motorizzazione, urbanizzazione e della crescita demografica, nonché di un sostanziale cambiamento nella densità di popolazione. Il traffico riduce l’uso delle infrastrutture dei trasporti ed aumenta il tempo di percorrenza, l’inquinamento ed il consumo di carburante.

Gli Intelligent Transportation Systems variano a seconda delle tecnologie applicate, dai sistemi di gestione base come navigatori satellitari, sistemi di controllo semaforici, o rilevatori di velocità per applicazioni di monitoraggio applicate a sistemi di telecamere a circuito chiuso, fino alle applicazioni avanzate che integrano dati in tempo reale provenienti da varie fonti esterne, tipo informazioni meteorologiche. L’esercizio dei Sistemi finora realizzati, sia a livello urbano che extraurbano, ha permesso di valutare in modo tangibile i benefici apportati dagli ITS. Dati della Commissione Europea rivelano che in diverse applicazioni realizzate in Paesi dell’Unione Europea sono state ottenute riduzioni dei tempi di spostamento nell’ordine del 20%, aumenti della capacità della rete del 5-10%, e

miglioramenti in termini di sicurezza del 10-15% grazie a strategie coordinate di informazione e controllo.

L'ITS è un settore attivo in Italia sin dagli anni ottanta, ma con notevoli sviluppi a partire dal decennio successivo, in parallelo con la crescita del settore negli altri maggiori paesi industrializzati.

La definizione dell'Architettura Telematica italiana *ARTIST*, all'inizio di questo decennio, ha segnato un passo decisivo per promuovere l'integrazione degli ITS, favorendo da una parte la diffusione del loro impiego e dall'altra la creazione di un mercato dei sistemi e dei servizi. *ARTIST*, tuttavia, fornisce un impianto generale, con i suoi schemi di riferimento che devono essere testati nella pratica da progetti pilota focalizzati su tematiche prioritarie per il sistema dei trasporti. Infatti, solo attraverso problemi concreti si possono identificare le criticità e gli ostacoli da rimuovere per pervenire a soluzioni affidabili e robuste, estendibili a tutto il territorio nazionale, presupposto indispensabile per creare le condizioni per il decollo di un "vero" mercato dei sistemi e dei servizi ITS in Italia. Come è già stato descritto, anche per il progetto TRIP è prevista una fase di test e simulazione di una piattaforma pilota. Questo servirà per studiarne i comportamenti e risolverne gli eventuali problemi che si manifesteranno nel corso della simulazione.

L'architettura *ARTIST* deve essere quindi considerata come il punto di partenza per una serie di azioni che consentano al nostro Paese di cogliere appieno le opportunità che gli ITS possono portare all'economia, alla competitività, alla qualità della vita.⁴

Il Quadro Nazionale.

L'Italia è uno dei Paesi Europei a più alta densità di traffico interno, con 947.205 milioni di passeggeri-Km/anno e di questi, il 92,40% sceglie la strada. Il traffico merci superiore ai 50 km sul territorio nazionale è di 211.206 milioni di tonnellate-km/anno, di cui il 65,33% avviene su strada⁵. L'Italia rappresenta un sistema, dunque, dalla mobilità intensa, ai limiti della saturazione. Su questo quadro, già di per sé preoccupante, grava il pesante squilibrio modale che rende sempre più concreto il rischio di collasso della mobilità stradale. Per quanto concerne la ripartizione modale, infatti, i dati di traffico attuali sul territorio nazionale confermano l'assoluto dominio dei trasporti su strada, sia per il traffico dei passeggeri (92,75%) che per quello delle merci superiore ai 50 km (68,25%). Le previsioni mostrano che questi numeri tendono ad aumentare nei prossimi anni. Sono dunque destinati ad aggravarsi pesantemente quel quadro di fenomeni che hanno segnato negativamente il sistema dei trasporti italiano negli ultimi decenni a causa dell'aumento dei volumi di traffico: la mancanza di efficienza, il peggioramento delle condizioni ambientali e la sicurezza. Questi fenomeni non incidono soltanto in maniera diretta sulla vita dei cittadini, ma comportano un elevatissimo costo a carico della collettività sia in termini sociali che di perdita di competitività economica del "Sistema Paese".

Il Quadro Internazionale.

Le prime attività di R&D sugli ITS risalgono agli anni Ottanta, principalmente negli Stati Uniti e in Giappone. A partire dai primi anni Novanta, anche la Commissione Europea ha finanziato numerosi programmi di ricerca e, successivamente, di implementazione di applicazioni ITS in tutti i paesi dell'Unione Europea, a sostegno anche delle diverse

⁴ Per una descrizione più dettagliata delle funzioni e degli obiettivi di *ARTIST*, si veda l'Appendice 1.

⁵ Conto Nazionale delle Infrastrutture e dei Trasporti, Dati 2004.

iniziative intraprese a livello nazionale e locale nei singoli Stati⁶. L'esercizio dei Sistemi finora realizzati in tutto il mondo, ha permesso di valutare in modo tangibile i benefici apportati dagli ITS. Riguardo all'Europa in particolare, la Commissione Europea riporta che in diverse applicazioni sono stati ottenuti i risultati che seguono⁷:

- Riduzione dei tempi di spostamento dell'ordine del 20%
- Aumento della capacità della rete del 5-10%
- Diminuzione del numero di incidenti del 20%
- Riduzione della congestione delle strade: 15%
- Emissioni inquinanti: 10%
- Consumi energetici: 12%

A livello dei singoli Stati, Paesi come Germania, Francia, Gran Bretagna, Olanda, Finlandia, Norvegia, Svezia, hanno avviato importanti programmi nazionali di Ricerca e Sviluppo per promuovere la diffusione su larga scala degli ITS in tutto il territorio. Si tratta di progetti pluriennali che prevedono il finanziamento di progetti pilota su applicazioni considerate strategiche per il sistema di trasporti nazionale, nei quali sono coinvolte le più importanti realtà, sia pubbliche che private, operanti nel mondo dei trasporti nei diversi Paesi. L'obiettivo è di studiare e realizzare dei sistemi fondati su soluzioni condivise che soddisfino pienamente le reali esigenze degli utenti finali, in modo anche da incentivare la crescita del mercato dei componenti, dei sistemi e dei servizi.

Il Quadro Tecnologico.

Le esperienze condotte in questi anni in numerosi Paesi Europei, negli Stati Uniti ed in Giappone⁸, hanno permesso di verificare che non è possibile dare risposte ai problemi della mobilità unicamente attraverso la realizzazione di nuove infrastrutture, che sono comunque assolutamente necessarie e previste. Occorre pertanto adottare un approccio strategico diverso, ovvero pensare ai trasporti in modo nuovo, ossia come un *sistema integrato*, in cui informazione, gestione e controllo operano in sinergia.

L'obiettivo è pertanto quello di intervenire direttamente sulla domanda, ottimizzando la gestione delle infrastrutture e delle piattaforme logistiche, e riorganizzando i flussi di traffico in modo da promuovere il riequilibrio fra i diversi mezzi di trasporto.

L'applicazione ai trasporti dei metodi e delle tecnologie proprie dell'informatica e delle comunicazioni ha permesso di sviluppare sistemi capaci di affrontare in modo "intelligente" i problemi della mobilità nella loro globalità: i Sistemi di Trasporto intelligenti, noti anche come "Sistemi Telematici per Trasporti". Per ITS si intende *"l'insieme delle procedure, dei sistemi e dei dispositivi che consentono, attraverso la raccolta, elaborazione e distribuzione di informazioni, di migliorare il trasporto e la mobilità di persone e merci"*. La commissione Europea classifica come ITS i sistemi per:

- La gestione del traffico e della mobilità.
- L'Informazione all'utente.
- La gestione del trasporto pubblico.
- La gestione delle flotte e del trasporto merci.
- Il pagamento automatico.
- Il controllo avanzato del veicolo .

⁶ Il progetto TRIP che verrà sviluppato a Torino, è in parte finanziato dalla Comunità Europea.

⁷ Libro Bianco "La politica europea dei trasporti fino al 2010: il momento delle scelte", Settembre 2001.

⁸ Progetti di infomobilità e mobilità sostenibile effettuati in questi Paesi sono descritti al Capitolo 3.

- La gestione delle emergenze e degli incidenti.

Sistemi ITS in Italia

L'Italia come detto sopra, è attiva nel settore dei Sistemi ITS dalla seconda metà degli anni Ottanta. Pubbliche Amministrazioni, aziende, istituti di ricerca e università italiane hanno preso parte a tutti i Programmi Quadro di Ricerca e Sviluppo della Commissione Europea con risultati significativi⁹. Sistemi ITS per la gestione del traffico e della mobilità sono in esercizio in numerose città italiane, tra cui Roma, Torino, Milano, Firenze, Bologna, Genova, Perugia, Napoli, Brescia, Salerno ecc... Inoltre, l'80% circa delle Aziende di trasporto pubblico locale sono dotate di sistemi di localizzazione e monitoraggio delle flotte mirati a migliorare l'offerta del servizio¹⁰.

Particolarmente interessanti sono i risultati dell'esperienza di Torino, dove è operativo un sistema telematico di controllo del traffico con il quale si è ottenuta una riduzione dei tempi di viaggio per il traffico privato nell'area controllata dal sistema del 20% circa, ed un aumento del 17% della velocità commerciale dei mezzi pubblici grazie alla gestione della priorità semaforica¹¹. A Roma il sistema integrato di gestione e controllo del traffico, messo in esercizio nel 2000, in occasione del Giubileo, ha permesso una diminuzione del 10% dei tempi di viaggio nelle zone gestite dalla Centrale di Controllo del Traffico¹².

Notevoli progressi sono stati fatti anche nel settore dei sistemi di bordo, al fine di aumentare il livello di sicurezza dei veicoli. L'iniziativa di maggiore rilevanza intrapresa in Italia per incentivare in modo sostanziale il pieno sviluppo degli ITS nel nostro Paese, e del relativo mercato dei sistemi e dei servizi all'utenza che tali sistemi rendono possibili, è, tuttavia, la definizione, da parte del Ministero delle Infrastrutture e dei Trasporti dell'Architettura ITS Nazionale: ARTIST¹³.

In termini di mobilità in Italia, tra gli interventi più efficaci si segnala innanzitutto il potenziamento del Trasporto pubblico locale (con corsie riservate e vie preferenziali, sistemi di integrazione tariffaria, strumenti per l'infomobilità). Inoltre, tra i nuovi interventi che stanno lentamente prendendo piede in Italia, si segnalano:

- Politiche di tariffazione e pricing: Pedaggio urbano (accesso a pagamento a strade o a particolari zone urbane), Park pricing (sosta su strada a pagamento); Park and ride (agevolazione nell'interscambio tra automobile e mezzo pubblico nelle città);
- Interventi di gestione della domanda: moderazione del traffico (traffic calming), limitazioni della circolazione veicolare, introduzione di servizi di car sharing e car pooling, introduzione della figura del mobility manager, information Technology (ITS) (gestione flussi veicolari, servizi non convenzionali);
- Realizzazione dei percorsi sicuri casa-scuola (con specifici progetti pilota è possibile incentivare la mobilità pedonale e ciclabile per gli spostamenti casa-scuola, ottenendo un deciso miglioramento del livello di accessibilità, vivibilità e sicurezza del territorio per i bambini ed i ragazzi);

⁹ Il progetto TRIP è finanziato dalla Regione Piemonte, la quale riceve finanziamenti dalla Comunità Europea.

¹⁰ TTS Italia, "Risultati sull'indagine sui Sistemi ITS e mobilità nelle Amministrazioni locali e nelle Aziende di trasporto pubblico: stato attuale e prospettive di sviluppo", Report aggiornato 2003.

¹¹ Fonte: 5T, Dati 2005.

¹² Fonte: Servizi per la Mobilità del Comune di Roma, Dati 2004.

¹³ Si veda l'Appendice 1 per una descrizione dettagliata di ARTIST

- Riduzione dell'inquinamento attraverso lo sviluppo della mobilità ciclabile;
- Uso del territorio ([Land-Use](#)) e trasporti: pianificazione integrata trasporti e territorio

La Regione Piemonte ha individuato la Piattaforma Mobilità Intelligente, o Infomobilità, per riunire e mettere a sistema tutti i diversi attori che siano, o possano risultare in futuro, specificatamente interessati nella diffusione di tecnologie e sistemi in ambito di infomobilità, nell'ottica di promuovere e sviluppare il valore del territorio. L'obiettivo della piattaforma è facilitare la connessione e ricercare le sinergie sui diversi progetti e tra gli attori che contribuiscono alle attività di sviluppo del settore infomobilità, interpretando e avvicinando le iniziative alle esigenze del settore automotive, ICT e tecnologie satellitari.

Gli ambiti rispetto a cui verranno perseguiti gli obiettivi di piattaforma saranno:

1. Gestione del traffico, finalizzata a permettere un utilizzo efficiente della risorsa viaria, un controllo adeguato della domanda di mobilità, una disponibilità di servizi complementari di supporto, quali pagamento automatico e prenotazione parcheggi, gestione congiunta in ambito urbano ed extraurbano;
2. Servizi di informazione per l'utente di mobilità riguardanti i mezzi (multimodalità) di trasporto e i percorsi a monte del trasporto, le situazioni relative alle congestioni e al superamento delle conseguenze durante lo spostamento e alla disponibilità di servizi complementari per i fornitori dei servizi, l'integrazione di dati di traffico con dati sul tempo e situazione delle strade;
3. Sistemi per la safety (sicurezza delle persone) quali gli strumenti informativi da parte degli Enti Gestori, i sistemi anticollisione per veicoli e i sistemi di assistenza alla guida;
4. Sistemi per la security (sicurezza degli oggetti e dei sistemi) quali i sistemi di telediagnosi dei veicoli, i sistemi di antifurto satellitare, i sistemi per la sicurezza informatica;
5. Sistemi di gestione delle emergenze, finalizzati a migliorare la tempestività e la qualità del servizio di emergenza attraverso l'utilizzo di sistemi di localizzazione del richiedente il soccorso o della squadra di soccorso;
6. Servizi di fleet e freight management, a supporto di una gestione efficiente delle missioni di trasporto per flotte di mezzi che movimentano persone o merci in ambito metropolitano.

Le tecnologie, in particolare quelle ICT, costituiscono un elemento abilitante per le politiche di piattaforma.

Il perseguimento dell'eccellenza e dell'innovazione dovrà contraddistinguere l'evoluzione da soluzioni standard, basate sul concetto di automazione, a soluzioni evolute, basate sul concetto di integrazione, fino a giungere a soluzioni di avanguardia, basate sul concetto di convergenza.

Gli attori della piattaforma infomobilità piemontese possono essere ricondotti a macro categorie:

- Autorità pubbliche, gestori autostrade, gestori del trasporto pubblico, gestori del trasporto merci, polizia, interessati a risolvere o diminuire determinati problemi (ad es. i problemi di congestione);
- Strutture di ricerca, integratori di sistemi, costruttori di veicoli, di componentistica, terminali e sensori, gestori di TLC, gestori di servizi e applicazioni, interessati a costruire e testare le soluzioni;

- Utenti che beneficiano direttamente delle soluzioni (guidatori/navigatori/trasportatori e relative strutture associative), e utenti che controllano i sistemi e forniscono gli input (gestori del controllo del traffico, i gestori dei servizi di emergenza, etc.) interessati a fruire e beneficiare dei servizi;
- Autorità che emettono le regole su come specificare, sviluppare e realizzare i sistemi e i servizi, locali (specifiche regolamentazioni e indirizzi di soluzioni) e nazionali (regole generali e architettura).

La piattaforma infomobilità dovrà perseguire gli obiettivi degli attori del territorio e più in generale del sistema Piemonte, attraverso la guida, l'ottimizzazione, l'integrazione e la sinergia delle iniziative, a beneficio della collettività e del tessuto imprenditoriale.

Il mercato mondiale ITS

Gli ITS costituiscono un'importante opportunità di mercato. Analisi condotte su scala internazionale riportano che nella sola Europa gli ITS nel 2002 hanno generato un mercato di circa 1,44 miliardi di dollari, con un aumento fino a 2,6 miliardi di dollari nel 2006. Per il 2002, il mercato degli ITS su scala mondiale è stato valutato in circa 5,7 miliardi di dollari. Nel 2010 il mercato mondiale è ipotizzato in circa 18,5 miliardi di dollari, con un trend di crescita annuo del 15,11% nel decennio 2000-2010.

La figura sottostante sintetizza l'andamento stimato del mercato ITS dal 2000 al 2010 in Europa, Stati Uniti, Giappone e Resto del Mondo¹⁴.

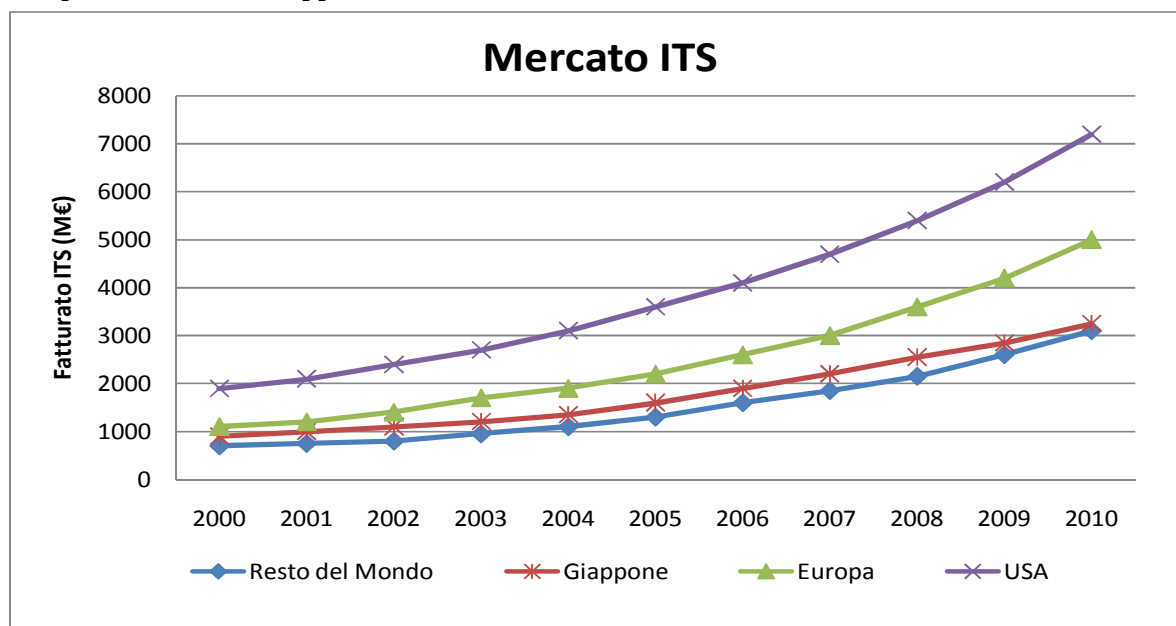


Fig. 11 - Il Mercato Mondiale dei sistemi ITS, valori attuali e previsioni future

Le proiezioni rivelano un mercato in forte espansione in tutte le aree geografiche considerate, caratterizzato da un tasso di crescita rapido, specie in Paesi emergenti come

¹⁴ Per Resto del mondo si intende un dato integrato che comprende Canada, Australia, Malesia, Cina, India, Corea del Sud, Brasile, Medio Oriente.

quelli asiatici, che può rappresentare, quindi, un'occasione estremamente interessante per aziende del settore, in particolare per quelle italiane.

Il comparto italiano degli ITS attualmente è costituito da una molteplicità di attori con interessi ed obiettivi differenti, riconducibili sostanzialmente a tre categorie: il mondo dell'offerta, composto dalle aziende che producono e forniscono sistemi e servizi ITS, quello della domanda, formato principalmente dagli Enti pubblici e privati principali utilizzatori di tali tecnologie, e quello della ricerca, attiva in numerosi progetti sugli ITS finanziati con fondi nazionali e comunitari. Per quanto riguarda il progetto TRIP, ad esempio, si possono identificare sul lato dell'offerta TSF (Tele Sistemi Ferroviari), CNR e Almagia CNR; sul lato della domanda la regione Piemonte e in particolare la città di Torino e infine, sul lato della ricerca IEIIT.

L'utente finale degli ITS è comunque la collettività, che è destinata ad assumere un ruolo sempre più incisivo sullo sviluppo soprattutto del mercato dei servizi legati agli ITS. Questo è ampiamente dimostrato dalla diffusione crescente registrata in Italia negli ultimi anni dei sistemi di tele-pedaggio e di diversi servizi di infomobilità, che rappresentano degli esempi significativi di un mercato giovane ma con grandi potenzialità.

Offerta italiana ITS

Gli ITS sono sistemi complessi, che richiedono tecnologie e competenze diverse, Questo è ampiamente rispecchiato dal panorama delle aziende che in Italia si occupano di tale settore, il quale risulta, infatti, contrassegnato da una molteplicità di attori, che agiscono su segmenti di mercato disparati.

Gli ITS comprendono un insieme molto vasto di sistemi e di servizi, che trovano applicazione in tutti i campi in cui è articolato il settore dei trasporti. Come riferimento di base per l'analisi strutturale del mercato, si è scelto di adottare la schematizzazione che classifica gli ITS nelle seguenti macroaree:

- Sistemi di controllo e la gestione del traffico e della mobilità
- Sistemi per l'informazione all'utenza per il trasporto stradale
- Sistemi per la gestione del trasporto pubblico
- Sistemi per la gestione delle flotte e del trasporto merci
- Sistemi per il pagamento automatico
- Sistemi per il controllo avanzato del veicolo e la navigazione

Se si parla, invece, di trasporto ferroviario, le applicazioni ITS comprendono sistemi di telelocalizzazione, monitoraggio e gestione del traffico, di informazione ai passeggeri e di bigliettizzazione elettronica. Per quanto riguarda il trasporto marittimo, sono stati considerati i sistemi di navigazione satellitare e i sistemi per il controllo del traffico marittimo e portuale delle merci, dei passeggeri e dei vettori.

Il segmento per cui si registra la maggiore concorrenza è quello degli ITS per le flotte. Prodotti e servizi legati al controllo e alla gestione delle flotte, insieme alla progettazione e realizzazione di sistemi di bordo per il controllo del veicolo e la navigazione, rappresentano i segmenti ITS a cui le aziende sono maggiormente interessate.

La figura sottostante rispecchia lo squilibrio del mercato ITS verso la "strada". Il mercato ITS si presenta, quindi, sostanzialmente incentrato sulle applicazioni per la modalità stradale. Queste, infatti, nel 2003 hanno prodotto circa l'81% dell'intero fatturato ITS,

mentre solo il 17% è venuto dalle applicazioni per la ferrovia. Del tutto trascurabile è, invece, la quota di mercato delle applicazioni ITS per il trasporto marittimo (2%).

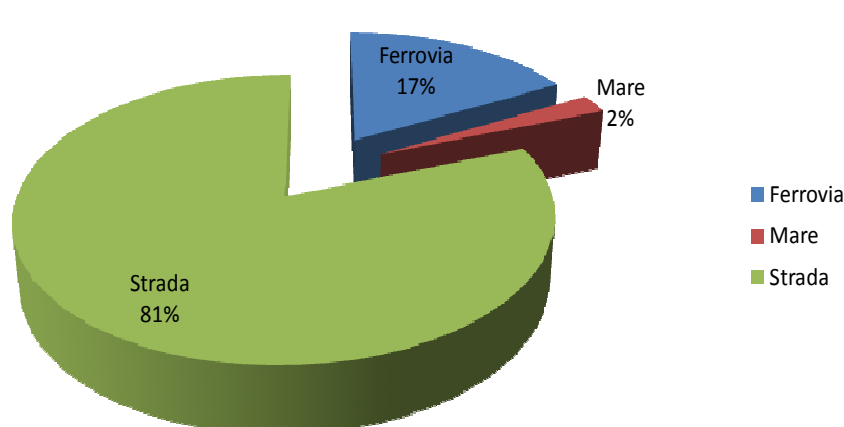


Fig. 12 - Mercato ITS per modalità di trasporto

Import-Export, un mercato ancora chiuso.

Riguardo alla situazione degli scambi commerciali, l'analisi di questo mercato mostra la sua attenzione verso una clientela nazionale. Fra i principali clienti istituzionali vi sono: Ministeri, Amministrazioni Locali e Forze dell'Ordine, mentre, nel settore privato, i maggiori acquirenti sono le aziende di trasporto pubblico, i gestori di infrastrutture autostradali e le aziende di trasporto merci.

Un discorso a parte meritano i servizi di infomobilità, per i quali stanno assumendo un ruolo sempre più rilevante gli utenti privati; al momento attuale, gli utenti possono ricevere servizi di infomobilità o attraverso un operatore telefonico, oppure sottoscrivendo un abbonamento direttamente con l'azienda stessa. In entrambi i casi, il servizio presuppone l'utilizzo di un sistema di telefonia mobile. Questo lascia prevedere che la diffusione su larga scala di tecnologie di comunicazione come il GPRS e l'UMTS possa determinare, a breve, una penetrazione rapida di questi servizi presso una larga fetta di pubblico.

Per quanto concerne la distribuzione delle esportazioni per i singoli segmenti di mercato, nel 2003 i settori di maggiore interesse per il mercato estero sono risultati gli ITS per la gestione delle flotte e gli ITS per il controllo del veicolo e la navigazione. Seguono i sistemi di gestione traffico e mobilità e le soluzioni ITS per l'infomobilità. Sono poche, invece, le aziende che hanno venduto all'estero sistemi di pagamento automatico¹⁵.

¹⁵ Allo scopo di ricavare un quadro il più possibile realistico dei valori attuali del mercato ITS italiano, è stata eseguita un'indagine basata sulla stima diretta delle grandezze significative, utilizzando dei questionari progettati ad hoc da sottoporre ad un campione rappresentativo del comparto ITS nazionale. L'indagine è stata eseguita da TTS Italia. Il campione di indagine è costituito da 160 aziende produttrici e fornitrici di ITS. Nell'indagine è stato richiesto alle aziende intervistate di fornire delle previsioni a breve-medio periodo sullo sviluppo del mercato ITS in Italia. Come orizzonti temporali sono stati considerati il 2005 e il 2009.

I Paesi europei destinatari dell'offerta ITS italiana di mercato sono stati:

- Gestione del traffico e della mobilità: Olanda, Portogallo, Svezia, Belgio.
- Informazione all'utenza: Germania, Francia, Inghilterra, Ungheria;
- Gestione flotte e merci: Grecia, Romania, Belgio, Spagna;
- Pagamento automatico: Grecia, Inghilterra, Austria;
- Controllo avanzato del veicolo: Francia, Spagna, Inghilterra, Belgio;
- ITS per il trasporto ferroviario: Germania, Svizzera, Austria;
- ITS per il trasporto marittimo: Svezia, Norvegia, Olanda, Francia.

Relativamente alle importazioni, dall'indagine eseguita sul mercato ITS, risulta che le aziende ITS hanno importato per lo più componenti per il controllo avanzato del veicolo, per i sistemi per la gestione del traffico e della mobilità e per le applicazioni ITS per il trasporto ferroviario. La maggior parte delle aziende ha importato i prodotti da Paesi europei; seguono l'Asia e il Nord America.

I principali Paesi europei da cui provengono le importazioni sono risultati:

- Gestione del traffico e della mobilità: Germania, Inghilterra, Olanda, Grecia.
- Informazione all'utenza: Germania, Inghilterra, Svezia.
- Gestione flotte e merci: Germania, Inghilterra, Belgio.
- Pagamento automatico: Inghilterra, Germania, Olanda, Francia.
- Controllo avanzato del veicolo: Francia, Spagna, Inghilterra, Belgio;
- ITS per il trasporto ferroviario: Germania.
- ITS per il trasporto marittimo: Svezia.

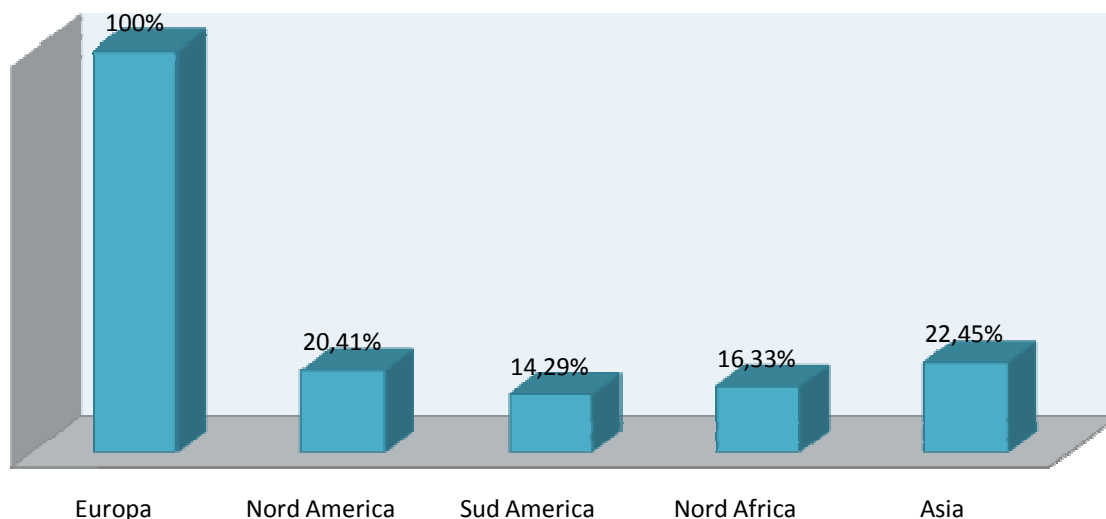


Fig. 13 - Paesi di destinazione delle esportazioni ITS

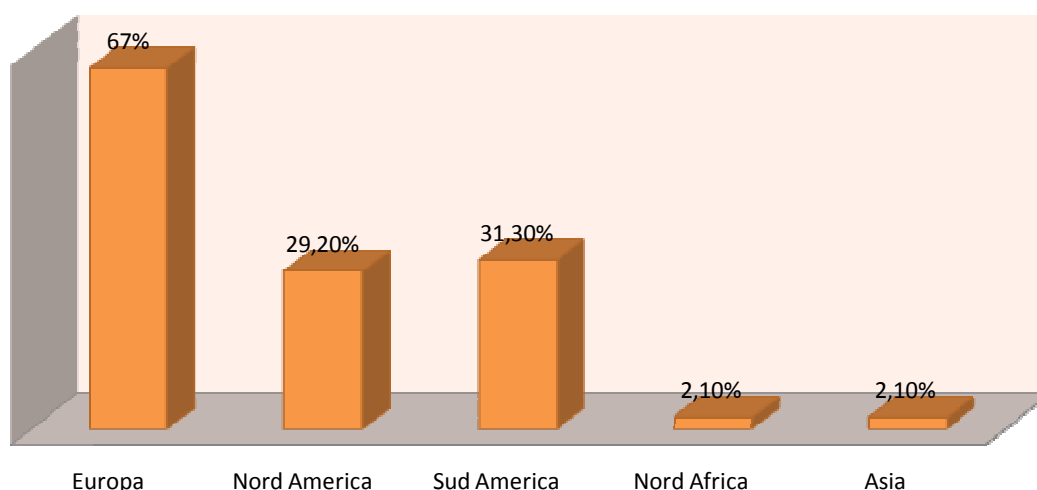


Fig. 14 - Paesi di provenienza delle importazioni ITS

Gli investimenti in R&S.

La ricerca e l'innovazione sono determinanti per la competitività delle aziende che operano negli ITS. Il livello elevato di concorrenza che caratterizza questo mercato, soprattutto in ambito internazionale, spinge, infatti, gli operatori italiani ad innovare continuamente i prodotti e a ricercare nuove soluzioni, per non perdere la propria posizione sul mercato.

Allo scopo di ricavare un quadro il più possibile realistico dei valori attuali del mercato ITS italiano, è stata eseguita un'indagine basata sulla stima diretta delle grandezze significative, utilizzando dei questionari progettati ad hoc da sottoporre ad un campione rappresentativo del comparto ITS nazionale. L'indagine è stata eseguita da TTS Italia. Il campione di indagine è costituito da 160 aziende produttrici e fornitrici di ITS. Nell'indagine è stato richiesto alle aziende intervistate di fornire delle previsioni a breve-medio periodo sullo sviluppo del mercato ITS in Italia. Come orizzonti temporali sono stati considerati il 2005 e il 2009, allo scopo di pervenire ad una visione quanto possibile realistica delle prospettive di questo settore e di cosa gli operatori si attendono nel breve-medio termine. La tabella seguente mostra la previsione di medio periodo degli investimenti ITS delle Aziende di trasporto pubblico (Milioni di €).

Investimenti ITS 2003	Investimenti ITS 2005	Previsione di Investimenti ITS 2009
7,9	28	39,8
	+254,43% rispetto al 2003	+42,19% rispetto al 2005

Tab. 3 - Investimenti ITS delle Aziende di trasporto pubblico

Le previsioni, fornite dalle aziende analizzate, confermano il trend crescente del mercato ITS. I numeri al 2009 appaiono molto incoraggianti: infatti secondo gli operatori, la diffusione su larga scala dell'UMTS e dei servizi satellitari disponibili EGNOS e GALILEO dovrebbe far crescere il mercato ITS. Il grafico seguente mostra l'andamento del mercato ITS italiano dal 2001 e con una previsione sino al 2009.

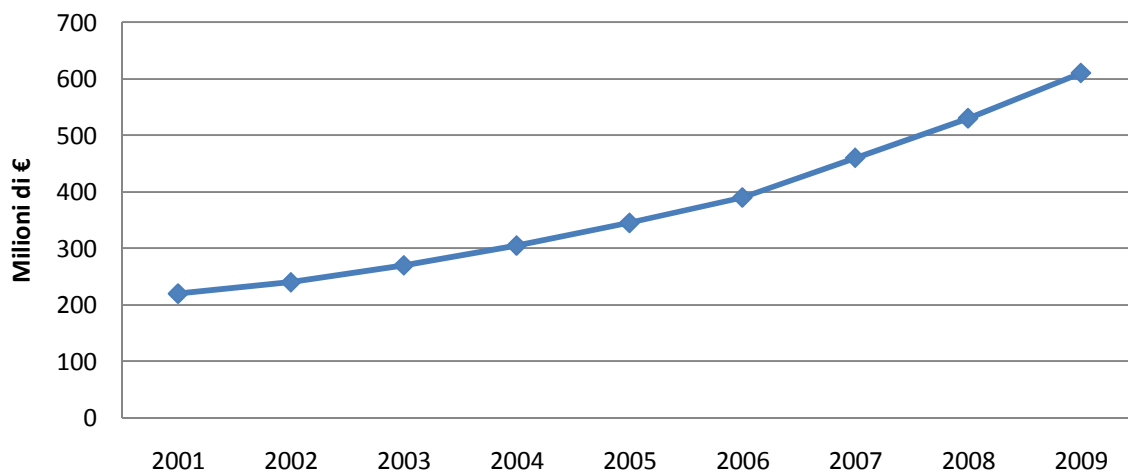


Fig. 15 - Mercato italiano ITS: analisi attuale e previsioni future

Passando, infine, ad esaminare le previsioni per i singoli segmenti di mercato, i dati relativi sono rappresentati nel seguente grafico.

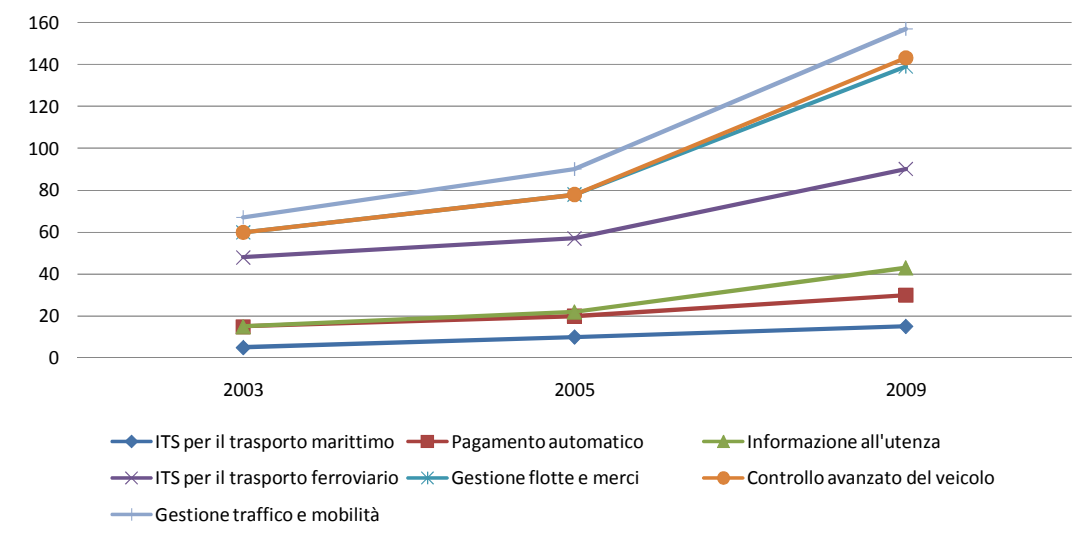


Fig. 16 - Previsione di fatturato ITS per segmento di mercato

Il grafico mostra che i settori con il maggior tasso di crescita risultano essere gli ITS per la gestione del traffico e della mobilità e gli ITS per l'informazione all'utenza. Seguono gli ITS per il controllo del veicolo, gli ITS per la gestione flotte e gli ITS per il trasporto ferroviario. Le previsioni al 2009 indicano che le tecnologie che avranno più mercato saranno i sistemi e servizi di infomobilità e gli ITS per il controllo del veicolo e la navigazione. Questi dati rispecchiano l'influenza che si prevede possa avere sugli ITS l'introduzione massiccia sul mercato delle nuove tecnologie di comunicazione mobili e satellitari, per la creazione e lo sviluppo di nuovi sistemi e servizi per la mobilità, sia per il trasporto dei passeggeri che per quello delle merci.

La domanda italiana degli ITS

La domanda degli ITS comprende un insieme estremamente variegato di attori: le pubbliche amministrazioni, i gestori delle infrastrutture, i costruttori di veicoli, le aziende del trasporto pubblico locale, le aziende di autotrasporto passeggeri e merci, gli operatori di flotte di diversa natura, i gestori di terminali e gli stessi cittadini, che rappresentano i principali utenti del servizio di trasporto.

Le amministrazioni comunali

I segmenti di mercato che più interessano le Amministrazioni Comunali sono:

- *I sistemi per il controllo e la gestione del traffico e della mobilità:* sistemi per il monitoraggio del traffico, sistemi VMS, sistemi semaforici avanzati per il controllo del traffico, sistemi per il controllo degli accessi in zone a traffico limitato, sistemi per il monitoraggio dell'ambiente, sistemi per la gestione della sosta.
- *I sistemi per l'informazione all'utenza per il trasporto stradale.*
- *I sistemi per la gestione delle flotte e del trasporto merci:* sistemi per la gestione delle flotte per la distribuzione urbana delle merci.

Allo scopo di conoscere nel dettaglio su quali tecnologie in particolare sono stati concentrati gli investimenti, il dato complessivo relativo agli ITS per la gestione del traffico e della mobilità è stato disaggregato nei diversi sottosistemi compresi in questo segmento. I risultati sono sintetizzati nel grafico seguente.

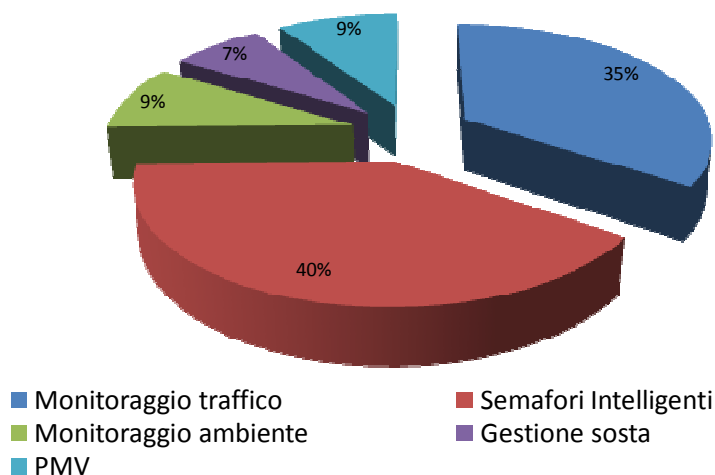


Fig. 17 - Investimenti delle Amministrazioni comunali per sistemi di gestione del traffico e della mobilità

Le previsioni di investimento per i singoli segmenti di mercato sono sintetizzate nel grafico seguente.

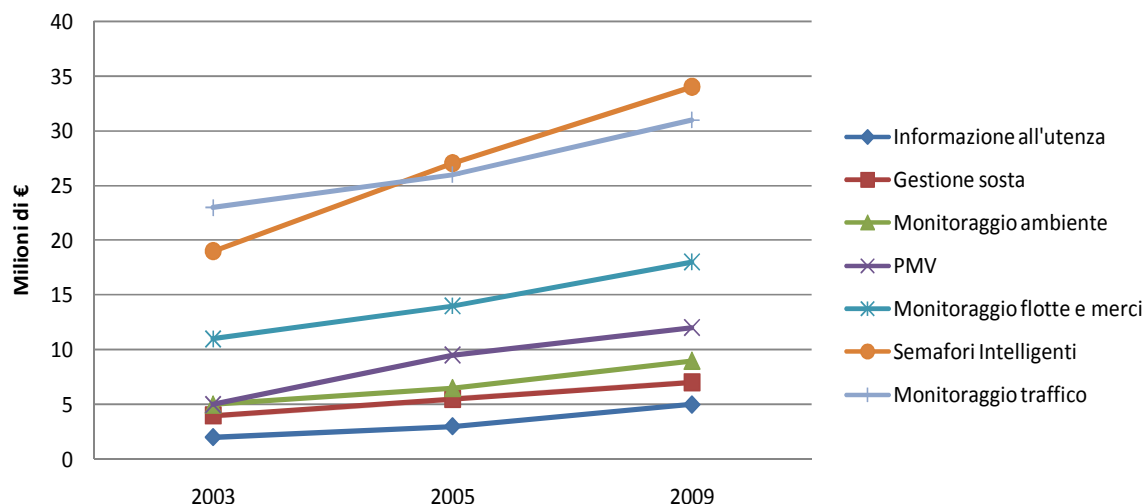


Fig. 18 - Previsioni investimenti ITS per segmento di mercato

Le tecnologie oggetto di maggiori investimenti saranno i sistemi per il monitoraggio del traffico ed i semafori intelligenti, mentre più ridotti saranno gli investimenti previsti per i sistemi di monitoraggio dell'ambiente e per la gestione della sosta.

Gli ITS dal punto di vista delle aziende di trasporto pubblico.

I primi investimenti sugli ITS risalgono alla seconda metà degli anni Ottanta e sono stati realizzati dalle aziende di trasporto pubblico delle principali città italiane. Le aziende che invece operano in città di piccole e medie dimensioni hanno iniziato ad investire sugli ITS a partire dalla fine degli anni Novanta.

Tutte le aziende concordano nell'affermare che con l'adozione degli ITS hanno ottenuto benefici significativi in termini di un miglioramento sia della qualità del servizio offerto, che della gestione delle risorse interne e della sicurezza del servizio di trasporto.

Le aziende di trasporto pubblico stanno rivolgendo una sempre maggiore attenzione agli ITS. Dall'analisi di questo settore, si evince che i segmenti di mercato che maggiormente interessano le aziende di trasporto pubblico sono:

- **Sistemi di informazione all'utenza:** informazioni pre-trip (internet, call center, televideo...), informazioni a bordo (display a messaggio variabile, messaggi precodificati...), informazioni alle fermate (paline intelligenti);
- **Sistemi di gestione del trasporto pubblico:** sistemi AVL¹⁶ e AVM¹⁷ per la localizzazione dei mezzi ed il monitoraggio della flotta, sistemi di bigliettazione elettronica, sistemi di monitoraggio delle corsie preferenziali, sistemi per la valutazione degli inquinanti.

¹⁶ Con l'espressione Automatic vehicle location (o AVL) si indica una tecnologia che consente di localizzare automaticamente i veicoli in diversi modi e per le necessità più svariate

¹⁷ L'Automatic Vehicle Monitoring (AVM), indicato anche come *telelocalizzazione*, è un sistema che consente di tracciare il movimento dei mezzi del trasporto pubblico locale seguendo una rotta predefinita.

Errore. Non si possono creare oggetti dalla modifica di codici di campo.

Fig. 19 - Investimenti ITS delle aziende di trasporto

I sistemi per il monitoraggio delle flotte hanno assorbito il 65% degli investimenti per questo segmento; seguono gli investimenti per la bigliettazione elettronica con il 31% circa e, infine, i sistemi per il monitoraggio dei passeggeri a bordo con il 4% circa degli investimenti.

Una crescita significativa sia nel breve che nel medio periodo è stata ipotizzata per i sistemi di informazione all'utenza, sia per quanto concerne l'informazione per la pianificazione dei viaggi (pre-trip) sia per le informazioni durante il viaggio (en-route). Nello specifico, per le informazioni pre-trip si è sviluppata in maniera efficace la diffusione dell'informazione via Internet. Per l'informazione en-route, invece, si è pensato a diffondere l'utilizzo delle paline intelligenti, impiegate per fornire agli utenti il tempo di attesa alle fermate, e dei sistemi per l'informazione a bordo, come ad esempio, i display a messaggio variabile.

Per quanto riguarda la bigliettazione elettronica, le tecnologie di maggiore interesse saranno quelle con smart-card e obliteratrici magnetiche e i sistemi contactless, in grado di dar luogo all'esazione attraverso il passaggio della carta vicino a un sensore che ne rileva la presenza addebitando l'importo dovuto senza alcun contatto.

Il grafico seguente mostra le previsioni di investimenti ITS delle aziende di trasporto pubblico suddivisi per segmenti di mercato.

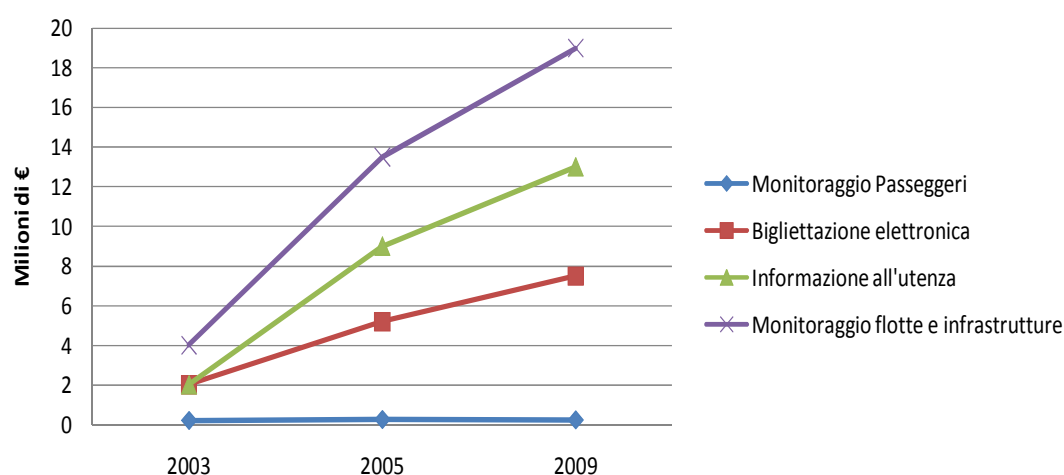


Fig. 20 - Investimenti ITS delle aziende di trasporto divise per segmenti di mercato

Conclusioni sulla situazione italiana.

Il trend crescente degli investimenti per gli ITS conferma l'interesse per le soluzioni fornite dall'innovazione tecnologica per affrontare i complessi problemi della mobilità.

Anche le amministrazioni comunali, con un tasso di crescita medio annuo di investimento del 19%, sono ormai consapevoli che non è più sufficiente limitarsi alla realizzazione di nuove infrastrutture, comunque necessarie, ma occorre gestire al meglio, attraverso l'uso degli ITS, quelle esistenti. Inoltre, da questo quadro emergono interventi per il trasporto pubblico, fondamentale per una migliore gestione del traffico in ambito urbano.

Il confronto con il mercato internazionale

Gli ITS costituiscono una realtà consolidata anche a livello internazionale, specie in quelle nazioni in cui l'esigenza di fronteggiare una domanda di mobilità in continuo aumento ha spinto le Pubbliche Amministrazioni ad intraprendere un processo di riorganizzazione del sistema dei trasporti, attraverso l'integrazione delle reti e l'ottimizzazione dell'utilizzo delle infrastrutture.

Sono sempre più numerosi, quindi, i Paesi che guardano con interesse agli ITS, e questo è confermato anche dalla crescita rapida registrata da questo mercato negli ultimi anni.

Regione/Paese	2000	2001	2003	2006	2010	Tasso di crescita annuo%
USA	1901	2117	2728	4064	7204	14,3
Europa	1115	1258	1668	2601	4920	16
Giappone	899	996	1269	1857	3207	13,6
Resto del Mondo	604	689	942	1539	3108	17,8

Tab. 4 - Mercato ITS mondiale per aree geografiche (Milioni di \$)

Nella tabella sono indicati i valori del mercato mondiale stimato per gli ITS dal 2000 al 2010.

Conclusioni sul mercato ITS

L'indagine sul mercato ITS ha permesso di dimostrare che gli ITS, oltre ad essere uno strumento indispensabile per l'attuazione delle politiche di mobilità volte ad un trasporto più efficiente, più competitivo, e, soprattutto, più sicuro, costituiscono anche un'importante opportunità di business.

I risultati presentano gli ITS come un settore in crescita costante, con prospettive di mercato del tutto positive nel breve e nel medio periodo nei diversi segmenti di mercato, capace quindi di generare occupazione e di stimolare la creazione di nuove iniziative imprenditoriali.

A fronte di un quadro dai toni decisamente incoraggianti, occorre tuttavia rimarcare degli elementi di difficoltà. Innanzitutto, la carenza di standard rischia di rallentare, se non di ostacolare, il pieno decollo del mercato dei sistemi e dei servizi. C'è quindi la necessità di linee guida generali per orientare i sistemi verso soluzioni aperte ed interoperabili, fondate su standard comuni che possano essere condivisi dai diversi attori della domanda e dell'offerta, al fine di stimolare lo sviluppo di un mercato concorrenziale dei servizi basati sugli ITS.

L'Architettura Nazionale ARTIST, rappresenta un primo, decisivo, passo in questa direzione. ARTIST, tuttavia, fornisce un impianto generale che deve essere necessariamente applicato in progetti concreti, in modo da identificare le eventuali barriere realizzative ed organizzative che frenano la diffusione degli ITS, e proporre interventi adeguati. ARTIST, quindi, deve essere considerata come un punto di partenza per tutta una serie di iniziative, mirate a favorire la diffusione degli ITS, anche capitalizzando ed ottimizzando i risultati delle esperienze già condotte e di quelle in corso, allo scopo di creare le condizioni per accelerare la crescita della domanda stessa degli ITS.

Un'ulteriore criticità riguarda, infine, la presenza ancora troppo scarsa dell'offerta ITS italiana sui mercati internazionali. Sarebbe quindi opportuno, da parte sia dei privati che

delle Istituzioni, promuovere una massiccia azione di internazionalizzazione di questo settore, con attenzione soprattutto a quei mercati che presentano attualmente i maggiori tassi di sviluppo e penetrazione di questi sistemi, come ad esempio il mercato asiatico, su cui è concentrata l'attenzione di tutti gli operatori internazionali.

La conclusione che emerge è che, perché gli ITS possono rappresentare un'opportunità, sociale od economica, importante e concreta per il nostro Paese, è necessario che tutti gli attori: Istituzioni, aziende, mondo della ricerca, collaborino insieme, sulla base di un programma strategico definito, per superare le barriere che ancora ne ostacolano la piena diffusione. A questo proposito, sull'esempio anche di quanto avvenuto in altri Paesi, sembra auspicabile promuovere anche in Italia un Piano strategico nazionale per gli ITS, nel quale siano indicati gli obiettivi e le priorità degli ITS in Italia nel breve-medio termine, le azioni necessarie per conseguirli attraverso lo strumento dell'Architettura Nazionale, e i benefici attesi, e che possa costituire anche da riferimento per le iniziative di internazionalizzazione di questo settore.

Questo permetterebbe di indirizzare gli investimenti in un quadro di maggiore certezza, e stimolare, quindi, la crescita del mercato tenendo conto anche delle sfide poste dalle esigenze di operabilità dei sistemi a livello europeo.

TECNOLOGIA A SUPPORTO DEL PROGETTO TRIP

Il progetto TRIP, può essere definito come un progetto di Information and Communication Technology. In questo caso, con il termine ICT si intende l'insieme dei sistemi (piattaforme hw/sw, logiche e servizi) costituiti da apparati mobili, elementi distribuiti sul territorio e unità centrali finalizzati a migliorare l'efficienza, la qualità, la sicurezza e l'impatto eco-sostenibile della mobilità dei veicoli, delle persone e delle merci.

Le nuove tecnologie oggi esistenti, rendono possibili soluzioni innovative per:

- la gestione del traffico urbano
- l'informazione all'utenza in movimento
- la gestione del trasporto pubblico
- la gestione delle flotte e del trasporto merci
- il pagamento automatico sui mezzi pubblici
- i sistemi avanzati anticollisione

Questo Capitolo si propone di descrivere le tecnologie che saranno alla base del successo del progetto TRIP. In particolare verranno descritte approfonditamente le seguenti tecnologie:

- *Wireless* (Wi-Fi, WiMax)
- *GPS*
- *GPRS*
- *UMTS*
- *EDGE*
- *DVB-H*
- *Sistemi di localizzazione: Galileo, Sistemi AVM* (Automatic Vehicle Monitoring) e *Sistemi AVL* (Automatic Vehicle Location).

Tecnologia wireless, un aiuto per incrementare i servizi ITS

Le reti wireless nascono all'inizio degli anni novanta, ma solamente verso la fine del decennio vengono standardizzate dall'IEEE come 802.11. Questa tecnologia, a quell'epoca, non permetteva l'intercompatibilità e si è arrivati fino alla seconda metà del 1999 per avere uno standard comune a tutti i produttori: l'IEEE 802.11b funzionante alla frequenza di 2,4 GHz e l'IEEE 802.11a funzionante alla frequenza di 5 GHz. La tecnologia wireless è nata con lo scopo di rispondere a esigenze quali:

- Eliminazione del cablaggio.
- Riduzione dei costi delle infrastrutture.
- Realizzazione di reti temporanee.
- Facilità di aggiungere nuovi utenti.

Questi punti sono stati ampiamente soddisfatti da questa tecnologia che inoltre ha risposto alle sempre maggiori richieste di velocità. Le reti wireless sono e ci si aspetta che saranno sempre più una importante forma di connessione per molte attività. Il mercato per i

dispositivi wireless è stimato in crescita. Il giro d'affari, di 300 milioni di dollari nel 1998, è passato ad 1,6 miliardi nel 2005.

Un tempo, a causa del prezzo degli apparecchi wireless, questa tecnologia veniva utilizzata solo in caso di condizioni in cui l'uso di cavi era difficile o impossibile. Al diminuire dei prezzi, le WLAN stanno entrando anche nelle case, permettendo la condivisione di dati e della connessione Internet (mediante Router) tra i diversi computer

Le tipologie di rete wireless sono tre:

- **PAN** (Personal Area Network) Le reti PAN sono composte da collegamenti a portata ridotta, tipicamente limitata agli oggetti indossati da una persona, o a quelli contenuti in una automobile. Una tecnologia molto popolare in questo campo è Bluetooth, e viene usata per abolire i collegamenti fisici (cavi) tra i dispositivi.
- **WLAN**. In informatica, Wireless Local Area Network, termine inglese abbreviato in WLAN, indica una “rete locale senza fili” che sfrutta la tecnologia wireless. Con la sigla WLAN si indicano genericamente tutte le reti locali di computer che non utilizzano dei collegamenti via cavo per connettere fra loro gli host della rete.
- **WAN wireless** (Wide Area Network)

Ultimamente stanno prendendo piede, invece, le reti wireless a larga banda e a copertura estesa per le quali è stato sviluppato lo standard IEEE 802.16 (WiMAX).

Wi-Fi

Wi-Fi, abbreviazione di *Wireless Fidelity*, è il nome commerciale delle reti locali senza fili (WLAN) basate sulle specifiche IEEE 802.11. Un dispositivo, anche se conforme a queste specifiche, non può utilizzare il logo ufficiale Wi-Fi se non ha superato le procedure di certificazione stabilite dal consorzio Wi-Fi Alliance (Wireless Ethernet Compatibility Alliance), che testa e certifica la compatibilità dei componenti wireless con gli standard 802.11x (della famiglia 802.11). La presenza del marchio Wi-Fi su di un dispositivo dovrebbe quindi garantirne l'interoperabilità con gli altri dispositivi certificati, anche se prodotti da aziende differenti.

Le reti Wi-Fi sono infrastrutture relativamente economiche e di veloce attivazione e permettono di realizzare sistemi flessibili per la trasmissione di dati usando frequenze radio, estendendo o collegando reti esistenti oppure creandone di nuove.

La fonte di connettività a banda larga può essere via cavo (ADSL o HDSL), oppure via satellite. Oggi esistono connessioni a internet satellitari bidirezionali, che consentono alte velocità di trasferimento dei dati sia in *download* che in *upload*¹⁸. A partire dalla fonte di banda, si può espandere la rete attraverso la tecnologia Wi-Fi utilizzando delle apposite antenne. Le coperture di queste antenne sono fondamentalmente di due tipi: omnidirezionali e direttive.

Le antenne omnidirezionali vengono utilizzate di norma per distribuire la connettività all'interno di uffici, o comunque in zone private e relativamente piccole. Oppure, con raggi d'azione più grandi, si possono coprire aree pubbliche (come aeroporti, centri commerciali, ecc.).

Con le antenne direttive è invece possibile coprire grandi distanze, definibili in termini di chilometri, e sono utili proprio per portare la banda larga nei territori scoperti dalla rete

¹⁸ Viene chiamata upload l'azione di *invio* alla rete di un file. L'azione inversa è chiamata download.

cablata. In questo caso, è possibile aggregare più reti in un'unica grande rete, portando la banda in zone altrimenti scollegate.

Vantaggi e svantaggi del wi-fi

Vantaggi:

- Molte reti riescono a fornire la cifratura dei dati e il roaming potendosi spostare dalla copertura di un access point ad un altro senza una caduta della connessione internet, al di fuori del raggio di azione che delimita un hot-spot.
- Diversamente dal cellulare, l'esistenza di uno standard certificato garantisce l'interoperabilità fra apparecchio e rete anche all'estero, senza i costi della cablatura (essendo tecnologia wireless) per una più rapida e facile installazione ed espansione successiva della rete.
- La presenza di parecchi produttori ha creato una notevole concorrenza abbassando di molto i prezzi iniziali di questa tecnologia.

Svantaggi:

- Il tempo di latenza delle schede wi-fi è leggermente superiore a quelle basate su cavo con una latenza massima nell'ordine di 1-3 ms (per cui questo particolare è trascurabile, a differenza delle connessioni GPRS/UMTS che hanno latenze nell'ordine di 200-400 ms). Uno svantaggio delle connessioni wi-fi 802.11a/g può essere la stabilità del servizio, che per via di disturbi sul segnale talvolta può essere discontinua (il segnale può ad esempio essere disturbato da forni a microonde nelle vicinanze che quando sono in funzione disturbano la frequenza operativa di 2,4-2,5 Ghz).

Wi-MAX

WiMAX, acronimo di Worldwide Interoperability for Microwave Access, è una tecnologia che consente l'accesso a reti di telecomunicazioni a banda larga e senza fili¹⁹. L'acronimo è stato definito da WiMAX Forum, consorzio formato da più di 420 aziende, il cui scopo è sviluppare, supervisionare, promuovere e testare la interoperabilità di sistemi basati sullo standard IEEE 802.16, conosciuto anche come WirelessMAN (wireless metropolitan area network).

WiMAX è una tecnologia di trasmissione senza fili d'accesso a banda larga, in grado di fornire elevate prestazioni, in termini di velocità di trasmissione di dati, a basso costo. La possibilità di essere utilizzato su qualsiasi tipo di territorio, a prescindere dalle caratteristiche geografiche (si possono utilizzare sistemi WiMAX, in tutti gli ambienti, dall'urbano al rurale), rende WiMAX competitivo sul mercato per ogni tipo di utenza.

La tecnologia supporta velocità di trasmissione di dati condivisi fino a 70 Mbit/s in aree metropolitane, utilizzando una tecnologia che non richiede la visibilità ottica tra le stazioni.

WiMAX ha un potenziale tale da consentire di allargare a molti milioni gli accessi ad Internet senza fili, proprio per il basso costo e la relativa facilità di implementazione della struttura: la copertura senza fili di WiMAX si misura in km², mentre la copertura Wi-Fi viene misurata in decine di m². Le stazioni-base WiMAX dovrebbero riuscire a coprire intere aree metropolitane collocando ciascuna area entro una WirelessMAN e consentendo

¹⁹ BWA - Broadband Wireless Access.

che all'interno di questa si realizzi una vera mobilità senza fili. I proponenti sperano che questa tecnologia nel giro di pochi anni venga adottata anche per i computer portatili e per i PDA²⁰. Di seguito vengono elencate alcune tra le caratteristiche che rendono lo standard WiMAX competitivo:

- *Flessibilità*: WiMAX è in grado di supportare sistemi punto-multipunto (P-MP) e multipunto-multipunto (MP-MP, detti anche MESH²¹).
- *Sicurezza*: WiMAX implementa diverse tecniche di crittografia, sicurezza ed autenticazione contro intrusioni da parte di terzi.
- *Qualità del Servizio (QoS)*: WiMAX supporta 4 tipologie di qualità del servizio: Unsolicited Grant Service (UGS) per sistemi real time di dimensione fissa (es. VoIP), Real-Time Polling Service (rtPS) per sistemi real time di dimensione variabile (es. applicazioni video), Non Real-Time Polling Service (nrtPS) per flussi di dati tolleranti al ritardo (es. applicazioni FTP), Best Effort (BE) per flussi di dati dove non è richiesto livello minimo di servizio.
- *Throughput*: utilizzando lo schema di modulazione IEEE 802.16 WiMAX trasporta una grossa quantità di traffico con un alto livello di efficienza.
- *Installazione*: WiMAX non richiede equipaggiamenti particolari.
- *Interoperabilità*: WiMAX è uno standard e quindi indipendente dal tipo di apparato o dal provider.
- *Mobilità*: grazie allo standard 802.16e WiMAX permette connessioni in ambienti mobili fino a 160 km/h.
- *Costi / Copertura*: lo standard di tipo "open", unito all'economia di scala dovuta alla produzione dei componenti WiMAX, dovrebbe ridurre notevolmente i costi per fornitori ed utenti.
- *NLOS (not line of sight)*: la capacità di trasmettere attraverso territori parzialmente ostruiti (es. monti), grazie alla modulazione utilizzata.

Inoltre, le potenzialità di WiMAX fanno sì che questa tecnologia venga utilizzata nelle seguenti applicazioni:

²⁰ Occorre però precisare che il vero roaming a banda larga di tipo cellulare senza fili si baserà sullo standard 802.20, peraltro compatibile con WiMAX.

²¹ *Wireless mesh networking* è una rete a maglie implementata tramite una wireless local area network. Mentre Internet è fondamentalmente una infrastruttura di comunicazione elettronica, cooperativa, che si serve per la trasmissione dei segnali di mezzi fisici quali fili, fibre ottiche, ecc., una rete a maglie è una rete di comunicazione senza fili cooperativa costituita da un gran numero di nodi che fungono da ricevitori, trasmettitori e ripetitori, costituendo, appunto, una di rete senza fili cooperativa.

Questo tipo di infrastruttura è decentralizzata (non ci sono server centrali), relativamente economica, molto adattabile e resistente, dal momento che ogni nodo deve solamente trasmettere un segnale al massimo fino al nodo successivo. I nodi fungono da ripetitori per trasmettere il segnale dai nodi più vicini ai peers (nodi equivalenti) che sono troppo distanti per essere raggiunti; in questo modo abbiamo una rete capace di coprire grandi distanze, specialmente su terreni accidentati o comunque "difficili". Le reti a maglie sono inoltre estremamente affidabili, poiché ogni nodo è connesso a molti altri nodi. Se un nodo viene meno alla rete, a causa di problemi hardware o qualunque altro motivo, i nodi vicini semplicemente cercano altri percorsi per trasmettere il segnale (si rivolgono ad altri nodi). Le capacità della rete possono essere incrementate semplicemente installando altri nodi. Le reti a maglie possono includere dispositivi sia fissi che mobili.

- Connessione tra hotspots Wi-Fi, e tra hotspot e rete Internet (con HotSpot si intendono i "punti di accesso" pubblici, come aeroporti o università, che offrono una connessione).
- servizi e connessione ad alta velocità per la trasmissione da apparecchi mobili (cellulari, PDA).

Sul mercato, i principali concorrenti del WiMAX sono i sistemi 3G, come UMTS e CDMA2000.

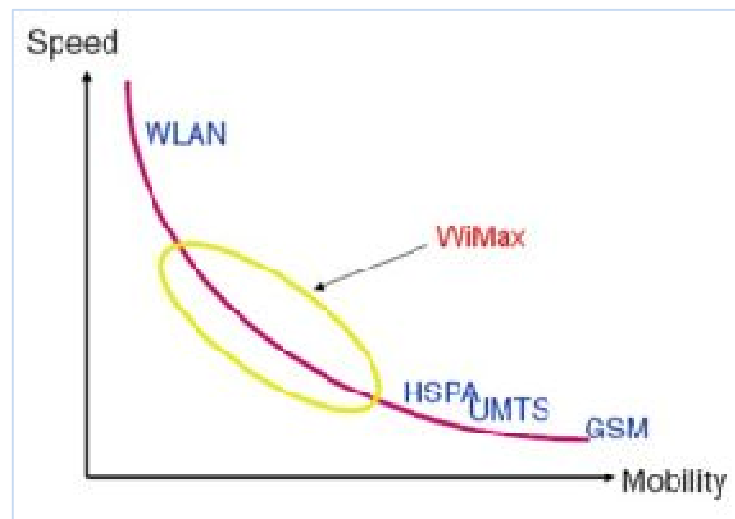


Fig. 21 - Confronto sintetico delle principali tecnologie wireless

Elementi di una Rete WiMAX

Una rete WiMAX è generalmente costituita da diversi elementi: la Base Station (BS), la Subscriber Station (SS), i Terminal Equipment (TE), la Repeater Station (RS).

La *Base Station* (stazione base) in una comunicazione radio, è una postazione di comunicazione fissa wireless che ha il compito di ricevere, amplificare e ritrasmettere i segnali provenienti da stazioni lontane. Il compito principale di una Base Station è quella di fornire copertura su un'area e raccogliere tutto il traffico proveniente dalle varie Subscriber Station connesse ad essa. La Base Station rappresenta il gateway (il passaggio di ingresso e di uscita) che permette agli utenti di connettersi alla rete WiMAX, con il vincolo di un unico punto di accesso alla volta.

Le Base Station sono collegate alla rete dalla quale ricevono/trasmettono flussi dati delle varie Subscriber Station. La Subscriber Station inoltra, quindi, il traffico proveniente dal Terminal Equipment a cui è connessa verso la Base Station che a sua volta inoltra il traffico verso la destinazione finale.

Il *Terminal Equipment* è l'apparato terminale tramite il quale l'utente si connette alla rete (es. PC, terminale mobile).

La *Repeater Station* è l'apparato che ripete le trame ricevute e serve a raggiungere utenze molto distanti dalla Base Station. Può essere usato anche come ripetitore per il traffico proveniente dal Terminal Equipment a cui è connessa verso la Base Station.

Standard di base

WiMAX si basa sulla famiglia di standard IEEE 802.16, noto anche come WirelessMAN, specializzato nell'accesso senza fili a banda larga del genere punto-multipunto.

Standard	Descrizione	Data di pubblicazione
IEEE std 802.16-2001	definisce reti metropolitane senza fili che utilizzano frequenze superiori a 10 GHz (fino a 66 GHz), con LoS	08-apr-02
IEEE std 802.16c-2002	definisce le opzioni per le reti che utilizzano frequenze tra 10 e 66 GHz.	15-gen-03
IEEE std 802.16a-2003	emendamento allo standard 802.16 per le frequenze tra 2 e 11 GHz, senza LoS.	01-apr-03
IEEE std 802.16-2004 (anche detto 802.16d)	attualizzazione e revisione degli standard di base 802.16, 802.16a e 802.16c.	01-ott-04
IEEE 802.16e	aggiunge allo standard componenti per la mobilità, fino a 122 Km/h, per cui è detto anche WiMAX mobile. Usa tecnologia OFDMA ²² e permette l'handover e quindi anche l'uso all'interno di veicoli in movimento.	07-dic-05
IEEE 802.16f	specifica la MIB (Management Information Base), per i livelli MAC (Media Access Control) e PHY (fisico)	22-gen-06

La versione "mobile" 802.16e

A ottobre 2005 è arrivato anche lo standard pensato per l'utilizzo di WiMAX in ambito mobile. Le caratteristiche principali del documento approvato dall'IEEE sono le seguenti:

- Frequenze radio utilizzate che coprono le bande dei 2,3 GHz, 2,5 GHz, 3,3 GHz, 3,5 GHz e 5,8 GHz.
- Modalità di modulazione con accesso multiplo in OFDM, che suddivide lo spazio di frequenze disponibili in diversi gruppi di portanti con sottocanali e riduce le interferenze per i terminali con antenna omnidirezionale (tipicamente i terminali mobili).
- Scalabilità dell'occupazione dei canali sulla base della disponibilità di banda.
- Supporto di antenne adattative, ossia con la focalizzazione in ricezione e trasmissione effettuate "formando" il fascio radio (modificando la direzionalità dell'antenna) con mezzi non meccanici ma elettronici.
- Miglioramento nella gestione dell'handover, ossia il passaggio di un terminale da una stazione base a un'altra senza perdita di connessione.
- Gestione del roaming, ossia la possibilità per un abbonato di un gestore di connettersi tramite un altro gestore.

²² Orthogonal Frequency Division Multiple Access

Tutto ciò dovrebbe consentire a un terminale su veicolo in movimento di rimanere collegato (trasferendo dati) fino a una velocità di circa 120 km/h, limite dettato dalle caratteristiche del protocollo di handover in base al rapporto di QoS (Quality of Service). Superando questa velocità si incorre inevitabilmente nella disconnessione del proprio terminale in quanto la potenza del segnale ed il rapporto segnale/rumore non soddisfa le necessarie caratteristiche qualitative di trasporto. Questo limite potrebbe risultare scomodo per coloro che usano un terminale (notebook, PDA, ecc.) in movimento a grande velocità, che non potranno sfruttare il WiMAX a meno che non vengano ulteriormente modificati i parametri dello standard IEEE 802.16 come tecnica di modulazione, larghezza di banda e altre caratteristiche del mezzo trasmissivo.

GSM

Il Global System for Mobile Communications (GSM) è attualmente lo standard di telefonia mobile più diffuso del mondo. Più di 2 miliardi di persone in 200 paesi usano telefoni cellulari GSM. La diffusione universale dello standard GSM ha fatto sì che la maggior parte degli operatori internazionali di telefonia mobile stipulassero fra di loro accordi per l'effettuazione del cosiddetto roaming (*commutazione automatica fra diverse reti*). La tecnologia alla base del GSM è significativamente diversa da quelle che la hanno preceduta, soprattutto per il fatto che sia il canale di identificazione che quello di conversazione sono digitali. Per questo motivo il nuovo standard è stato lanciato sul mercato come sistema di telefonia mobile di *seconda generazione* o, più sinteticamente 2G.

Il maggior punto di forza del sistema GSM è stata la possibilità, da parte degli utenti, di accedere a tutta una serie di nuovi servizi a costi molto contenuti. Ad esempio lo scambio di messaggi testuali (SMS) è stato sviluppato per la prima volta in assoluto in ambito GSM. Uno dei principali vantaggi per gli operatori è stato, invece, la possibilità di acquistare infrastrutture ed attrezzature a costi resi bassi dalla concorrenza fra i produttori. Per contro, una delle limitazioni più serie è derivata dal fatto che le reti GSM impiegano la tecnologia TDMA²³, considerata meno avanzata ed efficiente rispetto alla concorrente tecnologia CDMA²⁴.

L'origine dell'acronimo GSM deriva dal nome del gruppo francese che ne ha iniziato lo sviluppo (*Groupe Spécial Mobile*). In seguito si decise di mantenere la stessa sigla, cambiandone però il significato. In origine il gruppo francese fu sponsorizzato dal CEPT. Le specifiche di base del GSM furono definite nel 1987. Nel 1989 l'ETSI assunse il controllo del progetto, e ne pubblicò le specifiche complete in un volume di 6.000 pagine. La prima rete commerciale funzionante è stata la finlandese Radiolinja.

²³ La Time Division Multiple Access (*accesso multiplo a ripartizione nel tempo*, nota anche con l'acronimo TDMA) è una tecnica di moltiplicazione numerica in cui la condivisione del canale è realizzata mediante ripartizione del tempo di accesso allo stesso da parte degli utenti.

²⁴ La **Code Division Multiple Access** (*accesso multiplo a divisione di codice*, nota anche con l'acronimo CDMA) è una tecnica di accesso, da parte di più sorgenti di informazione, allo stesso canale di trasmissione, tramite l'impiego della stessa banda di frequenze, che consiste nell'associare all'informazione trasmessa da ciascuna di queste sorgenti, un codice che la identifichi univocamente rispetto a tutte le altre e che permetta ad un ricevitore, di estrarre selettivamente l'informazione associata a ciascuna di esse.

È una tecnologia alla base del funzionamento dei telefoni cellulari di terza generazione ("3G"), che funzionano secondo lo standard UMTS.

Servizi possibili con la rete GSM

Il servizio principale della rete GSM è chiaramente la comunicazione voce. Con il tempo però sono stati implementati altri servizi importanti quali gli SMS e la comunicazione dati. Attualmente con le tecnologie GPRS/EDGE è possibile effettuare traffico a commutazione di pacchetto ed utilizzare quindi un terminale GSM-GPRS/EDGE come modem per navigare sulla rete internet, scambiare file e immagini.

A partire dal 2006 la rete GSM permette di utilizzare il protocollo Dual Transfer Mode (DTM): un altro aspetto innovativo della rete GSM che la rende sempre più vicina a quella UMTS. Con il DTM un cellulare può contemporaneamente chiamare e trasmettere dati pacchetto. Questa nuova tecnologia rende tra l'altro possibile effettuare la videochiamata su rete GSM permettendo agli operatori telefonici di fornire servizi di terza generazione senza dover necessariamente migrare in toto sulla rete UMTS.

Reti GSM

Le reti GSM lavorano in diversi *range* di frequenza, e sono composte da un insieme di celle radio di varie dimensioni. Essenzialmente esistono 4 tipi di cella: macro, micro, pico e celle *a ombrello*. La copertura radio ottenibile con ciascun tipo di cella varia in funzione della situazione ambientale circostante. Nelle macro-celle l'antenna della stazione radio base è installata su un palo, o su una struttura, posti sul tetto di un edificio, mentre nelle micro-celle l'antenna è installata ad un livello più basso, situazione tipica delle aree urbane ad alta densità abitativa. Le pico-celle hanno dimensioni limitate, dell'ordine di poche dozzine di metri, e sono di solito usate in ambienti chiusi, mentre le celle *ad ombrello* sono usate per assicurare la copertura di zone lasciate scoperte da celle più piccole, o nei *gap* fra una cella e l'altra. Le antenne di queste celle sono di solito installate sulla sommità degli edifici più alti, o su altre strutture molto elevate.

Le dimensioni delle celle variano, in funzione dell'altezza dell'antenna, del guadagno dell'antenna stessa e delle condizioni di propagazione delle onde radio, da un minimo di circa 200 metri ad un massimo di parecchie decine di chilometri. La massima distanza fra due stazioni radio-base è praticamente di 35 km, sebbene le specifiche del sistema GSM prevedano distanze anche più che doppie. Il limite all'aumento della distanza è dato dalla difficoltà di centrare il cosiddetto *timeslot overlap* (dove il *timeslot* è il tempo allocato per ciascuna chiamata) quando il terminale si trova a grande distanza dalla stazione radio-base.

Il GSM, come sopra accennato, supporta anche le chiamate in ambienti chiusi. La copertura in ambienti interni può essere realizzata mediante piccoli ripetitori che inviano il segnale dall'antenna esterna ad un'antenna interna separata. Quando tutta la capacità, in termini di connessioni, deve essere concentrata in un unico ambiente al coperto, come ad esempio in centri commerciali, aeroporti, ecc., si adotta solitamente la soluzione di un'antenna ricevente installata direttamente all'interno dell'edificio. In aree urbane densamente popolate la copertura radio all'interno degli edifici è assicurata dalla penetrazione del segnale radio, senza la necessità di installare ricevitori interni.

Le frequenze usate dalla rete GSM sono 850, 900, 1800, 1900 MHz e variano a seconda degli stati in cui la rete stessa è installata. Tipicamente nelle nazioni europee si utilizzano le frequenze 900/1800 MHz, mentre negli Stati Uniti le frequenze 850/1900 MHz. La molteplicità delle portanti usabili e l'evoluzione dei sistemi di trasmissione hanno fatto in modo che le celle possano presentare configurazioni multifrequenza (dual band).

Struttura della rete

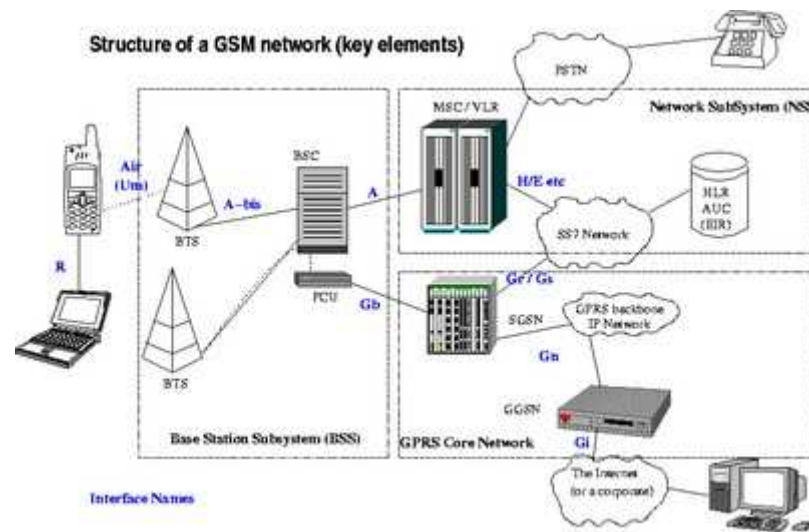


Fig. 22 - Elementi chiave della struttura di rete GSM

La struttura della rete che supporta il sistema GSM è vasta e complicata, perché deve essere in grado di fornire agli utenti tutta una serie di servizi e funzionalità. I componenti essenziali sono:

1. *Mobile Stations*: sono i terminali mobili a cui è destinato ogni servizio della rete.
2. *Access Network*: la rete d'accesso è di fatto il cuore dell'infrastruttura della rete cellulare ed implementa la comunicazione tra il terminale mobile e la rete telefonica fissa. In particolare comprende la BTS (*Base Transceiver Station*) che è l'interfaccia radio con i terminali mobili ed il BSC (*Base Station Controller*) che rappresenta il "cervello" della rete GSM, governando tutti gli aspetti del protocollo GSM e gestendo la comunicazione tra interfaccia radio e rete fissa.
3. *Core Network*: è l'interfaccia della rete fissa verso la rete cellulare. I due elementi fondamentali sono l'MSC (interfaccia per gli aspetti legati alla commutazione di circuito - chiamate voce) e l'SGSN (l'interfaccia per gli aspetti legati alla commutazione di pacchetto - chiamate dati)

GPRS

Il General Packet Radio Service (GPRS) è una delle tecnologie di telefonia mobile. Viene convenzionalmente definita di generazione 2.5, vale a dire una via di mezzo fra la seconda e la terza generazione.

È stato progettato per realizzare il trasferimento di dati a media velocità, usando i canali TDMA della rete GSM. Le specifiche di funzionamento del GPRS sono strettamente integrate a quelle del GSM. In un primo momento la definizione di tali specifiche era demandata all'ETSI, mentre attualmente è affidata al Comitato 3GPP.

Normalmente il costo delle comunicazioni GPRS viene calcolato in base ai kilobyte ritrasmessi, mentre nelle reti commutate il costo è in funzione del tempo di connessione, questo perché, in quest'ultimo tipo di rete, l'intera larghezza di banda disponibile è occupata anche quando nessun dato è in corso di trasferimento.

Il GPRS originariamente avrebbe dovuto supportare (almeno in teoria) i protocolli di rete IP, PPP²⁵ e X.25²⁶; quest'ultimo protocollo è stato usato soprattutto per applicazioni come terminali di pagamento *wireless*, ma attualmente non è più una funzionalità richiesta all'interno dello standard GPRS. Il protocollo X.25 può ancora essere supportato tramite PPP, o anche IP, ma ciò richiede la presenza di un router, o, in alternativa, di un software dedicato da installare a bordo del terminale.

La pacchettizzazione dei dati è realizzata dal GPRS occupando, per la trasmissione, le frequenze radio di una cella radio. Il massimo limite teorico per la velocità è di 171,2 kbit/s (considerando di utilizzare tutti gli 8 Time Slot con Code Scheme 4, ogni time slot ha un rate massimo di 21,4 kbps), ma un valore più realistico si attesta intorno a 30-70 kbit/s.

Un'evoluzione del modo in cui il GPRS utilizza le frequenze, denominato tecnologia EDGE²⁷, consente di raggiungere velocità maggiori, fra 20 e 200 kbit/s. La velocità dipende dal modello di cellulare/terminale usato (classe del cellulare) e dal numero di utenti collegati per cella fra cui è frazionata la banda che, a sua volta, è funzione della densità abitativa del luogo e della fascia oraria (statisticamente un picco di collegamenti si registra tra le 19.00 e le 21.00), dalla distanza fra il terminale e l'antenna più vicina.

Per ottenere le massime velocità di trasmissione teoriche è necessario utilizzare più di un *time slot* (letteralmente *fessura temporale*) contemporaneamente all'interno del cosiddetto *time frame* TDMA (letteralmente *cornice temporale*), tenendo però presente che a maggiori velocità corrispondono minori possibilità di correggere automaticamente gli errori di trasmissione. In linea di massima, la velocità decresce esponenzialmente all'aumentare della distanza dalla stazione radio base. Ciò non costituisce una limitazione in aree densamente popolate, coperte da una fitta maglia di celle radio, ma può diventare un problema serio in aree scarsamente abitate, come, ad esempio, le zone rurali.

Applicazioni del GPRS nel mondo reale

Alcuni gestori telefonici hanno adottato per il GPRS una politica di tariffe più convenienti rispetto ai precedenti sistemi di trasferimento dati attraverso reti GSM, noti con le sigle CSD (*Circuit Switched Data*) e HSCSD (*High Speed Circuit Switched Data*).

La massima velocità di una connessione GPRS è circa uguale a quella ottenibile tramite modem collegato alla normale rete telefonica analogica, e cioè fra 4 e 5 kB/s, a seconda del modem impiegato. Il tempo di latenza è piuttosto alto: per averne un'idea di grandezza si consideri che un'operazione di ping (acronimo di *Packet INternet Groper*) dura di solito 600-700 ms (a volte anche 1 s). L'operazione di *ping* consiste nello stabilire se un indirizzo IP è accessibile o meno, e viene realizzata inviando pacchetti di dati (ICMP) all'indirizzo specificato e aspettando la ricezione della risposta (invio di un'ICMP *echo request* e attesa di un'ICMP *echo reply*).

²⁵ **Point-to-Point Protocol**, o **PPP**, è un protocollo di rete di Livello datalink, comunemente usato nello stabilire connessioni dirette tra due nodi.

²⁶ Protocollo di rete a commutazione di pacchetto. La **commutazione di pacchetto** è una tecnica di trasmissione nella quale è la rete stessa a ridirigere il singolo pacchetto di dati verso la corretta destinazione in base all'indirizzo contenuto nel pacchetto stesso

²⁷ L'EDGE (acronimo di Enhanced Data rates for GSM Evolution) o EGPRS (*Enhanced GPRS*) è un'evoluzione dello standard GPRS per il trasferimento dati sulla rete cellulare GSM che consente maggiori velocità di trasferimento dei dati. Con l'EDGE la velocità di trasmissione dati passa dai 40 Kilobit/sec del GPRS a 200 Kbit/sec, 5 volte superiore

Normalmente viene data una priorità più bassa al GPRS rispetto al canale audio, e per questo motivo, la qualità effettiva della connessione varia grandemente. In effetti la maggior parte delle limitazioni non sono di natura tecnica, e quindi le connessioni GPRS potrebbero, in teoria, essere molto più performanti di quanto non siano nelle attuali implementazioni nel mondo reale.

Connessione WAP e connessione web

Solitamente utilizzando il solo cellulare GPRS/EDGE/UMTS non collegato ad altro terminale (PC, portatile o palmare) sono accessibili solo siti WAP ovvero siti appositamente progettati per i cellulari: si tratta di siti semplificati, non contenenti applet java, pagine dinamiche o applicazioni in HTML4.

Connettendo il cellulare ad un altro terminale, tramite Bluetooth o cavo USB 2.0 (che non rallenta la connessione nel tratto finale dal momento che supporta velocità di trasmissione fino a 480 Mbit/s), si stabilisce una normale connessione in cui sono disponibili i medesimi servizi e protocolli accessibili con il modem analogico o via ADSL. Il cellulare funziona da modem, e tutti i bit vengono inviati al terminale per la decodifica e la visualizzazione

DVB-H

Il DVB-H, acronimo di Digital Video Broadcasting - Handheld, è lo standard del consorzio europeo DVB per una modalità di radiodiffusione terrestre studiata per trasmettere programmi TV, radio e contenuti multimediali ai dispositivi *handheld*, come i più comuni smartphone i palmari e i telefoni cellulari evoluti. Questo standard è ufficialmente supportato dall'Unione Europea.

Si tratta di uno standard derivato dal DVB-T e funziona combinando gli standard del video digitale con l'Internet Protocol in modo da suddividere i contenuti in pacchetti di dati da trasferire sul cellulare e leggibili da parte dell'utente. In questo modo il peso dei dati si riduce ed è possibile trasferirli conservando un'alta qualità video. L'uso del protocollo IP, inoltre, rende possibile la trasmissione simultanea sullo stesso canale di pacchetti video (stream DVB) e pacchetti dati sfruttabili da applicazioni presenti sul terminale ricevente (IP Datacast). Queste applicazioni includono:

- feed di news,
- informazioni finanziarie,
- contenuto multimediale,
- trasferimento files,
- accesso ad Internet ad alta velocità.

Grazie allo standard DVB-H sarà possibile guardare la televisione sul cellulare in maniera interattiva e dunque rendere concreto il concetto di portabilità.

Il sistema è equivalente a quello del DVB-T. A differenza di quest'ultimo, pensato per condizioni di propagazione analoghe a quelle della TV analogica e ricevitore statico, lo standard DVB-H aggiunge un layer al di sopra del livello di trasporto (assimilabile ad un MAC layer del protocollo Ethernet) per meglio adattarsi alle particolari condizioni d'uso di un ricevitore portatile. Esistono infatti profonde analogie tra il modello di propagazione DVB-H e quello delle reti cellulari che rendono necessaria l'introduzione di tecniche di

correzione di errore per compensare l'effetto degli echi multipli e dell'effetto Doppler²⁸ tipici della ricezione ad altezza d'uomo e in movimento. Inoltre esiste la necessità di minimizzare il consumo dovuto alle sezioni radio del ricevitore allo scopo di prolungare al massimo la durata della batteria. Per migliorare quindi le prestazioni e diminuire i consumi di potenza, a livello di MPEG-2 Transport Stream, si utilizzano le tecniche di *time slicing* e di *Forward Error Correction for Multiprotocol Encapsulated Data* (MPE-FEC).

- MPE-FEC: la modalità dati dei sistemi DVB viene utilizzata per veicolare datagrammi IP al ricevitore. Questi possono trasportare video e audio con la modalità di compressione preferita. Inoltre, per fornire maggiore protezione in caso di trasmissione difficoltosa, si raggruppano i datagrammi in *trame* di lunghezza variabile, ed a queste si applica un codice a blocchi RS(255,191) per la correzione di eventuali errori. Inoltre, un controllo di parità aggiuntivo CRC-32 permette di capire se i dati ricevuti sono affetti o meno da errori residui.
- Time slicing: poiché le bit rate necessarie per trasmettere segnali audio/video sullo schermo di un terminale portatile sono molto minori rispetto a quelle trasportate dal segnale DVB-T nella sua interezza, i dati di ogni programma vengono trasmessi in gruppi ben isolati (*burst*). In questo modo, il terminale in ricezione si attiverà solo in corrispondenza della presenza dei burst, portando ad un notevole risparmio nei consumi d'energia: valori tipici possono andare dall'80% al 90%.

In ricezione, il terminale contiene tutti i circuiti necessari ad effettuare la demodulazione e la decodifica del segnale DVB-H. Il demodulatore DVB-H è in pratica un demodulatore DVB-T con la possibilità aggiuntiva di demodulare segnali trasmessi in modo 4K (dimensione della FFT pari a 4096): infatti lo standard DVB-T prevede che i segnali vengano elaborati con una FFT di dimensione pari a 2048 (modo 2K) o 8192 (modo 8K). In più, il time-slicing prevede un controllo sull'attivazione completa del ricevitore.

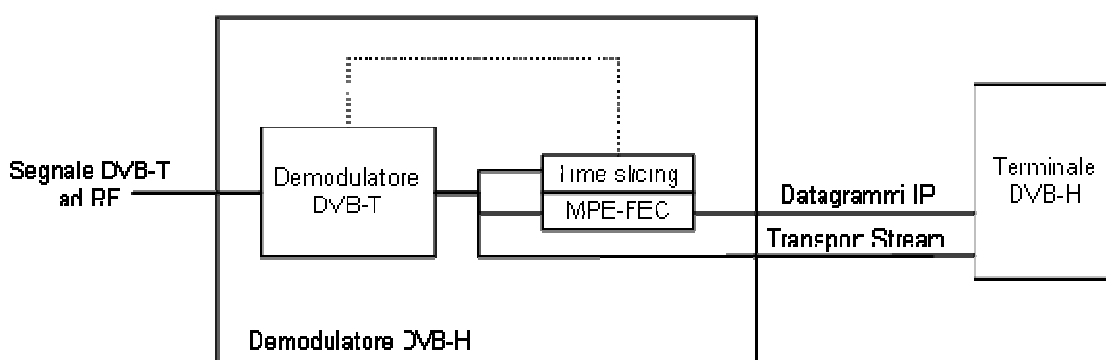


Fig. 23 - Demodulatore DVB-H

²⁸ L'effetto Doppler è un cambiamento apparente della frequenza o della lunghezza d'onda di un'onda percepita da un osservatore che si trova in movimento rispetto alla sorgente delle onde.

UMTS

Universal Mobile Telecommunications System (UMTS) è la tecnologia di telefonia mobile di terza generazione (3G), successore del GSM. Tale tecnologia impiega lo standard base W-CDMA²⁹ come interfaccia di trasmissione. L'UMTS è a volte lanciato sul mercato con la sigla 3GSM per mettere in evidenza la combinazione fra la tecnologia 3G e lo standard GSM di cui dovrebbe in futuro prendere il posto.

Caratteristiche tecniche

Il sistema UMTS supporta un transfer rate (letteralmente: *tasso di trasferimento*) massimo di 1920 kbit/s. Le applicazioni tipiche attualmente implementate, usate ad esempio dalla reti UMTS in Italia, sono tre: voce, videoconferenza e trasmissione dati a pacchetto. Ad ognuno di questi tre servizi è assegnato uno specifico transfer rate, per la voce 12,2 Kbit/s, 64 Kbit/s per la videoconferenza e 384 Kbit/s per trasmissioni di tipo dati (scarico suonerie, accesso al portale,...). UMTS è quindi in grado, potenzialmente, di consentire per la prima volta l'accesso, a costi contenuti, di dispositivi mobili al World Wide Web di Internet.

Dal 2004 sono presenti anche in Italia l'UMTS 2 e l'UMTS 2+ (si legge "*2 plus*"), due estensioni del protocollo UMTS, che funzionano sulle attuali reti UMTS e raggiungono velocità rispettivamente di 1.8 e 3 megabit/secondo.

In un prossimo futuro le attuali reti UMTS potranno essere potenziate mediante il sistema di accesso denominato HSDPA (*High Speed Downlink Packet Access*), con una velocità massima teorica di scaricamento dati di 10 Mbit/s. Gli operatori interessati al lancio sul mercato di questo sistema hanno preannunciato la possibilità di fornire servizi di videoconferenza tramite dispositivi mobili. Rimane tuttavia ancora da dimostrare l'esistenza, ad oggi, di un mercato di massa per questo tipo di servizi. Un'altra applicazione possibile è lo scarico di brani musicali.

In parole semplici, l'UMTS è una combinazione delle seguenti interfacce trasmissive (dove per *interfaccia trasmissiva* si intende il protocollo che definisce lo scambio di dati fra i dispositivi mobili e le stazioni radio base):

- W-CDMA
- MAP del GSM (acronimo di Mobile Application Part), protocollo che fornisce funzionalità varie ai dispositivi mobili, come ad esempio l'instradamento delle chiamate *da* e *per* i gestori.
- La famiglia di codec (*codec = algoritmo software di codifica-decodifica*) del GSM come ad esempio i protocolli MR e EFR, che definiscono il modo in cui il segnale audio è digitalizzato, compresso e codificato.

Da un punto di vista tecnico il W-CDMA è solo l'interfaccia verso la rete di accesso, mentre l'UMTS è l'insieme completo dello stack di protocolli di comunicazione progettati per i sistemi 3G, successori diretti del GSM. Comunque l'acronimo W-CDMA è spesso usato come termine generico per comprendere tutta la famiglia di standard 3G che utilizzano l'interfaccia trasmissiva W-CDMA, compresi i sistemi UMTS, FOMA e J-Phone.

²⁹ **W-CDMA (Wideband Code Division Multiple Access)** è una tecnologia per reti cellulari di terza generazione (3G). Il W-CDMA è un'interfaccia wireless a banda larga e ampio spettro per la comunicazione mobile che utilizza il CDMA.

Come altre implementazioni del W-CDMA, l'UMTS utilizza una coppia di canali a 5 MHz di larghezza di banda, uno nel range 1900 MHz per la trasmissione e uno nel range 2100 MHz per la ricezione. L'UMTS viene spesso criticato per la grande larghezza di banda di cui necessita.

Per gli operatori GSM esistenti la migrazione all'UMTS è relativamente semplice ma anche costosa: la maggior parte delle infrastrutture esistenti può essere riutilizzata, ma la spesa per ottenere le concessioni per le nuove frequenze e per gestirle con le esistenti stazioni radio base, può ancora richiedere investimenti elevatissimi.

Una delle principali differenze rispetto al GSM è la configurazione dell'interfaccia trasmissiva (il cosiddetto GRAN, acronimo di *Generic Radio Access Network*). Sono possibili connessioni con le *backbone* di varie altre reti, come Internet, ISDN (acronimo di *Integrated Services Digital Network*), GSM o altre reti UMTS. L'interfaccia GRAN utilizza i 3 layer di più basso livello del cosiddetto modello OSI. Il layer di rete del protocollo OSI (precisamente il layer 3) rappresenta il cosiddetto protocollo RRM (acronimo di *Radio Resource Management*). L'insieme di questi protocolli ha la funzione di gestire i canali portanti fra i dispositivi mobili e le reti fisse, compresa la gestione delle commutazioni (*handover*) fra reti diverse.

Terminali UMTS

A livello di interfaccia trasmissiva, l'UMTS è di per sé compatibile con il GSM. Poiché tutti i telefoni cellulari UMTS immessi sul mercato fino ad oggi sono del tipo dual-mode UMTS/GSM, essi possono inviare e ricevere chiamate attraverso l'esistente rete GSM. Quando un utente UMTS si sposta verso un'area non coperta dalla rete UMTS, un terminale UMTS commuta automaticamente al GSM (con eventuale addebito delle tariffe per il *roaming*). Se l'utente esce dalla zona di copertura UMTS durante una chiamata, la chiamata stessa sarà presa in carico dalla rete GSM in modo *trasparente* (cioè senza che l'utente se ne accorga). Al contrario i terminali GSM non possono essere usati all'interno di reti UMTS.

Considerando che il 70% degli italiani vive in centri con meno di 10000 abitanti, la maggioranza del Paese rimane esclusa dalla copertura UMTS e, senza esserne informata, acquista un cellulare dal quale non può ricevere alcun servizio.

La Japan Telecom (ora divenuta Softbank) ha realizzato una rete 3G che utilizza la tecnologia W-CDMA ed è compatibile con l'UMTS, divenendo così il primo esempio di rete globale totalmente wireless. L'attuale standard GSM, per contro, è disponibile in tutto il mondo, tranne che in Giappone e in Corea.

Problemi e dibattiti aperti.

Alcuni dei principali problemi che gli operatori devono tuttora risolvere sono:

- la crescente opposizione degli abitanti per quanto riguarda l'installazione di nuove antenne e verificare la possibilità che quest'ultime siano dannose per la salute;
- ridurre il peso eccessivo dei terminali portatili, aumentare la capacità delle batterie;
- i terminali dotati di tutte le funzionalità, compresi i servizi di Video on Demand, necessitano di una stazione radio base ogni 100 metri, cosa realizzabile a costi contenuti in aree urbane ad alta concentrazione abitativa, ma attualmente impossibile in zone urbane a bassa densità di popolazione e in zone rurali. Pur volendo limitare il carico della cella al 50% e la velocità a 128 kbit/s il raggio massimo di una cella non va oltre i 400 ÷ 500 m;
- fronteggiare la concorrenza della tecnologia Wi-Fi nel settore broadband (*accesso a banda larga*);

- ridurre i costi d'impianto che, attualmente, a parità di copertura base fornita per il servizio voce, va da 4 a 10 volte il costo della rete GSM;
- verificare l'effettiva possibilità di ricavare ulteriori profitti dai nuovi servizi a *valore aggiunto*. Alcuni analisti prevedono che l'aumento della concorrenza, la maggiore diffusione della rete e la possibilità di accedere a Internet dai terminali mobili finiranno con l'assorbire anche i margini di profitto attualmente esistenti.

Automatic Vehicle Location

Con l'espressione Automatic vehicle location (o AVL) si indica una tecnologia che consente di localizzare automaticamente i veicoli in diversi modi e per le necessità più svariate.

Una delle principali applicazioni della tecnologia AVL è quella legata all'uso dei sistemi AVM, cioè di Automatic vehicle monitoring, in uso su numerose realtà del trasporto pubblico locale. In genere, la localizzazione è effettuata con il GPS, ed il meccanismo di trasmissione è un satellite, una radio o una connessione via cellulare. Un'altra possibilità per determinare la posizione, ad esempio dove il GPS non è utilizzabile è l'uso di lettori RFID.

I dati sul tracciamento vengono poi trasmessi utilizzando un qualsiasi sistema di telemetria. Il GSM e l'EVDO sono le tecnologie più comunemente utilizzate per la telemetria, grazie al basso quantitativo di dati necessario per l'AVL, al relativo basso costo ed alla facile accessibilità di queste reti pubbliche. Il basso impiego di banda consente anche alla tecnologia satellitare di ricevere dati di telemetria a costi moderatamente superiori, ma attraverso una copertura globale dell'area in luoghi remoti non adeguatamente coperti da segnale radio terrestre.

L'AVL è un potente strumento per gestire flotte di veicoli, siano essi di servizio, di emergenza o di squadre di costruzione, oltre che legati al trasporto pubblico (bus e treni).

Un tipico sistema è basato sull'esame del territorio e utilizzato per tenere traccia di flotte di veicoli. Il principale scopo del tracciamento è fornire un servizio ben calibrato o gestire grandi quantità di veicoli. Ad esempio, per gestire una flotta di ambulanze un sistema AVL consente di individuare il mezzo di volta in volta più vicino al punto nel quale è richiesto l'intervento.

BIBLIOGRAFIA

- Business Data Networks and Telecommunications, 6th edition, Panco, Ed. Prentice Hall, 2006.
- Business Driven Technology, Haag, Baltzan, and Phillips, McGraw-Hill, 2006
- Commissione Europea, “Intelligent Transport Systems – Intelligence at the Service of Transport Networks”, Novembre 2003.
- Commissione Europea, Libro Bianco “La politica europea dei trasporti fino al 2010: il momento delle scelte”
- TTS Italia, “Risultati sull’indagine su Sistemi ITS e mobilità nelle Amministrazioni Locali e nelle Aziende di trasporto pubblico: stato attuale e prospettive di sviluppo”
- Ministero delle Infrastrutture e dei Trasporti, “ARTIST – Architettura Telematica Italiana per il Sistema dei Trasporti”
- Frost & Sullivan, “European Intelligent Transportation Systems Markets”, 2002
- www.ttsitalia.it
- <http://www.howstuffwork.com/>
- <http://www.its-artist.rupa.it/home.htm>
- <http://www.wbsolutions.it/tecnologia/tech.htm#>
- <http://cordis.europa.eu>
- <http://www.euromobilityforum.com/>

PIATTAFORMA DI COMUNICAZIONE MOBILE SHINE

Attività simulatori

Una delle principali aree di ricerca in cui il WiLab si contraddistingue a livello internazionale da diversi anni è lo studio di reti di accesso wireless eterogenee in scenari realistici. L'approccio adottato prevede, anche attraverso tecniche di simulazione, uno studio integrato cross-layer e inter-sistema.

Allo scopo è stata sviluppata una piattaforma di simulazione denominata **SHINE**: "Simulation Platform for Heterogeneous Interworking Networks". Tale piattaforma permette di riprodurre in uno scenario realistico, che va dalle dimensioni ridotte di un appartamento ad una regione territoriale ampia (ad esempio un comune), la presenza di una o più reti di accesso e di verificarne le prestazioni in presenza di utenti che richiedono un servizio.

Si riproducono tutti gli aspetti e i livelli protocollari; in particolare:

- la posizione di ogni utente nel tempo (con relativi spostamenti) sul territorio, tenendo conto anche di percorsi stradali/pedonali;
- la sessione di ciascun utente (inizio chiamata, traffico durante la chiamata, terminazione della chiamata), con il dettaglio di ciascun pacchetto generato a livello applicazione, poi opportunamente frammentato ai livelli inferiori;
- i protocolli di trasporto (UDP e TCP);
- gli effetti di attraversamento della core network;
- l'accesso al canale radio;
- l'integrità dei dati, pacchetto per pacchetto, tenendo conto di tutti gli aspetti di livello fisico, compresa una modellizzazione accurata del canale wireless correlato in tempo e frequenza.

La piattaforma è costituita da un server e uno o più client, come indicato in Fig. 24.

Il server (ULS) si occupa della gestione dei protocolli end-to-end (livello applicazione e di trasporto) e dell'attraversamento della core-network (livello rete dell'infrastruttura cablata), indipendenti dalla tecnologia d'accesso considerata.

I client (LLS, ogni client corrisponde ad una tecnologia di accesso) si occupano dei protocolli link-to-link dei collegamenti wireless delle reti d'accesso, che dipendono dalle specifiche tecnologie simulate.

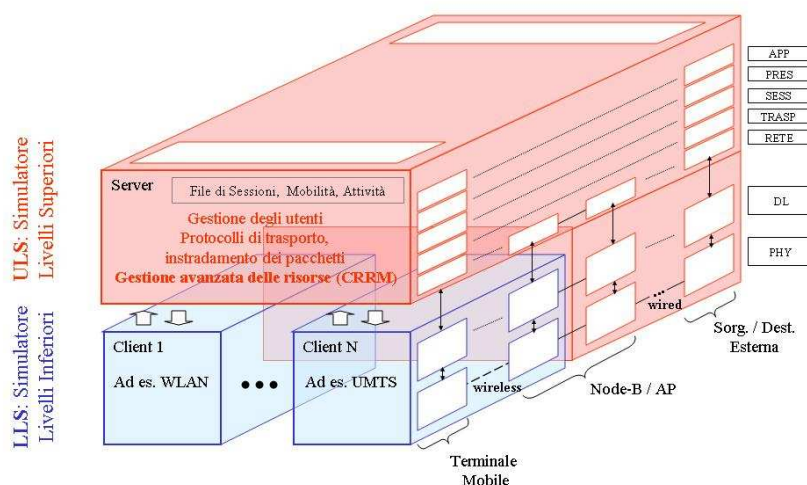


Fig. 24 - schema a blocchi di SHINE

In rosso il server ULS, in blu i client LLS; sull'asse verticale sono indicati i livelli della pila protocollare; sull'asse di profondità alcuni dei nodi che collegano il terminale mobile alla sorgente/destinazione esterna allo scenario

Ad oggi, sono stati sviluppati client per le seguenti tecnologie: WLAN IEEE802.11a/g/e/p, WiMax IEEE802.16e, UMTS WCDMA-FDD, UMTS TD-SCDMA, GSM-GPRS.

A titolo di esempio, si mostrano in

Fig. 25 i risultati di una simulazione effettuata tramite SHINE, in cui si indaga il vantaggio, in termini di utenti voce serviti, ottenibile grazie all'aggiunta di un hot-spot WLAN in un'area con elevato traffico servita da sistema UMTS TD-SCDMA. Sono stati considerati gli standard IEEE 802.11a ed 802.11e, in presenza ed in assenza di un algoritmo di controllo degli accessi (CAC).

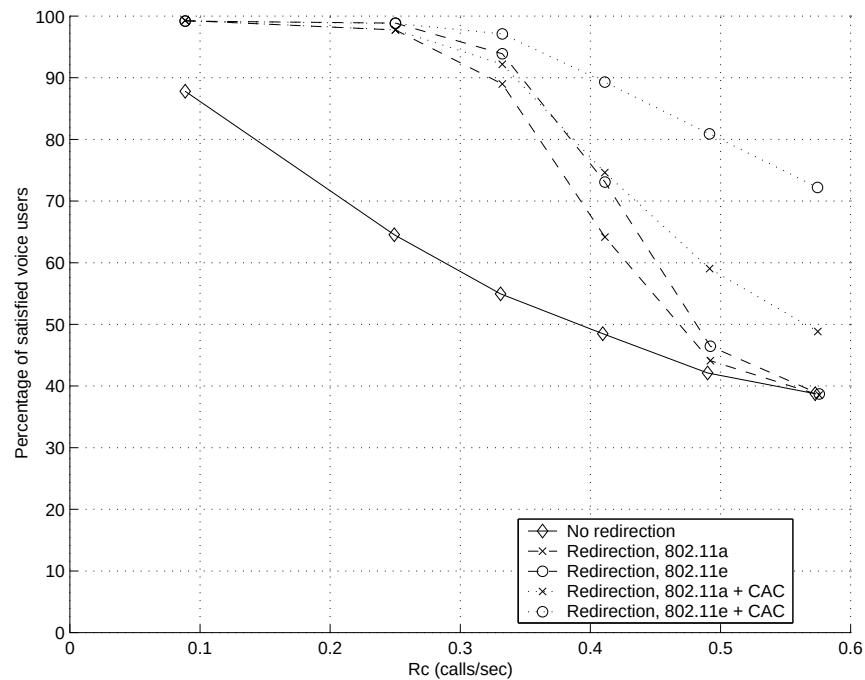


Fig. 25 - risultati di simulazione

La riga continua indica la percentuale di utenti voce serviti con qualità di servizio soddisfacente in sola presenza del sistema UMTS TD-SCDMA. Le curve tratteggiate rappresentano il miglioramento ottenuto tramite l'introduzione di un hot-spot WLAN, senza alcun controllo degli accessi. Le curve rappresentate da punti indicano l'ulteriore miglioramento apportato dall'introduzione di un controllo intelligente dell'accesso al servizio. Le 'x' rappresentano il caso di IEEE802.11a, i 'O' il caso IEEE802.11e.

STUDIO DELL'IMPATTO DEL SISTEMA SULL'INFRASTRUTTURA DI COMUNICAZIONE MOBILE

In questo rapporto sono descritte le caratteristiche fondamentali relative all'accesso nella tratta uplink (dal mobile alla base) degli attuali sistemi di comunicazione mobile allo scopo di calcolare il carico che la rete può supportare se i terminali connessi trasmettono piccole quantità di dati.

Saranno presi in considerazione sia il sistema di seconda generazione (2G) Global System for Mobile Communications (GSM), sia il General Packet Radio Service (GPRS) noto come sistema di generazione 2.5G, sia la terza generazione (3G) con l'Universal Mobile Telecommunications System (UMTS).

Per ciascuno di essi verrà fatta una breve introduzione per poi passare alla descrizione dei parametri che caratterizzano l'accesso alla risorsa radio nella tratta uplink.

Obiettivo finale è quello di chiarire al lettore come avvenga l'accesso nei diversi sistemi e quantificare il numero di utenti che possono contemporaneamente accedere al sistema senza degradarne le prestazioni.

Supponendo, infatti, come esempio, di voler tenere traccia di posizione e altre informazioni per una flotta di veicoli (ad esempio per fornire all'utente finale informazioni in tempo reale relative allo spostamento dei mezzi), verrà valutato per ogni tecnologia presa in esame, il numero massimo di veicoli che possono essere equipaggiati con tale sistema e contemporaneamente attivi.

GSM

In questa sezione vengono introdotte le caratteristiche fondamentali del GSM, nonché i canali fisici e logici che garantiscono le comunicazioni tra mobile (MS) e base (BS). Tra questi verrà posta particolare attenzione sul canale per l'accesso in uplink e valutato il numero massimo di utenti che possono accedere alla risorsa radio in un'unità di tempo.

Breve introduzione al GSM

Il Global System for Mobile Communications (GSM) è lo standard di telefonia mobile più diffuso del mondo. Nato in Europa come standard di seconda generazione garantisce dai primi anni '90 due servizi fondamentali in mobilità: la chiamata vocale e lo Short Message Service (SMS).

L'architettura di rete prevede tre sottosistemi (o tre livelli gerarchici) come mostrato in Fig. 26. Il primo livello è composto dalla stazione base (BS) che individua la cella elementare e garantisce la copertura a tutte le mobile station (MS) in una certa porzione di territorio. Gruppi di stazioni base vengono controllate dal Base Station Controller (BSC), il secondo livello gerarchico, che costituisce la vera e propria intelligenza e si occupa dell'assegnamento delle risorse agli utenti e delle comunicazioni con il Mobile services Switching Centre (MSC), terzo livello gerarchico. Al MSC sono delegate tutte le funzionalità di rete e di commutazione.

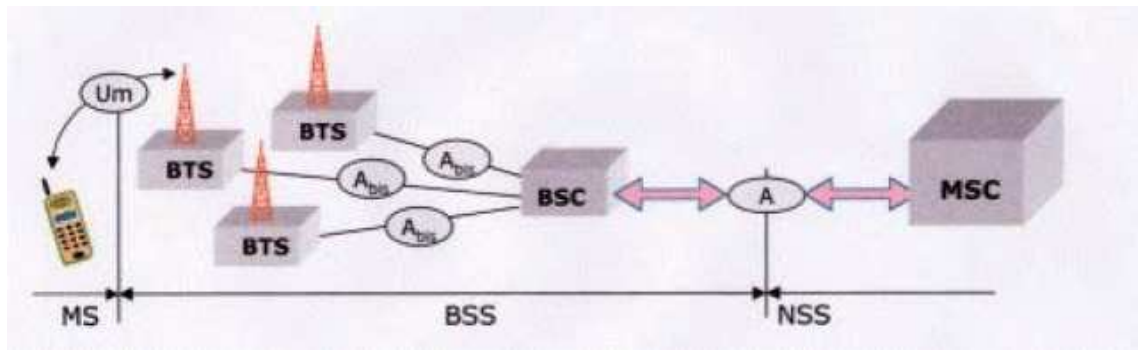


Fig. 26 - Architettura delle rete GSM.

Le principali funzionalità del sistema sono le seguenti:

- il roaming, relativo allo spostamento dell'utente da un'area di localizzazione ad un'altra o alla rete di un altro operatore;
- la location updating per l'aggiornamento della localizzazione
- l'handover che garantisce il collegamento del mobile ad una seconda BS qualora si allontanano troppo con relativa perdita di segnale dalla BS a cui era precedentemente collegato.

Il GSM utilizza la modulazione Gaussian Minimum Shift Keying (GMSK) e la tecnica frequency division duplex (FDD): uplink e downlink sono trasmessi su bande di frequenza disgiunte. Le frequenze del GSM sono allocate in due regioni dello spettro: due bande da 25 MHz ciascuna per uplink e downlink intorno a 900 MHz e 2 bande da 75 MHz intorno a 1800 MHz.

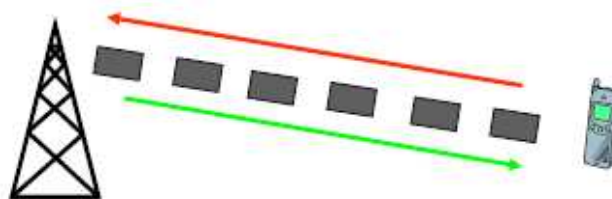


Fig. 27 - Tecnica FDD: trasmissioni simultanee in uplink e downlink in due bande diverse di frequenza

Lo schema di accesso multiplo è una combinazione di Time Division Multiple Access (TDMA) e Frequency Division Multiple Access (FDMA). FDMA: la banda complessiva viene suddivisa in canali da 200 KHz; TDMA: ciascuno di questi canali viene condiviso nel tempo da 8 utenti.

La portante uplink e quella downlink per un utente sono sempre separate da 45 MHz.

Per combattere interferenza da accesso multiplo e fading, ad ogni trama la frequenza portante viene

cambiata con un algoritmo di Frequency Hopping (FH).

Il rate a cui vengono trasmessi i dati è 270.833 kbit/s che equivale ad un rate per utente di $270.833/8=33.854$ kbit/s che diventa 24.7 kbit/s se si eliminano overhead di sequenza pilota e guard interval (2×57 bit ogni 4.615 ms). Il rate effettivo si riduce infine a 22.8 kbit/s a causa della banda concessa al canale di controllo Slow Associated Control CHannel (SACCH) multiplexato con i dati, di cui si dirà più ampiamente nel seguito.

Canali logici nel GSM

L'interfaccia radio del sistema GSM, definita interfaccia Um deve soddisfare tutte le necessità di segnalazione e di trasmissione dati che sono tipiche di una rete cellulare. Il livello fisico si interfaccia con il livello collegamento (data link layer) attraverso un certo numero di canali logici costituiti sia da canali di traffico voce e dati, che da canali di segnalazione e di controllo. In particolare si può distinguere tra:

- canali di diffusione (Broadcast Common Control Channel - BCCH)
- canali di controllo comuni (Common Control Channel - CCCH)
- canali di controllo dedicati (Dedicated Control Channel - DCCH)
- canali riservati al traffico (Traffic Channel - TCH).

Tra questi possiamo individuare l'insieme dei canali logici di traffico (TCH) che portano le informazioni utili degli utenti e i canali di controllo (CCH) per le informazioni di segnalazione di rete e di utente, come mostrato in Fig. 28.

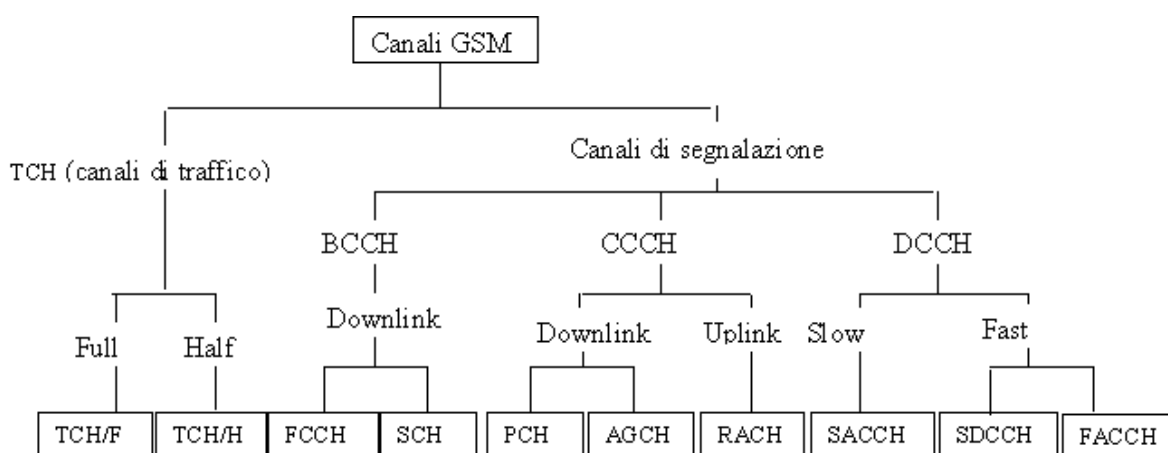


Fig. 28 - Canali logici nel GSM

Ai fini di questo rapporto ci interessa porre l'attenzione sui canali di controllo comuni unidirezionali (PCH, AGCH e RACH). E' infatti il Random Access Channel (RACH) quello utilizzato in uplink dagli utenti per poter richiedere l'accesso alla risorsa radio o poter accedere ad un qualunque servizio (il RACH è anche utilizzato per rispondere ad una chiamata di paging). Il RACH è un canale a contesa di tipo Slotted-Aloha in cui sono possibili collisioni fra le richieste dei diversi utenti. In particolare, il terminale mobile invia un burst in uno degli slot assegnati al RACH senza coordinarsi con altri utenti. Se arriva una risposta della BS sull'AGCH significa che non c'è stata collisione e la richiesta è andata a buon fine, diversamente ripete la trasmissione.

Canali fisici nel GSM

Per trasportare i canali logici si utilizzano i canali fisici. Abbiamo visto che la risorsa radio è suddivisa in frequenza (portanti con banda a disposizione di 200KHz) e nel tempo. Nel tempo, si hanno trame lunghe 4.615ms suddivise in 8 time slot. Quindi, ogni singolo canale fisico è identificato dalla combinazione sottoportante/time slot all'interno di una trama. Risultano così 8 canali fisici per ogni sottoportante.

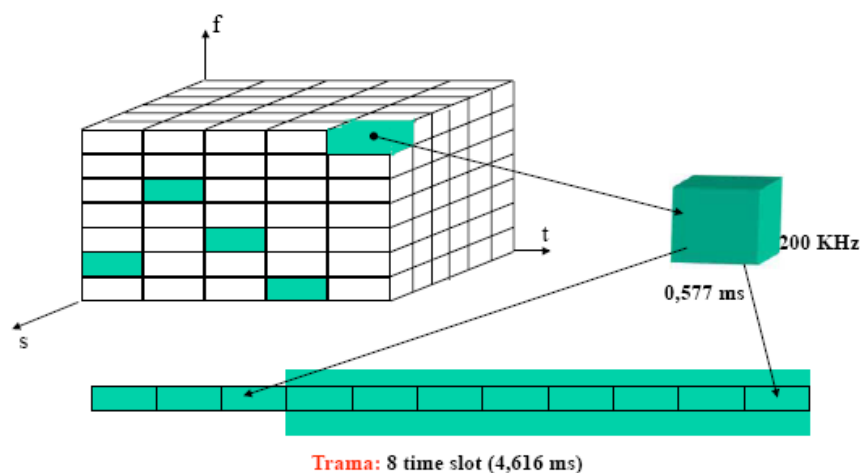


Fig. 29 - Struttura accesso radio nel GSM

Ogni canale fisico può essere di dati o di segnalazione. All'interno del canale fisico si inseriscono poi i canali logici, multiplexati nel tempo.

Per consentire l'individuazione, da parte degli utenti mobili e della radiobase, dei diversi canali logici presenti nel flusso informativo trasmesso, il sistema cellulare GSM prevede la definizione di strutture multitrama; in particolare i canali TCH sono organizzati in una multitrama costituita da 26 trame mentre i canali di controllo sono organizzati in una multitrama costituita da 51 trame. **Multitrama associata ai canali TCH:** delle 26 trame disponibili le prime 12 sono destinate ai canali di traffico TCH, la tredicesima è riservata ai canali SACCH associati ai canali TCH, le successive dodici trame sono nuovamente disponibili per i canali di traffico e, infine, l'ultima trama risulta inutilizzata ed è disponibile per futuri utilizzi (si veda Fig. 30). Il singolo canale fisico di dati potrà trasportare ad esempio una connessione a circuito.

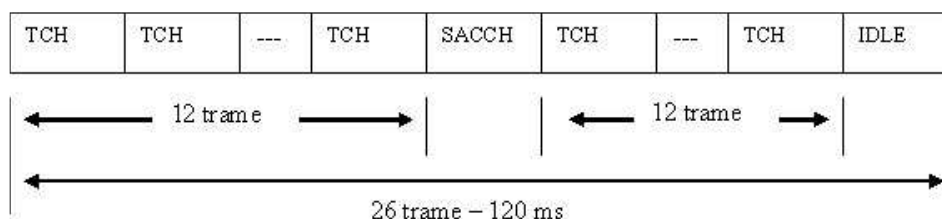


Fig. 30 - Multitrama associata ai canali TCH

Multitrama associata ai canali di controllo: tanto nell'uplink quanto nel downlink risulta definita una multitrama di 51 trame; all'interno di tale struttura trovano collocazione, in posizioni note alla rete ed ai terminali mobili, i canali di controllo. Tra i canali logici ci saranno quello di broadcast delle informazioni di cella, quello di paging, quello di random access in uplink, ecc.

All'interno di uno slot sarà trasmesso un burst. Normalmente si trasmette il normal burst che porta 114 bit di informazione lordi (cioè dopo la codifica, ma già depurati dell'overhead di livello fisico).

Nel GSM, per poter trasmettere un burst in uplink all'interno dei confini dello slot, una stazione deve in qualche modo stimare la sua distanza dalla base tramite il meccanismo di timing advance.

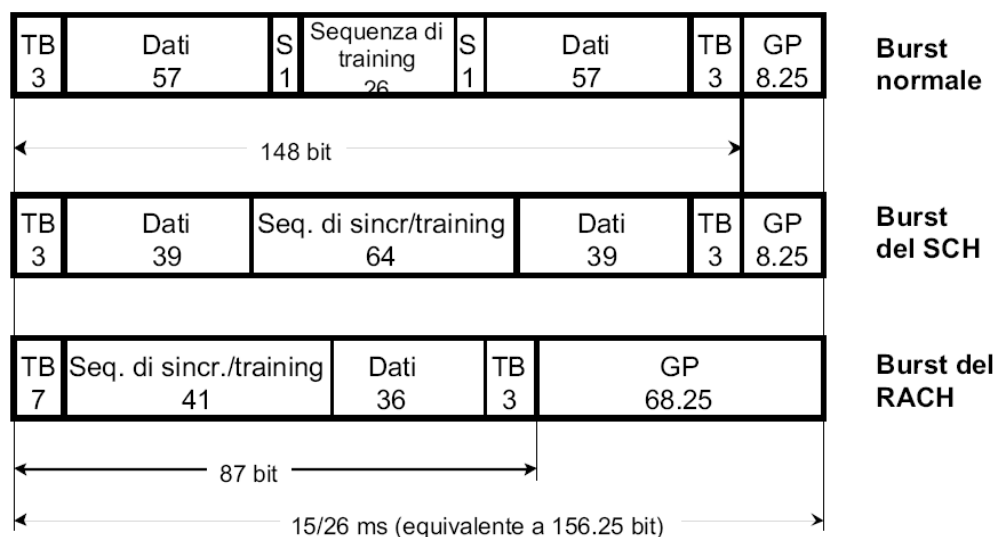


Fig. 31 - Struttura dei burst radio nel GSM

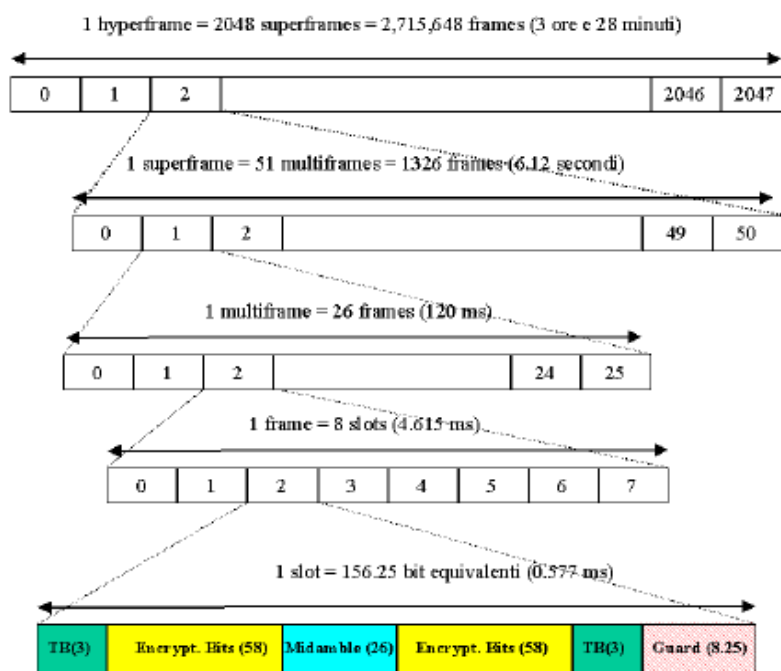


Fig. 32 - Canali fisici nel GSM

L'accesso in uplink nel GSM

Per l'accesso a qualunque servizio in uplink e per rispondere ad una richiesta di paging, si usa il canale RACH. E' un canale a contesa di tipo Slotted-Aloha in cui sono possibili collisioni fra le richieste dei diversi mobili.

Sul RACH l'unico burst utilizzabile è l'*access burst* [1]. L'*access burst* è un pacchetto di livello MAC di piccole dimensioni per permettere un funzionamento senza conoscenza perfetta del timing advance.

L'*access burst*, infatti, porta solo 8 bit di informazione.

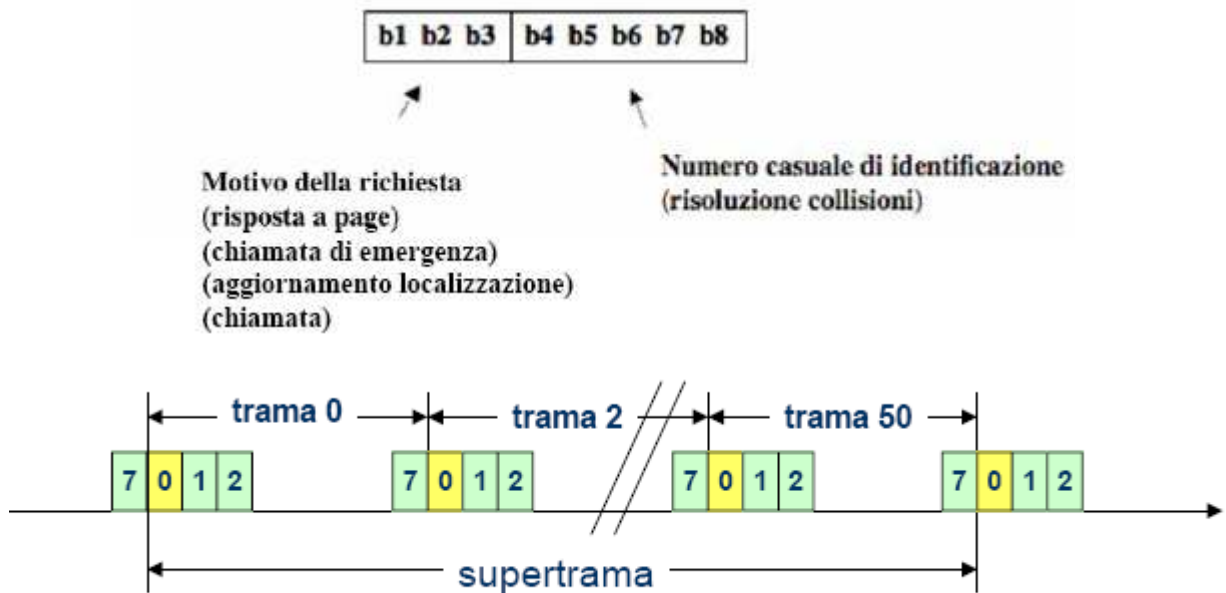


Fig. 33 - Gli 8 bit dell'*access burst*

Il RACH è un canale di segnalazione che occupa uno slot ogni otto su una portante: quindi, si trasportano

$$8 * \left[\frac{1}{8 * 557 \mu s} \right] = 1.733 Kbps.$$

Ci sono quindi,

$$\frac{1}{8 * 557 \mu s} = 216 \text{ slot} / s$$

circa per la richiesta d'accesso per ogni canale RACH.

Anche con un singolo CCCH per cella, ci sono circa 780.000 time-slot RACH disponibili all'ora (sono $216.64 * 3600 = 779.904$).

Nel caso peggiore il CCCH è combinato con SDCCH: in questo caso, in una multitrama di 51, 24 di 51 slot sono usati per SDCCH, quindi, "solo" 413.000 slot RACH circa sono disponibili in un'ora.

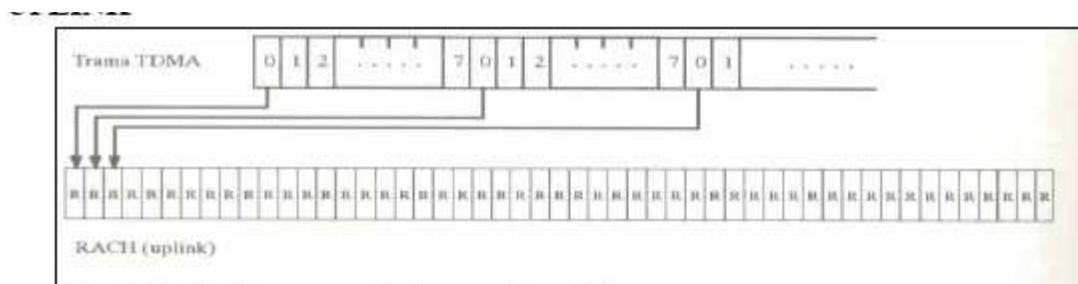


Fig. 34 - Multi-trama di segnalazione in uplink nel GSM

Se teniamo conto del fatto che l'efficienza dello Slotted Aloha non supera il 36% [2], possono essere effettuate fino a circa 77 richieste d'accesso al secondo.

Nel GSM il numero massimo di RACH è fissato a 4 (normalmente se ne prevede 1 solo).

Come si potrà capire in seguito, questi numeri mostrerebbero che il RACH non rappresenta un limite per l'accesso alla risorsa, anche se viene utilizzato in modo condiviso per tutte le richieste di servizio, non solo per gli SMS.

Per quanto riguarda l'invio di SMS, le operazioni da eseguire sono connection-setup, invio, connection-release. Una volta effettuata la richiesta ed ottenuta la risorsa, l'invio vero e proprio avviene su SDCCH (per una descrizione accurata dell'intera procedura il lettore può fare riferimento a [3]).

Per la mappatura dei canali SDCCH vengono utilizzati i canali fisici TACH/8 o TACH/4 che sono costituiti da 8 o 4 slot per la trasmissione dei dati più 4 slot per la trasmissione di segnalazione all'interno di una struttura multitrama di 102 frame di durata 0.47 s.

Così come nei canali di controllo, anche i canali TACH/8 e TACH/4 utilizzano un solo slot (quello numero zero per default) mentre lasciano gli altri liberi per altri scopi (canali di traffico).

I pacchetti vengono costruiti come per il caso ACCH: 456 bit vengono generati da 184 bit informativi. Nel caso TACH/4 si ottiene un bit rate di circa $184/0.47 = 391$ b/s (molto basso), mentre nel caso TACH/8 si ha un bit rate doppio.

Ogni canale SDCCH/8 ha una capacità di 8 slot in una multitrama lunga 102 trame. Ogni trama misura 4.615ms.

Ogni time slot trasferisce 114 bit (lordi); poiché la codifica trasforma 184 bit in 456, ogni time slot trasferisce $114 \cdot 184 / 456 = 46$ bit di informazione.

Oltre al payload (max 140 byte) sono aggiunti fino a 37 byte. Se ipotizzo di trasmettere 32 byte, la dimensione complessiva del pacchetto risulta quindi di 69 byte.

Per trasferire 69 byte servono $69 \cdot 8 / 46 = 12$ slot. Ciò significa che ogni SDCCH/8 ci mette una multitrama e mezza a trasferire tutto il pacchetto, ovvero $4.615 \cdot 102 \cdot 1.5 = 706$ ms.

Se considero 8 canali SDCCH/8, ogni secondo posso trasferire al massimo $8 / 0.706 = 11$ SMS circa contenenti il nostro piccolo payload (si noti che il massimo valore per un SMS è 140 byte, mentre in questa sede supponiamo di trasferire solo 32 byte).

Quindi, per ogni cella e per ogni canale di segnalazione configurato interamente ad SDCCH/8 ed utilizzato solo per il servizio in oggetto, si supporterebbero fino a 110 utenti ipotizzando un pacchetto ogni 10 secondi (nel calcolo non si è tenuto in considerazione l'overhead protocollare per apertura e rilascio della connessione).

Il costo di un canale di segnalazione utilizzato solo per questo scopo è una chiamata a circuito in meno.

GPRS

Breve introduzione al GPRS

Il GPRS è lo standard che permette l'introduzione della trasmissione dati a commutazione di pacchetto nel sistema GSM; grazie a questa possibilità il gestore è in condizione di offrire agli utenti mobili l'accesso alle reti esterne basate su IP (ad esempio Internet o le reti Intranet aziendali).

Il principio fondamentale alla base delle scelte effettuate in sede di definizione dello standard GPRS prevede il mantenimento dell'interfaccia radio del sistema cellulare GSM, così da lasciare inalterata la modalità di accesso alla rete e la condivisione della risorsa radio, garantendo, al contempo, un'adeguata qualità del servizio. Lo standard GPRS condivide, quindi, le scelte di livello fisico e la modalità di accesso al mezzo radio del sistema GSM originario, ed in particolare: la banda di frequenze, la tecnica di modulazione GMSK, la tecnica TDMA di accesso multiplo a divisione di tempo, la struttura degli slot temporali.

Dal punto di vista sistemistico e dell'architettura di rete, l'estensione dei servizi GPRS alla rete GSM richiede, pertanto, semplicemente l'inserimento di due nuove entità di rete preposte alla gestione delle nuove funzionalità: Serving GPRS Support Node (SGSN) e Gateway GPRS Support Node (GGSN): queste entità agiscono come nodi di commutazione e gestiscono la mobilità degli utenti GPRS e l'interconnessione con reti dati esterne.

Canali logici nel GPRS

Oltre ai canali logici definiti per il GSM, il sistema GPRS introduce ulteriori canali logici che si affiancano a quelli definiti in ambito GSM:

Packet Broadcast Control CHannel (PBCCH) che trasmette informazioni di segnalazione a tutti gli utenti GPRS all'interno di una cella.

Il *Packet Common Control CHannel (CCCH)* – include:

Packet Random Access CHannel (PRACH): utilizzato in uplink dalle stazioni mobili per richiedere l'accesso alle risorse e per rispondere ad un paging,

Packet Paging CHannel (PPCH): utilizzato per segnalare alla stazione mobile la presenza di una chiamata in arrivo (downlink).

Packet Access Grant CHannel (PAGCH): utilizzato nella fase di definizione della connessione per assegnare la risorsa alla stazione mobile o per notificare l'accoglimento della richiesta di accesso (downlink).

Packet Notification CHannel (PNCH): utilizzato per inviare informazioni ad un gruppo di stazioni mobili prima di dare inizio ad un trasferimento punto-multipunto (downlink).

Packet Data Transfer CHannel (PDTCH): utilizzato per il trasferimento dei dati di utente a commutazione di pacchetto. Viene temporaneamente assegnato ad una o più stazioni mobili (in tal caso si tratta, evidentemente, di trasmissione punto-multipunto). Nel caso di operazioni multislot una stazione mobile può utilizzare più di un PDTCH in parallelo (uplink e downlink): in tal caso al relativo utente risulta assegnata una capacità multipla della capacità del singolo PDTCH.

L'obiettivo di incrementare la velocità di trasmissione garantita al singolo utente viene pertanto conseguito attribuendo, a chi abbia bisogno di elevate capacità, un numero di canali PDTCH sufficiente a soddisfare la richiesta (fino ad un massimo di otto).

Packet Associated Control CHannel (PACCH): utilizzato per trasmettere informazioni di segnalazione relative ad una particolare trasmissione, ad esempio gli acknowledgment e il livello di potenza ricevuta (uplink e downlink).

In Fig. 35 è possibile osservare la mappatura degli ulteriori canali logici del GPRS sul canale fisico PDCH.

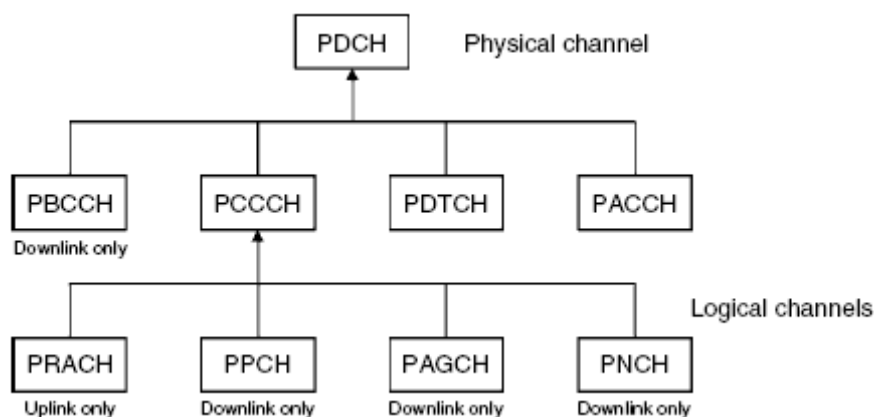


Fig. 35 - Canali logici nel GPRS che si aggiungono a quelli del GSM

L'accesso in uplink nel GPRS

Le modalità con cui un utente GPRS richiede la risorsa radio per dare inizio ad una trasmissione di dati a commutazione di pacchetto, sono sostanzialmente analoghe a quelle previste dal sistema GSM; un'importante eccezione riguarda l'eventualità che non vi siano canali disponibili: in tal caso la stazione base notifica al richiedente GPRS che la richiesta è stata correttamente ricevuta e che, pertanto, la risorsa verrà assegnata non appena disponibile. Questa soluzione evita che il chiamante inoltri sul canale ad accesso casuale (PRACH) una nuova richiesta, con il rischio di incorrere in collisioni con altri richiedenti.

Gli aspetti più innovativi si trovano comunque nella fase di trasmissione vera e propria. Quando la stazione base dispone delle risorse necessarie per accogliere la richiesta dell'utente mobile GPRS, risponde comunicando la lista dei canali assegnati (si ricorda che GPRS offre la possibilità di sfruttare simultaneamente più canali, così da incrementare la bit-rate per singolo utente) e la sequenza di USF (*Uplink State Flag*) che lo identifica e consente, pertanto, l'associazione univoca fra utente e canale ad esso assegnato.

Poichè la stazione base può assegnare uno stesso canale uplink fisico contemporaneamente a più utenti, l'identificativo USF, inviato sulla tratta downlink, segnala a quali utenti mobili verranno assegnate le risorse elementari (4 slot temporali in 4 trame successive) sul corrispondente canale uplink. Il flag USF è costituito da 3 bit, pertanto sono possibili 8 configurazioni utilizzate per gli assegnamenti ai terminali mobili. Questa soluzione consente di moltiplicare, in modo efficiente, burst appartenenti a mobili diversi sullo stesso canale e consente, inoltre, la possibilità di interrompere una trasmissione a favore di un messaggio con priorità più elevata (tipo voce).

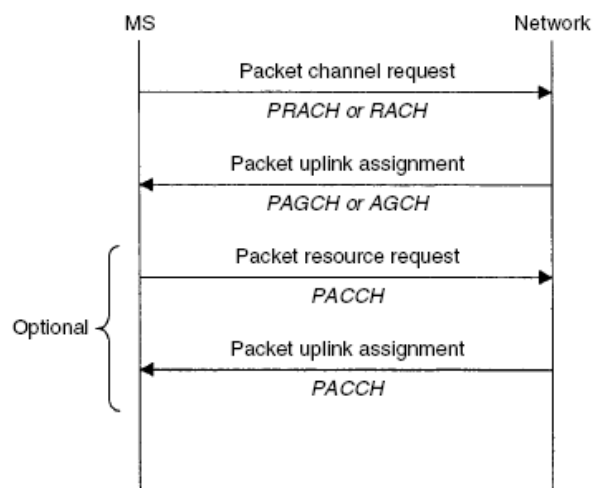


Fig. 36 - Accesso in uplink sul RACH o PRACH nel GPRS

In Fig. 36 è possibile osservare la procedura di richiesta d'accesso sul RACH o sul PRACH e la risposta della rete sul PAGCH o sul AGCH.

Il GPRS aggiunge alle due strutture multiframe del GSM (da 26 e 51 multiframe) un canale per trasmissione più efficiente dei dati. Tale canale si ripete ogni 52 trame (52 multiframe). Osservando le 52 trame, esse sono suddivise in 12 block di 4 slot ciascuno più 4 slot per altri usi. Più precisamente, si alternano 3 block ed uno slot. Le trasmissioni dei dati hanno come unità i block.

Essendo la struttura su 52 trame, la risorsa unitaria (il block) corrisponde a 4 slot GSM e all'interno di un multiframe sono trasferiti 12 block in circa 240ms ($4.615\text{ms} \times 52\text{trame} = 239.98\text{ ms}$).

Anche per il GPRS, tutti i messaggi trasferiti dal livello MAC al livello fisico saranno inviati come una sequenza di quattro *Normal Burst* definiti per il sistema GSM; questo significa che ogni blocco conterrà un singolo messaggio che verrà trasferito in quattro trame consecutive sullo stesso PDCH. Questa tipologia di codifica e di trasmissione vale sia per i messaggi di controllo che per i messaggi dati.

L'unico canale a costituire eccezione è il PRACH in cui i messaggi trasferiti vengono codificati mediante *Access Burst* di tipo GSM. Questo significa che in ogni blocco dedicato al PRACH possono essere presenti fino a quattro messaggi diversi, uno per trama che costituisce il blocco, visto che un *Access Burst* è codificato in modo tale da occupare un solo *timeslot*.

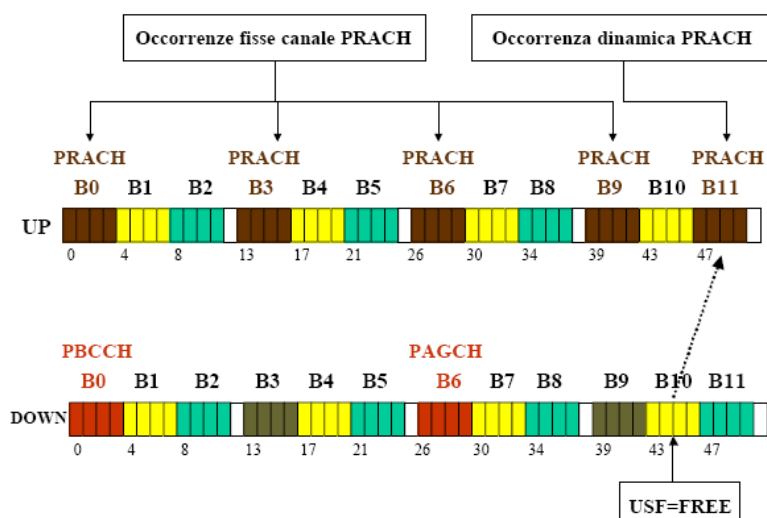


Fig. 37 - Schema di allocazione dei blocchi

Con il GPRS l'*access burst* può portare 11 bit di informazione (da usare sul Packet RACH, PRACH); quindi, si possono trasportare al massimo 8 bit sul RACH e 8 oppure 11 sul PRACH; in quest'ultimo caso si parla di *extended packet access burst*. Ciò, tuttavia, non cambia le cose in modo sostanziale rispetto al caso del GSM.

Sul singolo canale fisico, in 240ms possono essere serviti al massimo 12 utenti. Ciò non è particolarmente restrittivo, in quanto permetterebbe $12 \text{ (utenti)} / 0.240 \text{ (secondi)} = 50$ utenti al secondo sul singolo canale fisico. Anche in questo caso, ciò non rappresenta un grosso limite nell'accesso multiplo.

Come premesso, la 52 multiframe è divisa in 12 burst. Ogni burst trasferisce 456 bit di informazione.

Per trasferire 32 byte più ad esempio 50 byte di header (TCP+UDP+altro), mi servono $82 \cdot 8 / 456 = 1.44$ burst, ovvero per arrotondamento 2 burst. Posso quindi trasferire 6 pacchetti ogni 52 multiframe. Il che significa $6 / (52 \cdot 0.004615) = 25$ pacchetti al secondo.

Quindi, per ogni cella e per ogni slot configurato come canale PTCH GPRS si possono supportare fino a 250 utenti (ipotizzando una trasmissione ogni 10 secondi). Anche con questi numeri, RACH/PRACH non dovrebbe costituire un limite nell'accesso multiplo alla risorsa.

Si noti che questa soluzione richiede un overhead di segnalazione nettamente inferiore rispetto al caso precedente.

UMTS

Breve introduzione all'UMTS

Lo Universal Mobile Telecommunications System (UMTS) rappresenta uno dei principali sistemi di telefonia mobile di terza generazione.

Rispetto ai sistemi 2G e 2.5G cambia l'interfaccia radio le cui caratteristiche principali sono di seguito riportate:

- supporto di servizi con bit rate fino a 144Kbit/s (384kbps) con completa mobilità e copertura e di servizi fino a 2Mbit/s in situazioni di limitata mobilità e copertura;
- frequenza portante a circa 2 GHz;
- la modulazione utilizzata è la QPSK;
- granularità nell'allocazione della bit rate per ottenere flessibilità e guadagno di moltiplicazione;
- accesso radio per tutti i possibili ambienti utilizzando una singola tecnologia;
- supporto di servizi dati a pacchetto ad alta capacità, servizi multimediali e asimmetrici;
- alta flessibilità nell'introduzione di nuovi servizi;
- migliore efficienza spettrale degli attuali sistemi di seconda generazione;
- cambiano gli elementi della rete fissa relativi all'accesso radio (RNC e node B).

L'UMTS, nella versione UTRA proposta da ETSI, supporta due diverse modalità di *duplexing* per le comunicazioni nelle tratte di uplink e downlink: modalità TDD (*Time Division Duplex*, a divisione di tempo) e modalità FDD (*Frequency Division Duplex*, a divisione di frequenza). Nel primo caso le fasi di uplink e di downlink avvengono nella medesima banda di frequenze ma in intervalli temporali differenti, nel secondo caso le due fasi avvengono simultaneamente ma su bande differenti. UTRA-TDD fa uso delle bande 1900-1920 MHz e 2010-2025 MHz, mentre UTRA-FDD fa uso della banda 1920-1980 MHz per l'uplink e 2110-2170 MHz per il downlink.

Dal punto di vista della tecnica di accesso al canale, le due soluzioni proposte da ETSI, pur diverse, prevedono l'adozione della tecnica CDMA.

In Fig. 38 sono riportate le caratteristiche fondamentali del livello fisico dell'UMTS.

	FDD	TDD
Tecnica di accesso	W - CDMA	TD - CDMA
Chip Rate	3.84 Mchip/s	
Canalizzazione	4.4 – 5 MHz	
Durata di trama	10 ms	
N. di slot per trama	15	
Modulazione	downlink: QPSK	downlink: QPSK
	uplink: dual code BPSK	uplink: QPSK
Velocità di trasmissione dell'informazione	Variabile (ogni trama). Velocità diverse possono essere ottenute variando lo spreading factor, oppure assegnando più codici al segnale oppure (nel solo caso TDD) affasciando più time slot.	

Fig. 38 - Caratteristiche fondamentali del livello fisico per lo standard UMTS

La trama UMTS ha durata 10 ms (nel GSM era 4.615 ms) ed è suddivisa in 15 slot ciascuno della durata di 2560 chip. La banda di frequenza associata ad un canale è di 5 Mhz (nel GSM era 200 kHz). Dovendo gestire una banda di frequenza molto larga, la frequenza con

cui il ricevitore WCDMA campiona il segnale ricevuto è molto elevata dando luogo a complessità e consumi maggiori rispetto al ricevitore GSM.

I canali nell'UMTS

I canali di trasporto dell'UMTS e la loro mappatura sui canali fisici sono illustrati in Fig. 39

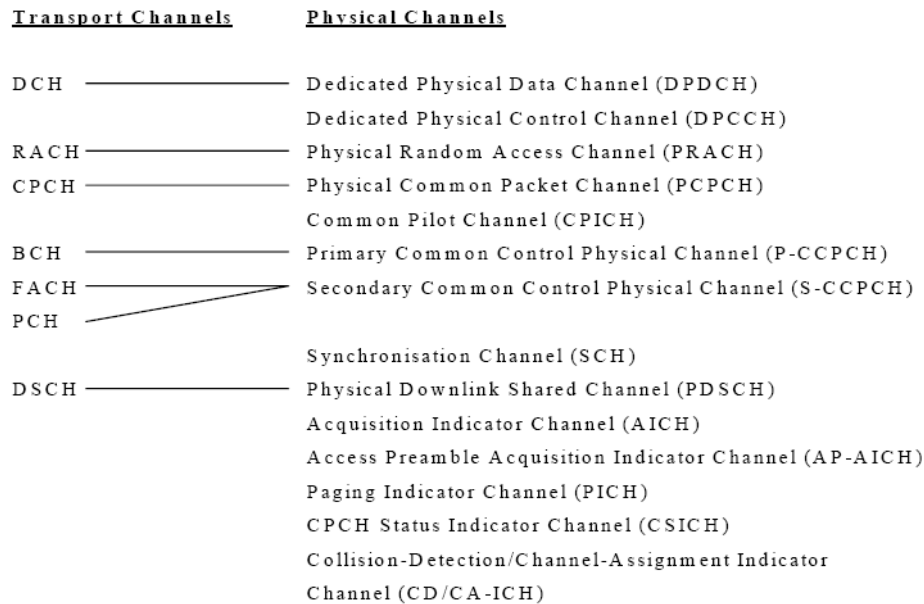


Fig. 39 - I canali dell'UMTS

In Fig. 40 è possibile, inoltre, osservare la mappatura dei canali tra il livello di rete (RLC), il MAC e il livello fisico.

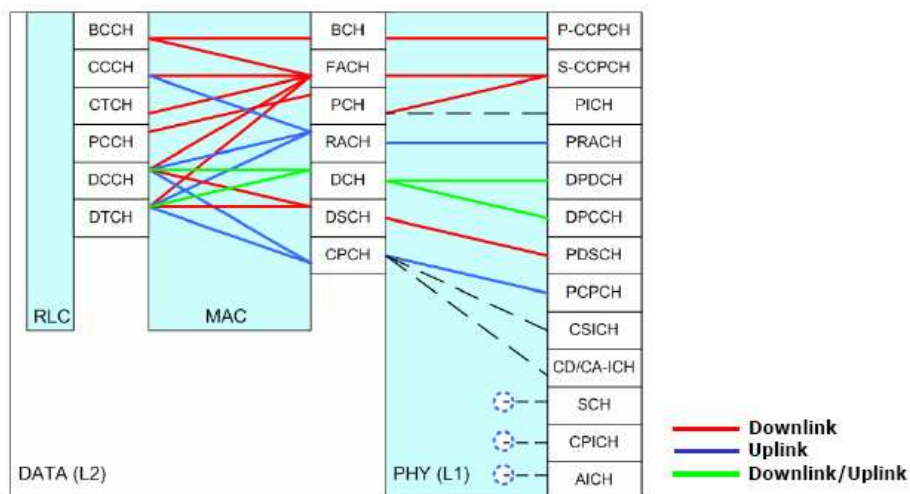


Fig. 40 - Mappatura dei canali nell'UMTS

L'accesso in uplink nell'UMTS

Ogni cella ha un massimo di 16 PRACH.

Il PRACH è diviso in slot di lunghezza doppia rispetto al GSM e al GPRS (20ms diviso 15, anziché 10ms diviso 15). Il canale PRACH è diviso in 12 sottocanali a divisione di tempo (uno slot PRACH per sottocanale). A seconda della classe di traffico, ciascuna sessione può disporre di un sottoinsieme di tali sottocanali [4][5].

Una stazione che deve accedere al PRACH seleziona casualmente uno dei sottocanali a disposizione e seleziona casualmente una tra 16 firme, il cui scopo è solo quello di rendere riconoscibile la sorgente del messaggio; poi, in base ad un parametro chiamato *persistent value*, determina casualmente se inviare la richiesta o rimandare al frame successivo. Il *persistent value* ha un valore compreso tra 0 ed 1 e corrisponde alla probabilità di trasmettere in quel frame; 1 significa che quando si ha la risorsa si prova a trasmettere, 0.5 che ci si prova solo una volta su due nel primo frame e altrimenti si rimanda al frame successivo o oltre, ecc. Più precisamente, il *persistent value* è una potenza negativa di 2 a sua volta moltiplicata per un parametro con valore compreso tra 0.2 e 1. Così tanti gradi di libertà sul *persistent value* consentono al progettista di differenziare in modo significativo la priorità delle varie classi di traffico.

E' possibile gestire fino a otto classi diverse di traffico mediante differenti parametri di configurazione per l'accesso. Ad ogni classe vengono concessi un sottoinsieme dei sottocanali d'accesso ed un valore per il *persistent value*.

Il protocollo adottato a livello MAC, anche grazie all'utilizzo di diversi sottocanali e di un inizio probabilistico della contesa, permette di ottenere un'efficienza massima di uso del canale anche oltre il 60-70% [5], ben oltre il valore pari al 36% ottenuto con il normale Slotted Aloha.

Il pacchetto di richiesta d'accesso è lungo 4096 chips (si noti che lo slot del PRACH, corrispondente a due slot normali, conterrebbe $3840 \times 2 = 7680$ chip). Viene trasmesso con $SF=256$. La struttura del random access burst per l'accesso è rappresentata in Fig. 41.

Una volta inviato il messaggio di richiesta di accesso si attende un ACK o un NACK sul canale AICH; se arriva l'ACK si trasmette il pacchetto di dati sul canale PRACH; se invece arriva un NACK, si fallisce a livello fisico e si torna a livello MAC; poi a livello MAC si effettuerà il backoff. E' anche possibile che non arrivi né un ACK né un NACK. In quest'ultimo caso, significa che la potenza utilizzata per la trasmissione era insufficiente e si prevede un meccanismo di ramping, ovvero di incremento progressivo della potenza come illustrato in Fig. 42 e Fig. 43: si effettua nuovamente l'accesso selezionando casualmente il sottocanale e la firma, a patto di non ritrasmettere con un intervallo tra i due preamboli minore di 3 o 4 slot PRACH (dipende dalla configurazione scelta per il canale considerato).

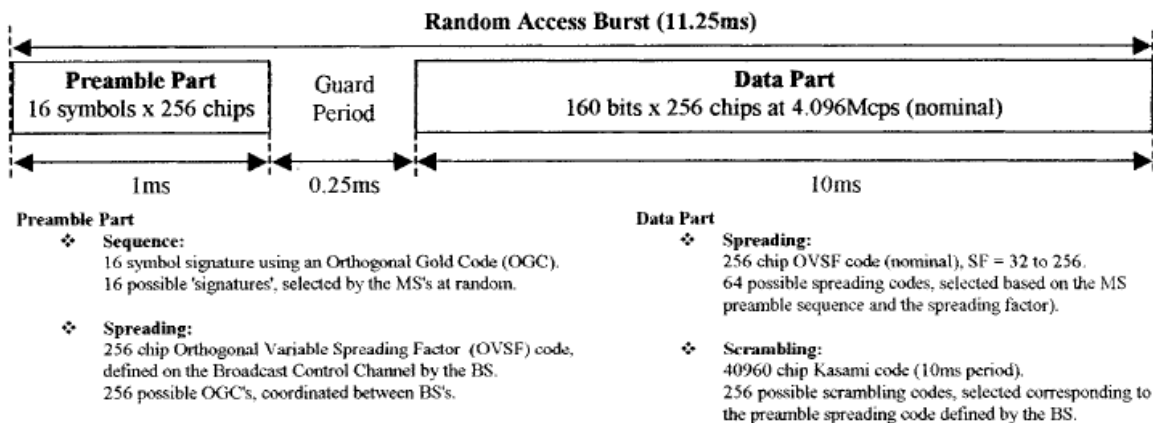


Fig. 41 - Struttura del random access burst per l'accesso nell'UMTS

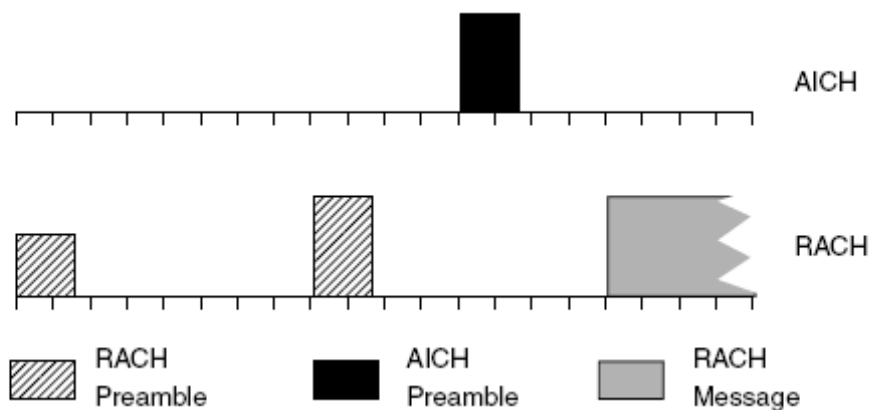


Fig. 42 - Ramping sul PRACH e trasmissione del messaggio

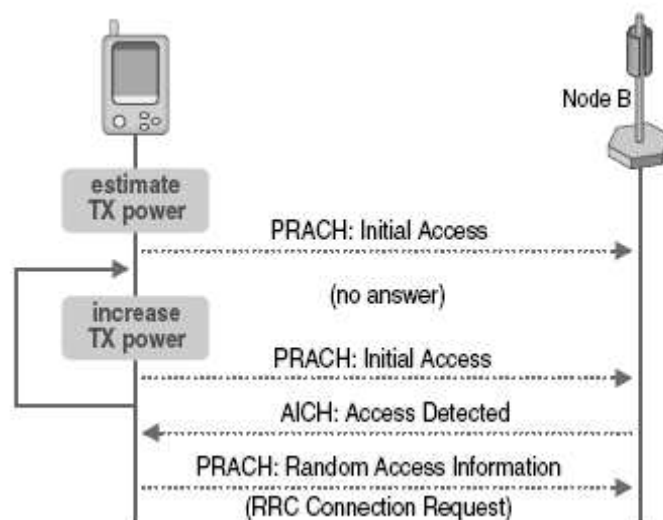


Fig. 43 - Procedura di accesso casuale in uplink

Come detto, tra due preamboli durante il ramping non possono esserci meno di 3 o 4 slot; corrispondentemente, tra un preambolo e il pacchetto dati che segue il messaggio di ACK

intercorrono o 3 o 4 access slot; la selezione tra questi 2 valori è effettuata da un parametro con valore 0 oppure 1, chiamato *AICH transmission timing*. Se si attendono 3 slot tra i due preamboli, il messaggio sull'AICH viene inviato dopo 1.5 slot PRACH dall'inizio della trasmissione del preambolo, altrimenti viene inviato dopo 2.5 slot PRACH.

I dati, di durata 10 o 20ms, possono essere trasmessi con SF da 32 a 256 come mostrato in Fig. 44. Poiché nel caso dell'UMTS in uplink i bit di dati sono mescolati I-Q con quelli di controllo di livello fisico, non è possibile inviare pacchetti più corti del valore massimo corrispondente alla durata selezionata (10ms o 20ms): se, ad esempio, la capacità del pacchetto è di 320 bit e si hanno meno di 320 bit da trasmettere, saranno inseriti bit di pudding, ignorati al ricevitore.

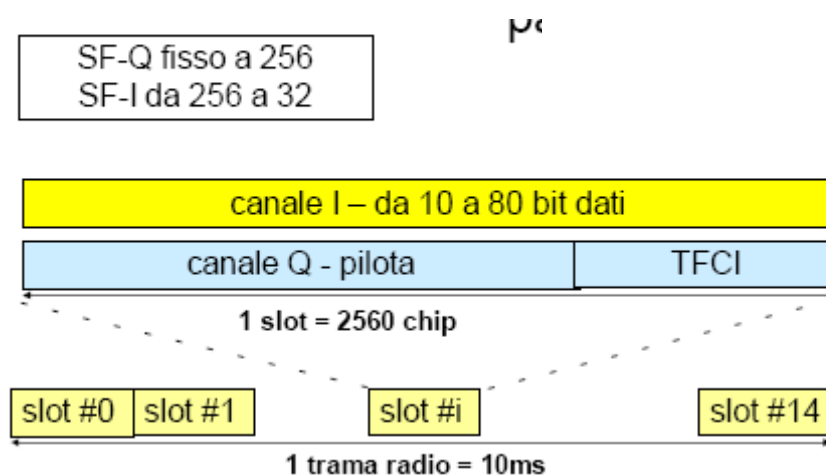


Fig. 44 - PRACH: formato dei dati nella parte di messaggio

Per completare il quadro riguardante la trasmissione di dati sul canale PRACH, in Fig. 45 è illustrata la struttura temporale del canale PRACH, mentre in Fig. 46 si mostra come sono combinati i bit di dato per la successiva modulazione all'interno degli slot temporali.

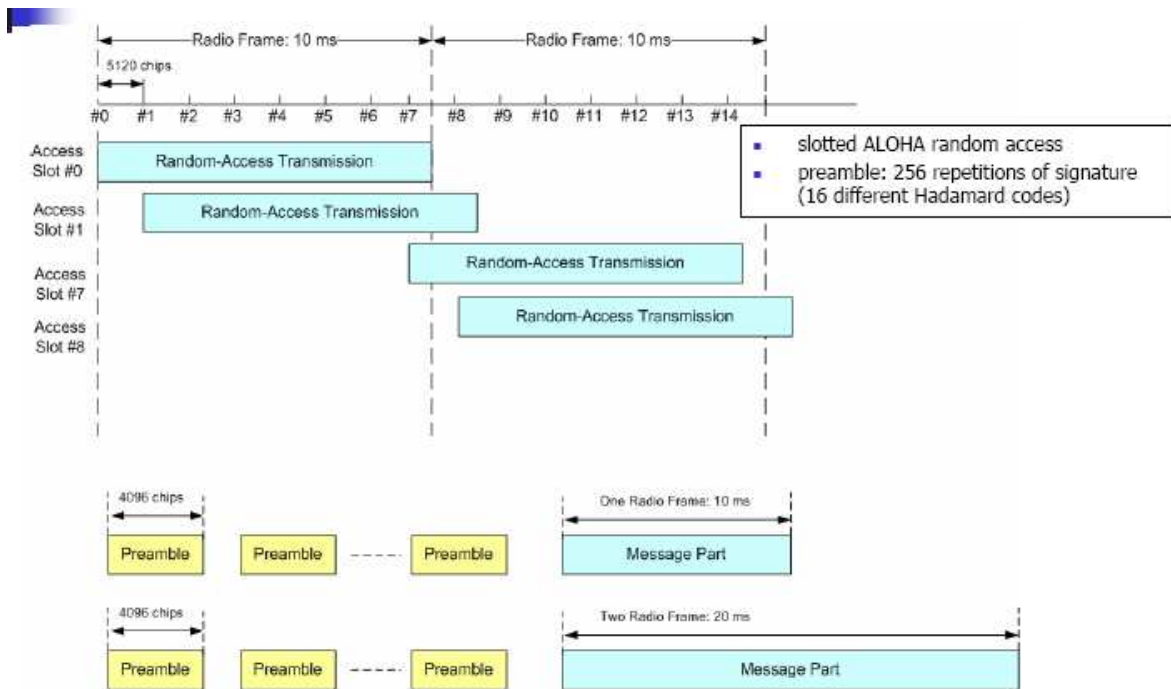


Fig. 45 - Accesso casuale in uplink

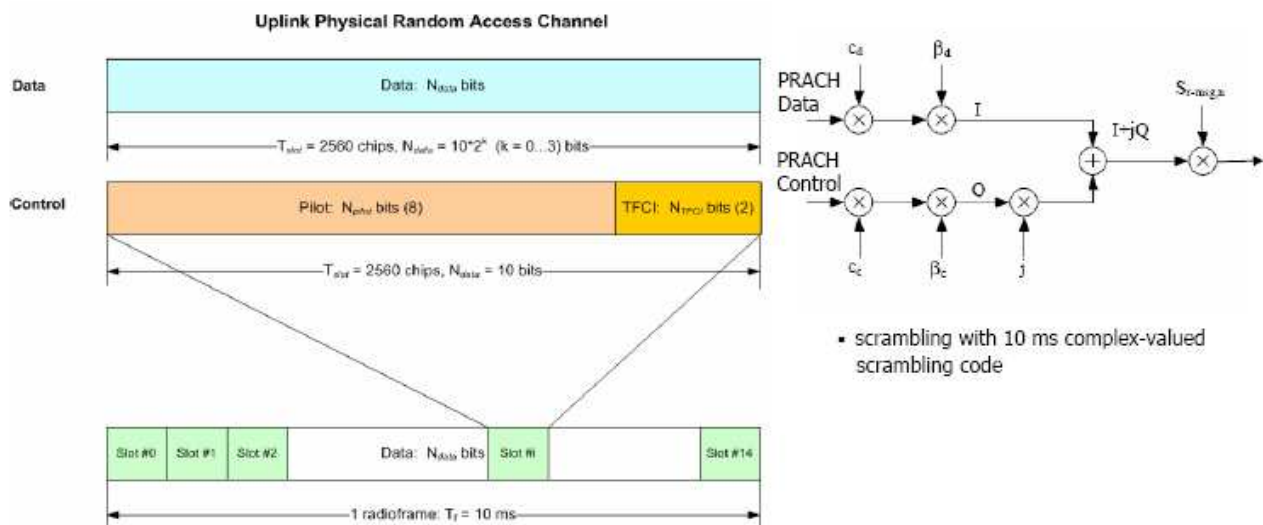


Fig. 46- Struttura del PRACH

Supponiamo ora di voler trasmettere (32+40) byte (576bit) come nel caso GPRS. Viene di seguito effettuato una prima valutazione di massima della capacità del canale utilizzato per poi arrivare a valutare i ritardi di accesso e le tempistiche di trasmissione.

Per quanto riguarda il calcolo della capacità del canale utilizzato, prendiamo in considerazione il numero di bit da trasmettere. Includendo tutti gli header di livello 2, un massimo di 320 bit possono essere trasmessi in 10ms utilizzando uno SF=32 oppure in 20ms con SF=64; servono quindi 20ms e SF=32 per poter inviare il pacchetto ipotizzato. Va evidenziato che l'utilizzo di uno SF pari a 32 implica che la trasmissione in questione produrrà all'interno del sistema un'interferenza per le altre connessioni relativamente alta; a

parità di distanza tra trasmettitore e ricevitore, infatti, uno SF basso provoca un ridotto coding gain ed è necessaria una potenza maggiore per la sua corretta ricezione; a questo va aggiunto il fatto che il controllo di potenza è approssimativo e non viene aggiustato durante la trasmissione del pacchetto.

Con alcune ipotesi semplificative ed indipendentemente dal numero di bit utili del payload e dal valore dello SF, è possibile effettuare un calcolo approssimativo della durata media complessiva della trasmissione su canale RACH da cui dedurre il numero massimo di utenti che possono essere serviti.

Se ipotizzo che il preambolo sia ricevuto correttamente e che si scelga di mandare il pacchetto dati dopo 4 slot e per la durata di 20ms, dall'inizio dell'operazione alla fine passano 1slot (preambolo) + 3 slot (distanza 4 slot è intesa dall'inizio del preambolo all'inizio del pacchetto) + 15 slot = 19 slot PRACH. Con un pacchetto ogni 19 slot ne consegue al massimo 50 (coppie di frame al secondo) * 15 (slot per coppia di frame) / 19 (slot necessari per un pacchetto) = 39.4737 pacchetti al secondo.

Quindi, con l'ipotesi dei 10 secondi tra un invio ed il successivo, riescono ad accedere al massimo 394 utenti.

Tale numero diminuisce se, più realisticamente, immaginiamo di non riuscire ad inviare subito il preambolo oppure che l'invio non avvenga con successo al primo tentativo. Immaginando di perdere nella procedura un tempo equivalente al fallimento di un preambolo (4 slot), gli slot totali per pacchetto diventano 23. E allora il numero di pacchetti al secondo diventano 32.6087 e il numero massimo di utenti che riescono a trasmettere scende a 326.

Si noti che, una volta stabilito il numero medio di slot necessario per la trasmissione di un pacchetto e la dimensione in bit di tale pacchetto, la capacità in funzione della dimensione del pacchetto da trasmettere sarà costante: finché i bit da trasmettere per pacchetto sono meno del massimo possibile, il pacchetto viene riempito con bit di padding e quindi non aumenta la capacità, oltre tale soglia non è possibile fornire il servizio se non dividendo i dati su più PDU PRACH.

In conclusione, nelle condizioni ipotizzate e trascurando le problematiche di interferenza e copertura, il limite risulta pari a 326 utenti per cella e pacchetti di al massimo 40 byte comprensivi degli header.

Volendo confrontare tale valore con i 250 utenti ottenuti per il GPRS va evidenziato che ci sono due differenze da tenere in conto. Per prima cosa, il numero di celle UMTS necessarie a coprire un territorio è approssimativamente pari al doppio rispetto al numero di celle GPRS. In secondo luogo, nel caso UMTS il canale di segnalazione utilizzato non corrisponde esattamente ad una connessione a circuito come nel caso GPRS; se si ipotizza per semplicità che un canale con SF=32 sia equivalente a due canali con SF=64, allora il canale di segnalazione UMTS corrisponde in media a due chiamate a circuito; tale ipotesi è tuttavia molto semplicistica nel caso UMTS e serve solo a chiarire che la risorsa occupata per la trasmissione è in questo caso maggiore rispetto a quella che occuperebbe un terminale con una sessione voce attiva.

Capacità di canale su rete di seconda e terza generazione

In questo paragrafo si vuole mostrare qualche grafico per riassumere quanto discusso in precedenza.

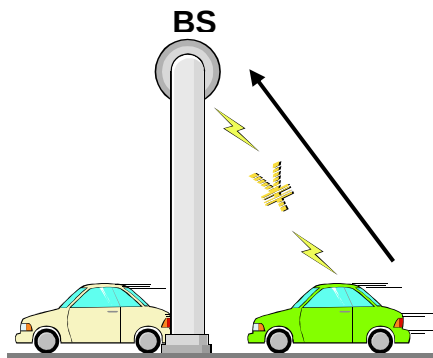


Fig. 47 - Periodico invio di informazioni dai veicoli alla rete cellulare

Si assuma che:

- ogni veicolo invii informazioni relative a posizione e velocità (più eventuali altre informazioni) ogni 10 secondi;
- l'invio dei dati avvenga via SMS (su rete GSM) oppure con invio di pacchetti di dati (su GPRS o UMTS);
- nel caso di invio di pacchetti dati sia 40 byte la dimensione dell'header dall'applicazione al livello di rete;
- venga utilizzato un intero canale di segnalazione per questo servizio.

Va precisato che i risultati illustrati derivano esclusivamente dalle considerazioni effettuate e tengono quindi conto dei canali a disposizione per le diverse tecnologie e della loro capacità, ma non considerano altri possibili limiti tecnologici quali, ad esempio, limiti sull'indirizzamento o problemi di copertura e interferenza.

Con queste ipotesi, seguono i risultati mostrati in Fig. 48 e Fig. 49.

In particolare, in Fig. 48 si mostra il numero massimo di terminali che possono essere serviti per ciascuna cella mentre in Fig. 49 si mostra il numero massimo di terminali che possono essere serviti in una città coperta da 50 celle GSM/GPRS oppure 100 celle UMTS. Entrambe le figure sono illustrate al variare della dimensione in byte del payload.

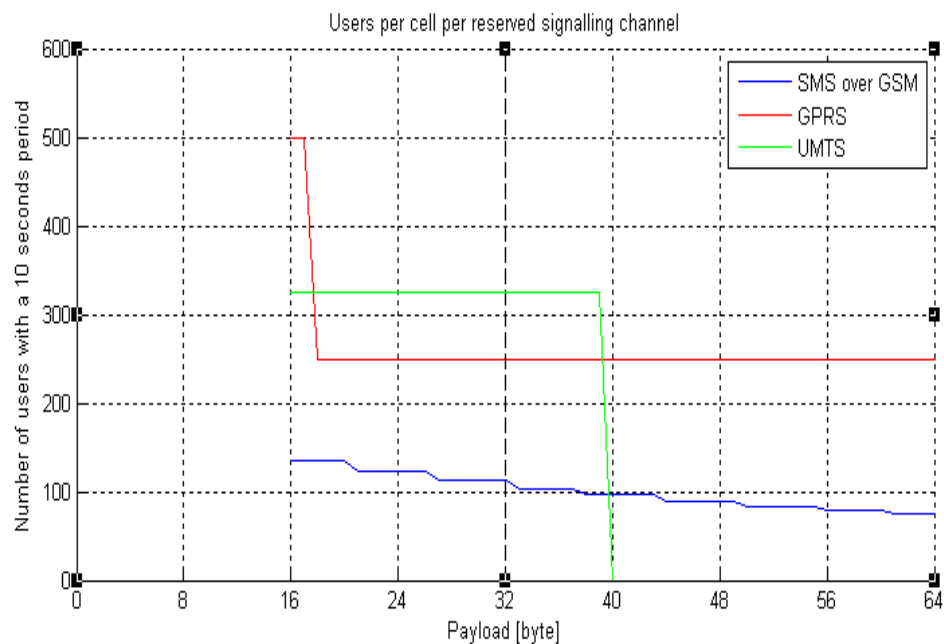


**Fig. 48 - Numero di utenti per cella per singolo canale di segnalazione.
Confronto tra rete GSM, GPRS ed UMTS**

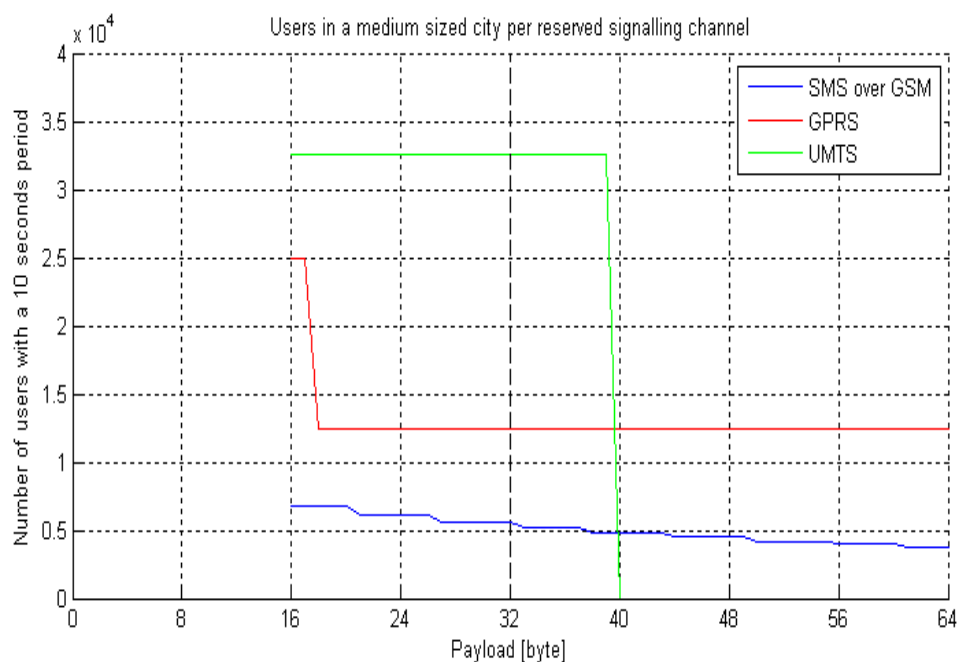


**Fig. 49 - Numero di utenti in una città di 50 celle per singolo canale di segnalazione.
Confronto tra rete GSM, GPRS ed UMTS**

Bibliografia

- [1] Libro intitolato “The GSM System for Mobile Communications”. Autori: [Michel Mouly](#), [Marie-Bernadette Pautet](#). Editore da Cell&Sys, France, 1992.
- [2] Libro intitolato “GPRS: Gateway to Third Generation Mobile Networks”. Autori: Gunnar Heine, Holger Sagkob. Editore da Artech House Inc., 685 Canton Street, Norwood, MA, 2003.
- [3] Report “SMS over GPRS”. Scritto da Charlotta Baath e Joanna Kuhn presso il Department of Microelectronics and Information Technology (IMIT), Royal Institute of Technology (KTH), Sweden. Aprile 2003.
- [4] Libro intitolato “WCDMA for UMTS, Radio Access for Third Generation Mobile Communication”, III Edition. Autori: Harri Holma and Antti Toskala. Editore da John Wiley and Sons.
- [5] Articolo intitolato “Performance analysis of the random access channel (RACH) in WCDMA”. Autori: I.N.Vukovic, T.Brown. Presente negli atti della conferenza VTC Spring 2001.
- [6] Documento 3GPP TS 34.108: “Technical Specification 3rd Generation Partnership Project; Technical Specification Group Terminals; Common test environments for User Equipment (UE) conformance testing (Release 1999).

MODULO GIS

Un Sistema Informativo Territoriale (SIT; in lingua inglese Geographic(al) Information System, abbreviato in GIS) è un sistema informativo computerizzato che permette l'acquisizione, la registrazione, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici (geo-referenziati).

Hardware

Il server necessario a rendere possibile lo sviluppo del sottosistema GIS deve mettere a disposizione una piattaforma stabile e performante in grado di supportare i seguenti pacchetti software:

- Server web (es. Apache)
- PHP module
- Postgresql
- Estensione PostGIS
- GeoServer
- Java Development Kit

Ovviamente è possibile che non tutti questi applicativi siano installati sulla stessa macchina, ma siano distribuiti su più macchine (anche virtuali).

In base alle condizioni di carico in cui si troveranno a lavorare i server è necessario dimensionare opportunamente le caratteristiche fisiche delle macchine (CPU, RAM, HD) tenendo in considerazione che le operazioni più gravose sono:

- Memorizzazione dei dati geografici nel database (stress HD)
- Rendering delle mappe (stress CPU+RAM)
- Accesso multiplo di utenti alle pagine web (stress CPU+RAM)

Architettura Software

L'architettura del sotto-sistema GIS adottato per il progetto TRIP è schematizzato nella Fig. 50:

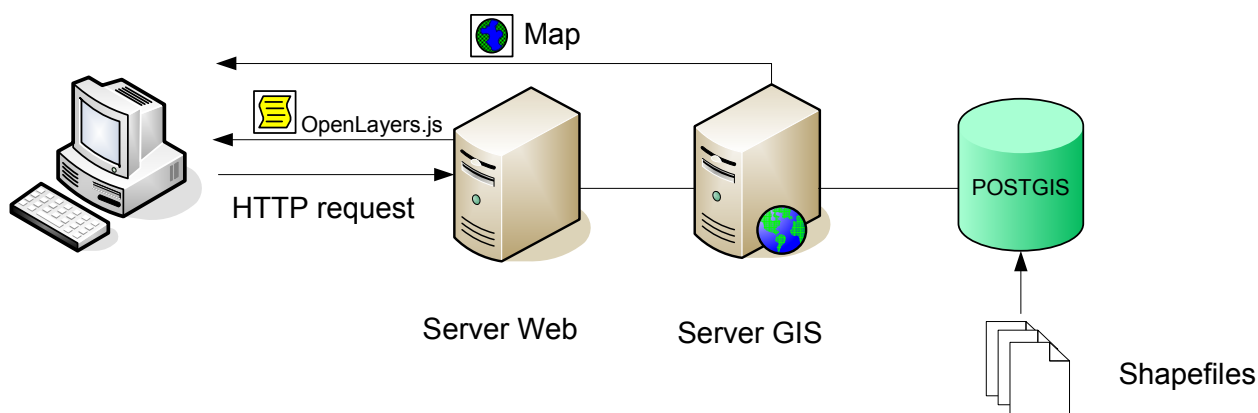


Fig. 50 - Sottosistema GIS

Per quanto riguarda l'architettura del modulo GIS, la figura mostra come a partire dai dati grezzi di TeleAtlas in formato shapefile sia possibile fornire all'utente una mappa dinamica e navigabile. E' possibile fornire delle mappe dettagliate all'utente, per un'innumerabile serie di applicazioni, per esempio:

- scegliere il punto di partenza/arrivo mediante click sulla mappa
- visualizzazione delle linee dei mezzi pubblici e delle fermate
- ricerca delle fermate più vicine ad un'indirizzo
- visualizzazione del percorso calcolato dal sistema
- visualizzazione dei punti di interesse (ristoranti, musei, ecc)

Ovviamente queste mappe, oltre ad essere il più dettagliate e precise possibile, devono essere esteticamente curate e avere la possibilità di essere navigabili e zoomate. Per ottenere entrambe queste qualità è necessario avere a disposizione sia un database per archiviare tutte le informazioni sia un server GIS per renderle fruibili all'utente.

In una prima fase è necessario quindi importare, mediante apposite procedure, all'interno del database con estensioni spaziali PostGIS gli shapefile, uno per ciascun livello che si vuole rappresentare sulla mappa (es. fiumi, boschi, edificato, strade, ferrovie, ecc). Ogni shapefile viene così importato in una tabella all'interno del database utilizzando lo script shp2pgsql che carica i dati *.shp e li trasforma in formato PostGIS. Questa operazione viene eseguita una-tantum in maniera off-line solo alla configurazione del sistema o se i dati necessitano di un aggiornamento.

A run-time invece, per ogni utente che si collega al server web, e che richiede una mappa, il server GIS elabora la richiesta e fornisce a ciascun utente nel più breve tempo possibile le informazioni richieste. Nello specifico il server GIS provvede a:

- recuperare i dati geografici memorizzati in formato testuale (es. MULTILINESTRING(7.45 45.45,...)) nelle tabelle del DB
- applicare a ciascuna geometria il suo stile (definito mediante un XML che descrive come deve essere renderizzata la geometria agendo sullo spessore delle linee, sui colori e sulle etichette, in base al tipo stesso di oggetto che la geometria rappresenta)
- sovrapporre i diversi livelli per creare infine la mappa come oggetto grafico vero e proprio (jpeg, png, png8).
- fornire all'utente una libreria Javascript (OpenLayers.js) per rendere la fruizione della mappa dinamica e navigabile (zoom, panning, ecc.).

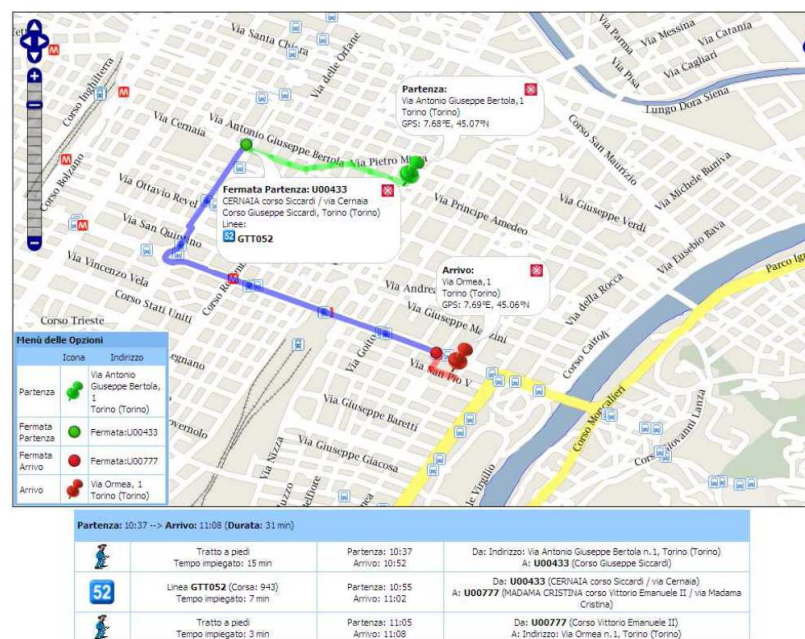


Fig. 51 - Esempio di mappa dinamica

Nella figura è rappresentato un esempio di mappa dinamica e di alcuni contenuti informativi che possono venire visualizzati, in risposta alla ricerca di un percorso.

Ovviamente l'utente può anche fare delle ricerche direttamente sui dati geografici presenti nel DB, senza passare dal server GIS, per esempio volendo cercare le fermate dei mezzi pubblici più vicini al suo punto di partenza o arrivo, per avere un elenco delle vie o quanto altro serve alle applicazioni TRIP. Queste richieste vengono processate direttamente dalle pagine web in cui l'utente effettua le sue richieste, mediante linguaggi di scripting server-side tipo PHP, che accedono a loro volta al database via web.

Di seguito vengono riportate le principali caratteristiche dei software utilizzati così come riportate dai siti dei rispettivi progetti open-source. Per i dettagli sui software, proprio per la loro natura open-source, si faccia riferimento alla documentazione presenti sui siti dei progetti stessi.

GeoServer 1.7.5

GeoServer is an open source software server written in Java that allows users to share and edit geospatial data. Designed for interoperability, it publishes data from any major spatial data source using open standards. Being a community-driven project, GeoServer is developed, tested, and supported by a diverse group of individuals and organizations from around the world. GeoServer is the reference implementation of the Open Geospatial Consortium (OGC) Web Feature Service (WFS) and Web Coverage Service (WCS) standards, as well as a high performance certified compliant Web Map Service (WMS). GeoServer forms a core component of the Geospatial Web.

GeoServer is a Java-based software server that allows users to view and edit geospatial data. Using open standards set forth by the Open Geospatial Consortium (OGC), GeoServer allows for great flexibility in map creation and data sharing. GeoServer allows you to display your spatial information to the world. Implementing the Web Map Service (WMS)

standard, GeoServer can create maps in a variety of output formats. OpenLayers, a free mapping library, is integrated into GeoServer, making map generation quick and easy. GeoServer is built on Geotools, an open source Java GIS toolkit.

There is much more to GeoServer than nicely styled maps, though. GeoServer also conforms to the Web Feature Service (WFS) standard, which permits the actual sharing and editing of the data that is used to generate the maps. Others can incorporate your data into their websites and applications, freeing your data and permitting greater transparency. GeoServer is free software. This significantly lowers the financial barrier to entry when compared to traditional GIS products. In addition, not only is it available free of charge, it is also open source. Bug fixes and feature improvements in open source software are greatly accelerated when compared to traditional software solutions. Leveraging GeoServer in your organization also prevents software lock-in, saving costly support contracts down the road. GeoServer can display data on any of the popular mapping applications such as Google Maps, Google Earth, Yahoo Maps, and Microsoft Virtual Earth. In addition, GeoServer can connect with traditional GIS architectures such as ESRI ArcGIS. GeoServer has a large and vibrant community consisting of users and developers from all over the world. Support is available through a variety of sources, such as email lists and IRC (the latter with real time access to software developers). In addition, commercial support is available from a variety of providers. With GeoServer, you are always in good company.

[Fonte: <http://geoserver.org>]

OpenLayers

OpenLayers makes it easy to put a dynamic map in any web page. It can display map tiles and markers loaded from any source. MetaCarta developed the initial version of OpenLayers and gave it to the public to further the use of geographic information of all kinds. OpenLayers is completely free, Open Source JavaScript, released under a BSD-style License. OpenLayers is developed and supported by a number of organizations around the world. We are also looking for sponsors to help support the community. If you are in a position where you want to support the development of OpenLayers, but do not have development resources to share, you may be interested in supporting through our sponsorship program.

OpenLayers is a pure JavaScript library for displaying map data in most modern web browsers, with no server-side dependencies. OpenLayers implements a (still-developing) JavaScript API for building rich web-based geographic applications, similar to the Google Maps and MSN Virtual Earth APIs, with one important difference -- OpenLayers is Free Software, developed for and by the Open Source software community. Furthermore, OpenLayers implements industry-standard methods for geographic data access, such as the OpenGIS Consortium's Web Mapping Service (WMS) and Web Feature Service (WFS) protocols. Under the hood, OpenLayers is written in object-oriented JavaScript, using components from Prototype.js and the Rico library. The OpenLayers code base already has hundreds of unit tests, via the Test.AnotherWay framework. As a framework, OpenLayers is intended to separate map *tools* from map *data* so that all the tools can operate on all the data sources. This separation breaks the proprietary silos that earlier GIS revolutions have taught civilization to avoid. The mapping revolution on the public Web should benefit from the experience of history.

[Fonte: <http://openlayers.org/>]

Postgresql

PostgreSQL is a powerful, open source object-relational database system. It has more than 15 years of active development and a proven architecture that has earned it a strong reputation for reliability, data integrity, and correctness. It runs on all major operating systems, including Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), and Windows. It is fully ACID compliant, has full support for foreign keys, joins, views, triggers, and stored procedures (in multiple languages). It includes most SQL92 and SQL99 data types, including INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, and TIMESTAMP. It also supports storage of binary large objects, including pictures, sounds, or video. It has native programming interfaces for C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, among others.

[Fonte: <http://www.postgresql.org/about>]

PostGIS

PostGIS adds support for geographic objects to the PostgreSQL object-relational database. In effect, PostGIS "spatially enables" the PostgreSQL server, allowing it to be used as a backend spatial database for geographic information systems (GIS), much like ESRI's SDE or Oracle's Spatial extension. PostGIS follows the OpenGIS "Simple Features Specification for SQL" and has been certified as compliant with the "Types and Functions" profile.

PostGIS has been developed by Refrations Research as a project in open source spatial database technology. PostGIS is released under the GNU General Public License. We continue to develop PostGIS, and have added user interface tools, basic topology support, data validation, coordinate transformation, programming APIs and much more. Our list of future projects includes full topology support, raster support, networks and routing, three dimensional surfaces, curves and splines and other features.

[Fonte <http://postgis.refrations.net/>]

Shapefile

The ESRI Shapefile or simply a shapefile is a popular geospatial vector data format for geographic information systems software. It is developed and regulated by ESRI as a (mostly) open specification for data interoperability among ESRI and other software products.[1] A "shapefile" commonly refers to a collection of files with ".shp", ".shx", ".dbf", and other extensions on a common prefix name (e.g., "lakes.*"). The actual shapefile relates specifically to files with the ".shp" extension, however this file alone is incomplete for distribution, as the other supporting files are required.

Shapefiles spatially describe geometries: points, polylines, and polygons. These, for example, could represent water wells, rivers, and lakes, respectively. Each item may also have attributes that describe the items, such as the name or temperature.

A shapefile is a digital vector storage format for storing geometric location and associated attribute information. This format lacks the capacity to store topological information. The shapefile format was introduced with ArcView GIS version 2 in the beginning of the 1990s. It is now possible to read and write shapefiles using a variety of free and non-free programs. Shapefiles are simple because they store primitive geometrical data types of points, lines, and polygons. These primitives are of limited use without any attributes to specify what they represent. Therefore, a table of records will store properties/attributes for each primitive shape in the shapefile. Shapes (points/lines/polygons) together with data attributes can

create infinitely many representations about geographical data. Representation provides the ability for powerful and accurate computations.

While the term "shapefile" is quite common, a "shapefile" is actually a set of several files. Three individual files are normally mandatory to store the core data that comprises a shapefile. There are a further eight optional files which store primarily index data to improve performance. Each individual file should conform to the MS DOS 8.3 filenameing convention (8 character filename prefix, fullstop, 3 character filename suffix such as shapefil.shp) in order to be compatible with past applications that handle shapefiles. For this same reason, all files should be located in the same folder.

Mandatory files :

- .shp — shape format; the feature geometry itself
- .shx — shape index format; a positional index of the feature geometry to allow seeking forwards and backwards quickly
- .dbf — attribute format; columnar attributes for each shape, in dBase III format

[Fonte wikipedia.en]

TeleAtlas Shapefile

Tele Atlas delivers the digital maps and dynamic content that power some of the world's most essential navigation and location-based services. The information is the foundation for a wide range of personal and in-car navigation systems and mobile and Internet map applications that help users find the people, places, and services they need, wherever they are.

The Network (_NW) theme contains all of the topological Transportation Elements (Roads, Ferries, Address Area Boundaries Elements) within the MultiNet Shapefile product set. The Network theme is useful for many GIS applications, for example:

- Displaying all Transportation Elements in one theme
- Making queries on all Transportation Elements
- Supporting vehicle routing applications

Other important layers are:

- _BU: Built-Up Area
- _LC: Land Coverage
- _WA: Water Area
- _RR: Railroad

[Fonte <http://www.teleatlas.com/index.htm>]

INTEGRAZIONE SOFTWARE GIS

Il sottosistema GIS integra i seguenti pacchetti software:

- Server web (es. Apache)
- PHP module
- Postgresql
- Estensione PostGIS
- GeoServer
- Java Development Kit

Ovviamente è possibile che non tutti questi applicativi siano installati sulla stessa macchina, ma siano distribuiti su più macchine (anche virtuali).

La corretta configurazione e integrazione dei diversi software, che rende possibile il corretto funzionamento del sistema, è un aspetto fondamentale e deve essere progettato e realizzato con particolare accuratezza.

Configurazione Server

Vediamo in dettaglio come devono essere configurati i diversi applicativi software, per permettere lo scambio dei dati tra loro e verso l'utente finale. In Fig. 52 è rappresentato uno schema riassuntivo.

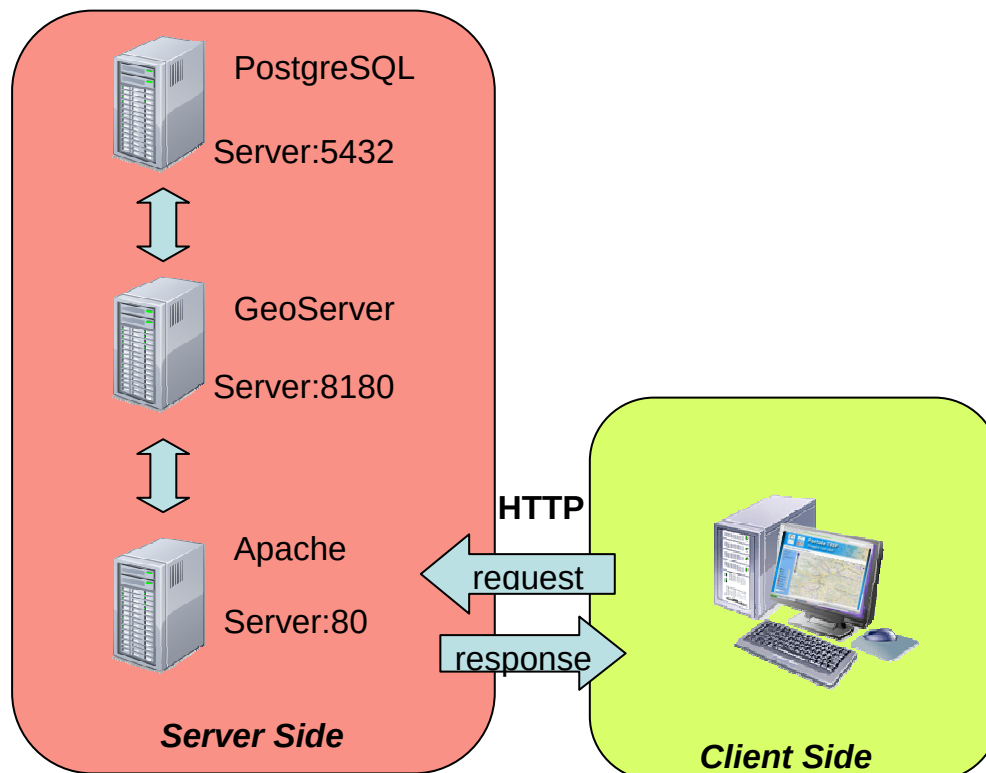


Fig. 52 - Schema riassuntivo configurazione del Server

L'utente (Client Side) accede al sistema (Server Side) attraverso richieste il protocollo HTTP (Hypertext Transfer Protocol) che possono venire effettuate sia mediante PC che terminali mobili compatibili (via WiFi, UMTS,...). Le richieste vengono intercettate e gestite dal server web Apache in ascolto sulla porta 80. Tutte le richieste riguardanti mappe vengono inoltrate al server GIS GeoServer in ascolto sulla porta 8180, installato sulla piattaforma Java di Apache Tomcat. Le ulteriori richieste di accesso ai dati georeferenziati vengono effettuate direttamente a partire dal database PostgreSQL sulla porta 5432. Le risposte alle richieste seguono il flusso inverso dei dati, fin qui descritto. La risposta finale all'utente viene inoltrata sempre adottando il protocollo HTTP.

Ovviamente tutti i pacchetti software devono venire correttamente installati e configurati risolvendo tutte le problematiche di sicurezza inerenti ad ogni singolo software. Tutte le relazioni tra i software devono venire impostate a mano configurando correttamente anche tutte le credenziali di accesso (indirizzo IP, numero di porta username, password), per esempio impostando dentro GeoServer dove si trova il database con i dati geografici ecc.

Importazione degli shapefiles

I dati geografici (strade, civici, confini amministrativi, centri abitati, uso del territorio, ferrovie e corsi d'acqua) sono stati acquistati da TeleAtlas in formato shapefile, uno per ciascuno dei diversi livelli geografici di interesse.

I singoli shapefile sono stati convertiti, attraverso un opportuno tool software chiamato `shp2pgsql`, in formato sql, pronti per essere importati direttamente da PostgreSQL, come riportato in figura.

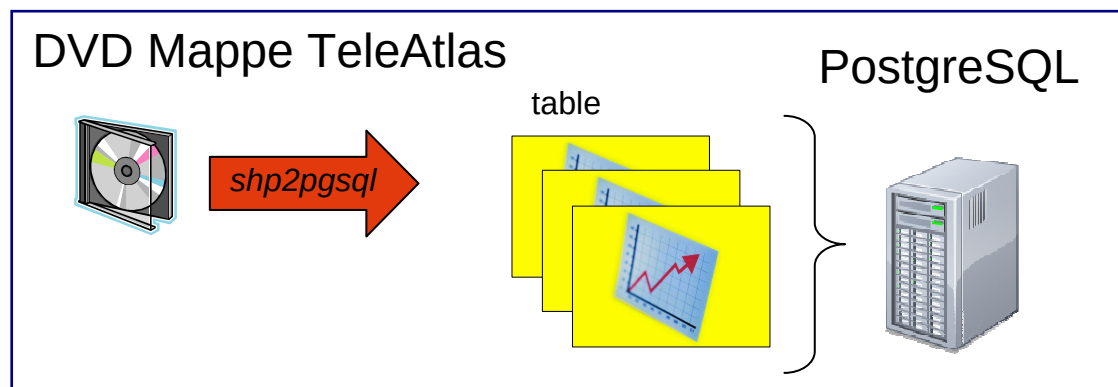


Fig. 53 - Da shapefile a formato sql

La sintassi con cui viene eseguita la conversione è:
`shp2pgsql -s 4326 -W UTF8 '$input.' > '$output.'.sql'.`

Sintassi

`shp2pgsql [<opzioni>] <shapefile> [<schema>.]<tabella>`

Opzioni

`-s <srid>` Definisce il campo *SRID*. Se non è specificato, viene automaticamente definito `-1`.

(-d|a|c|p) Queste sono opzioni che si escludono reciprocamente:

- d** Cancella la tabella, quindi la ricrea e la popola con i dati del file shape corrente.
- a** Accoda il file shape nella tabella corrente, che deve essere esattamente la stessa tabella dello schema.
- c** Crea una nuova tabella e la popola (questa è l'azione predefinita se non viene specificata alcuna opzione).
- p** Modalità di preparazione, viene solamente creata la tabella.
- g** **<geometry_column>** Specifica il nome della colonna delle geometrie (usato principalmente in modalità di accodamento).
- D** Usa il formato dei dump di postgresql (se non viene specificato nulla vengono impostate le istruzioni di inserimento sql).
- k** Keep postgresql identifiers case.
- i** Usa il tipo *int4* per tutti i campi del file *dbf* con numeri interi.
- I** Create a GiST index on the geometry column.
- w** Usa il formato wkt (for postgis-0.x support - drops M - drifts coordinates).
- S** (Dalla versione 1.1.5) generare geometrie *S*ingole anziche' multiple (linee e polygons).

Per ulteriori informazioni fare riferimento a <http://wiki.gfoss.it/index.php/Shp2pgsql>.

Per ogni provincia del Piemonte sono stati elaborati i seguenti livelli geografici:

- A8: comuni
- Bu: centri abitati
- Gc: nomi strade
- Jc: interconnessione rami stradali
- Lc: territorio (monti, spiagge)
- Nw: grafo stradale
- Pi: points of interest
- Rr: ferrovie
- Wa: corsi d'acqua

in modo da ottenere una descrizione del territorio il più accurata possibile.

Ogni singolo shapefile convertito viene importato in una corrispondente tabella in PostgreSQL come riportato in Fig. 54

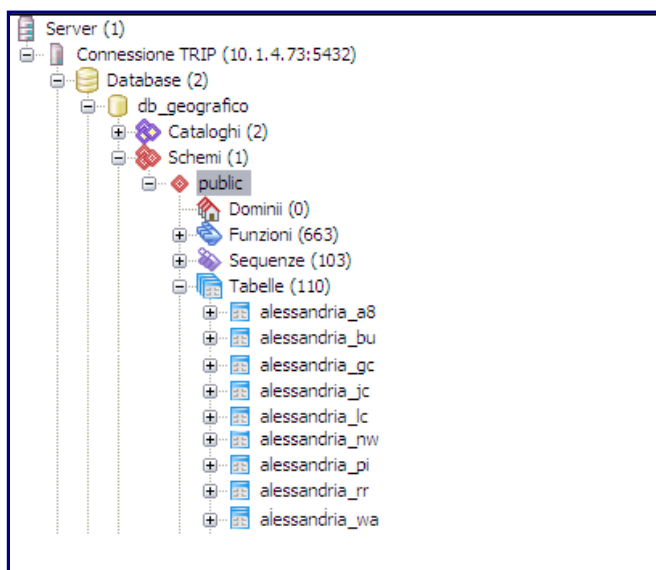


Fig. 54 - Esempio di tabelle contenute nel DB PostgreSQL

Configurazione GeoServer

GeoServer è il cuore del sottosistema GIS, ed è un software open source che permette la pubblicazione e l'editing di dati geospaziali utilizzando standard di comunicazione aperti (WFS-T, WMS e altri). Per la sua installazione e descrizione del funzionamento si faccia riferimento al sito geoserver.org ed alla opportuna documentazione. In questo paragrafo verranno illustrate soltanto alcune configurazioni particolari che sono state adottate per il Progetto TRIP.

In Fig. 55 è illustrato il meccanismo base delle richieste di dati georeferenziati, gestite mediante GeoServer, attraverso interrogazione al database con dati geografici PostgreSQL

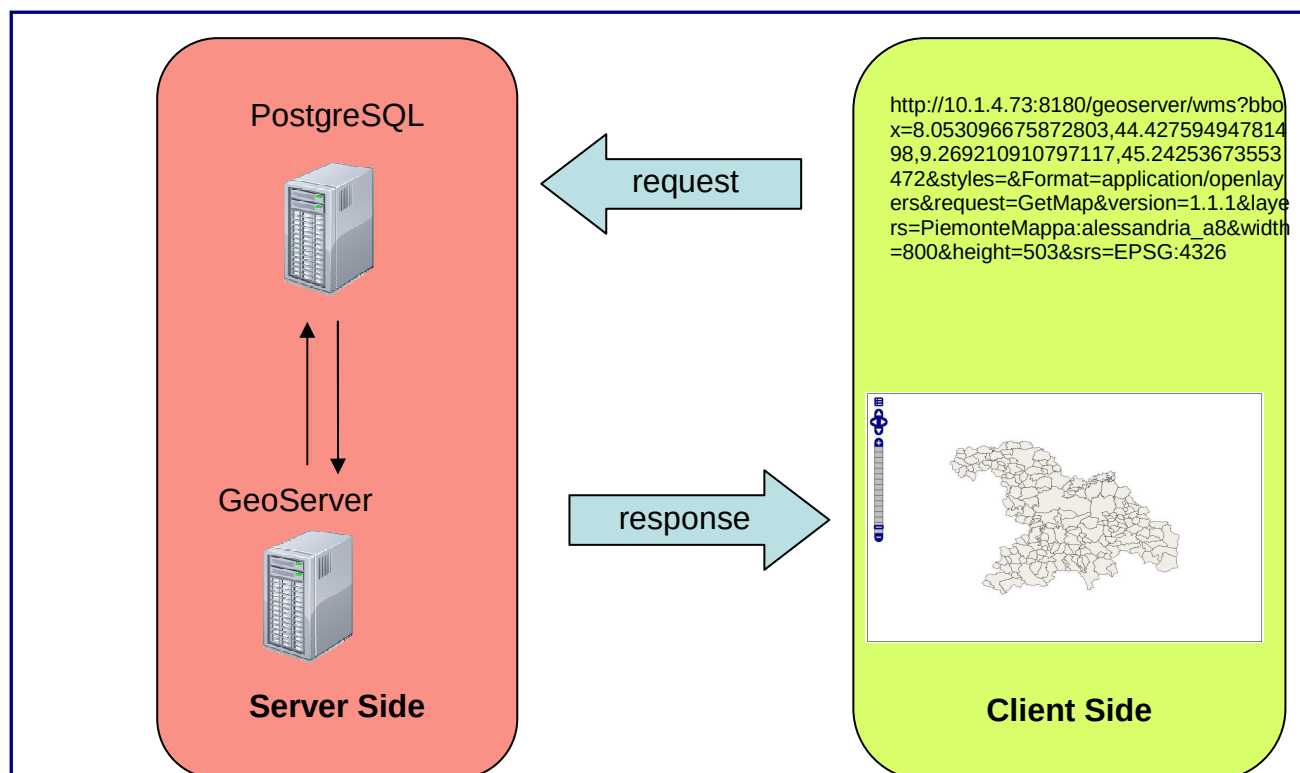


Fig. 55 - Richiesta di dati georeferenziati

Questo scambio di dati è possibile grazie alla configurazione, all'interno di Geoserver di un opportuno Datastore (Welcome → Config → Data → DataStores → Edit) che si occupa di recuperare i dati dal database. Una schermata di esempio in cui si effettua la configurazione è riportata di seguito.

Welcome | Config | Data | DataStores | Edit Logout

Feature Data Set Editor

Edit a source of spatial information

Feature Data Set ID: DBgeografico

Enabled: ☒

Namespace: PiemonteMappa ▼

Description:

* host: localhost

* port: 5432

schema: public

* database: db_geografico

* user: admin

passwd:

max connections: 10

min connections: 4

validate connections: false ▼

wkb enabled: true ▼

loose bbox: true ▼

estimated extent: false ▼

* = required field

Fig. 56 – Configurazione di GeoServer

Come è possibile notare, i parametri fondamentali che devono essere inseriti sono:

- Namespace
- Description
- host:
- port:
- schema:
- database:
- user:
- passwd:

e fanno ovviamente riferimento al server in cui è installato PostgreSQL e su cui si trovano i dati georeferenziati

Per la creazione di un layer in cui vengono rappresentati i dati di una tabella PostgreSQL, relativa ad un singolo shapefile, è necessario creare un FeatureType (Welcome → Config → Data → FeatureTypes → Edit), in cui vengono fornite informazioni per la visualizzazione del layer (stile e colori), come riportato nella schermata seguente.

My GeoServer

Credits Contact: Claudius Ptolemaeus

Welcome | Config | Data | FeatureTypes | Edit Logout

FeatureType Editor

Edit Feature Type definition and schema

Schema Base:

Name:
 Alias:
 Style: [Create new SLD](#)
 Additional Styles:

Bosco
 Comune
 Edificato
 FERMATE
 Ferrovia
 Fiume
 PoL_Trip
 Strade

[>>](#) [<<](#)

SRS: [Lookup SRS](#) [SRS Help](#) - [SRS List](#)

SRS WKT: `GEOGCS["WGS 84", DATUM["World Geodetic System 1984", SPHEROID["WGS 84", 6378137.0, 298.257223563, AUTHORITY["EPSG","7030"]], AUTHORITY["EPSG","6326"]], PRIMEM["Greenwich", 0.0, AUTHORITY["EPSG","8901"]], UNIT["degree", 0.017453292519943295], AXIS["Geodetic longitude", EAST], AXIS["Geodetic latitude", NORTH], AUTHORITY["EPSG","4326"]]`

Native SRS WKT: `GEOGCS["WGS 84", DATUM["World Geodetic System 1984", SPHEROID["WGS 84", 6378137.0, 298.257223563, AUTHORITY["EPSG","7030"]], AUTHORITY["EPSG","6326"]], PRIMEM["Greenwich", 0.0, AUTHORITY["EPSG","8901"]], UNIT["degree", 0.017453292519943295], AXIS["Geodetic longitude", EAST], AXIS["Geodetic latitude", NORTH], AUTHORITY["EPSG","4326"]]`

SRS handling:

Title:
 Bounding Box: [Generate](#)

Data min X:	8.10837459564209	Data min Y:	44.4646377563477
Data max X:	9.21393299102783	Data max Y:	45.205493927002

Min Long: Min Lat:
 Max Long: Max Lat:

Keywords:
 Abstract:
 WMS Path:

Type	Format	URL
FGDC	text/plain	
FGDC	text/plain	

Caching Enabled?: ☐
 Cache Time:
 GeoSearch indexing Enabled?: ☐
 KML Regionating Attribute:
 KML Regionating Strategy:
 KML Feature Limit:
 Title Attribute: * Changes to this setting will not take effect until you save.
 Maximum Features:

-- [Change](#)

id:	Long	nillable:false	min:0	max:1
feattyp:	Short	nillable:false	min:0	max:1
order00:	String	nillable:false	min:0	max:1
order01:	String	nillable:false	min:0	max:1
order02:	String	nillable:false	min:0	max:1
order03:	String	nillable:false	min:0	max:1
order04:	String	nillable:false	min:0	max:1
order05:	String	nillable:false	min:0	max:1
order06:	String	nillable:false	min:0	max:1
order07:	String	nillable:false	min:0	max:1
order08:	String	nillable:false	min:0	max:1
featarea:	Long	nillable:false	min:0	max:1
featperim:	Long	nillable:false	min:0	max:1
name:	String	nillable:false	min:0	max:1
nameI:	String	nillable:false	min:0	max:1
popclass:	Short	nillable:false	min:0	max:1
pop:	Long	nillable:false	min:0	max:1
citycenter:	Long	nillable:false	min:0	max:1
the_geom:	MultiPolygon	nillable:false	min:0	max:1

[Submit](#) [Reset](#)

Fig. 57 - Informazioni per la visualizzazione dei layer

Tutte le informazioni per lo stile da applicare ad ogni singolo layer vengono inserite e modificate con lo Style Editor (Welcome → Config → Data → Style → Edit), che consiste nell'editare un semplice file di testo XML, specificando le direttive sui colori e gli stili da applicare. Per la corretta sintassi della stringhe dei file SLD (Style Layer Descriptor) si faccia riferimento alla documentazione ufficiale.

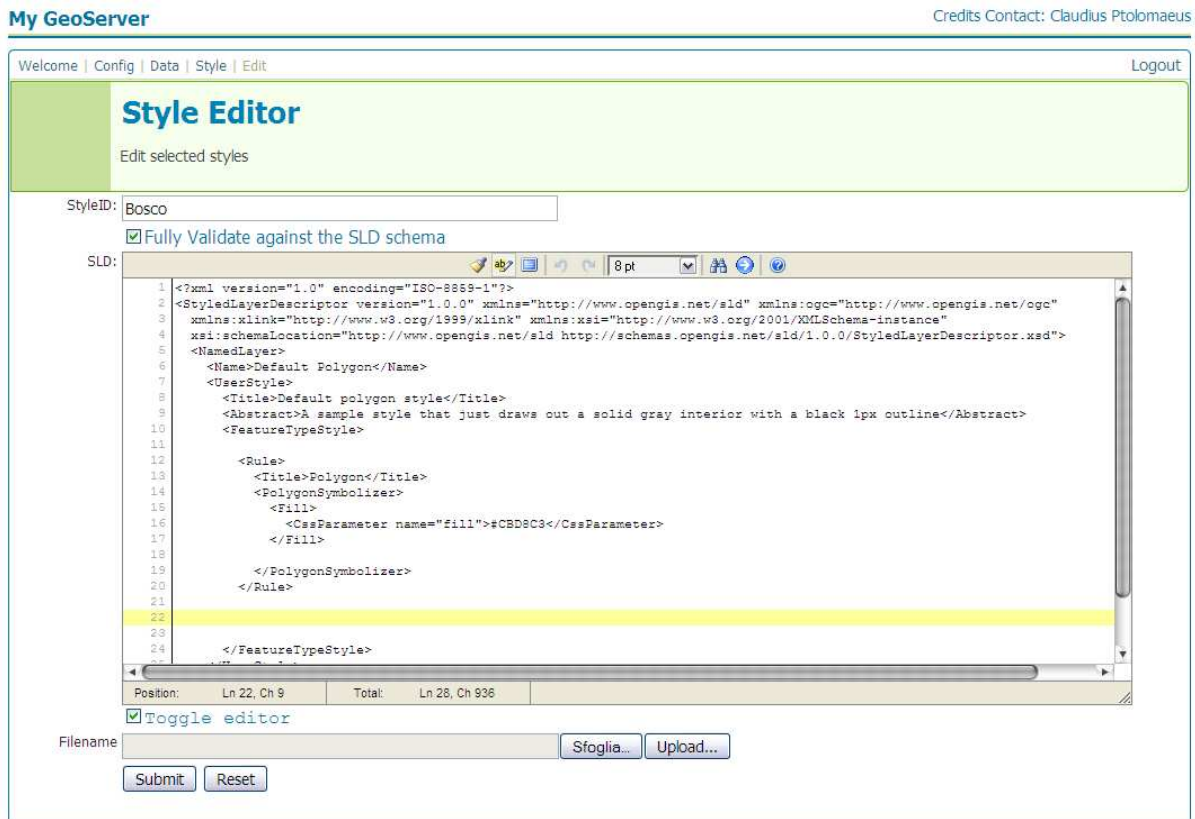


Fig. 58 - GeoServer Style Editor

Come esempio viene riportato lo SLD dell'edificato urbano:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.0.0"          xmlns="http://www.opengis.net/sld"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/sld
http://schemas.opengis.net/sld/1.0.0/StyledLayerDescriptor.xsd">
  <NamedLayer>
    <Name>Default Polygon</Name>
    <UserStyle>
      <Title>Default polygon style</Title>
      <Abstract>A sample style that just draws out a solid gray interior with a black 1px
outline</Abstract>
      <FeatureTypeStyle>
        <Rule>
```

```

    <Title>Polygon</Title>
    <PolygonSymbolizer>
      <Fill>
        <CssParameter name="fill">#EDEAE2</CssParameter>
      </Fill>
    </PolygonSymbolizer>
    <TextSymbolizer>
      <Label>
        <ogc:PropertyName>name</ogc:PropertyName>
      </Label>
      <Font>
        <CssParameter name="font-family">Tahoma</CssParameter>
        <CssParameter name="font-style">Normal</CssParameter>
        <CssParameter name="font-size">20</CssParameter>
      </Font>
      <LabelPlacement>
        <LinePlacement>
          <PerpendicularOffset>0</PerpendicularOffset>
        </LinePlacement>
      </LabelPlacement>
      <Halo>
        <Radius>2</Radius>
      </Halo>
      <Fill>
        <CssParameter name="fill">#FFFFFF</CssParameter>
        <CssParameter name="fill-opacity">0.80</CssParameter>
      </Fill>
      <Halo>
        <Fill>
          <CssParameter name="fill">#000000</CssParameter>
        </Fill>
        <VendorOption name="group">true</VendorOption>
        <VendorOption name="followLine">true</VendorOption>
        <VendorOption name="maxDisplacement">50</VendorOption>
        <VendorOption name="repeat">300</VendorOption>
      </Halo>
    </TextSymbolizer>
  </Rule>
</FeatureTypeStyle>
</UserStyle>
</NamedLayer>
</StyledLayerDescriptor>

```

Infine per creare una mappa generale di tutta la regione Piemonte, in cui vengano riportati tutti i livelli geografici per ogni provincia, è necessario creare un WMS (Web Map Server Content) in cui vengano elencate tutti i layer geografici provincia per provincia, separati da virgola:

PiemonteMappa:alessandria_a8,PiemonteMappa:asti_a8,PiemonteMappa:biella_a8,PiemonteMappa:cuneo_a8,PiemonteMappa:novara_a8,PiemonteMappa:torino_a8,PiemonteMappa:vercelli_a8,PiemonteMappa:verbano_cusio_ossola_a8,PiemonteMappa:alessandria_bu,

PiemonteMappa:asti_bu,PiemonteMappa:biella_bu,PiemonteMappa:cuneo_bu,PiemonteMappa:novara_bu,PiemonteMappa:torino_bu,PiemonteMappa:vercelli_bu,PiemonteMappa:verbano_cusio_ossola_bu,PiemonteMappa:alessandria_lc,PiemonteMappa:asti_lc,PiemonteMappa:biella_lc,PiemonteMappa:cuneo_lc,PiemonteMappa:novara_lc,PiemonteMappa:torino_lc,PiemonteMappa:vercelli_lc,PiemonteMappa:verbano_cusio_ossola_lc,PiemonteMappa:alessandria_wa,PiemonteMappa:asti_wa,PiemonteMappa:biella_wa,PiemonteMappa:cuneo_wa,PiemonteMappa:novara_wa,PiemonteMappa:torino_wa,PiemonteMappa:vercelli_wa,PiemonteMappa:verbano_cusio_ossola_wa,PiemonteMappa:alessandria_rr,PiemonteMappa:asti_rr,PiemonteMappa:biella_rr,PiemonteMappa:cuneo_rr,PiemonteMappa:novara_rr,PiemonteMappa:torino_rr,PiemonteMappa:vercelli_rr,PiemonteMappa:verbano_cusio_ossola_rr,PiemonteMappa:alessandria_nw,PiemonteMappa:asti_nw,PiemonteMappa:biella_nw,PiemonteMappa:cuneo_nw,PiemonteMappa:vercelli_nw,PiemonteMappa:verbano_cusio_ossola_nw,PiemonteMappa:novara_nw,PiemonteMappa:torino_nw;

così come riportato nella schermata di esempio in Fig. 59

My GeoServer

Credits Contact: Claudius Ptolomaeus

Welcome | Config | WMS | Contents Logout

Web Map Server Contents

Configuration of Web Map Server functionality

Enabled: ☒

Online Resource:

Limited Capabilities CRS List:

Base Maps:

Layer-group Name: Remove

Base Map Layers:

Base Map Styles:

SRS: [SRS Help](#) - [SRS List](#)

Envelope: Generate

Min X: Min Y:

Max X: Max Y:

Submit
Reset
Add New Layer-Group

Fig. 59 - Gestione dei layers

Il server GIS GeoServer quindi sovrappone tutti i livelli geografici fino a creare la mappa navigabile dell'intero Piemonte. Il procedimento è illustrato in Fig. 60.

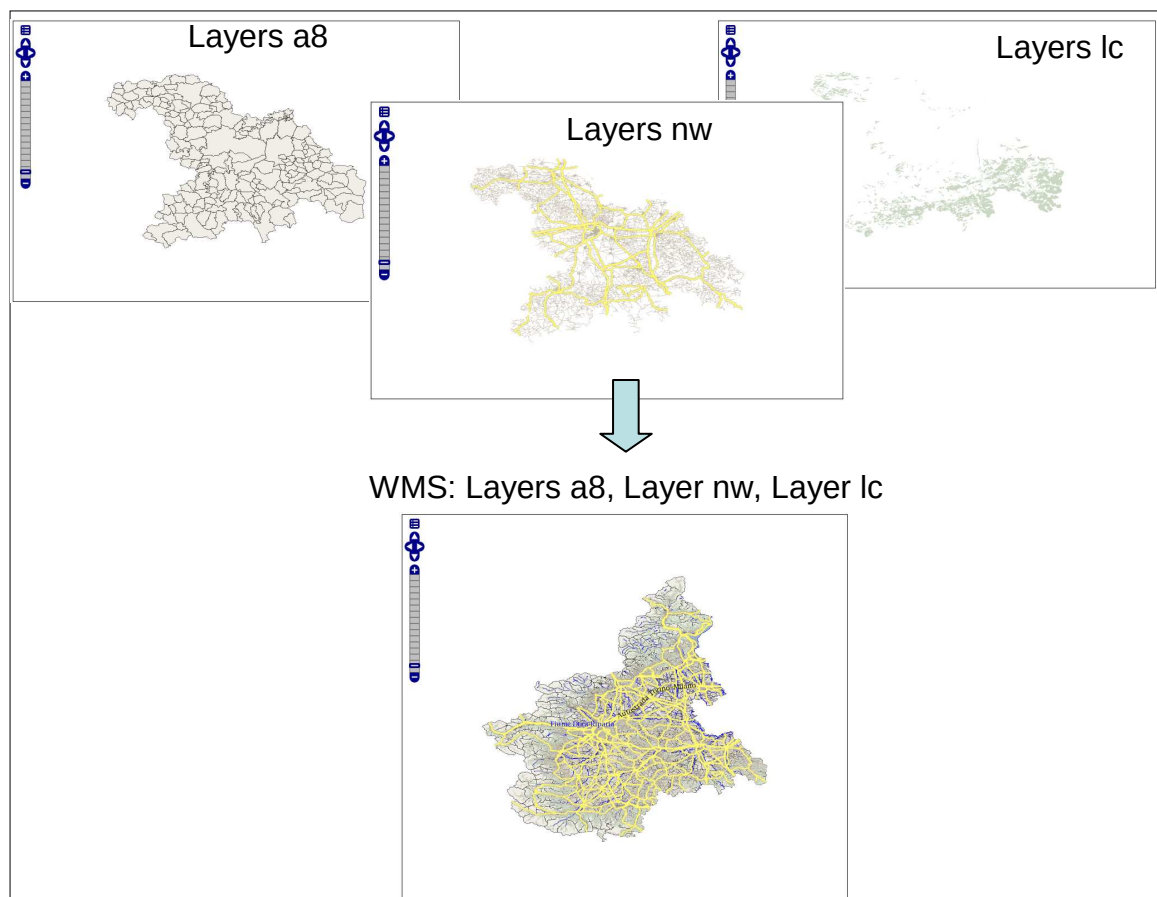


Fig. 60 - Unione di diversi layer a formare una mappa completa

Livello Punti di Interesse (PoI)

Un livello geografico che viene sovrapposto alla mappa completa del Piemonte è quella dei Punti di Interesse, anch'essi ricavati dallo shapefile fornito da TeleAtlas. Applicando un opportuno SLD (Style Layer Descriptor) è possibile renderizzare una diversa icona per ciascuna tipologia di punto di interesse, per esempio per riconoscere immediatamente e intuitivamente i cinema, i ristoranti, i parcheggi ecc. In Fig. 61 e Fig. 62 seguenti è rappresentata uno zoom della mappa del Piemonte su Torino, con e senza la visualizzazione dei punti di interesse.

Le tipologie di punti di interesse trattati sono:

- Bank
- Camping Ground
- Camping Site Ground
- Car Dealer
- Car Repair Facility
- Cash Dispenser
- Cinema
- Dentist
- Doctor
- Hospital/Polyclinic
- Hotel or Motel

- Hotel or Motel Ground
- Library
- Library Ground
- Museum
- Museum Ground
- Nightlife
- Open Parking Area
- Petrol Station
- Petrol Station Ground
- Pharmacy
- Police Office Ground
- Police Station
- Post Office
- Post Office Ground
- Postal District
- Railway Station
- Railway Station Ground
- Rent-a-Car Facility
- Rent-a-Car Parking
- Restaurant
- Restaurant Area
- Tennis Court
- Theatre
- Theatre Ground
- Tourist Information Office
- Veterinarian

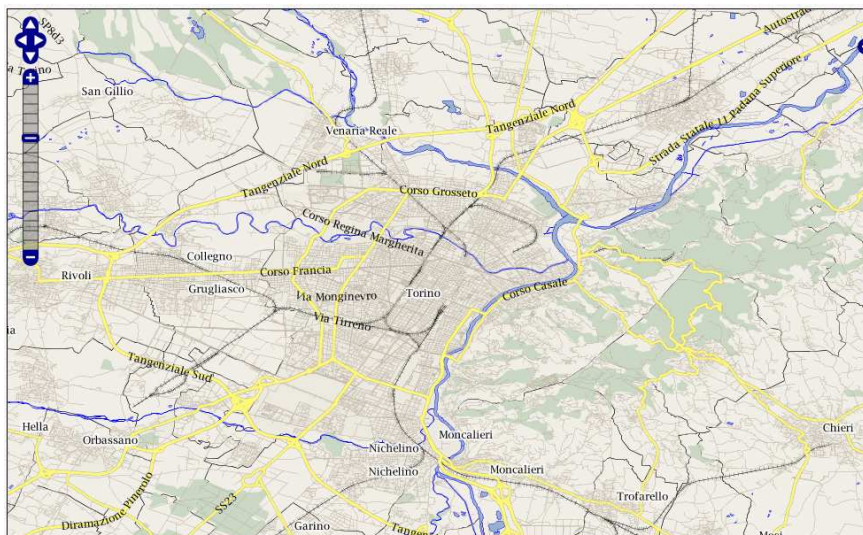


Fig. 61 - Esempio di mappa senza visualizzazione di Punti di Interesse

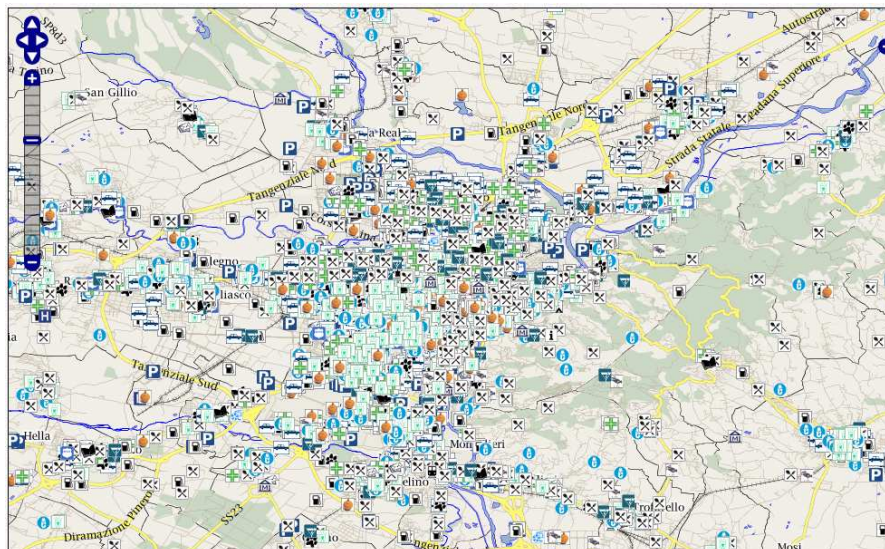


Fig. 62 - Esempio di mappa con visualizzazione di Punti di Interesse

Funzionalità GetFeatureInfo

Grazie alla funzionalità di GetFeatureInfo messo a disposizione da GeoServer è possibile, dopo aver cliccato su uno qualsiasi dei punti di interesse ottenere le informazioni relative, come indirizzo e numeri di telefono. Con questa funzionalità si riesce a creare un meccanismo che in tempo reale, sulla effettiva richiesta dell'utente, va a recuperare i dati nel database geografico e provvede a pubblicarli in una nuvoletta (personalizzabile).

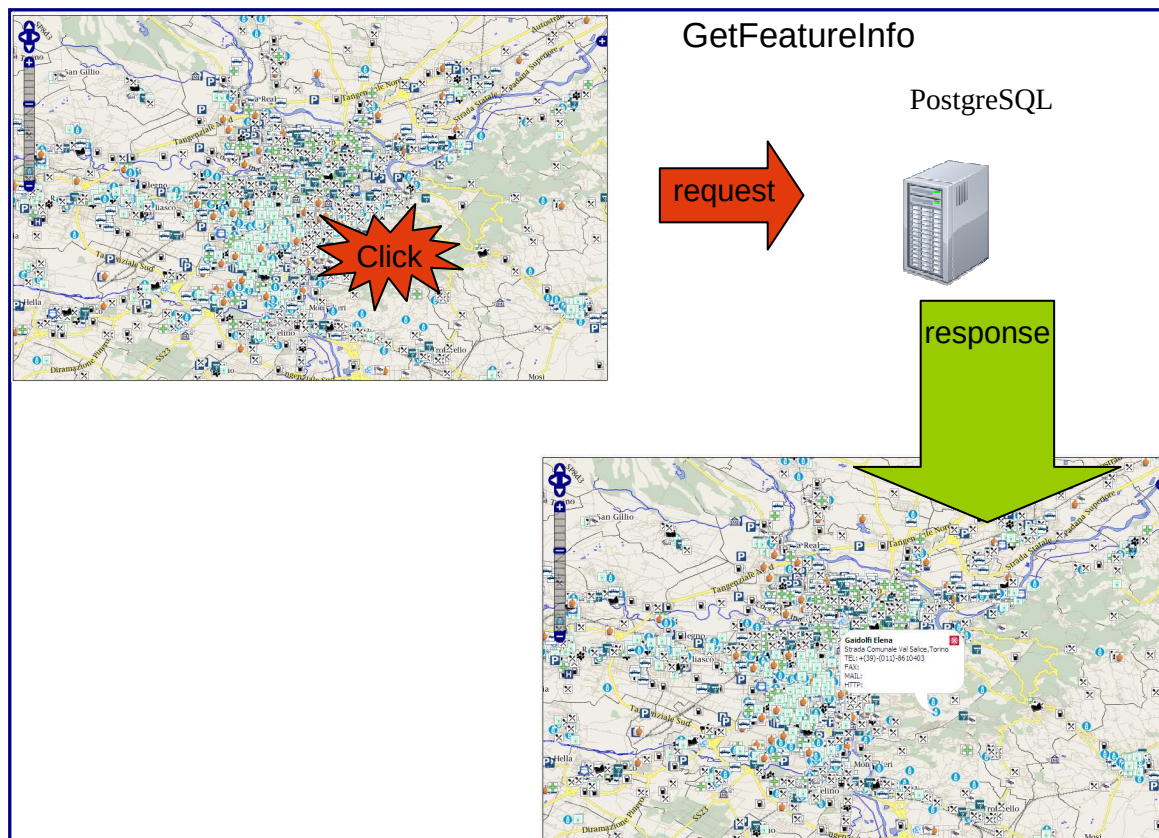


Fig. 63 - Funzione GetFeatureInfo

Configurazione GeoWebCache

Per rendere più veloci ed efficienti le richieste delle mappe, che sono molto onerose in termini di carico computazionale e di accesso al database se devono essere calcolate runtime per ogni utente, è stato inserito un middleware di tipo software in grado di effettuare un caching delle mappe.

Dividendo infatti le singole immagini in piccole mattonelle o 'tile' e archiviandole opportunamente, il software è in grado di velocizzare il rendering delle mappe nelle diverse funzionalità che le richiedono all'interno del Portale. Il software ottimale è stato individuato in GeoWebCache, che risulta, nelle ultime versioni di GeoServer, già installato nel pacchetto base.

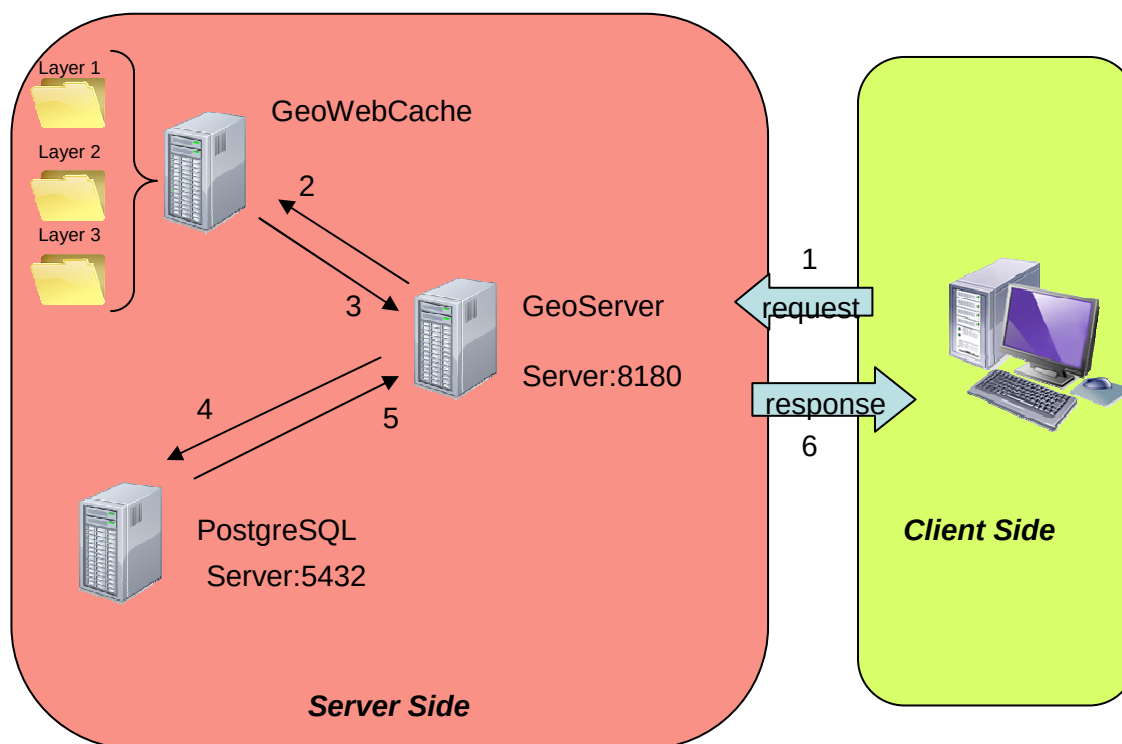


Fig. 64 - Integrazione del sistema con GeoWebCache

Grazie a questo potente software è possibile far sì che le richieste delle mappe, fatte dall'utente attraverso il portale, vengano intercettate da GeoServer che prima di provvedere alla creazione e al rendering della stesse, controlla attraverso GeoWebCache se ne esiste già una versione memorizzata in cache. In questo caso la mappa viene prontamente fornita all'utente, senza che debba attendere la creazione ex novo della mappa. Il procedimento è illustrato in Fig. 64.

Attraverso un apposito portale di amministrazione è possibile controllare quali tra tutti i livelli geografici, presenti sul server GIS, sono cachati e in che percentuale. E' ovviamente possibile forzare il caching totale di tutto un livello o di un intero WMS (come la mappa completa del Piemonte), cancellarlo e generare solo i 'tile' mancanti. è riportato una

schermata di esempio di un elenco di livelli all'interno del portale di amministrazione del tool GeoWebCache.



Reload Configuration:

You can reload the configuration by pressing the following button. The username / password is configured in WEB-INF/user.properties, or the admin user in GeoServer if you are using the plugin.

[Reload Configuration](#)

Known Layers:

- This is just a quick demo, the bounds are likely to be less than perfect.
- You can append &format=image/jpeg to the URLs in the table to change the output format.

Layer name:	OpenLayers:	Google Earth:	Custom:
PiemonteMappa:cuneo_wa	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:novara_lc	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:vercelli_nw	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:alessandria_a8	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:novara_bu	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:torino_a8	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:novara_pi	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:verbano_cusio_ossola_lc	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:vercelli_wa	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:cuneo_nw	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:biella_bu	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:verbano_cusio_ossola_wa	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:verbano_cusio_ossola_pi	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		
PiemonteMappa:biella_rr	EPSG:4326 EPSG:900913 KML (PNG) KML (vector) <i>none</i> Seed this layer		

Fig. 65 - Elenco di livelli all'interno del portale di amministrazione del tool GeoWebCache

Come anticipato attraverso questo portale è possibile lanciare un processo di caching, che crea tutte le 'tile' che compongono la mappa a tutti i livelli di zoom richiesti, affinché vengano memorizzate nel server direttamente come immagini png8, png o jpeg. Il processo è quello illustrato in Fig. 66. Attraverso il comando id caching (1), il server lancia il processo che attraverso una serie di query al database geografico, recupera tutte le informazioni, effettua il rendering della 'tile' e la archivia opportunamente in cartelle diverse a seconda del livello di zoom.

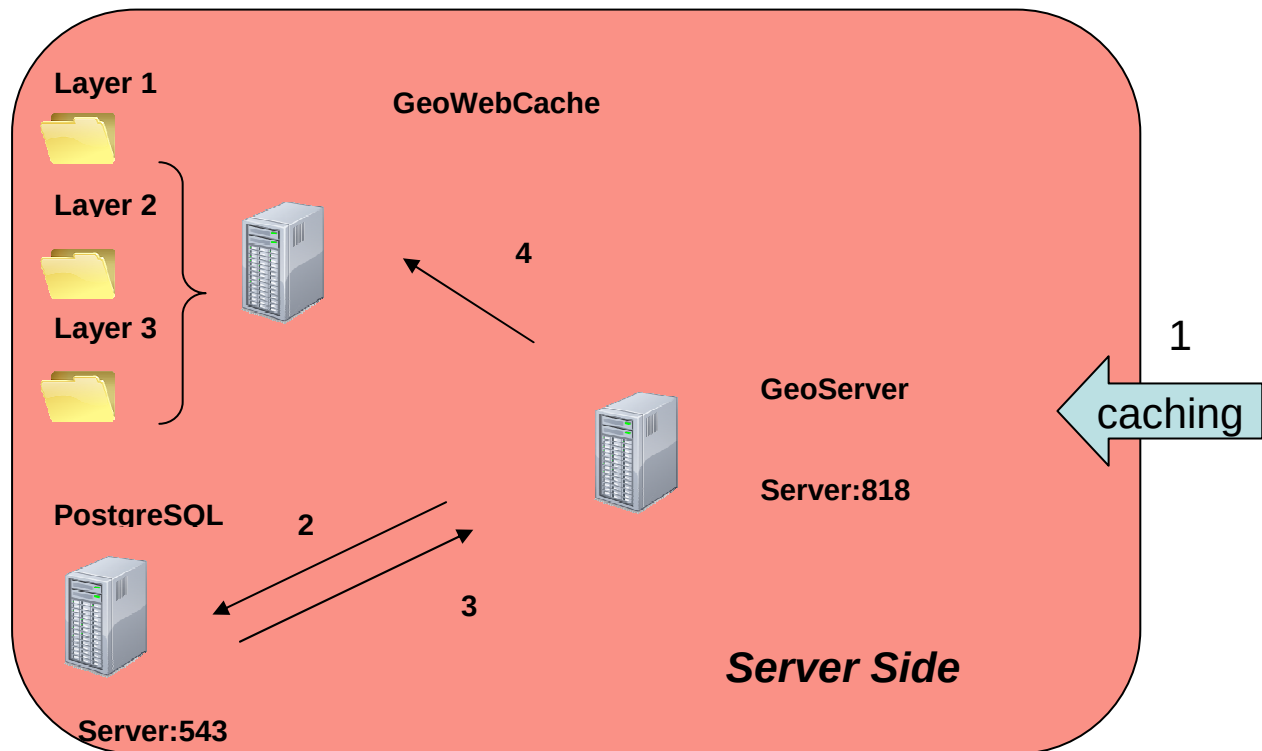


Fig. 66 - Fase di caching

Di seguito è riportata la lista delle opzioni per la fase di caching, in cui viene richiesto di inserire:

- Number of threads to use: il numero di processi lato server da attivare
- Type of operation: il tipo di operazione (cancellare, completare, creare)
- SRS: Spatial referencing systems o sistema spaziale di riferimento
- Format: il formato delle immagini (png, png8, jpg)
- Zoom start: il livello di zoom di partenza
- Zoom stop: il livello di zoom di arrivo
- Bounding box: le coordinate del rettangolo che contiene l'area della mappa da cachare.

**List of currently executing tasks:**

- *none*

[Refresh list](#)**Please note:**

- This minimalistic interface does not check for correctness.
- Seeding past zoomlevel 20 is usually not recommended.
- Truncating KML will also truncate all KMZ archives.
- Please check the logs of the container to look for error messages and progress indicators.

Here are the max bounds, if you do not specify bounds these will be used.

- EPSG:900913: 881215.1621232651,5743057.782850856,969709.9723643009,5854368.397702601
- EPSG:4326: 7.91609048843384,45.7668037414551,8.71105289459229,46.4599685668945

Create a new task:

Number of threads to use:	01
Type of operation:	Reseed - regenerate all tiles
SRS:	EPSG:4326
Format:	image/png
Zoom start:	00
Zoom stop:	15
Bounding box:	

These are optional, approximate values are fine.

Database Geografico: PostgreSQL e PostGIS

Come indicato nell'architettura del sistema, i principali gruppi di informazioni sono: mappe, informazioni su autobus, eventuali altri mezzi pubblici o punti di interesse, dati in tempo reale sul traffico, utenti, loro dati e percorsi loro suggeriti nel tempo.

I dati relativi agli utenti sono tenuti su un server MySQL, quelli geografici su un server PostgreSQL. Per quanto riguarda la parte geografica, per ogni provincia ci sono nove tabelle di descrizione morfologica (layer), come gli incroci e gli archi stradali, i punti di interesse (POI), le ferrovie, i corsi d'acqua, ecc.

Per ogni provincia i livelli di interesse sono:

- A8: comuni
- Bu: centri abitati
- Gc: nomi strade
- Jc: interconnessione rami stradali
- Lc: territorio (monti, spiagge)
- Nw: grafo stradale
- Pi: points of interest
- Rr: ferrovie
- Wa: corsi d'acqua

Di seguito è riportata la schermata del pannello di amministrazione di PGAdmin, lo strumento di amministrazione del database PostgreSQL. In evidenza, per la provincia di Alessandria, le nove tabelle importate direttamente dalle mappe TeleAtlas, secondo la procedura riportata nei precedenti capitoli.

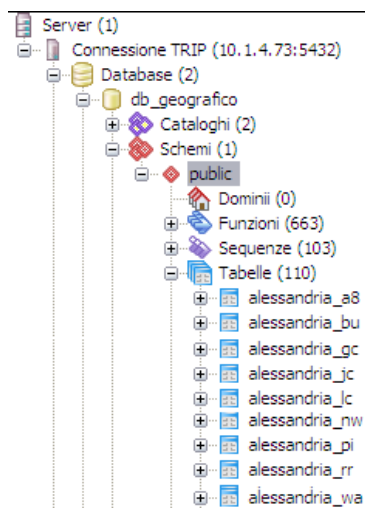


Fig. 67 - Tabelle nel DB PostgreSQL per la provincia di Alessandria

Oltre a questi dati che risultano ad una prima analisi palesemente georeferenziati, nel database PostgreSQL sono memorizzate anche altre informazioni come le fermate e i percorsi complessivi dei mezzi pubblici e tutti i tratti intermedi da fermata a fermata. Anche tutti questi dati presentano caratteristiche tali da far ritenere opportuno la memorizzazione in un database con estensioni spaziali. Le fermate sono quindi memorizzate in un tabella, in cui vengono rappresentate dalle loro coordinate GPS a tutte le informazioni sul passaggio delle linee.

Ci sono anche tabelle che memorizzano le linee per ogni fermata, le linee, i percorsi fermata-fermata, le tratte ed altre tabelle per i parcheggi, ecc.

In dettaglio sono necessarie tutta una serie di tabelle per memorizzare:

- descrizione fermata
- descrizione linee
- linee che passano alle diverse fermate
- tratti di strada percorsi dalla line
- tratti pedonali da fermata a fermata

La schermata di PGAdmin con le tabelle in evidenza sono riportate di seguito:

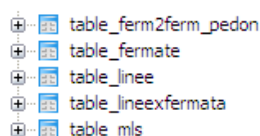


Fig. 68 - Tabelle relative a percorsi di mezzi e percorsi pedonali

Le relazioni tra queste tabelle sono riassunte nel grafico sottostante in cui si evidenziano i campi comuni delle tabelle, che fungono quindi da chiave per la ricerca delle linee o fermate. Ovviamente per ogni linea esistono una serie di fermate che appartengono alla linea, ed è ugualmente possibile che linee diverse condividano alcune fermate.

Come già anticipato vengono memorizzati anche tutti gli oggetti geometrici che rappresentano gli archi di strada in cui un singolo mezzo transita, nel suo percorso da fermata a fermata.

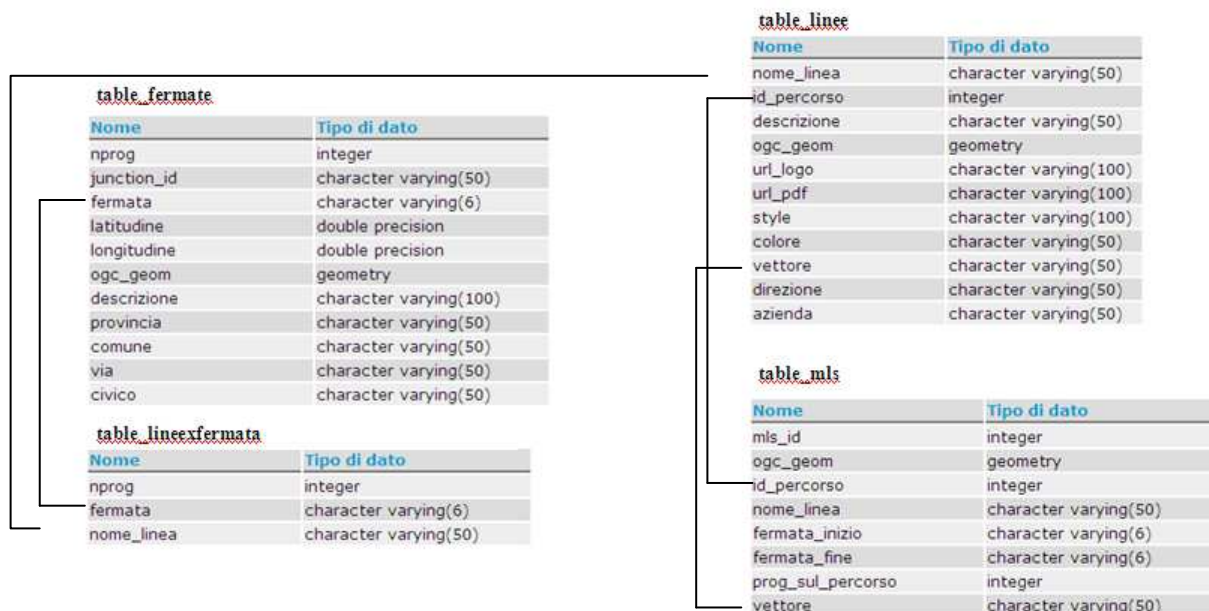


Fig. 69 - Campi delle tabelle relative ai percorsi e linee

Nello schema riportato in alto è anche possibile leggere il tipo di dato associato a ciascun campo della tabell. E' da notare che gli oggetti geometrici che memorizzano le informazioni spaziali hanno un tipo di dato indicato come 'geometry'.

Database Generico: MySQL

Sul server MySQL sono memorizzate tutte le informazioni sugli utenti e le loro richieste e sugli eventi. Gli eventi in generale sono intesi come ritardi, eventi significativi o eventi pendenti. Nella schermata seguente è possibile vedere, attraverso il tool di amministrazione del database MySQL, l'elenco di tutte le tabelle presenti.

Tabella	Azione	Record	Tipo	Collation	Dimensione	In eccesso
☐ table_admin		2	InnoDB	latin1_swedish_ci	16,0 KiB	-
☐ table_chiamate_motore		13	MyISAM	latin1_bin	21,8 KiB	-
☐ table_comuni_piemonte		1.206	MyISAM	latin1_swedish_ci	44,9 KiB	-
☐ table_corse		1.506	MyISAM	latin1_swedish_ci	175,9 KiB	-
☐ table_date_richieste		81	MyISAM	latin1_swedish_ci	5,2 KiB	-
☐ table_eventi_pendenti		0	MyISAM	latin1_bin	1,2 KiB	248 B
☐ table_eventi_ritardo		134	MyISAM	latin1_bin	23,6 KiB	-
☐ table_eventi_significativi		78	MyISAM	latin1_bin	3,6 KiB	-
☐ table_feattyp TO REMOVE		246	MyISAM	latin1_swedish_ci	12,8 KiB	-
☐ table_province_piemonte		8	MyISAM	latin1_swedish_ci	1,2 KiB	-
☐ table_richieste		1.471	MyISAM	latin1_swedish_ci	71,1 KiB	-
☐ table_risposte_motore		13	MyISAM	latin1_bin	251,0 KiB	212 B
☐ table_session_dashboard		1.939	InnoDB	latin1_swedish_ci	464,0 KiB	-
☐ table_tempi_xml		1.076	MyISAM	latin1_bin	64,0 KiB	-
☐ table_utente		11	InnoDB	latin1_swedish_ci	16,0 KiB	-
15 tabella(e)	Totali	7.784	MyISAM	latin1_bin	1,1 MiB	460 B

Fig. 70 - Tabelle del DB MySQL

Senza entrare eccessivamente nel dettaglio, di seguito verranno fornite alcune informazioni sul contenuto e sul formato di alcune tabelle, in modo da cercare di rendere più evidente possibile il flusso dei dati e i meccanismi di memorizzazione e ricerca delle informazioni.

Nella tabella 'table corse' vengono memorizzati tutti i dati sulle corse come il capolinea di partenza e quello di destinazione, i giorni di circolazione, eventuali informazioni aggiuntive e gli orari di partenza e arrivo. Per tutti i campi, sempre osservando la schermata seguente è possibile ricavare il formato dei dati utilizzati.

phpMyAdmin

Database: db_TRIP_TO (15)

db_TRIP_TO (15)

- table_admin
- table_chiamate_motore
- table_comuni_piemonte
- table_corse**
- table_date_richieste
- table_eventi_pendenti
- table_eventi_significativi
- table_feattyp TO REMOVE
- table_province_piemonte
- table_richieste
- table_risposte_motore
- table_session_dashboard
- table_tempi_xml
- table_utente

Server: localhost Database: db_TRIP_TO Tabella: table_corse

Mostra Struttura SQL Cerca Inserisci Esporta Importa Operazioni Svuota Elimina

Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐ orario	varchar(50)	latin1_swedish_ci		No			
☐ corsa	int(10)			No			
☐ origine	varchar(50)	latin1_swedish_ci		No			
☐ destinazione	varchar(50)	latin1_swedish_ci		No			
☐ validoda	varchar(50)	latin1_swedish_ci		No			
☐ validoa	varchar(50)	latin1_swedish_ci		No			
☐ linea	varchar(50)	latin1_swedish_ci		No			
☐ circola	varchar(50)	latin1_swedish_ci		No			
☐ classifica	varchar(50)	latin1_swedish_ci		No			
☐ percorso	varchar(50)	latin1_swedish_ci		No			
☐ metri	varchar(50)	latin1_swedish_ci		No			
☐ orapartenza	int(11)			No			
☐ oraarrivo	int(11)			No			
☐ stato	varchar(50)	latin1_swedish_ci		No			
☐ tipoop	varchar(50)	latin1_swedish_ci		No			

Seleziona tutti / Deseleziona tutti Se selezionati:

Fig. 71 - Tabella Corse

Nella 'table evento ritardo' vengono gestite le informazioni sui ritardi, in particolare su quale corsa e quale mezzo è interessato dal ritardo. Ovviamente oltre a funzioni di routine questa tabella fornisce una ottima base per l'analisi statistica dei ritardi e per processi di data fusion successivi.

Le informazioni memorizzate in questa tabella servono inoltre a rendere funzionale e operativo tutto il meccanismo di ricalcolo del percorso e segnalazione all'utente dei percorsi interessati da ritardo o che necessitano di aggiornamento. Un campo sul tipo di evento discrimina le diverse tipologie di informazione associate, se si tratta per esempio di un ritardo previsto sulla prossima fermata o un ritardo consolidato su un passaggio ad una fermata. Per tutti i campi, sempre osservando la schermata seguente è possibile ricavare il formato dei dati utilizzati.

phpMyAdmin

Database: db_TRIP_TO (15)

db_TRIP_TO (15)

- table_admin
- table_chiamate_motore
- table_comuni_piemonte
- table_corse
- table_date_richieste
- table_eventi_pendenti
- table_eventi_ritardo**
- table_eventi_significativi
- table_feattyp TO REMOVE
- table_province_piemonte
- table_richieste
- table_risposte_motore
- table_session_dashboard
- table_tempi_xml
- table_utente

Server: localhost Database: db_TRIP_TO Tabella: table_eventi_ritardo

Mostra Struttura SQL Cerca Inserisci Esporta Importa Operazioni Svuota Elimina

Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐ id_evento	varchar(20)	latin1_bin		No			
☐ data_evento	varchar(20)	latin1_bin		No			
☐ ora_evento	varchar(20)	latin1_bin		No			
☐ orario	varchar(20)	latin1_bin		No			
☐ corsa	varchar(10)	latin1_bin		No			
☐ nome_linea	varchar(50)	latin1_bin		No			
☐ id_percorso	int(11)			No			
☐ originecorsa	varchar(6)	latin1_bin		No			
☐ destinazionecorsa	varchar(6)	latin1_bin		No			
☐ validoda	varchar(20)	latin1_bin		No			
☐ validoa	varchar(20)	latin1_bin		No			
☐ progressivofermata	int(5)			No			
☐ datainiziorcorsa	varchar(20)	latin1_bin		No			
☐ datapassaggio	varchar(20)	latin1_bin		No			
☐ secondipassaggio	int(10)			No			
☐ secondiritardo	int(10)			No			
☐ previsto_effettuato	varchar(1)	latin1_bin		No			
☐ tipoevento	varchar(1)	latin1_bin		No			
☐ indicatorepropagazione	varchar(1)	latin1_bin		No			
☐ valutato	int(11)			No	0		

Seleziona tutti / Deseleziona tutti Se selezionati:

Fig. 72 - Tabella Evento Ritardo

Nella tabella ‘table richieste’ vengono memorizzate tutte le richieste fatte dagli utenti, in modo tale da poter ricontrollare in caso di ritardo se è opportuno ripianificare il viaggio in quanto qualche evento ritardo va ad impattare pesantemente sul percorso suggerito ad un utente. Anche l’analisi di questa tabella con processi di data fusion permette di ottenere importanti informazioni sulle dinamiche dei percorsi: quali i punti di partenza/arrivo maggiormente utilizzati per potenziare il servizio o informazioni di tipo aggregato sulla modalità di utilizzo del servizio da parte degli utenti per capire quali sono i percorsi maggiormente richiesti. Per tutti i campi, sempre osservando la schermata seguente è possibile ricavare il formato dei dati utilizzati.



Server: localhost ▶ Database: db_TRIP_TO ▶ Tabella: table_richieste

Mostra Struttura SQL Cerca Inserisci Esporta Importa Operazioni Svuota Elimina

	Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐	ID	int(30)			No		auto_increment	
☐	user	varchar(30)	latin1_swedish_ci		No			
☐	Transiti_ID	int(30)			No			
☐	corsa	varchar(10)	latin1_swedish_ci		No			
☐	id_percorso	int(5)			No			
☐	numero_risposta	varchar(30)	latin1_swedish_ci		No			
☐	ID_richiesta	int(30)			No			
☐	percorso	varchar(3)	latin1_swedish_ci		Sì	NULL		
☐	influenza_evento	int(11)			No	0		

Seleziona tutti / Deseleziona tutti Se selezionati:

Fig. 73 - Tabella Richieste

La tabella ‘table session dashboard’ è il cuore dell’archiviazione dei punti di partenza e arrivo scelti dagli utenti per le loro richieste. Poiché le possibilità di fissare un punto di partenza/arrivo è molto ampia e l’utente può scegliere se partire da un indirizzo, una fermata, un punto di interesse o un punto sulla mappa, il sistema deve garantire la flessibilità necessaria a supportare il sistema. Vengono memorizzate sia le date in cui vengono effettuate le richieste sia le date per cui la richiesta viene effettuata, oltre ai dati sui mezzi e su tutte le preferenze espresse dall’utente. Per tutti i campi, sempre osservando la schermata seguente è possibile ricavare il formato dei dati utilizzati.

Analisi di tipo statistico possono portare importanti risultati anche per questa tabella in cui maggiormente sono archiviati i dati sensibili e le scelte degli utenti.

Server: localhost ▶ Database: db_TRIP_TO ▶ Tabella: table_session_dashboard "InnoDB free: 18432 kB"

Mostra Struttura SQL Cerca Inserisci Esporta Importa Operazioni Svuota Elimina

	Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐	Nprog	int(11)			No		auto_increment	
☐	SESSION_Identifier	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	user	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	IP	varchar(20)	latin1_swedish_ci		No			
☐	istante_richiesta	varchar(20)	latin1_swedish_ci		No			
☐	tipo_part	varchar(20)	latin1_swedish_ci		No			
☐	tipo_arr	varchar(20)	latin1_swedish_ci		No			
☐	ProvinciaFROM	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	ComuneFROM	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	ViaFROM	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	CivicoFROM	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	ProvinciaTO	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	ComuneTO	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	ViaTO	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	CivicoTO	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	FermataFROM	varchar(6)	latin1_swedish_ci		No			
☐	FermataTO	varchar(6)	latin1_swedish_ci		No			
☐	PoiFROM	varchar(50)	latin1_swedish_ci		No			
☐	PoiTO	varchar(50)	latin1_swedish_ci		No			
☐	year	varchar(4)	latin1_swedish_ci		Si	NULL		
☐	month	varchar(2)	latin1_swedish_ci		Si	NULL		
☐	day	varchar(2)	latin1_swedish_ci		Si	NULL		
☐	hour	varchar(2)	latin1_swedish_ci		Si	NULL		
☐	min	varchar(2)	latin1_swedish_ci		Si	NULL		
☐	modo	varchar(50)	latin1_swedish_ci		Si	NULL		
☐	azienda	varchar(6)	latin1_swedish_ci		Si	NULL		
☐	mezzo	varchar(10)	latin1_swedish_ci		Si	NULL		
☐	pedonale	varchar(1)	latin1_swedish_ci		No			
☐	cambi	varchar(10)	latin1_swedish_ci		Si	NULL		
☐	visibile_in_storico	int(1)			No			

Fig. 74 - Tabella Session dashboard

Infine la tabella 'table utente' memorizza le informazioni proprie dell'utente, la sua anagrafica, username e password (salvata cifrata per criteri di sicurezza) e il recapito telefonico a cui inviare le informazioni per l'aggiornamento dei percorsi in real-time.

Server: localhost ▶ Database: db_TRIP_TO ▶ Tabella: table_utente "InnoDB free: 18432 kB"

Mostra Struttura SQL Cerca Inserisci Esporta Importa Operazioni Svuota Elimina

	Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐	nome	varchar(30)	latin1_swedish_ci		No			
☐	cognome	varchar(30)	latin1_swedish_ci		No			
☐	indirizzo	varchar(30)	latin1_swedish_ci		No			
☐	civico	int(5)			No			
☐	cap	varchar(50)	latin1_swedish_ci		No			
☐	comune	varchar(30)	latin1_swedish_ci		No			
☐	provincia	varchar(30)	latin1_swedish_ci		No			
☐	eMail	varchar(40)	latin1_swedish_ci		No			
☐	cellulare	varchar(50)	latin1_swedish_ci		No			
☐	user	varchar(30)	latin1_swedish_ci		No			
☐	password	varchar(50)	latin1_swedish_ci		No			
☐	data_reg	varchar(10)	latin1_swedish_ci		No			

Seleziona tutti / Deseleziona tutti Se selezionati:

Fig. 75 - Tabella Utente

MODULO STIMA RITARDO

Grazie alla rete di telecomunicazione eterogenea pervasiva, ogni mezzo pubblico può trasmettere in tempo reale la propria posizione, grazie ad un apparato GPS installato a bordo del veicolo, ed un proprio identificativo (p. es la targa o l'identificativo della corsa che il mezzo sta effettuando).

Grazie a queste informazioni che arrivano in maniera continuativa e in real time al centro di controllo, grazie alla rete di telecomunicazione, il sistema può valutare e stimare l'eventuale ritardo accumulato del mezzo nell'esecuzione del proprio percorso.

Algoritmo per la Stima del Ritardo

E' da notare inoltre che per ogni singolo mezzo che trasmette la propria posizione è possibile ottenere a priori il percorso che il veicolo dovrà percorrere, e conoscere quindi il punto in cui si trova, più precisamente tra quali fermate si trova e quale sarà la fermata successiva, conoscendo ovviamente la lista delle fermate e il suo ordine progressivo.

Associando una percentuale che va dallo 0 al 100% all'intero percorso, è possibile calcolare la percentuale di percorso a cui si trovano le diverse fermate, come rappresentato in Fig. 76.



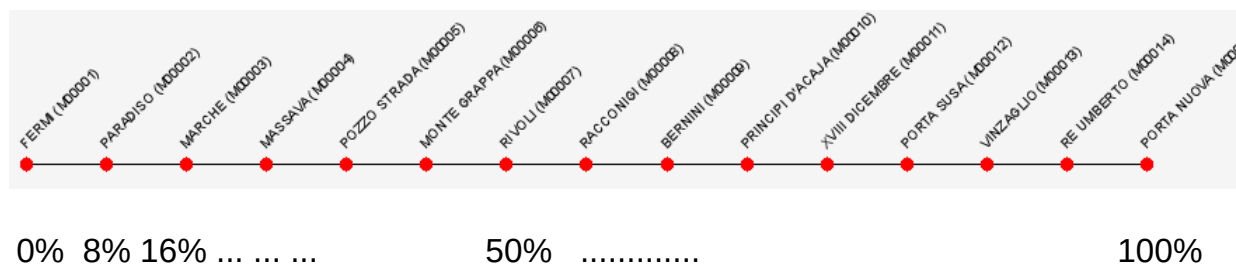


Fig. 76 – Fermate indicate come percentuale di un percorso

Questa percentuale può esser calcolata grazie ad una funzione spaziale messa a disposizione da PostGis per cui viene valutata la percentuale a cui si trova un determinato punto appartenente ad una spezzata. Fornendo ovviamente le coordinate GPS di tutte le fermate e l'intero percorso è possibile calcolare le percentuali che le singole fermate rappresentano sul percorso complessivo della linea.

La funzione PostGIS è:

```
float ST_Line_Locate_Point(geometry a_linestring, geometry a_point)
```

che accetta in ingresso l'oggetto geometrico che rappresenta il percorso della linea in esame *linea_percorso* e le coordinate GPS e fornisce la percentuale in cui si trova il punto rispetto all'intero percorso.

[Documentazione PostGis: Returns a float between 0 and 1 representing the location of the closest point on LineString to the given Point, as a fraction of total 2d line length. You can use the returned location to extract a Point (ST_Line_Interpolate_Point) or a substring (ST_Line_Substring). This is useful for approximating numbers of addresses]

Ovviamente anche per la posizione GPS comunicata dal mezzo al centro di controllo è possibile calcolare la percentuale e capire tra quali altre percentuali che fanno riferimento alle fermate si trova

Supponiamo per fare un esempio numerico che valgano le relazioni espresse in Fig. 77 e le coordinate GPS fornite dal mezzo indichino una percentuale di percorso effettuato pari al 52.1%.

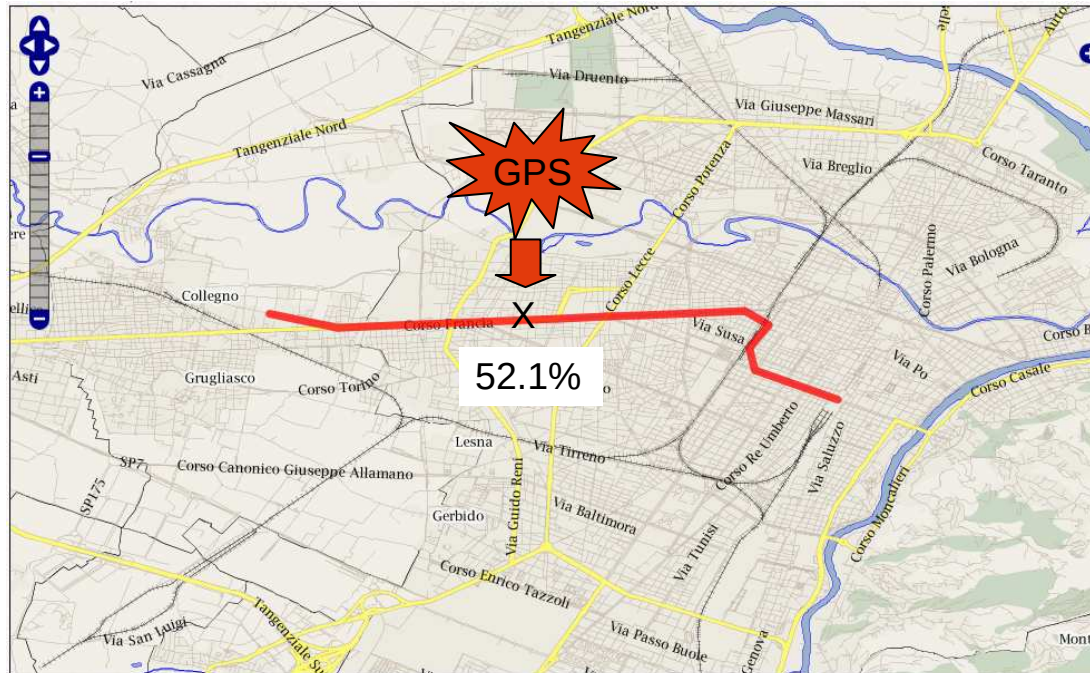


Fig. 77 - Dato GPS che indica una posizione al 52,1% del percorso

E' quindi possibile individuare in maniera immediata le due fermate che sono agli estremi della tratta che il mezzo sta percorrendo, ovvero quale fermata ha appena lasciato e verso quale fermata si sta dirigendo il mezzo.

Nel caso in esame la percentuale del dato GPS comunicato dal singolo mezzo si colloca al 52.1% del percorso totale, risulta immediatamente che la fermata appena passata è la M00007 (50%), mentre la prossima è la M00008 (53%).

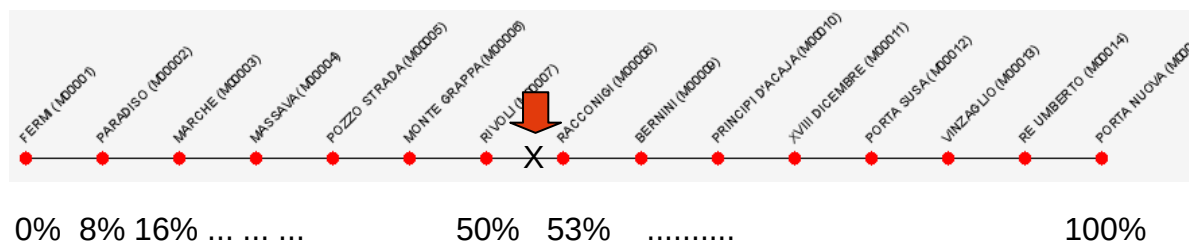


Fig. 78 - Posizione GPS rispetto alle fermate di un percorso

A questo punto della procedura di stima del ritardo per ciascuna delle due fermate viene effettuata una richiesta degli orari di passaggio della corsa in oggetto, in modo da ottenere gli orari pianificati in cui la corsa deve transitare nelle due fermate.

Supponiamo di ottenere le seguenti due letture:

- Fermata M00007 passaggio ore 14.30 (t_fermata1)
- Fermata M00008 passaggio ore 14.35 (t_fermata2)

In questo modo è possibile calcolare il tempo di percorrenza da orario della tratta in cui si trova il mezzo: nel nostro caso 5 minuti. Facendo altri calcoli è possibile ricavare la

percentuale di percorso fatta e da fare all'interno della singola tratta, effettuando le differenze tra le percentuali già note.

Nel nostro esempio si ricava che il percorso fatto all'interno della singola tratta è il 70%, mentre ovviamente resta ancora il 30% da percorrere.

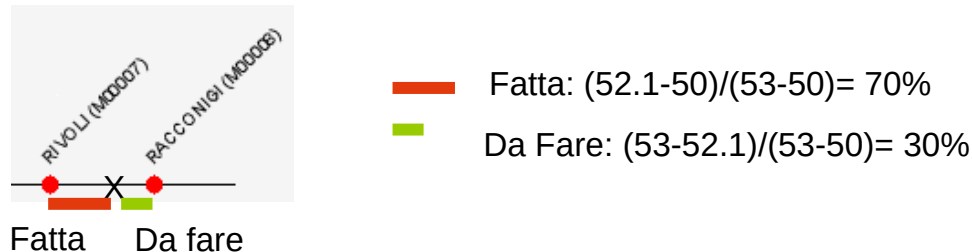


Fig. 79 - Percentuale di percorso fatto all'interno della singola tratta

E' possibile quindi calcolare il tempo necessario a terminare di percorrere la tratta come percentuale del tempo totale di percorrenza, calcolato precedentemente. Nel nostro caso abbiamo:

$$\text{Tempo\_residuo} = 5 \text{ minuti} * 30\% = 1.5 \text{ minuti}$$

Risulta quindi anche possibile calcolare l'ora di arrivo alla prossima fermata, come somma del tempo residuo all'istante attuale (o a quello che fa riferimento il dato GPS comunicato dall'autobus). Vale quindi la relazione:

$$\text{Tarrivo} = \text{Tattuale} + \text{Tempo\_residuo}$$

Infine l'ultimo passo del nostro algoritmo permette di calcolare il ritardo o anticipo con cui il mezzo transiterà alla prossima fermata utilizzando la relazione:

$$\text{Ritardo} = \text{Tarrivo} - \text{t\_fermata2}$$

All'interno del portale TRIP è stata implementata una demo di questo algoritmo che in realtà è sempre attivo e processa tutte le informazioni con i dati GPS inviati dai mezzi al centro di controllo, calcolando per ciascuno di essi l'eventuale ritardo e provvedendo eventualmente a scatenare tutte le procedure per avvisare l'utente in caso di pesanti ritardi o comunque ritardi che inficiano il percorso scelto dall'utente.

Nella pagina di demo è possibile inserire tutti i dati che caratterizzano la corsa, in maniera da individuare il mezzo che simula l'invio del dato con la propria posizione e ovviamente i dati GPS. A questo punto viene immediatamente calcolato il ritardo, secondo l'algoritmo appena presentato. Si ricorda che la richiesta degli orari di passaggio alla fermata segue le modalità di interrogazione e accesso ai dati presenti sul database degli orari (TSF) con le logiche di domanda/risposta basate sullo scambio di file XML.

[HOME](#) | [PIANIFICA IL TUO VIAGGIO](#) | [INFORMAZIONI](#) | [REGISTRAZIONE](#)



Portale TRIP

Viaggia con noi!

PORTALE DI RILEVAZIONE POSIZIONE MEZZI E GENERAZIONE EVENTI RITARDO

In questa pagina puoi trovare una demo del Portale di Rilevazione Posizione Mezzi:

GENERATORE EVENTI RITARDO

Linea*:

Corsa:

Percorso*:

Longitudine:

Latitudine:

Data:

Ora:

Percentuale Attuale: 0.536233175793% del percorso totale
Fermata già passata: U03426 (0% del percorso totale)
Fermata da raggiungere: U00408 (3.66663657556% del percorso totale)
Percentuale parziale sulla tratta: 14.6246611777% da fare: 85.3753388223%

Username:

Password:

Non sei ancora registrato? [Registrati ora!](#)
[Perché registrarsi?](#)

Partner Progetto Trip:



SCRIPTING PER AUTOMATIZZAZIONE SERVIZI

In un sistema operativo si definisce "servizio" (o anche "demone") un processo in background che gira autonomamente, senza intervento da parte dell'utente, o comunque con una interazione ridotta al minimo. Un esempio di servizio è il server web Apache: il server viene controllato dal demone "httpd" che gira in background, resta in ascolto sulla porta indicata e serve le pagine richieste.

All'interno dell'architettura TRIP alcune applicazioni devono operare in questo modo, ossia come servizio o demone, in quanto dovranno reagire a determinate condizioni che spesso non sono dirette conseguenze di un intervento da parte di un utente.

Per realizzare tali servizi non ci si affida a linguaggi per lo sviluppo di pagine web dinamiche, ma ci si affida ad altri linguaggi che possono produrre applicativi facilmente automatizzabili.

Nel caso di TRIP questi servizi risiedono su macchine con sistema operativo Linux, scelto per la grande affidabilità e gestibilità a fronte di basse richieste in termini di occupazione di memoria e potenza di calcolo richiesta. Per realizzare quindi i servizi necessari su piattaforma Linux si è scelto il linguaggio di scripting Perl.

Linguaggio di Scripting

In informatica un linguaggio di scripting è un linguaggio di programmazione interpretato (cioè che non viene compilato) destinato in genere a compiti di automazione del sistema (batch) o delle applicazioni (macro), o ad essere usato all'interno delle pagine web.

La differenza rispetto ai linguaggi interpretati (il più comune dei quali è il BASIC) è data più che altro da ragioni "storiche". I primi linguaggi di scripting erano molto rudimentali e permettevano poche e semplici operazioni, e non erano adatti alla scrittura di veri programmi. Oggi che i linguaggi di scripting hanno una potenza equivalente agli altri linguaggi la distinzione resta solo in base all'uso che si fa del linguaggio.

Esempi di linguaggi di scripting sono Perl, JavaScript, VBScript, Shell scripting (Unix), PHP, Python e Ruby.

Nei linguaggi di scripting il programmatore generalmente si disinteressa delle risorse di sistema che il programma finito dovrà consumare, demandando il tutto al sistema stesso. Per risorse si intendono, per esempio, la gestione della allocazione e deallocazione della memoria, la conversione tra tipi, l'inizializzazione e la chiusura dell'applicazione.

In questo modo si evitano molti problemi tipici della programmazione tradizionale, che risulta essere soggetta ad errori non facilmente individuabili e pericolosi, e inoltre costringe il programmatore ad occuparsi di problematiche non strettamente connesse con l'obiettivo del software che deve creare. L'utilizzo di un linguaggio di scripting permette di concentrarsi direttamente sulla soluzione del problema.

Perl

Perl è comunemente ritenuto un linguaggio interpretato, ossia che per essere eseguito viene interpretato al momento dell'esecuzione. In realtà, la prima cosa che fa l'interprete è di trasformare il codice sorgente in bytecode, un po' come Java; sul bytecode crea un grafo intermedio sul quale applica ottimizzazioni, ed è questo grafo ad essere interpretato. Questo

approccio permette di limitare la lentezza tipica dei linguaggi interpretati. La versione del Perl 6, in via di sviluppo da alcuni anni, divide l'esecutore del bytecode (o virtual machine) dal linguaggio in modo tale da permetterne l'utilizzo anche da compilatori di altri linguaggi, tra cui Tcl, Python, Java, etc.

La virtual machine che ne deriva sarà una nuova base per lo sviluppo di linguaggi liberi e consentirà livelli di ottimizzazioni molto elevati e indipendenza dallo sviluppo dei linguaggi sovrastanti. Nonostante il Perl 6 non sia ancora rilasciato, alcune idee sono percolate nella versione stabile del Perl 5.8 e molte altre avranno un riscontro in Perl 5.10. Un prototipo del Perl 6, chiamato Pugs, è stato sviluppato in Haskell e si sta verificando una interessante fertilizzazione reciproca tra le due comunità (Perl ed Haskell).

Ancora una volta è stato messo in evidenza come il Perl sia un linguaggio "socievole", che tende quindi a interagire con gli altri linguaggi e ambienti di sviluppo, siano essi: dialetti shell, altri linguaggi interpretati, linguaggi specializzati (come l'SQL), o i più comuni linguaggi compilati. Questa è la ragione che determina il successo di Perl nell'integrazione di sistemi diversi.

Perl è nato in ambiente Unix e distribuito contemporaneamente con due licenze liberali, la GPL e la Licenza Artistica, è disponibile anche per i sistemi operativi Microsoft Windows e MacOS precedente alla versione MacOS X (che appartiene alla famiglia Unix). L'implementazione per Windows più diffusa viene distribuita da una società, la ActiveState, che da un lato offre con una licenza open source moduli specifici per il sistema operativo Windows, dall'altro vende degli ambienti di sviluppo integrati, sia per Perl che per altri linguaggi open source quali ad esempio Python e Tcl.

Benché il Perl sia stato una delle grandi novità nel campo della programmazione, il giudizio su di esso da parte della comunità di programmatori è vario:

- da un lato viene giudicato negativamente per il fatto che facilita la scrittura di programmi difficili da leggere e quindi rendendo complicata la loro manutenzione (al punto che il nome del linguaggio è stato reinterpretato come Pathologically Eclectic Rubbish Lister);
- dall'altro viene apprezzato per la facilità di scrivere programmi potenti ma semplici, per la libertà semantica che lascia al programmatore al punto che "non c'è un unico modo di fare le cose" è uno dei modi di dire legati a Perl.

Wall - che per formazione è un linguista - ritiene questa libertà semantica un pregio, in quanto più simile al linguaggio umano. Un ulteriore aspetto positivo che attrae i programmatori è l'ampia disponibilità di moduli rilasciati con licenze open source, quasi sempre le stesse di Perl. Moduli solitamente documentati bene, in quanto il linguaggio stesso offre il Pod, un modo di includere la documentazione all'interno del codice, garantendo così che assieme al modulo ci sia pure la documentazione. La comunità ha creato un sito particolare, chiamato CPAN, il quale organizza per argomenti i moduli ritenuti particolarmente validi. I moduli stessi non sono archiviati in quel sito, ma rimangono sui siti scelti dai loro autori. Infine, in quanto linguaggio interpretato e dunque sempre distribuito con il codice sorgente visibile, favorisce la pratica liberale.

Altre caratteristiche importanti di Perl sono:

- le variabili di default che sono definite per molte funzioni e operatori builtin del Perl
- la sensibilità del contesto negli assegnamenti, dove Perl sa riconoscere cosa restituire in base al left value
- le espressioni regolari, che permettono la ricerca e la sostituzione di stringhe di testo descritte con caratteri speciali
- le chiusure

- la possibilità di applicare paradigmi di programmazione diversi, come quello funzionale o quello ad oggetti.

Alcune caratteristiche del Perl, tra cui la sintassi, permettono una sintesi raramente possibile con altri linguaggi ed impossibile con linguaggi tipo Java e i sorgenti possono dunque essere molto densi di significato, tanto da risultare criptici a chi non ne conosca i rudimenti. In compenso su Internet c'è così tanta documentazione sul Perl che è possibile avvicinarsi rapidamente al linguaggio e con opportuni testi di riferimento iniziarne la strada dell'apprendimento.

Il linguaggio e l'interprete vengono sviluppati da un gruppo di circa cento sviluppatori, guidati da Wall, il quale prende le decisioni finali su cosa includere nel codice. Gli sviluppatori hanno creato il Perl Institute per facilitare lo sviluppo di Perl e migliorarne la visibilità e organizzano conferenze. Wall stesso lavora per la O'Reilly, una casa editrice che sostiene attivamente il movimento Open Source.

Perl fa parte degli strumenti standard dei sistemi operativi Unix. Il suo aggiornamento può essere fatto sia da Internet che tramite le distribuzioni dei sistemi operativi liberi, quali GNU/Linux, BSD, ecc. Le versioni per Windows possono anch'esse essere scaricate da Internet. Ad alcuni manuali di Perl viene allegato un CD-Rom comprendente le versioni per diversi sistemi operativi e tutti i moduli compresi nel CPAN.

La comunità Perl è sovente attaccata per l'assenza di un IDE come quelli presenti per Java che rendano semplice l'avvicinamento al linguaggio da parte dei neofiti. In realtà la ricchezza espressiva del Perl rendono complicata la realizzazione di un IDE che evidenzi errori di costruzione degli statement, e dati i numerosi valori di default delle funzioni base risulterebbe oltremodo difficile comprendere in automatico cosa vuole ottenere il programmatore, dove c'è un errore o forse no. Tuttavia recentemente ad esempio Eclipse contiene una estensione per il Perl e forse altri seguiranno.

Gli sviluppatori Perl esperti in verità non sentono l'esigenza di strumenti diversi da vi o Emacs, e quest'ultimo in particolare permette di effettuare il debugging di ogni programma Perl, con narrowing, colorazione della sintassi, macro, breakpoint, single step etc.

SCRIPT GESTIONE SOCKET

Questo script, oltre a aprire un canale di comunicazione tra portale TRIP e motore di calcolo del percorso si occupa del controllo della correttezza formale delle stringhe XML che le due entità si scambiano.

Iniziamo descrivendo la natura della socket. Si tratta di una canale TCP che viene aperto verso un determinato indirizzo formato da IP e numero di porta. Su tale porta si trova in ascolto un server che accetta tale connessione.

Lo script prevede una fase di inizializzazione in cui si instaura tale connessione seguita dalla fase operativa in cui vengono istanziati tre thread che operano in maniera concorrente per:

- Leggere dalla socket e costruire i file XML opportuni in base alla tipologia di XML, che verranno poi utilizzati da altre intelligenze;
- Verificare la presenza di nuovi file xml da inviare, leggerne il contenuto e inviare la stringa XML attraverso la socket;
- Monitorare lo stato della socket

Tipologie di stringhe XML

Le tipologie di XML scambiati tra portale TRIP e motore sono:

- Richiesta percorso
- Risposta percorso
- Richiesta orari
- Risposta orari
- Evento Ritardo
- Ack Evento Ritardo
- Aggiornamento Evento

Lo script di gestione della socket non si occupa di utilizzare i dati forniti dalle stringhe XML. Lo script si occupa di controllarne la correttezza formale e di effettuare opportune operazioni in seguito alla ricezione ed invio delle stringhe.

Lettura Socket

Lo script di gestione della socket avvia un processo sempre attivo che controlla la ricezione sul canale di dati. Se tali dati formano una stringa XML corretta tale stringa viene salvata sottoforma di file in un'opportuna cartella in modo che sia disponibile per le altre entità che compongono il sistema TRIP.

Lo script svolge inoltre un controllo su eventuali casi di ricezione di stringhe incomplete o errate.

Scrittura Socket

Lo script di gestione della socket avvia un processo sempre attivo che controlla la presenza di file XML, generati da altre entità della piattaforma TRIP. Quando un file XML viene

trovato viene letta come stringa il contenuto e la stringa XML che ne risulta viene inoltrata in maniera trasparente al motore di calcolo del percorso.

Per non sovraccaricare il motore di calcolo del percorso minimo con troppe richieste contemporanee, lo script di gestione della socket svolge una funzione di temporizzatore. Mantenendo il conteggio delle richieste inoltrate pendenti e il numero di richieste che sono state processate lo script tiene monitorato il numero di richieste in fase di elaborazione da parte del motore di calcolo del percorso. Se tale numero raggiunge un certo valore limite, lo script smette di inoltrare ulteriori richieste e attende che il motore smaltisca le richieste pendenti fino a scendere sotto al limite prefissato.

Monitoraggio Socket

Lo script di gestione della socket avvia un processo sempre attivo che effettua il monitoraggio dello stato della socket. Tale monitoraggio si effettua perché in talune condizioni (ad esempio reset della macchina remota) non è possibile valutare con un semplice comando se la connessione con la socket sia abilitata o meno.

Per essere certi del funzionamento della socket si provvede all'inoltro con una certa cadenza di un stringa XML di controllo. Se entro un certo periodo di tempo non si riceve una stringa di conferma si considera la connessione interrotta. In tal caso si procede alla terminazione dei tre processi inerenti al socket: questo per evitare di instaurare cicli di lettura o scrittura su socket non abilitata che possono portare a comportamenti instabili del sistema (ad esempio utilizzo del 100% delle risorse di calcolo della CPU).

La condizione di chiusura dei processi è inoltre individuabile facilmente dal sistema e quindi può essere indicato dalle pagine web create per il controllo dello script.

Scenario tipico di utilizzo

Un esempio di utilizzo: un utente accede al portale TRIP e richiede le informazioni per un particolare viaggio. Il portale raccoglie i dati della richiesta e li inoltra al motore di calcolo del percorso attraverso la socket (scrittura su socket). Dopo aver elaborato i dati il motore di calcolo del percorso fornisce un insieme di percorsi ottimali sotto forma di stringa XML. Tale stringa viaggia attraverso la socket e viene intercettata dallo script (lettura da socket). La stringa viene salvata in una cartella opportuna a cui il portale può accedere; il portale elabora le informazioni ricevute e le propone all'utente in un formato facilmente fruibile (testo impaginato e grafica).

Pagine di controllo dello script

All'interno del portale TRIP è stata realizzata una pagina che permette di monitorare il funzionamento dello script di gestione della Socket. Si tratta di una semplice pagina web dinamica in cui è possibile attraverso un tasto aprire o chiudere la connessione tramite socket verso il motore di calcolo del percorso.

Lo stato dello script di gestione della socket (e quindi lo stato della connessione) è indicato in maniera sintetica attraverso un riquadro di stato:

- Riquadro verde lo script è attivo e la connessione è attiva oppure è in fase di attivazione

- Riquadro giallo lo script non è attivo e tanto meno lo è la connessione

Una textarea mostra in maniera più precisa lo stato dello script riportando le ultime note scritte sul file di log: si possono quindi leggere le ultime operazioni eseguite con relativa data ed ora ed eventuali messaggi di errore sempre corredati di coordinate temporali.

Uno screenshot della pagina di gestione dello script all'interno del portale TRIP è riportata in Fig. 80

Fig. 80 - Screenshot della pagina del Portale TRIP per il controllo dello script di gestione della socket

SCRIPT GESTIONE SMS

La gestione principale dei SMS avviene sul Server TRIP attraverso uno script Perl. Lo script si occupa della gestione della scheda SIM con cui si accede alla rete GSM per l'invio e ricezione di SMS.

Hardware:

Il server è in grado di operare sulla scheda SIM attraverso un interfaccia hardware. Nel caso specifico si tratta di una scheda PCMCIA che offre accesso alla rete GSM/GPRS/UMTS.

I driver per le operazioni elementari sulla scheda SIM sono stati implementati all'interno del progetto TRIP e sono integrati all'interno dello script Perl.

In particolare l'accesso alla scheda si effettua come accesso ad un modem seriale che accetta comandi di tipo AT. Attraverso tali tipologie di comandi elementari si è riuscito a eseguire operazioni come:

- Lettura di SMS in arrivo sulla SIM;
- Scrittura di SMS sulla SIM per inoltrare sulla rete GSM
- Controllo presenza di SMS sulla memoria della SIM
- Eliminazione di messaggi dalla memoria della SIM

Nel procedere all'implementazione di tali comandi di accesso alla scheda SIM, si sono affrontati due scopi:

- ricerca della giusta temporizzazione nell'accesso alla scheda SIM: oltre ad impostare un'opportuna baudrate, si sono dovute trovare ed impostare le giuste temporizzazioni per la scrittura di SMS lunghi, per permettere il corretto accesso alla rete GSM, per effettuare l'alternanza tra fasi di lettura e fasi di scrittura sulla SIM
- Utilizzare la formattazione PDU per la scrittura dei SMS: tale scelta permette un maggiore controllo sulle opzioni dei SMS ed inoltre permette di utilizzare tutti i caratteri per la scrittura di un SMS (160 caratteri contro i 140 disponibili utilizzando il metodo di scrittura tipo testo). Tali vantaggi vengono al costo di una fase di codifica del testo non intuitiva.

Script

Lo script è sviluppato per operare in diverse fasi che si alternano nell'accesso alla risorsa hardware, ossia il modem seriale che permette l'accesso alla SIM. Le capacità di elaborazione a bordo del dispositivo sono limitate e quindi si preclude un accesso multiplo da varie entità software.

In breve diverse fasi sono 3:

1. fase di inizializzazione: si registra la SIM sulla rete GSM e si cancellano eventuali messaggi memorizzati, questa fase viene eseguita una sola volta all'avvio del sistema SMS-TRIP e non viene ripetuta
2. fase di lettura: si accede alla SIM per controllare l'arrivo di nuovi SMS; in caso affermativo si procede con la loro elaborazione
3. fase di scrittura: si cercano eventuali file SMS da inviare; in caso affermativo si accede alla scheda SIM per inoltrare tali SMS verso i rispettivi destinatari.

Dopo la fase di inizializzazione la fase 2 e 3 si alternano in modo da palleggiarsi eventuali accessi alla scheda SIM.

Fase di inizializzazione

In questa fase si attiva la scheda SIM e la si registra sulla rete GSM.

La registrazione avviene tramite comando AT e si attende la stringa di conferma.

Una volta registrata la SIM si procede con la cancellazione dei SMS in memoria. Tale azione ha lo scopo di facilitare la fase di lettura, infatti ad ogni ciclo di tale fase tutti i messaggi trovati in memoria verranno considerati nuovi messaggi. Inoltre, agendo in questo senso, si previene la totale occupazione della memoria della SIM da parte di messaggi accumulati, che avrebbe la conseguenza di impedire la ricezione di nuovi messaggi.

Si assume che la SIM disponga di 30 celle di memorizzazione: si accede ad ogni cella e si cancella l'eventuale SMS presente: questa operazione comporta un tempo di circa 1 secondo per cella, e quindi un totale di circa 30 secondi per l'intera fase di inizializzazione.

Fase di lettura da SIM

Questa fase, insieme a quella di scrittura, sono le fasi principali che si alternano l'accesso della SIM per poter scambiare SMS con l'utenza.

Si accede alla SIM e si leggono tutti i SMS presenti in memoria. Tali SMS vengono organizzati in un vettore. Scorrendo tale vettore si opera su ognuno dei SMS ricevuti. Dopo aver analizzato un SMS lo si cancella, in modo che al termine della fase nessun messaggio risieda all'interno della memoria della SIM; in tal modo al prossimo ciclo tutti i SMS trovati sulla SIM sono nuovi messaggi da prendere in carico.

Naturalmente ogni SMS è formato da due campi: numero dell'utente e testo del messaggio.

Una volta letto dalla SIM ogni SMS viene passato ad un filtro che cerca di capire quale sia la richiesta da parte dell'utente.

Interpretazione SMS

Il testo del SMS viene analizzato per riconoscere le diverse tipologie possibili di messaggio. La tipologia del messaggio viene dedotta da alcune parole chiave che sono presenti o meno nel messaggio.

1. richiesta DA:... A:...

In questo caso la stringa che compone è così composta

DA:	campo partenza	A:	campo arrivo

Lo script si occupa di individuare campo di partenza e campo di arrivo e di determinare anche per questi campi il contenuto; le opzioni sono

Comune, Via, numero civico

Comune, Via/Punto di Interesse

Codice Fermata

In questo caso il riconoscimento del tipo di informazione avviene attraverso il conteggio delle virgole di separazione e dal tipo alfanumerico che individua un numero civico o il codice di una fermata.

2. codice per eliminare ambiguità

In questo caso la stringa che compone è così composta

COD: **codice alfanumerico**

Il codice alfanumerico viene inviato dall'utente nel caso di punto di partenza e/o arrivo ambigui. In tal caso vengono proposte all'utente varie alternative. L'utente rispondendo con un messaggio di questo tipo indica al sistema la scelta corretta.

3. richiesta orario passaggi

In questo caso la stringa che compone è così composta

PASSAGGI FERMATA: **codice fermata**

Richiesta la portale

Una volta che lo script ha individuato la tipologia di richiesta posta dall'utente accede al portale TRIP interrogando opportunamente la pagina dinamica PHP che si occupa del calcolo del percorso ottimale. Tale pagina restituisce il percorso sottoforma di file di interscambio XML.

Il file XML contiene tutte le informazioni da inviare all'utente.

Creazione del messaggio

Questa fase implica unicamente la trasformazione dell'informazione contenuta nel file XML in un formato da proporre all'utente finale. La stringa creata viene salvata come file SMS ossia:

1° riga: numero destinatario

2° riga: testo

Tale file viene salvato nella cartella dei file SMS da inviare e verrà preso inviato realmente solo nella fase di scrittura su SIM

Fase di scrittura su SIM

In questa fase si accede ad una specifica cartella per controllare se esistono file SMS da inviare. Da notare che i file presenti in questa cartella possono essere stati generati sia in risposta ad un SMS ricevuto nella fase di lettura da SIM, sia da altre entità del sistema TRIP, come ad esempio nel caso di un SMS da inviare per avvertire l'utente di un evento realtime.

Se sono presenti file SMS ognuno di essi viene letto per ottenere numero del destinatario e testo.

Tali informazioni vengono processate da un filtro che trasforma queste informazioni in formato testuale in una stringa di formato PDU.

In seguito attraverso opportuni comandi AT la stringa viene inviata tramite la scheda SIM sulla rete GSM come normale SMS. Attraverso la rete GSM raggiungerà l'utente destinatario.

Richiesta da SMS

Di seguito viene descritta la sequenza di eventi che rende possibile la richiesta di un utente attraverso SMS via terminale GSM.

Tale sequenza di eventi è illustrata anche in Fig. 81.

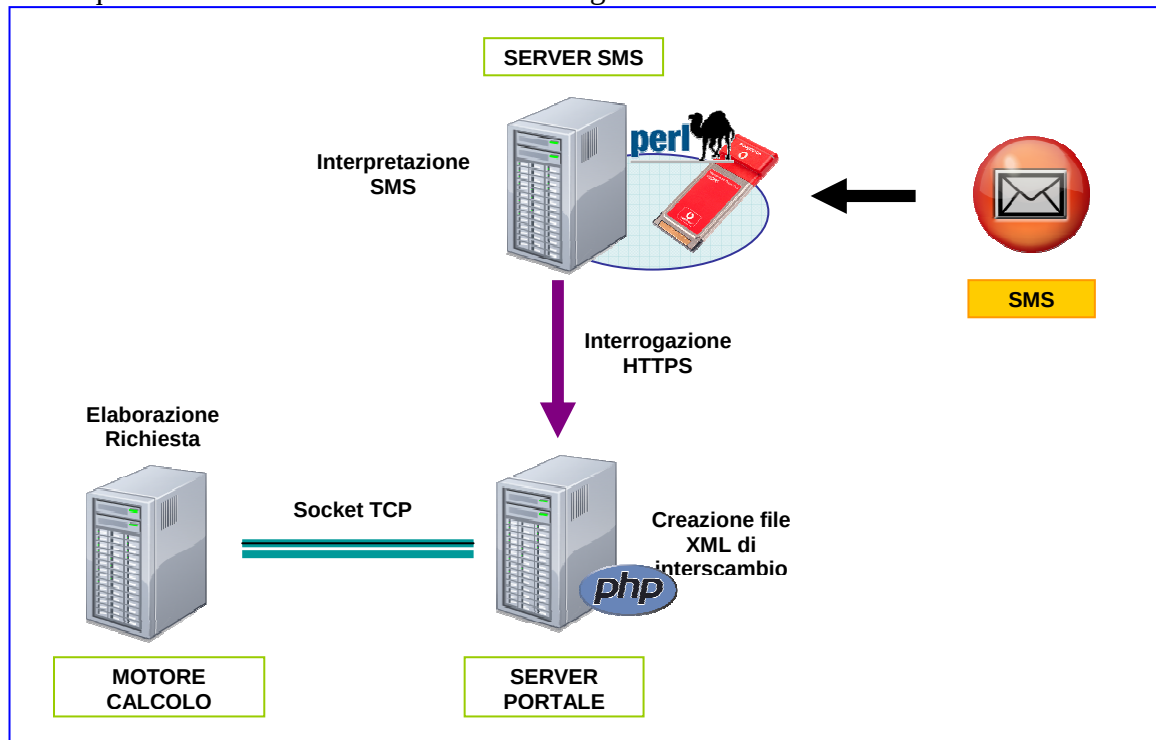


Fig. 81 - Richiesta al sistema TRIP attraverso SMS

Una richiesta tramite SMS prevede l'utilizzo delle funzioni descritte in precedenza in questa configurazione:

- l'utente invia il messaggio al numero telefonico preposto, quindi il messaggio raggiunge la SIM
- nella fase di lettura da SIM, lo script Perl si accorge del messaggio presente sulla SIM e lo legge come stringa recuperando al contempo il numero di cellulare dell'utente
- il testo del messaggio viene interpretato e attraverso i campi forniti si procede ad una richiesta verso il portale attraverso file di interscambio XML
- l'interrogazione sul portale scatena a sua volta un'interrogazione verso il motore per ottenere un percorso o orari di mezzi pubblici

A questo punto il flusso informativo si inverte e i dati generati percorrono la strada inverso fino all'utente con dispositivo GSM, come in Fig. 82:

- il motore calcola il percorso ottimale e lo invia la portale nella forma di file di interscambio XML usando l'apposita socket
- la pagina PHP che è stata interrogata dallo script Perl restituisce un file XML arricchito da ulteriori informazioni
- lo script Perl, ancora nella fase di lettura da SIM, riceve il file XML, lo interpreta e costruisce il file SMS

- nella fase di scrittura su SIM lo script Perl trova il file SMS appena generato e lo invia attraverso la scheda all'utente mobile.

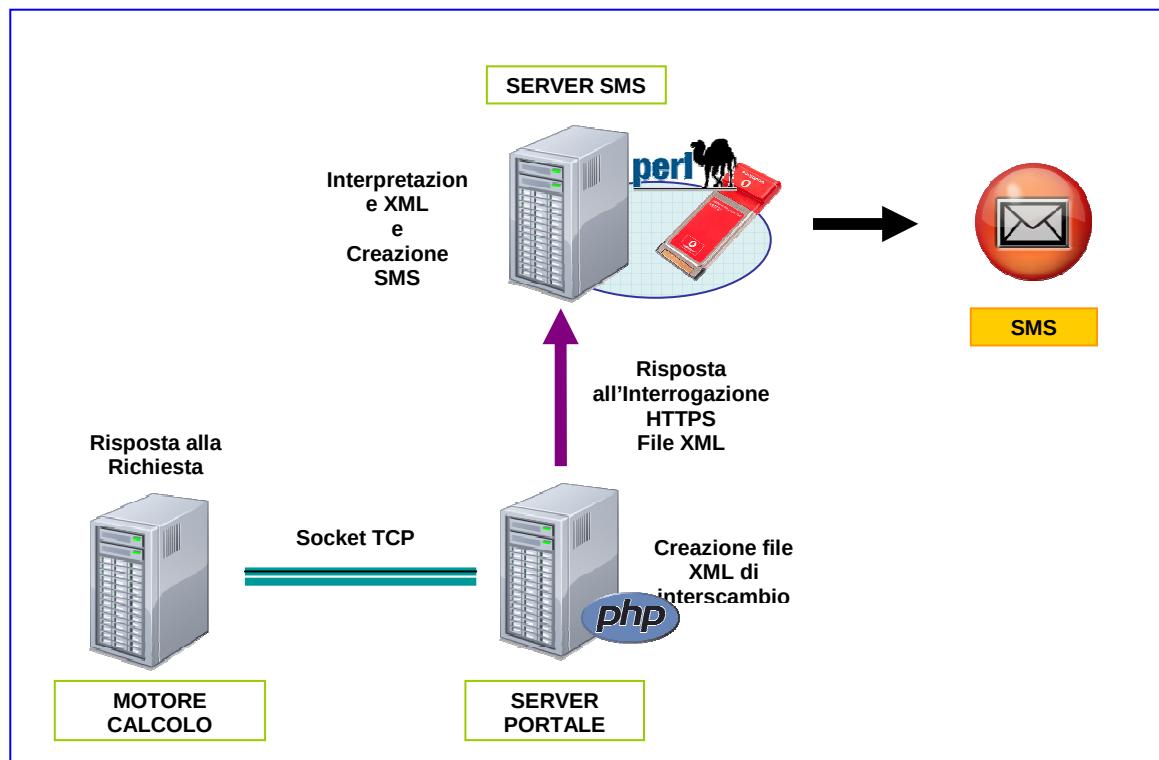


Fig. 82 - Risposta ad una richiesta via SMS

Invio SMS dal portale

A seguito di alcune operazioni sul browser è possibile causare l'invio di SMS.

Le possibilità sono:

- registrazione di un nuovo utente: quando un nuovo utente si registra, una notifica dell'avvenuta registrazione viene spedita attraverso SMS
- dalla pagina Console SMS il gestore del portale può scrivere un SMS da inviare a tutti gli utenti registrati nel sistema
- la pagina demo SMS simula l'invio di un SMS di richiesta informazioni e viene restituita a video il risultato di tale richiesta (il SMS viene generato seguendo la normale catena di passaggi, ma non viene spedito attraverso SIM, viene invece intercettato dalla pagina web)

In tutti questi casi l'utente (amministratore o meno che sia) compie operazioni su pagine WEB dinamiche che vanno poi ad interfacciarsi con lo script di gestione SMS.

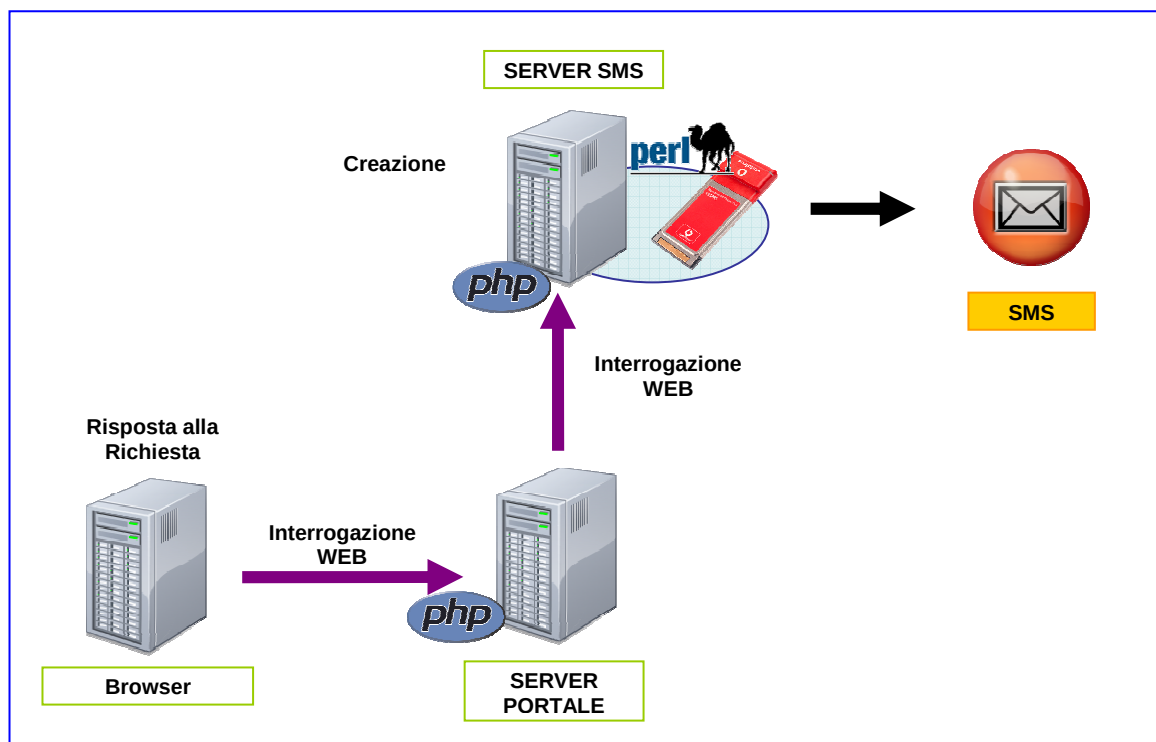


Fig. 83 - Invio SMS a seguito di operazioni sul Portale TRIP

Da notare che questo scambio di informazioni e di controllo interessa due server differenti: il Server del Portale TRIP e il Server SMS.

Per realizzare tale scambio il server SMS mette a disposizione una pagina PHP. Tale pagina per la sua stessa natura di oggetto web può essere richiamata da remoto (e quindi dal server TRIP). Tale pagina viene richiamata passandogli come argomenti testo e numero di destinazione del messaggio SMS. La pagina si occupa di trasformare tali informazioni in un file .sms che verrà poi gestito dallo script Perl come un normale file SMS uscente.

Il funzionamento appena descritto si applica per i primi due casi. Il terzo caso corrisponde ad un diversa catena di eventi dato che il messaggio non deve essere inviato all'utente, ma è un messaggio di interrogazione da inviare al motore, la cui risposta verrà reindirizzata verso il browser stesso. Per realizzare tale passaggio si utilizza una pagina php differente sempre residente sul server SMS. Tale pagina genera il file SMS con una numerazione particolare (invece del numero si trova la stringa INTERNO seguita da un identificativo della sessione dell'utente che ha utilizzato la pagina del browser, questo per evitare sovrapposizioni). Lo script Perl interpreta questo file SMS come una richiesta e quindi interroga in modo appropriato il motore. La pagina php resta in attesa di tale risposta e quando la intercetta la restituisce come output al richiedente originale, ossia il browser TRIP che la riproduce su schermo per l'utente finale.

Pagina Demo SMS

All'interno del portale TRIP è stata realizzata una pagina che permette di testare il funzionamento dello script di gestione SMS. Si tratta di una semplice pagina web dinamica

in cui è possibile scrivere un testo (sono già pronti alcuni esempi di testo preconfezionati) ed inviarli al sistema come se fossero stati inviati da un cellulare.

Il testo inviato seguirà esattamente gli stessi processi che seguirebbe con l'invio effettivo da cellulare. L'unica differenza sostanziale è che la risposta del sistema non viaggia effettivamente attraverso la SIM, ma viene reinstradata alla pagina web che la mostra graficamente a video.

Uno screenshot della pagina di test dello script all'interno del portale TRIP è riportata in Fig. 84

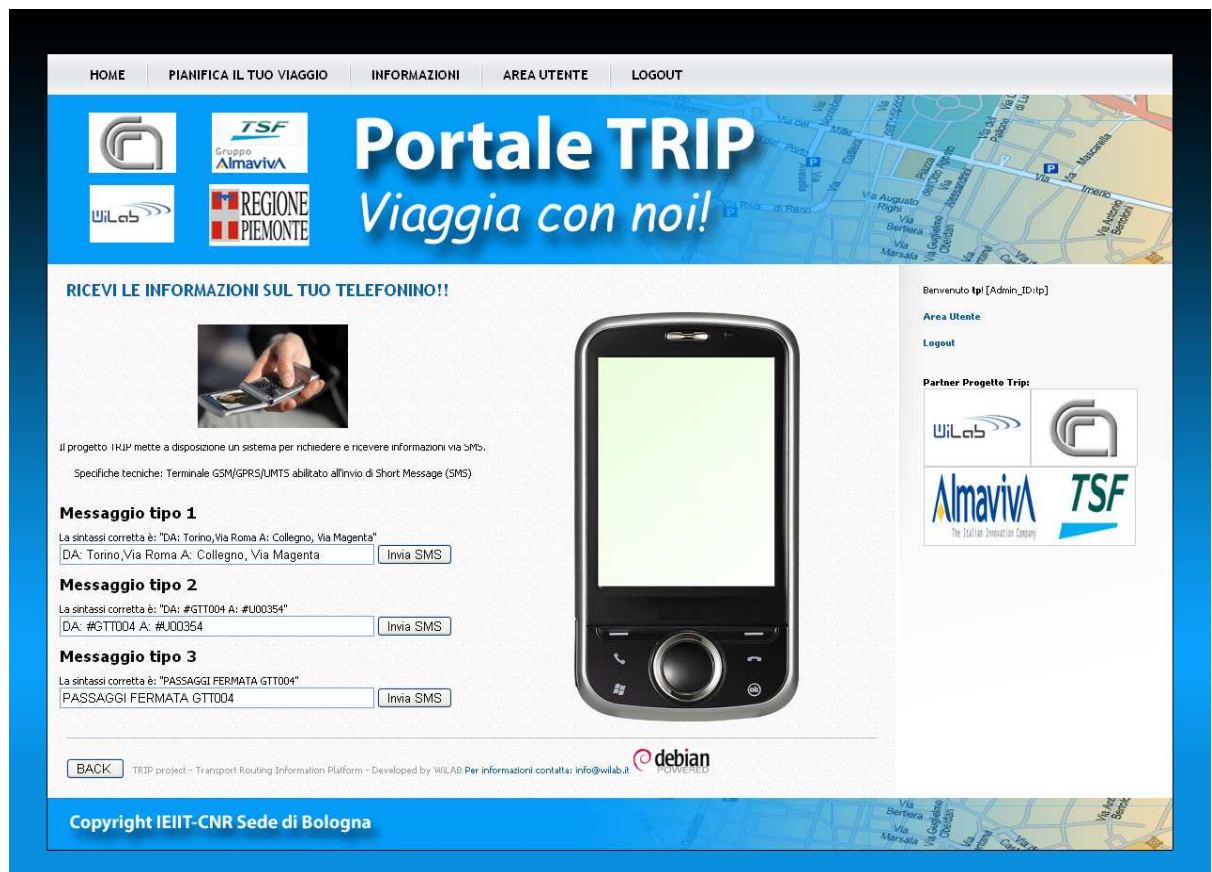


Fig. 84 - Screenshot della pagina del Portale TRIP per uso demo dello script di gestione SMS

SCRIPT REAL TIME

La generazione di un evento ritardo avviene in seguito alla raccolta da parte del centro di controllo di un dato di posizione GPS di un mezzo pubblico.

Confrontando posizione attuale del mezzo con la posizione che dovrebbe avere affidandosi agli orari da tabellone si determina una stima del ritardo che affligge quel mezzo (nella descrizione dell'algoritmo di calcolo del ritardo questa procedura è descritta in maggior dettaglio).

Descrizione di insieme: questo script si occupa di intercettare gli eventi che influiscono sul traffico dei mezzi e di valutare la loro influenza sugli itinerari memorizzati da parte degli utenti. Nel caso di itinerari interessati da uno o più eventi si procede al ricalcolo di un nuovo percorso e si informa l'utente di eventuali variazioni del tragitto ottimale per raggiungere la destinazione prefissata.

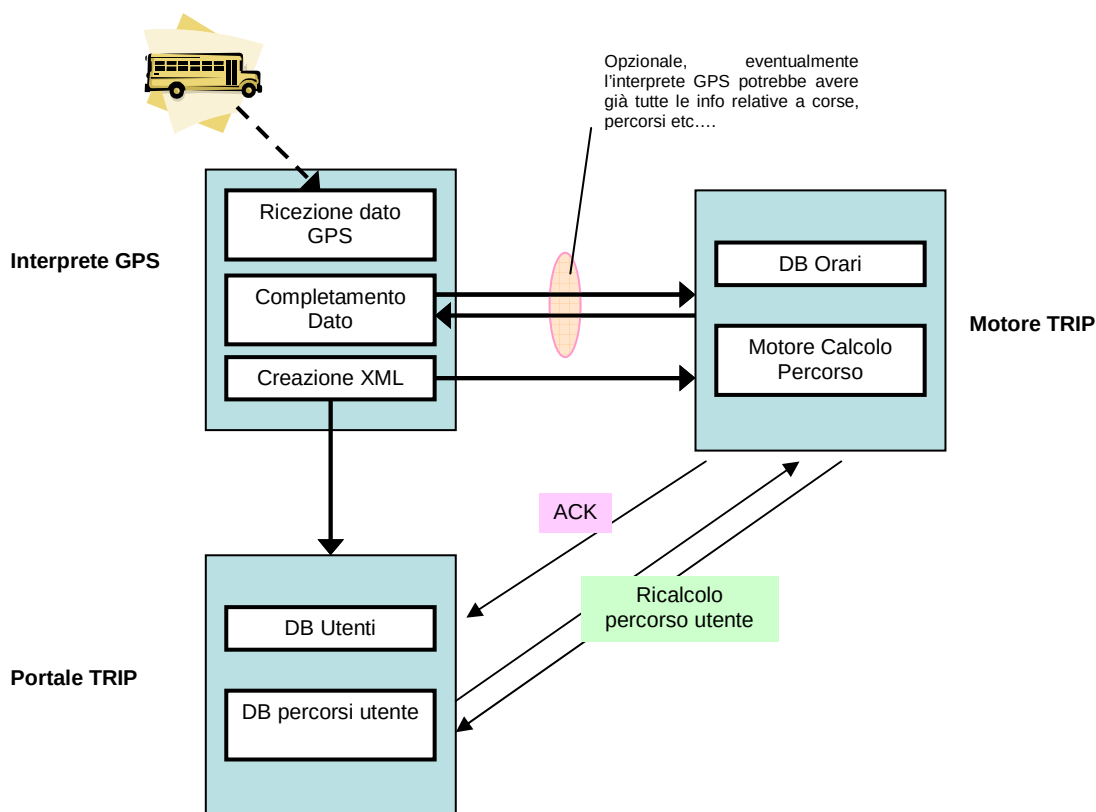


Fig. 85 - Schema di base del processo di notifica realtime

Struttura dello script: lo script utilizza due thread separati che si occupano rispettivamente:

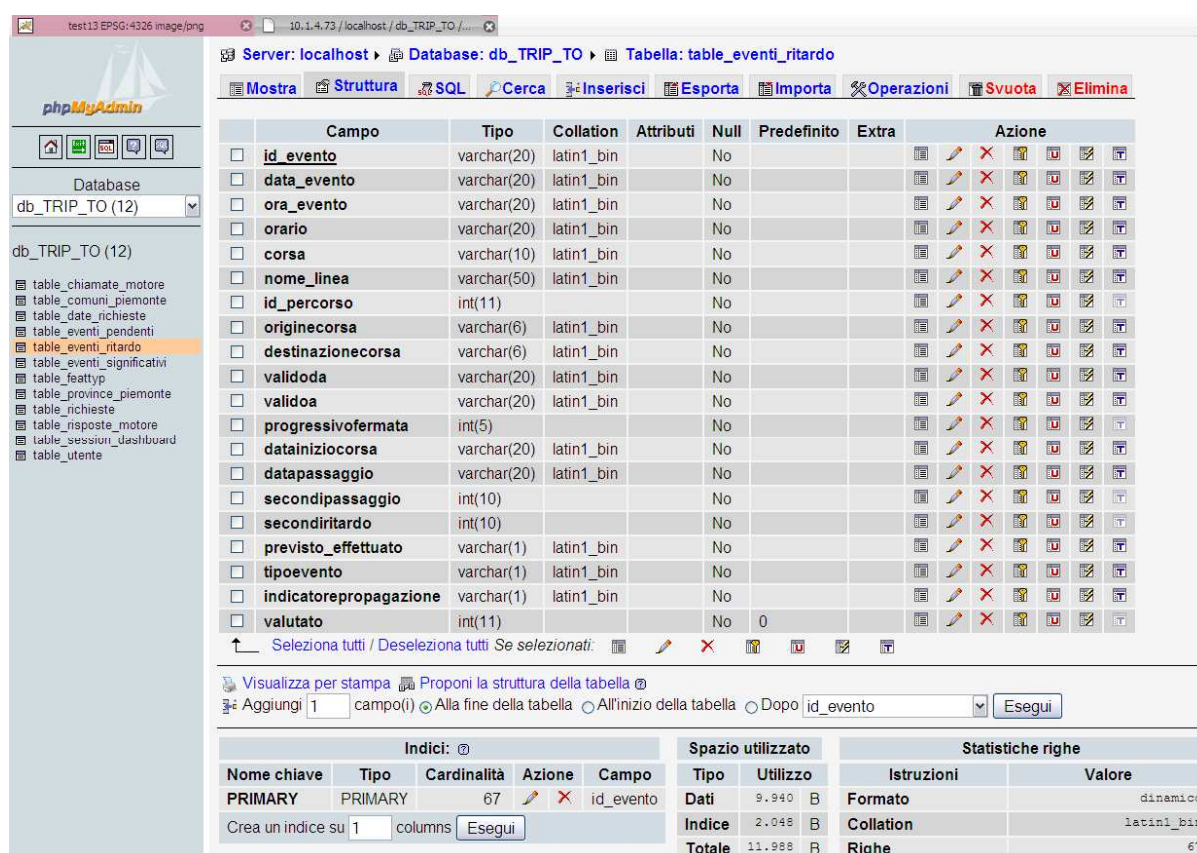
- accedere al DB per intercettare l'arrivo di nuovi eventi di traffico; nel caso di nuovo evento valutare quali percorsi memorizzati dagli utenti sono interessati da tali eventi e registrare questa condizione attraverso aggiornamento del DB.
- Attende che gli eventi vengano presi in carico dal Motore; quando tutti gli eventi che influenzano un itinerario memorizzato vengono presi in carico si passa al ricalcolo di un nuovo percorso (attraverso interrogazione al Motore); dal confronto tra itinerario

memorizzato e nuovo itinerario calcolato si stabilisce quale messaggio inviare all'utente.

Primo Thread

Il thread intercetta l'arrivo di nuovi eventi di traffico. Per farlo legge con cadenza regolare il DB mysql db_TRIP_TO, tabella table_eventi_ritardo. La tabella contiene le informazioni di tutti gli elementi evento. In particolare ogni evento ha un campo valutato che vale 0 se l'evento deve essere ancora preso in considerazione dallo script realtime e vale 1 se è già stato valutato.

Un esempio dei dati raccolti per la descrizione degli eventi è riportata in Fig. 86.



Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐ id_evento	varchar(20)	latin1_bin		No			
☐ data_evento	varchar(20)	latin1_bin		No			
☐ ora_evento	varchar(20)	latin1_bin		No			
☐ orario	varchar(20)	latin1_bin		No			
☐ corsa	varchar(10)	latin1_bin		No			
☐ nome_linea	varchar(50)	latin1_bin		No			
☐ id_percorso	int(11)			No			
☐ originecorsa	varchar(6)	latin1_bin		No			
☐ destinazionecorsa	varchar(6)	latin1_bin		No			
☐ validoda	varchar(20)	latin1_bin		No			
☐ validoa	varchar(20)	latin1_bin		No			
☐ progressivofermata	int(5)			No			
☐ datainiziocorsa	varchar(20)	latin1_bin		No			
☐ datapassaggio	varchar(20)	latin1_bin		No			
☐ secondipassaggio	int(10)			No			
☐ secondiritardo	int(10)			No			
☐ previsto_effettuato	varchar(1)	latin1_bin		No			
☐ tipoevento	varchar(1)	latin1_bin		No			
☐ indicatorepropagazione	varchar(1)	latin1_bin		No			
☐ valutato	int(11)			No	0		

Indici:					Spazio utilizzato		Statistiche righe	
Nome chiave	Tipo	Cardinalità	Azione	Campo	Tipo	Utilizzo	Istruzioni	Valore
PRIMARY	PRIMARY	67		id_evento	Dati	9.940 B	Formato	dinamico
Crea un indice su 1 columns					Indice	2.048 B	Collation	latin1_bin
					Totale	11.988 B	Righe	67

Fig. 86 - Campi che identificano l'oggetto Evento Ritardo

Lo script quindi recupera le informazioni di tutti gli eventi che hanno il campo valutato a 0. Ogni evento interessa una particolare corsa: si accede al DB tabella table_richieste e si cercano quegli itinerari memorizzati dagli utenti che hanno quella corsa come parte del tragitto. Per ogni richiesta memorizzata si valuta se l'evento impatta confrontando le date di validità dell'itinerario e la data dell'evento. Nel caso di evento impattante si aggiunge una riga nel database tabella table_eventi_pendenti che sta ad indicare che per una determinata richiesta ed in una particolare data c'è un evento che ha influenza e che si è in attesa di un riscontro da parte del motore per sapere se tale evento è già stato preso in carico dall'entità di calcolo dei percorsi (campo ack=0).

Dopo aver aggiornato la lista degli eventi pendenti il thread termina il suo compito. Sarà il secondo thread a rimanere in attesa degli ack da parte del motore e poi procedere con il ricalcolo.

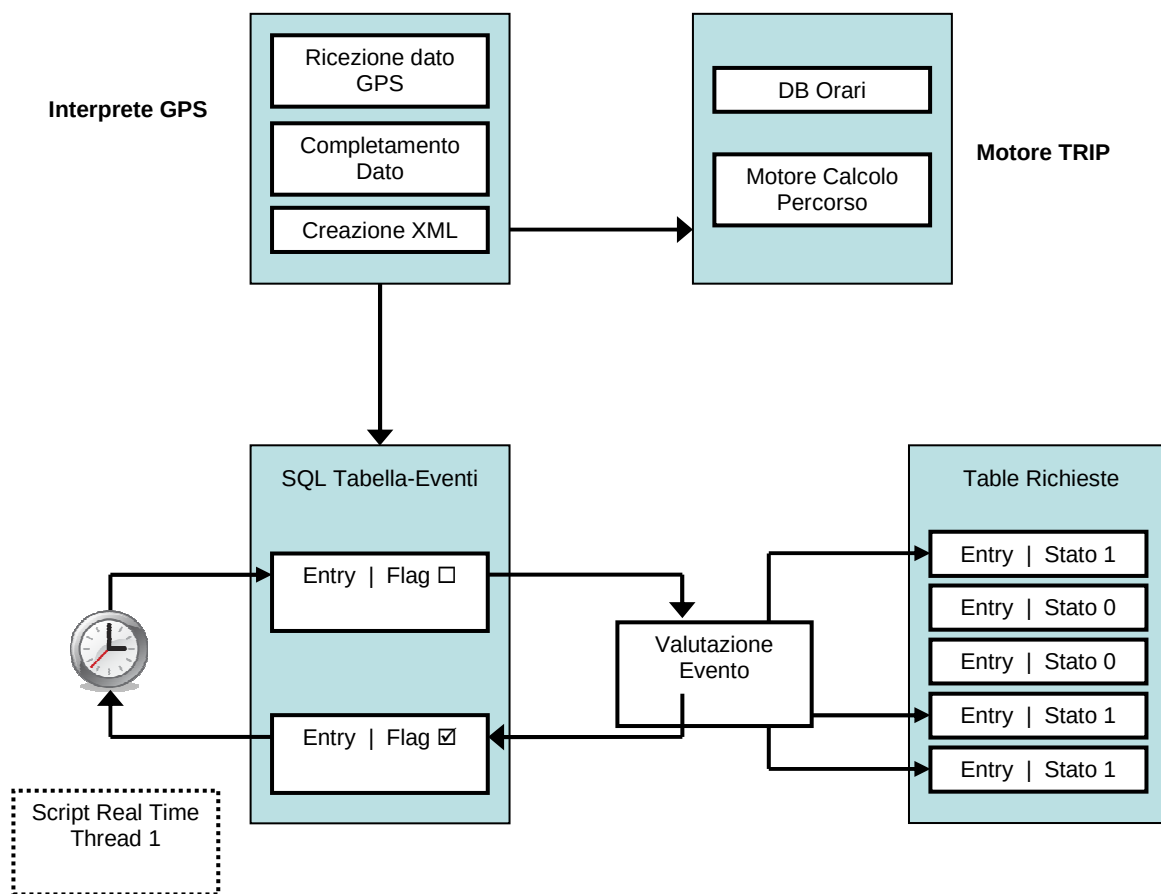


Fig. 87 - Schema a blocchi del primo thread dello script realtime

Secondo Thread

Il thread verifica l'arrivo degli ack da parte del motore. L'intelligenza legata alla socket di comunicazione portale-motore alla ricezione di un XML di ack aggiorna sul DB la tabella table_eventi_pendenti. Quindi dopo la ricezione di un ACK gli eventi presi in carico dal motore hanno il campo ack=1.

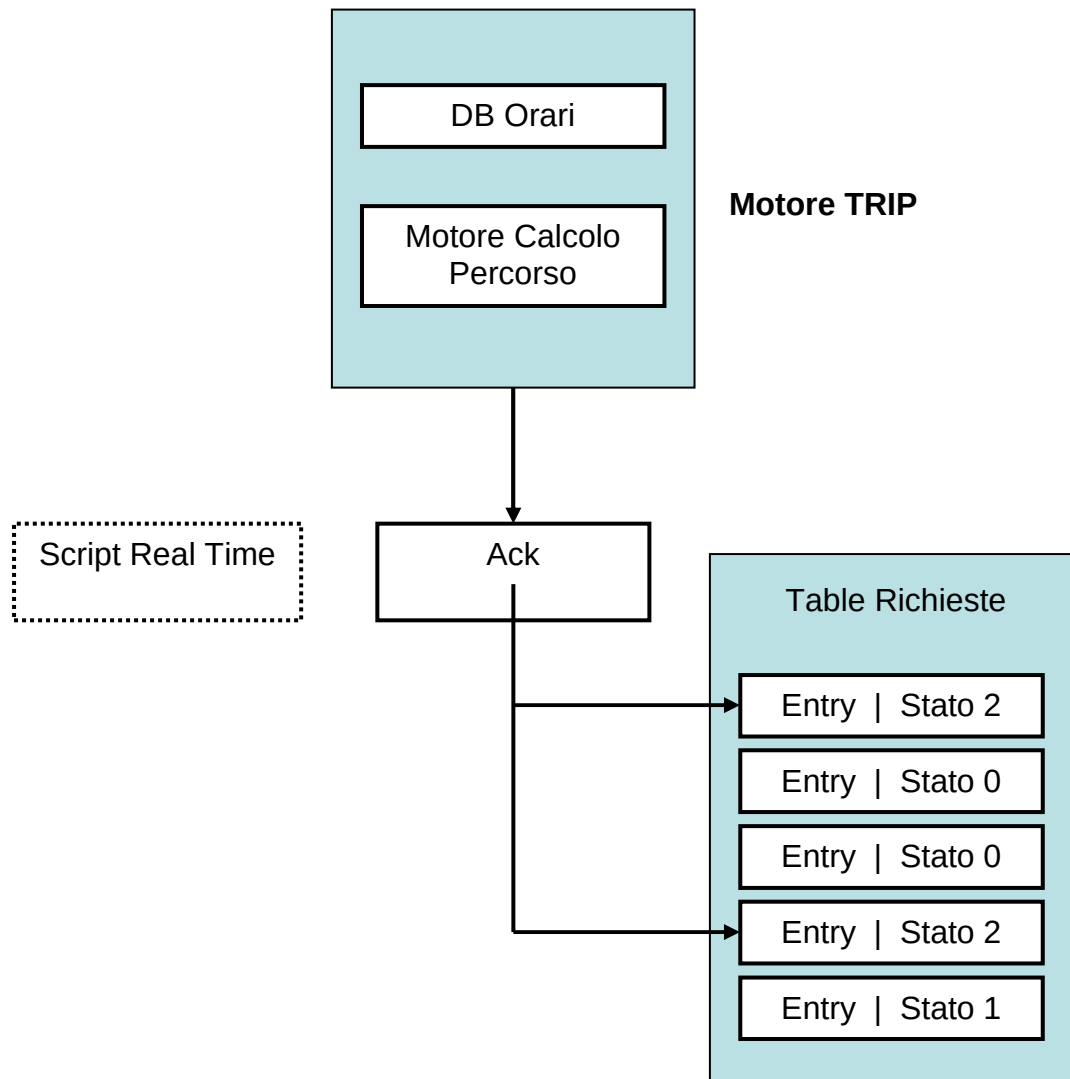


Fig. 88 - Attesa di conferma presa in carico dell'evento ritardo da parte del motore di calcolo

Il thread controlla ciclicamente il valore di ack degli eventi pendenti. Quando tutti gli eventi pendenti afferenti ad una richiesta hanno ack=1, si procede con la fase di ricalcolo dell'itinerario.

Viene salvata una copia del percorso iniziale.

Si costruisce un nuovo file XML di richiesta a partire dal file di richiesta originario, tenendo in conto diversi aspetti (si veda sotto).

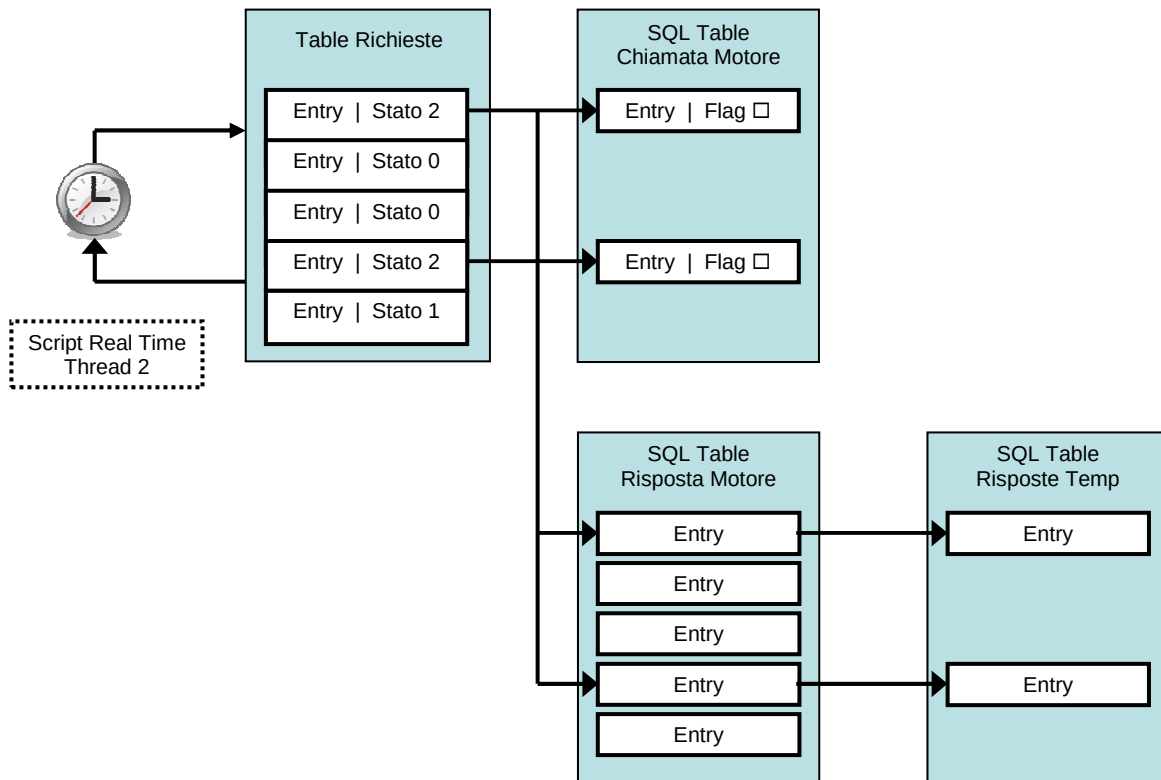


Fig. 89 - Analisi dei percorsi affetti dagli eventi ritardo già presi in carico

La nuova richiesta viene inoltrata al motore. Quando si riceve la risposta si confrontano i due itinerari ottenuti (pre e post eventi traffico).

Se non vi sono differenze non viene inviata alcuna segnalazione. Se invece vi sono differenze (diverso mezzo da prendere oppure differenza su orari di partenza e/o arrivo) si procede con l'invio all'utente di un SMS che informi del fatto e che riporta le istruzioni per il nuovo itinerario.

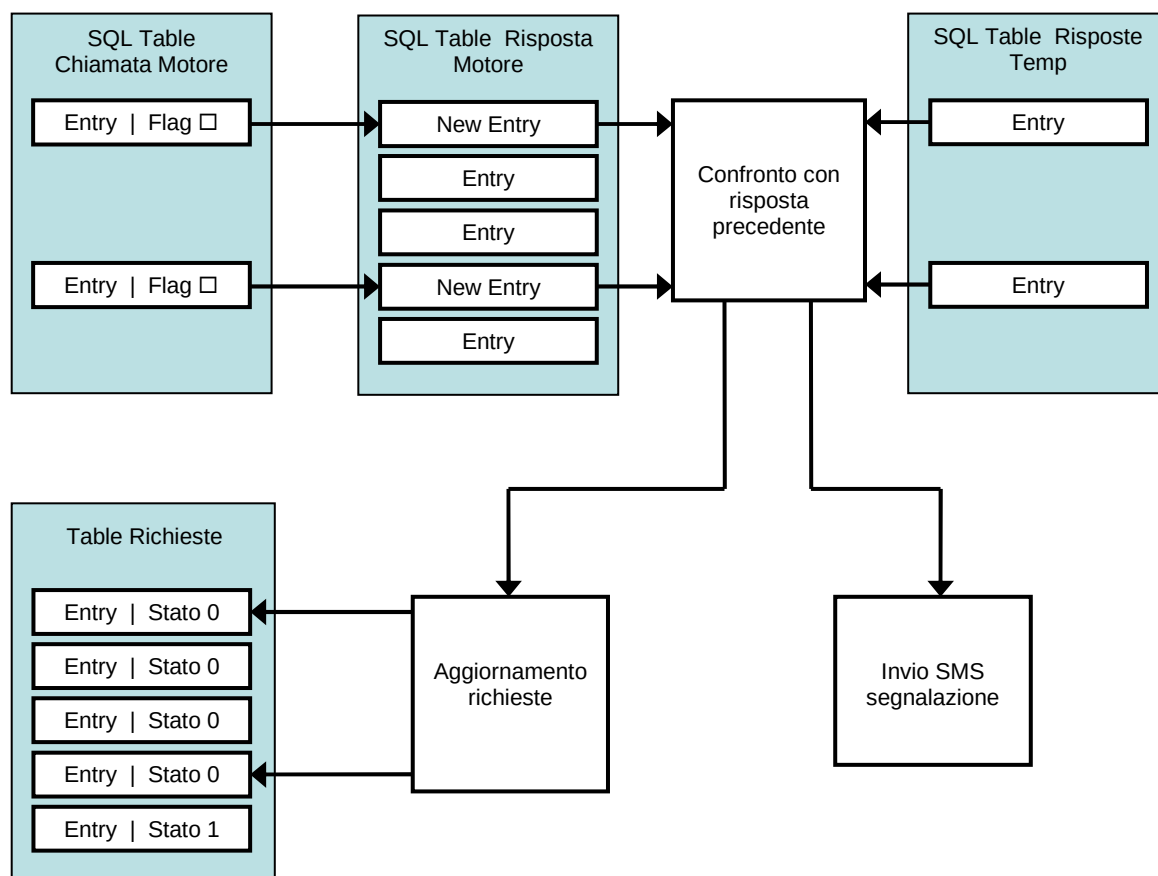


Fig. 90 - Ricalcolo del percorso e confronto con percorso suggerito in precedenza

Eventi significativi

La generazione di un evento ritardo avviene in seguito alla raccolta da parte del centro di controllo di un dato di posizione GPS di un mezzo pubblico.

Confrontando posizione attuale del mezzo con la posizione che dovrebbe avere affidandosi agli orari da tabellone si determina una stima del ritardo che affligge quel mezzo (nella descrizione dell'algoritmo di calcolo del ritardo questa procedura è descritta in maggior dettaglio).

Dato che per monitorare in maniera continua un mezzo esso invierà più volte al minuto le informazioni relative alla sua posizioni GPS, si può capire come sia necessario un'operazione di filtraggio per impedire che un flusso costante di eventi ritardo porti ad una mole di richieste al motore di calcolo del percorso minimo.

Il filtraggio avviene in prima istanza attraverso una valutazione a soglia. Fissata tale soglia a 300 secondi ritardi sotto questo valore sono considerati non significativi, quelli sopra tale soglia viceversa lo sono. La scelta di tale valore di soglia non è casuale: questo è lo stesso valore che il motore per il calcolo del percorso utilizza come margine per poter effettuare un cambio mezzo. Quindi se il mezzo che si sta utilizzando ha un ritardo inferiore ai 300 secondi è ancora possibile raggiungere il secondo mezzo senza perdere la coincidenza e quindi senza deviare dal percorso fissato.

Quando invece il ritardo è oltre tale margine la coincidenza verrà persa e quindi si avrà un impatto su tutto il percorso, quindi in questo caso è opportuno creare l'evento ritardo, segnalare all'intero sistema tale condizione e ricalcolare il percorso per gli utenti che devono utilizzare quel mezzo. Sarà poi il motore, valutando tutti gli eventi significativi

ricevuti fino a quel momento a costruire un nuovo percorso ottimo che andrà poi confrontato con il percorso già definito per determinare quale info inviare all'utente.

Non è tutto. Le considerazioni appena esposte partono dal presupposto di considerare un mezzo in orario che inizia ad accumulare ritardo, ma un mezzo che ha già accumulato più di 300 secondi di ritardo invierà da quel momento in poi segnalazioni al centro di controllo che sono tutte considerate ritardi 300 secondi e quindi eventi significativi che genereranno conseguenti numerose richieste di ricalcolo del percorso.

Tale situazione si aggira memorizzando per ogni mezzo (corsa) l'ultimo evento significativo (si veda Fig. 91).

Server: localhost Database: db_TRIP_TO Tabella: table_eventi_significativi

Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
☐ <u>corsa</u>	varchar(10)	latin1_bin		No			
☐ id_evento_significativo	varchar(20)	latin1_bin		Sì	NULL		
☐ id_ultimo_evento	varchar(20)	latin1_bin		Sì	NULL		

Visualizza per stampa Proponi la struttura della tabella

Aggiungi 1 campo(i) Alla fine della tabella All'inizio della tabella Dopo corsa Esegui

Indici					Spazio utilizzato		Statistiche righe	
Nome chiave	Tipo	Cardinalità	Azione	Campo	Tipo	Utilizzo	Istruzioni	Valore
PRIMARY	PRIMARY	27		corsa	Dati	580 B	Formato	dinamico
Crea un indice su 1 columns Esegui					Indice	2.048 B	Collation	latin1_bin
					Totale	2.628 B	Righe	27
							Lunghezza riga	21
							Dimensione riga	97 B
							Creazione	25 Mar, 2009 at 01:01 PM
							Ultimo cambiamento	07 Mag, 2009 at 05:52 PM

Fig. 91 - Tabella evento significativo

In tal modo una nuova segnalazione è un evento significativo se lo scostamento tra nuovo evento e vecchio evento è maggiore di 300 secondi. Quindi un evento è significativo se il ritardo di un mezzo cresce o si riduce di 300 secondi rispetto all'ultimo evento significativo. Questo approccio è valido perché si considera che il motore di calcolo dei percorsi, una volta memorizzato un evento ritardo significativo, continuerà a fornire lo stesso percorso ottimale a meno che negli orari non ci sia uno scostamento oltre il margine di sicurezza.

Procedendo in questo modo un mezzo che continua ad accumulare ritardo invia delle segnalazioni che diventeranno eventi significativi al massimo una volta ogni 5 minuti. In realtà tale limite non dovrebbe essere raggiunto in quanto si può immaginare che un mezzo impieghi più di 5 minuti per accumulare un ritardo di 5 minuti (questo caso specifico si verifica solo se il mezzo resta immobile per 5 minuti).

Da notare che la cadenza degli eventi significativi non si traduce direttamente nella cadenza di segnalazioni (SMS) verso l'utente. Infatti l'utente viene avvertito solo nel caso che i ritardi portino a:

- Un ritardo che si propaga fino all'arrivo (es: l'utente non deve modificare il percorso visto che non ne esistono di migliori, ma visti i ritardi dei mezzi arriverà in ritardo rispetto al previsto);
- Un ritardo che implica il cambio del percorso ottimale (es: una linea di un autobus è fortemente ritardata, per non perdere il treno all'utente viene indicato un percorso alternativo che utilizza un altro mezzo per raggiungere la stazioni dei treni)

In altri casi gli eventi non portano ad alcun effetto di sorta sul viaggio dell'utente (es: l'utente è su un tram che arriva in stazione dei treni 20 minuti prima della partenza del treno che vuole prendere, se il tram accumula anche 10 minuti di ritardo, non ci saranno ripercussioni sul viaggio dell'utente e quindi il sistema non lo avverte).

Costruzione nuovo file richiesta

La generazione di una nuova richiesta per la generazione di un percorso ottimale parte dalla costruzione di un nuovo file di richiesta.

Si possono immaginare tre possibili casi di evento che impatta su un percorso che influiscono in maniera differente sulla creazione di un nuovo file di richiesta:

1. Il sistema viene informato di uno o più eventi significativi prima che l'utente effettui la richiesta di informazioni sul viaggio
2. Il sistema viene informato di uno o più eventi significativi nel periodo che intercorre tra la prenotazione del viaggio e l'effettivo orario di partenza dell'utente
3. Il sistema viene informato di uno o più eventi significativi dopo la partenza dell'utente per il viaggio registrato.

Le conseguenze sono:

1. L'utente riceve indicazioni per il viaggio già influenzate da tali eventi. Per questi eventi in particolare non si procede dunque a nessun ricalcolo. Si procederà al ricalcolo solo per altri eventi significativi che possono arrivare in seguito e quindi ricadono negli altri due casi
2. Il sistema effettua il ricalcolo del percorso. In questo caso semplicemente riutilizza il file di richiesta originario. In base al confronto tra nuovo percorso suggerito e percorso già definito il sistema procederà ad avvertire o meno l'utente.
3. Il sistema effettua il ricalcolo del percorso. In questo caso il sistema deve tener conto del fatto che l'utente ha già iniziato il suo viaggio. In questo caso ci sono due ulteriori possibilità e il file per la richiesta di ricalcolo deve essere ricostruita opportunamente in base alla condizione dell'utente:
 - a. L'utente non si trova su un mezzo, ma essendo all'interno di un percorso costruito con movimenti su mezzi pubblici, si trova nei pressi di una fermata/stazione. In tal caso il file di richiesta viene modificato imponendo come partenza quella fermata e come orario di partenza l'ora attuale
 - b. L'utente si trova su un mezzo. In tal caso si modifica il file di richiesta indicando appunto tale condizione (un opportuno campo del file XML indica se la partenza avviene da un punto geografico o a bordo di un mezzo), indicando inoltre su quale corsa si trova l'utente.

Naturalmente le considerazioni di questo terzo caso partono dal presupposto che l'utente stia effettivamente seguendo il percorso suggerito all'origine. Solo come evoluzione futura si può immaginare l'utente che possiede un terminale dotato di modulo GPS e per il quale è possibile risalire all'effettiva posizione anche nel caso non segua il percorso suggerito inizialmente.

Pagine di gestione dello script

Per poter monitorare e controllare il funzionamento dello script che gestisce il realtime è stata creata una pagina web (accessibile dal portale TRIP in un'area riservata agli amministratori del sistema).

In questa pagina, visibile in Fig. 92, è disponibile un pulsante per avviare/bloccare tutti i processi relativi allo script. Lo stato di attività (attivo o non attivo) dello script è visibile graficamente attraverso una label colorata in verde (attivo) o giallo (non attivo).

In questa pagina è anche possibile seguire le operazioni svolte dallo script leggendo l'output dei log in un formo costantemente aggiornato.

Da questa pagina si è anche in grado di generare un evento di ritardo fittizio. Attraverso opportuno form è possibile scegliere la corsa su cui intervenire, data dell'evento e ammontare del ritardo da applicare in secondi. Premendo il pulsante Genera XML Evento viene costruito un file XML strutturato esattamente come se l'evento si fosse generato da un effettivo ritardo del mezzo. Tale file viene quindi inoltrato al sistema TRIP che lo processerà opportunamente.

Uno screenshot della pagina di gestione dello script all'interno del portale TRIP è riportata in Fig. 92.

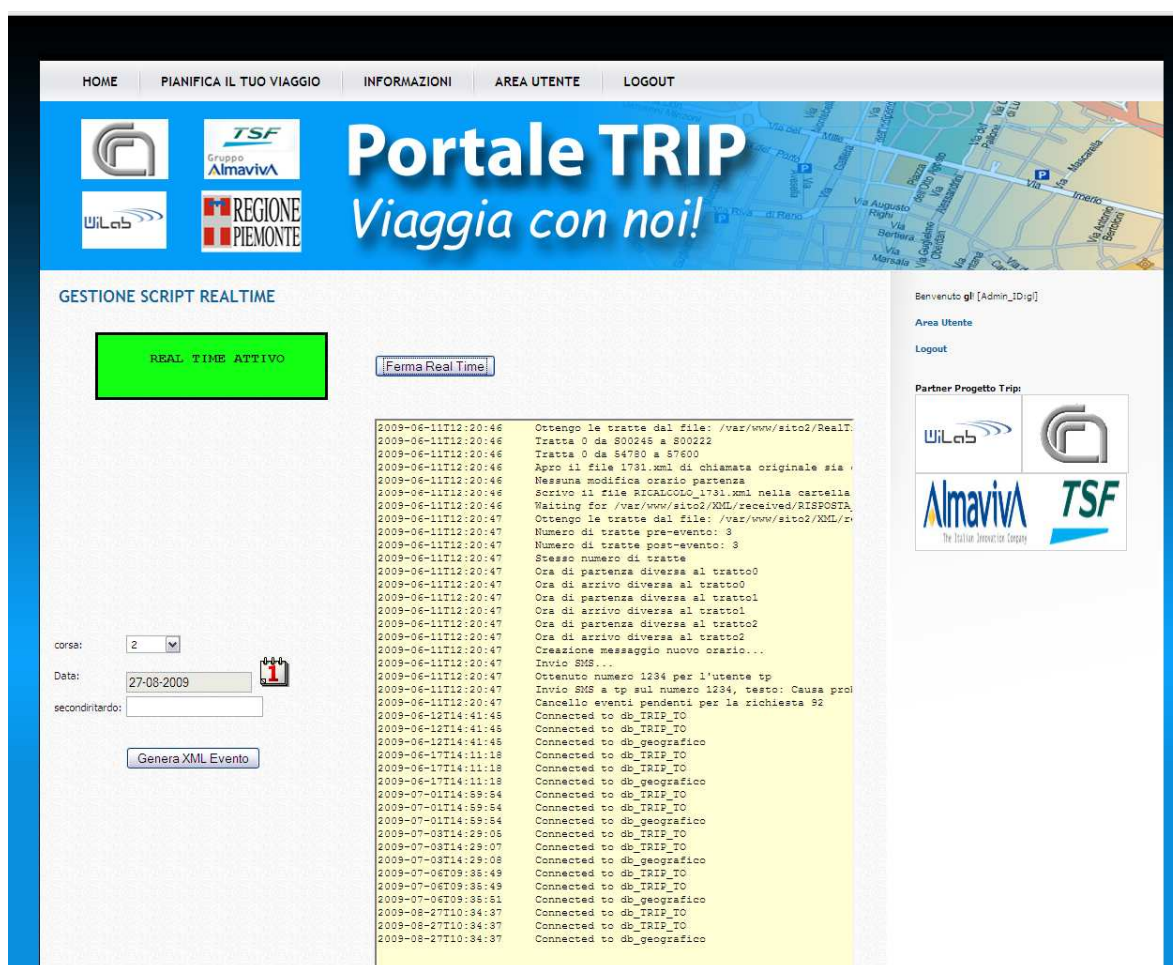


Fig. 92 - Screenshot della pagina del Portale TRIP per il controllo dello script realtime

Scenario tipico

Lo scenario tipico in cui si attivano le funzionalità di controllo degli eventi in real time prevede il coinvolgimento di tutte le strutture che compongono l'infrastruttura TRIP.

9. Si prevede che almeno un utente del sistema registri un percorso su cui vuole avere informazioni aggiornate in real time.
10. Le posizioni di tutti i mezzi pubblici sono monitorate costantemente attraverso un modulo GPS in grado di inviare dati verso un Centro di Controllo.
11. Il centro di controllo aggrega i dati dai mezzi e attraverso l'analisi di tali dati è in grado di fornire informazioni su eventi di ritardo (effettivi e/o presunti).
12. La generazione di un evento di ritardo implica un aggiornamento degli orari dei mezzi per tener conto dell'effettiva posizione e velocità dei mezzi: dopo l'aggiornamento eventuali richieste terranno in considerazione tali ritardi.
13. Lo script di gestione real-time controllerà quindi se esistono dei percorsi registrati dagli utenti che sono affetti dai ritardi segnalati; in caso affermativo si procede a valutare se il percorso registrato è ancora il percorso ottimale.
14. Una nuova richiesta viene quindi inoltrata al motore di calcolo del percorso ottimale, eventualmente tenendo conto del fatto che l'utente può essere già in viaggio e quindi trovarsi in una posizione di partenza diversa o addirittura già a bordo di un mezzo.
15. La nuova soluzione fornita viene confrontata con quella già messa a disposizione dell'utente. Se la differenza tra le due è sostanziale, il sistema si occupa di informare l'utente.
16. Generalmente il sistema prevede di inviare all'utente un SMS contenente le informazioni sul nuovo percorso da seguire in modo da contrastare l'effetto dell'evento.

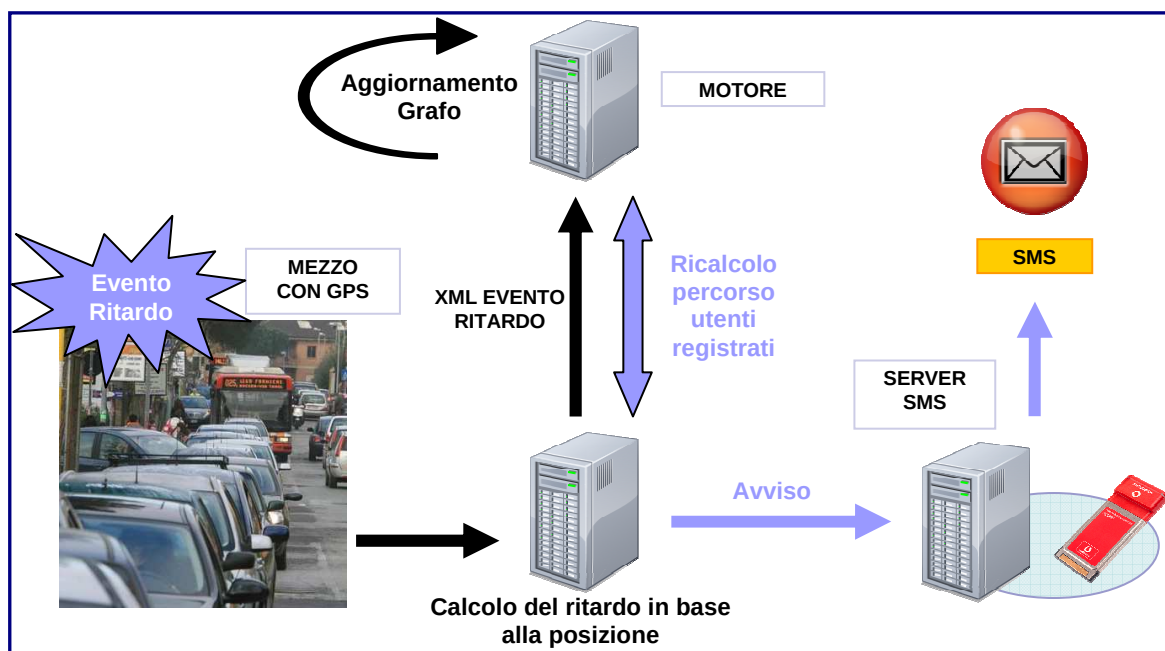


Fig. 93 - Scenario tipico per lo script realtime

ALTRI SOTTOSISTEMI

Facendo riferimento alla illustrazione riportata che ne evidenzia i legami funzionali, vengono descritti i sottosistemi di TRIP relativi al calcolo del percorso e al modulo amministrativo.

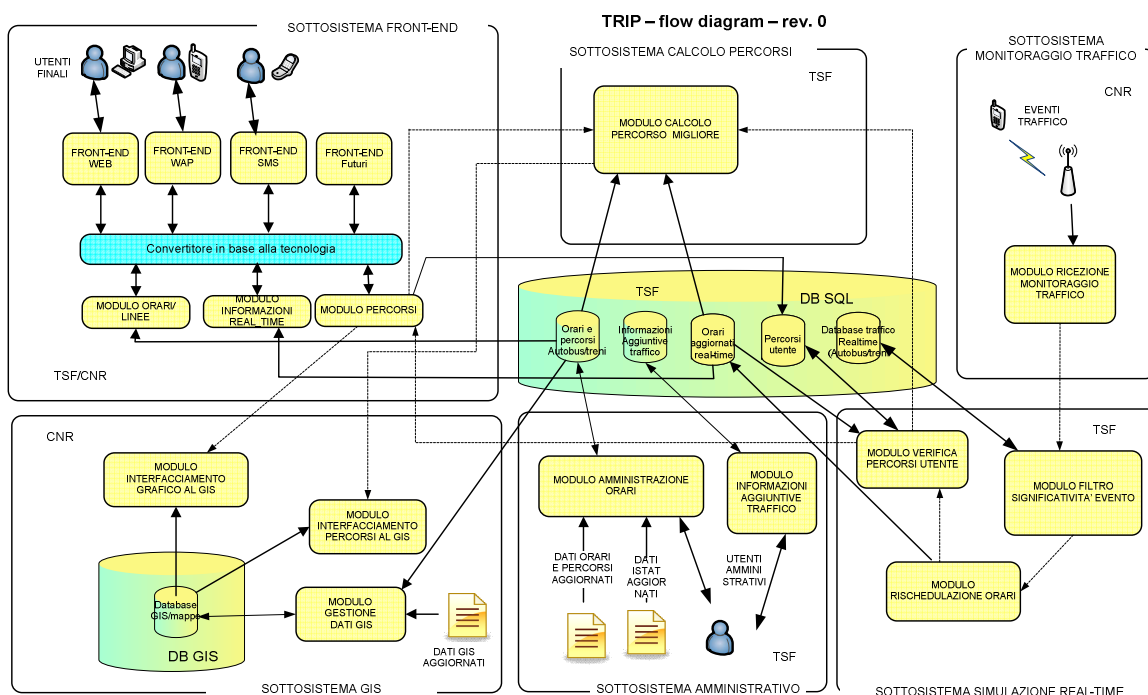


Fig. 94 - Architettura generale della piattaforma TRIP

Sottosistema calcolo percorsi

Il sottosistema “calcolo dei percorsi” realizza il motore di ricerca dei percorsi ottimi in un contesto di trasporto pubblico multi operatore e multi vettore. Per far questo considera tutti i tipi di vettore (Autobus, Treno, Tram, Metro, etc.) gestiti dai diversi operatori che integra in un grafo in cui ne rappresenta la rete complessiva.

Il motore, derivato dall’algoritmo di Dijkstra opportunamente modificato per adattarlo ai requisiti di TRIP, ricerca i percorsi ottimi in funzione del tempo di percorrenza e, a parità di tempi, privilegia quelli con il minor numero di cambi.

Mentre l’algoritmo di Dijkstra, nella sua forma originale, opera su di un grafo in cui ricerca il percorso ottimo a partire da un suo nodo specifico, l’algoritmo TRIP amplia questa impostazione considerando le soluzioni comprese in un range temporale e gestendo la partenza e l’arrivo con un insieme di fermate (fermate limitrofe) invece di una singola fermata.

Una volta fissati questi confini, l'algoritmo ricerca tutte le soluzioni, comprese tra i due insiemi di fermate di partenza e di arrivo, che siano successive all' "Orario di partenza al più presto" e che siano compatibili con l' "Orario di arrivo al più tardi". Tra tutte le soluzioni trovate, l'algoritmo mantiene solo quelle che non sono sicuramente surclassate da un'altra soluzione migliore, le condizioni in base a cui una soluzione viene considerata surclassata sono descritte in dettaglio nel documento "TRIP-ASW-2008002 V. 1.0 – Specifiche del motore per il calcolo dei percorsi".

Tra tutte le soluzioni non surclassate il motore dichiara vincenti quelle che soddisfano le seguenti condizioni:

1. minimo numero di corse utilizzate;
2. minimo numero di trasferimenti a piedi a parità di numero corse.

Il motore può venire attivato:

1. in modalità door to door da una richiesta utente che proviene dal sottosistema "front-end";
2. in modalità real-time dal sottosistema "simulazione real time".

Un calcolo del percorso in modalità door to door nasce da una richiesta utente raccolta dal sottosistema "front-end", che, interagendo con il sottosistema "GIS", individua tutte le fermate limitrofe al punto di partenza e al punto di arrivo richiesti dall'utente, ogni fermata è corredata dal tempo necessario a raggiungerla. Il motore provvede, allora, a creare sul grafo due punti fittizi, un punto di partenza e un punto di arrivo, ed esegue la ricerca delle soluzioni migliori.

Un calcolo del percorso in modalità real time nasce da una o più variazioni apportate al grafo del motore di TRIP in seguito al ricevimento di eventi raccolti sul campo dal sottosistema "monitoraggio traffico" e filtrati dal sottosistema "simulazione real time". Una volta aggiornati gli orari i percorsi utente, già calcolati in modalità door to door, vengono verificati (ricalcolati) e, se necessario, sostituiti da nuovi percorsi ottimali che il sottosistema "front-end" provvede a comunicare agli utenti interessati.

Documenti relativi al sottosistema:

- TRIP-ASW-2008001 V. 5.0 – TRIP Base dati sottosistema calcolo del percorso
- TRIP-ASW-2008002 V.1.0 – Specifiche del motore per il calcolo dei percorsi
- TRIP-RST-2009001 V. 1.0 – Requisiti di integrazione del sistema
- TRIP-MAN-2009001 V. 1.0 – Manuale installazione del software

Sottosistema "simulazione real-time"

Il sottosistema gestisce gli eventi di variazione agli orari statici, orari che sono la base della modalità di calcolo door to door.

Gli eventi ricevuti provengono sia come dati raccolti sul campo dal sottosistema "monitoraggio traffico" sia come variazioni inserite dagli operatori quando, per qualche motivo, devono alterare gli orari pianificati.

Il sottosistema effettua su questi eventi una azione di selezione e filtro trattenendo solo quelli significativi.

Utilizzando gli eventi selezionati il sottosistema modifica gli orari relativi alla o alle fermate impattate e ne propaga le conseguenze su tutte le fermate a valle ricalcolando gli orari di passaggio dei diversi mezzi e aggiornando il grafo real-time. Quest'ultimo è una copia del grafo base utilizzato dal motore di ricerca che evolve con l'evolversi della situazione reale del traffico. Periodicamente il grafo real time viene riallineato al grafo statico.

Documenti relativi al sottosistema:

- TRIP-ASW-2008001 V. 5.0 – TRIP Base dati sottosistema calcolo del percorso
- TRIP-ASW-2008005 V.1.0 – Specifiche del sottosistema real time

Sottosistema "amministrativo"

Il sottosistema è orientato agli amministratori di TRIP e agli operatori di trasporto pubblico locale e consente la creazione e la manutenzione della base dati del sistema, che è la base di riferimento per gli altri sottosistemi, per quanto riguarda:

- gestione della sicurezza;
- gestione delle reti;
- gestione degli orari;
- gestione dei dati di sistema.

Il sottosistema fornisce agli attori coinvolti nel funzionamento di TRIP strumenti per:

- acquisire in modo massivo e controllato i dati di rete e orario dagli operatori di trasporto pubblico attraverso dei formati standard di comunicazione;
- acquisire in modo puntuale e controllato le variazioni alla base dati attraverso applicazioni web based;
- interagire con il modulo GIS per consentire la designazione diretta delle linee.
- rendere disponibili i dati acquisiti al motore senza interferire nella sua operatività

Documenti relativi al sottosistema:

- TRIP-ASW-2008001 V. 5.0 – TRIP Base dati sottosistema calcolo del percorso
- TRIP-ASW-2008004 V. 1.0 – TRIP Base dati sottosistema amministrativo
- TRIP-ASW-2008003 V. 2.0 – Sottosistema amministrativo analisi di dettaglio
- TRIP-ASW-2008004 V. 1.0 – TRIP Base dati sottosistema amministrativo
- TRIP-SUS-2008001 V. 2.0 – Tracciati per lo scambio dati aziende TPL/TRIP
- TRIP-SUS-2008002 V. 1.0 – Interazione utente/sistema nelle funzioni on-line
- TRIP-SPF-2008001 V. 2.0 – Caricamento massivo degli orari
- TRIP-SPF-2008002 V. 1.0 – Validazione degli orari
- TRIP-SPF-2008003 V1.0 – Trasferimento dei dati validati
- TRIP-SPF-2008004 V. 1.0 – Calcolo distanze tra impianti fisici
- TRIP-SPF-2008006 V. 1.0 – Aggiornamento dati Istat
- TRIP-SPF-2008007 V. 2.0 – Funzioni amministrative via web
- TRIP-SPF-2008008 V. 1.0 – Estrazione dati per produzione orari in PDF
- TRIP-SPF-2008009 V. 1.0 – Cancellazione orari

MANUALE DI UTILIZZO DEL PORTALE TRIP

Il progetto TRIP si interfaccia verso il mondo esterno e quindi verso l'utente finale attraverso un portale web in grado di supportare tutti i servizi di infomobilità avanzati previsti. Il portale attraverso una serie di pagine web che sfruttano le più avanzate tecnologie e linguaggi di programmazione come AJAX e JAVASCRIPT, è in grado di fornire una solida base a tutti i servizi implementati, ma grazie alla sua solidità e flessibilità anche a quelli futuri. Il sistema inoltre sfrutta una base dati geografica proprietaria e soluzioni di navigazioni delle mappe assolutamente all'avanguardia. Funzioni di completamento automatico e di ricerca avanzata completano il quadro degli strumenti assolutamente innovativi che sono stati utilizzati in questo progetto.

Portale di Accesso

Il portale principale, mostrato in Fig. 95, è stato disegnato e implementato in modo da far risaltare l'obiettivo del portale come supporto alla mobilità pubblica. Attraverso il portale principale, l'utente può accedere ai servizi erogati dal Portale del Progetto TRIP. La pagina principale è raggiungibile semplicemente digitando nella barra degli indirizzi del proprio browser installato sul PC dell'utente, l'indirizzo del sito.

Attualmente non è stata effettuata la registrazione su nessun dominio. Il portale è disponibile in locale attraverso il web server messo a disposizione dal progetto. Ovviamente è immediatamente renderlo disponibile al pubblico fornendo uno spazio con dominio registrato.

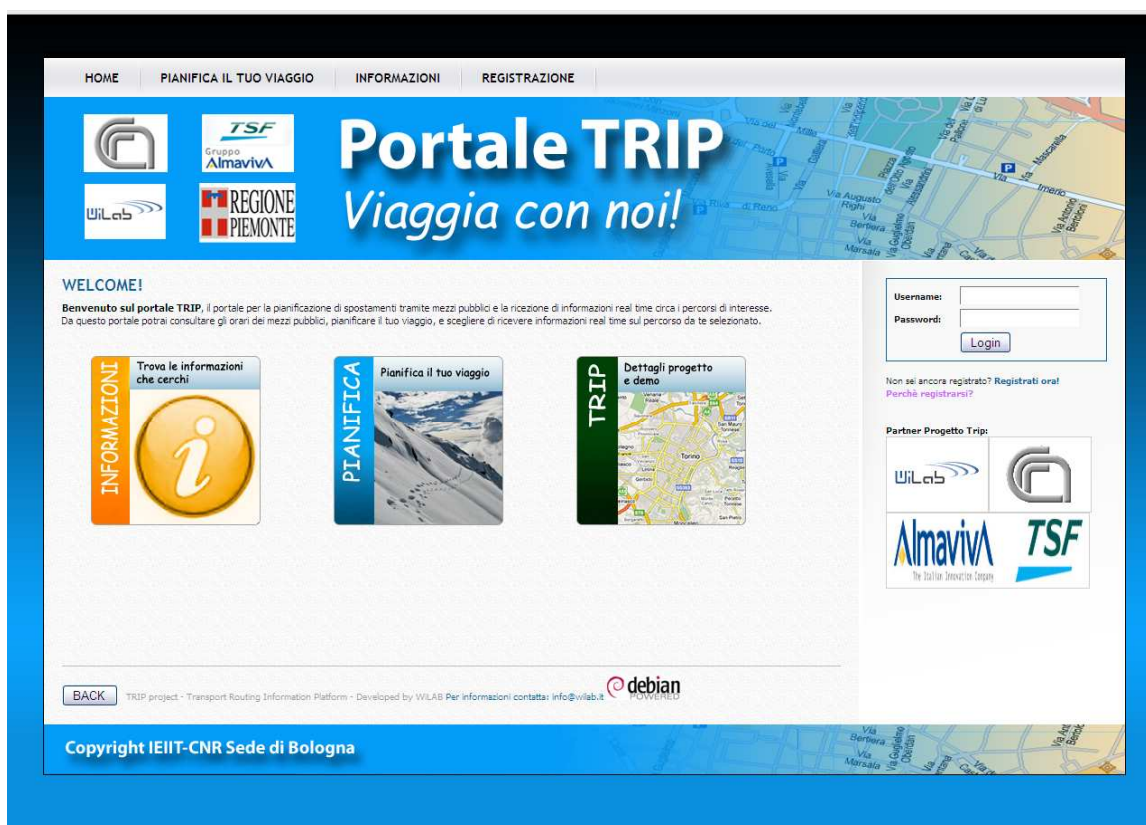


Fig. 95 - Homepage del portale TRIP

Il portale principale permette prevalentemente di spostarsi nei tre diversi sottoportali:

- **Informazioni:** per avere informazioni sul percorso e gli orari di passaggio dei mezzi del Trasporto Pubblico Locale, ma anche sulle segnalazioni di ritardo in tempo reale e tanti altri servizi di informazioni varie.
- **Pianifica:** per programmare un viaggio, fissando partenza e destinazione in tante maniere differenti, per trovare la soluzione di viaggio più comoda e funzionale
- **Dettagli progetto e demo:** per avere spiegazioni sul progetto, sulle tecnologie messe in campo e le soluzioni software adottate.

Ovviamente dal portale principale è possibile anche registrarsi per poter accedere ai diversi servizi personalizzati per l'ottimizzazione dei viaggi in real time.

Portale di Accesso (versione Amministratori)

La versione del portale per gli amministratori che devono provvedere all'inserimento dei dati, alla gestione della base dati e dell'aggiornamento del sito dispone, dopo l'accesso con account di amministratore (inserendo username e password opportune nella finestra di login), di un apposito sottoportale di Management che permette di accedere a diversi servizi di gestione e amministrazione.



Fig. 96 - Homepage del portale TRIP dopo l'accesso come amministratore

Portale di accesso (versione scalata)

Il progetto TRIP per andare incontro alle esigenze di mobilità degli utenti, così come stabilito nel bando e come realizzato nel corso del progetto, prevede la possibilità per gli utenti che accedono da remoto utilizzando terminali mobili con connessione a internet a banda larga, di avere a disposizione un portale scalato per migliorare la leggibilità nei display limitati dei cellulari. In Fig. 97 è riportato la versione scalata del portale principale di accesso.

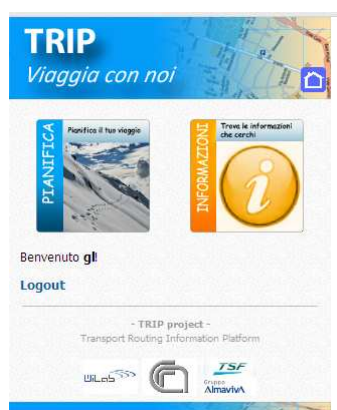


Fig. 97 - Portale in versione scalata

Ovviamente tutti i terminali accedono allo stesso indirizzo internet riservato agli utenti fissi: è il server che si accorge del differente tipo di connessione e decide autonomamente di ridurre e scalare opportunamente i contenuti. Attualmente sono presenti due diverse tipologie di accesso scalato: una con dimensione dello schermo a 240 pixel (terminali con display non troppo grande) e a 320 pixel, così come per l'iPhone della Apple.

Portale di Pianificazione Viaggio

Nella pagina di pianificazione del viaggio è possibile impostare un punto di partenza e di arrivo che possono essere caratterizzati come indirizzo (via, civico, comune, provincia), fermata (nome della fermata, comune, provincia) o punto di interesse (nome del POI, comune, provincia). Nel caso in cui questi campi vengano inseriti manualmente verrà eseguito un controllo sulla stringa immessa, in modo tale da proporre all'utente la soluzione più simile a quella digitata se la stringa inserita non dovesse corrispondere a nessuna già presente nel database. Per ogni tipologia di punto di partenza/arrivo è comunque previsto un inserimento tramite scelta dall'elenco di vie del comune (caso indirizzo), oppure tramite scelta cliccando direttamente sulla mappa geografica in cui sono messi in evidenza i possibili punti (caso fermata e POI). In Fig. 98 è riportato il portale nella sua versione standard, ovvero con la scelta di tutti i parametri di viaggio non visibili.

The screenshot shows the 'Portale TRIP' web application. At the top, there is a navigation bar with links: HOME, PIANIFICA IL TUO VIAGGIO, INFORMAZIONI, and REGISTRAZIONE. Below this, the header features logos for WiLab, Gruppo Almaviva, and REGIONE PIEMONTE, alongside the title 'Portale TRIP' and the slogan 'Viaggia con noi!'. The main content area is titled 'PIANIFICA IL TUO VIAGGIO!' and is divided into three columns: 'Partenza', 'Arrivo', and 'Parametri'. Each column contains dropdown menus for Province and Comune, a search field, and a list of options: 'Scegli un indirizzo', 'Scegli una fermata', 'Scegli un punto di interesse', and 'Scegli un punto sulla mappa'. The 'Parametri' column includes a date field set to '26-08-2009' and radio buttons for 'Partenza alle:' and 'Arrivo prima delle:'. A 'Login' form is located on the right side. At the bottom, there is a 'Calcola Percorso!' button, a 'BACK' button, and a footer with copyright information for IEIT-CNR Sede di Bologna and a 'powered by' logo for debian.

Fig. 98 - Form di pianificazione del viaggio

Nel form della pagina di pianificazione, è presente anche la possibilità di specificare alcuni parametri avanzati del viaggio, cliccando su Ulteriori Preferenze. L'utente potrà a quel punto scegliere:

- Data: data per la quale si vuole cercare una soluzione di viaggio (se non inserita, di default viene assunta la data in cui si sta effettuando la ricerca);

- Partenza alle/Arrivo prima delle: orario dopo il quale si vuole partire o entro il quale si vuole arrivare (se non inserito, di default viene assunto l'orario in cui si sta effettuando la ricerca);
- Aziende preferite: elenco delle aziende di trasporto pubblico presenti sul territorio; l'utente può preferire alcune aziende rispetto ad altre in base ad abbonamenti o promozioni (se non selezionate, di default vengono assunte tutte);
- Scelta dei mezzi e delle combinazioni: elenco dei tipi di mezzi di trasporto pubblico presenti sul territorio; l'utente può preferire alcuni mezzi rispetto ad altri (se non selezionati, di default vengono assunti tutti);
- Disponibilità per eventuale tragitto a piedi: l'utente può scegliere di limitare il più possibile gli spostamenti a piedi (se non selezionato, di default viene accettato);
- Scelta del massimo numero di cambi: l'utente può preferire soluzioni di viaggio che non prevedano più di un certo numero di cambi di mezzo (se non selezionato, di default viene assunto un numero illimitato di cambi).

In Fig. 99 è riportato il portale nella sua versione avanzata, ovvero con la scelta di tutti i parametri di viaggio visibili.

The screenshot displays the 'Portale TRIP' web application interface. At the top, there is a navigation bar with links: HOME, PIANIFICA IL TUO VIAGGIO, INFORMAZIONI, and REGISTRAZIONE. Below this, the main header features logos for Gruppo Almaviva, REGIONE PIEMONTE, and WiLab, alongside the title 'Portale TRIP' and the slogan 'Viaggia con noi!'. The main content area is titled 'PIANIFICA IL TUO VIAGGIO!' and is divided into three columns: 'Partenza', 'Arrivo', and 'Parametri'. The 'Partenza' column includes dropdown menus for 'Provincia' (Torino) and 'Comune' (Torino), a search field with 'Corso Francia' entered, and a list of options to choose a starting point (indirizzo, fermata, punto di interesse, or punto sulla mappa). The 'Arrivo' column has similar dropdowns for 'Provincia' and 'Comune', a search field, and options to choose an arrival point. The 'Parametri' column contains a date field set to '26-08-2009', radio buttons for 'Partenza alle:' and 'Arrivo prima delle:', a button for 'Ulteriori Preferenze', checkboxes for preferred companies (RF, GTT, Trenitalia), checkboxes for transport modes (Autobus, Tram, Metro, Treno), radio buttons for 'Disponibilità per eventuale tragitto a piedi' (SI, No), a dropdown for 'Scelta del massimo numero di cambi' (quellunque), and checkboxes for search type (Statica, Dinamica). A 'Calcola Percorso!' button is at the bottom left. On the right side, there is a login section with 'Username' and 'Password' fields, a 'Login' button, and a registration link. Below the login section, it lists 'Partner Progetto Trip' with logos for Almaviva and TSF. The footer includes a 'BACK' button, project information, and the Debian logo.

Fig. 99 - Form di pianificazione del viaggio con preferenze avanzate

Per effettuare la scelta della provincia, sia per la partenza che per l'arrivo del proprio viaggio, l'utente deve utilizzare il menù a tendina apposito. Ovviamente è possibile

specificare una provincia diversa per partenza e arrivo. La scelta di una provincia aggiornerà dinamicamente il contenuto del menù a tendina con la lista dei comuni, appartenenti alla provincia scelta.

In Fig. 100 sono riportati due esempi di scelta attraverso il menù a tendina della provincia di Torino e del comune di Torino stesso.

Fig. 100 - Scelta di provincia e comune

A questo punto per fissare con precisione il punto di partenza e/o arrivo è necessario introdurre l'indirizzo esatto o sceglierlo da un elenco (tipo stardario), oppure selezionare una fermata, sia direttamente sulla mappa che da un apposito elenco, oppure scegliere un

punto di interesse (cinema, teatro, museo, ..) o selezionare le coordinate GPS di un punto preciso cliccando direttamente sulla mappa.

Nel caso di inserimento della via direttamente da tastiera, il portale effettua un controllo costante, carattere per carattere inserito, per verificare la somiglianza del testo inserito con le vie memorizzate ed effettivamente disponibili nel comune. Il sistema, molto sofisticato e preciso è di grande importanza in quanto permette di risparmiare tempo, in quanto la via corretta viene suggerita e completata ancor prima della sua effettiva digitazione o in caso di errore in fase di inserimento (errore ortografico o di battitura).

Nella schermata riportata di seguito è illustrato il funzionamento.

The screenshot displays the 'Portale TRIP' web application. At the top, there is a navigation bar with links: HOME, PIANIFICA IL TUO VIAGGIO, INFORMAZIONI, and REGISTRAZIONE. Below this, the main header features logos for WiLab, Gruppo Almaviva, and REGIONE PIEMONTE, alongside the title 'Portale TRIP' and the slogan 'Viaggia con noi!'. The main content area is titled 'PIANIFICA IL TUO VIAGGIO!' and is divided into three sections: 'Partenza', 'Arrivo', and 'Parametri'. The 'Partenza' section includes dropdown menus for 'Provincia' (Torino) and 'Comune' (Torino), a text input for 'Inserisci un campo di ricerca:', and a list of suggested addresses starting with 'Via Rom'. The 'Arrivo' section has similar dropdowns and a search input. The 'Parametri' section shows a date picker set to '26-08-2009' and radio buttons for 'Partenza alle:' and 'Arrivo prima delle:'. On the right side, there is a login section with 'Username:' and 'Password:' fields, a 'Login' button, and a link to 'Registrati oral'. Below the login section, there are logos for 'Partner Progetto Trip: WiLab, Gruppo Almaviva, and TSF'. At the bottom, there is a 'Copyright IEIT-CNR Sede di Bologna' notice and a 'BACK' button.

Fig. 101 - Introduzione di un campo di ricerca

Come anticipato, è possibile scegliere la via cliccando direttamente su un elenco alfabetico di tutte le vie fatto a modo di stradario: cliccando sul bottone “Scegli un indirizzo di partenza/arrivo” comparirà una finestra come quella riportata in Fig. 102.

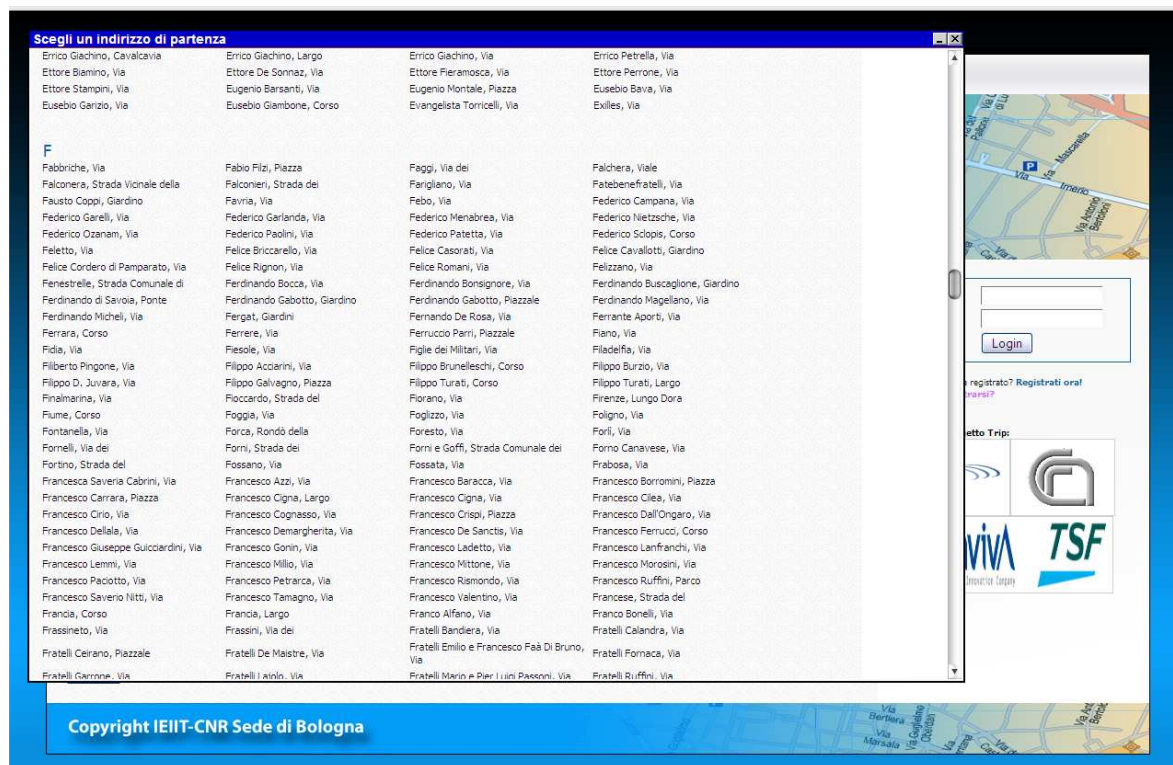


Fig. 102 - Scelto di un indirizzo di partenza da una lista

Se l'utente decide di voler partire/arrivare da una fermata, può inserire il nome o il codice della fermata direttamente nel campo di ricerca. Anche in questo caso è attivo il meccanismo di completamento e correzione automatica. L'alternativa è cliccare il bottone "Scegli una fermata di partenza/arrivo" e scegliere se si vuole selezionare la fermata direttamente dalla mappa (l'utente potrebbe conoscere approssimativamente la zona in cui andare, senza conoscere precisamente le fermate) o da un elenco di fermate con i rispettivi indirizzi a cui si trovano. Di seguito è riportata la maschera di scelta per la fermata.

Fig. 103 – Doppia possibilità per la scelta di una fermata di partenza

Nel caso l'utente decida di partire/arrivare ad una fermata e la voglia indicare direttamente sulla mappa, si aprirà una finestra con la mappa navigabile, centrata sul comune scelto, in cui sono evidenziate le linee che transitano in zona. Aumentando il livello di zoom è possibile visualizzare anche le singole fermate delle linee e cliccando direttamente sulla cartina sulla fermata di interesse è possibile, oltre a leggere alcune informazioni sulla fermata (indirizzo, orari, linee in transito), anche fissare la fermata come punto di partenza/arrivo. Nel caso in cui l'utente voglia scegliere una fermata in un elenco, verrà fornito un apposito form, in cui sono raccolte e catalogate tutte le fermate del comune. In Fig. 104 sono riportate i due casi precedentemente discussi.

HOME PIANIFICA IL TUO VIAGGIO INFORMAZIONI REGISTRAZIONE

 **Portale TRIP**
Viaggia con noi!

PIANIFICAZIONE DEL PERCORSO SCEGLIENDO LE FERMATE

In questo portale puoi ottenere informazioni sulle fermate dei mezzi pubblici.
Clicca sulle fermate per scegliere la tua fermata direttamente dalla mappa!

Menù delle opzioni

Provincia: Torino
Comune: Torino

Visualizza i Poi
Cambia il Comune!

Mostra:

- 2 GTT002
- 9 GTT009
- 52 GTT052
- CAN GTTCanavesana
- CHA GTTCiriLanzo
- 1 GTTHT01
- TOM Torino-Milano
- TO-SU Torino-Susa

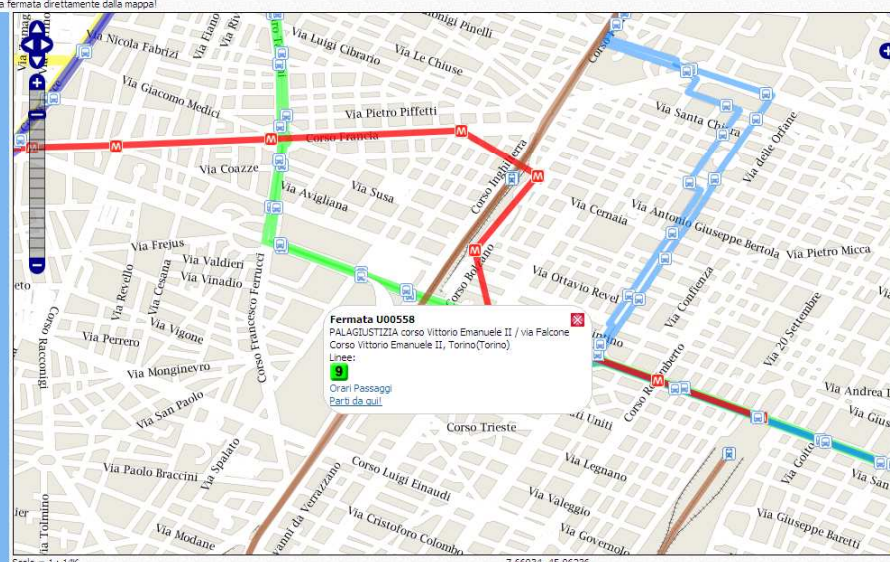
Cerca una fermata (codice): Cerca!

Visualizza lo schema Linee: 

Viaggia da una fermata

Arriva a una fermata

Help Mappa



Scale = 1 : 14K 7.66034, 45.06236

Fermate

ELENCO DELLE FERMATE TORINO (TORINO)

Fermata	Descrizione	Indirizzo
GTT004	STAZIONE TORINO LINGOTTO	Via Mario Pannunzio, Torino (Torino)
GTT005	STAZIONE TORINO PORTA SUSA SOTTERRANEA	Piazza 18 Dicembre, Torino (Torino)
GTT006	STAZIONE TORINO PORTA SUSA	Piazza 18 Dicembre, Torino (Torino)
GTT007	STAZIONE TORINO DORA	Corso Mortara, Torino (Torino)
GTT008	STAZIONE TORINO STURA	Strada Vicinale dell'Abbadia di Stura, Torino (Torino)
GTT021	STAZIONE MADONNA DI CAMPAGNA	Via Lucento, Torino (Torino)
M00003	MARCHE	Corso Francia, Torino (Torino)
M00004	MASSAIA	Corso Francia, Torino (Torino)
M00005	POZZO STRADA	Via Pozzo Strada, Torino (Torino)
M00006	MONTE GRAPPA	Corso Monte Cucco, Torino (Torino)
M00007	RIVOLI	Corso Francia, Torino (Torino)
M00008	RACCONIGI	Corso Francia, Torino (Torino)
M00009	BERNINI	Piazza Lorenzo Bernini, Torino (Torino)
M00010	PRINCIPI D'ACAJA	Corso Francia, Torino (Torino)
M00011	XVIII DICEMBRE	Piazza 18 Dicembre, Torino (Torino)
M00012	PORTA SUSA	Corso Bolzano, Torino (Torino)
M00013	VINZAGLIO	Corso Vittorio Emanuele II, Torino (Torino)
M00014	RE UMBERTO	Corso Re Umberto, Torino (Torino)
M00015	PORTA NUOVA	Corso Vittorio Emanuele II, Torino (Torino)
S00219	STAZIONE TORINO PORTA NUOVA	Via Paolo Sacchi, Torino (Torino)
S00222	STAZIONE TORINO PORTA SUSA	Piazza 18 Dicembre, Torino (Torino)
U00033	FERRARIS corso Vittorio Emanuele II n.93 A	Corso Vittorio Emanuele II, Torino (Torino)
U00034	FERRARIS corso Vittorio Emanuele II / corso Galleo Ferraris	Corso Vittorio Emanuele II, Torino (Torino)
U00035	RE UMBERTO corso Vittorio Emanuele II / via Massena	Corso Vittorio Emanuele II, Torino (Torino)
U00036	RE UMBERTO corso Vittorio Emanuele II / via Parini	Corso Vittorio Emanuele II, Torino (Torino)
U00039	PORTA NUOVA FS corso Vittorio Emanuele II - Porta Nuova F.S.	Corso Vittorio Emanuele II, Torino (Torino)
U00040	PORTA NUOVA FS corso Vittorio Emanuele II fronte Porta Nuova	Corso Vittorio Emanuele II, Torino (Torino)
U00069	PONCHIELLI CAPOLINEA via Monteverdi/via Cimarosa	Via Claudio Monteverdi, Torino (Torino)
U00070	BOLOGNA via Gottardo n.271	Via Gottardo, Torino (Torino)

Copyright IEIT-CNR Sede di Bologna



Fig. 104 - Scegliere una fermata di partenza: da mappa o da lista

Come anticipato è stata implementata anche la possibilità per l'utente di scegliere un punto di interesse (Point of Interest PoI) come un museo, un teatro, un cinema,...come proprio punto di partenza/arrivo. La scelta del PoI viene effettuata direttamente sulla mappa. Anche in questo caso si apre una pagina con la mappa navigabile centrata nel comune di interesse e tutti i PoI visibili. Ovviamente è anche possibile filtrare la visualizzazione dei PoI per categoria, agendo sulla barra laterale in cui è visibile una serie di radio buttons. In Fig. 105 un esempio di un PoI (Stazione di servizio) scelta come partenza.

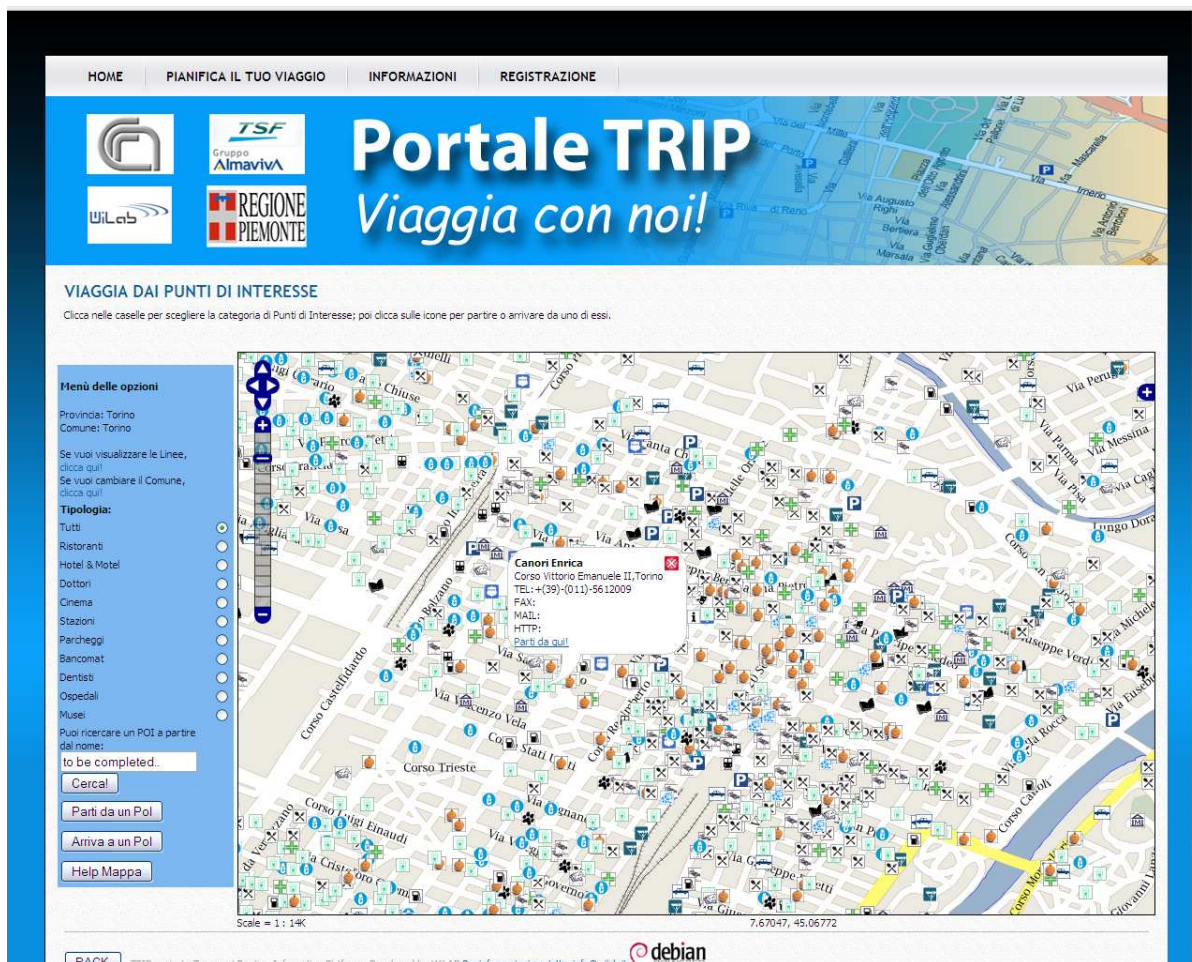


Fig. 105 - Punto di interesse come punto di partenza

L'ultima modalità messa a disposizione dal portale per la scelta del punto di partenza/arrivo è la possibilità di cliccare direttamente sulla mappa e fissare un punto geografico inteso come coordinate longitudine e latitudine. Cliccando sulla mappa navigabile, centrata sul comune di interesse, compare un popup con le coordinate GPS del punto per chiedere conferma all'utente.

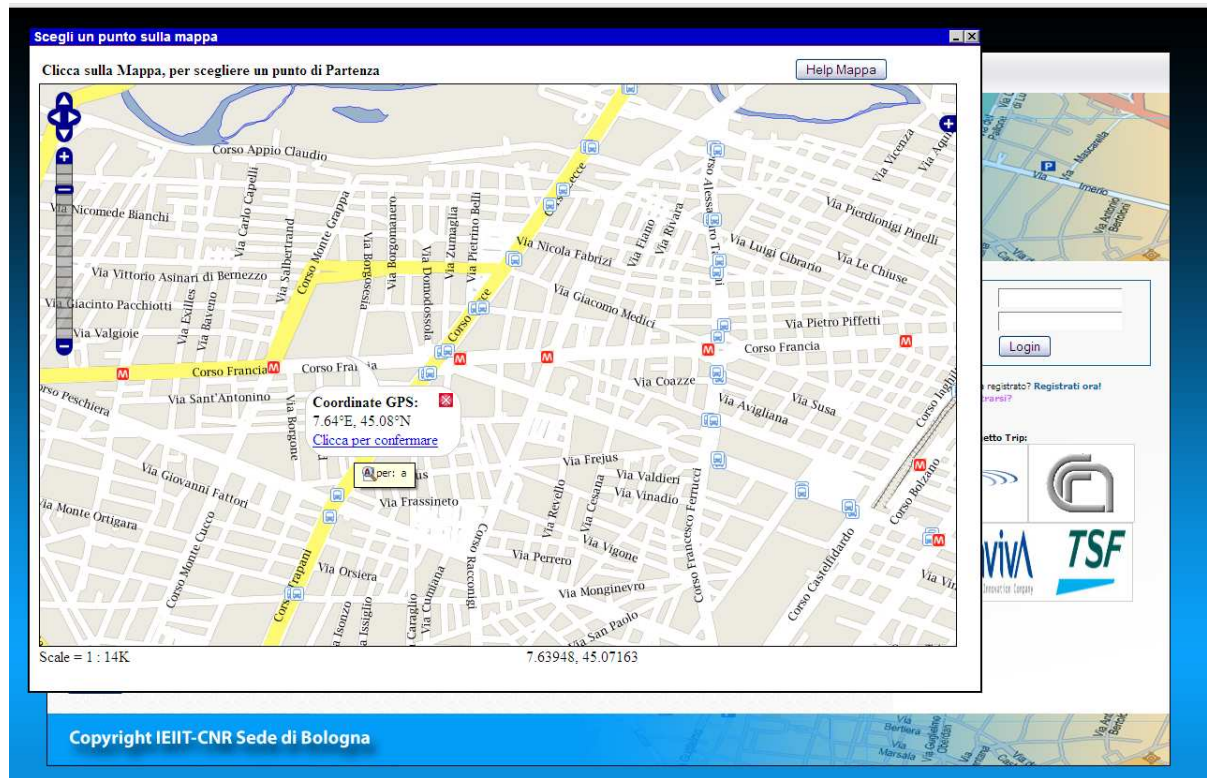


Fig. 106 - Partenza da punto GPS

Una volta che il punto di partenza/arrivo è stato identificato e fissato, compare un bottone (Clicca qui per vedere le fermate vicine) in grado di mostrare direttamente sulla mappa, centrata nel punto esatto di partenza/arrivo rappresentato da una puntina verde/rosso, tutte le fermate di mezzi pubblici. Viene di default evidenziata e quindi aperto il rispettivo popup della fermata più vicina, considerando opportunamente il percorso pedonale, e non solo effettuando un calcolo banale di distanza euclidea. Per tutte le fermate vengono riportate all'interno del popup, attivabile cliccando sulle stesse, una serie di informazioni utili sulle fermate stesse.

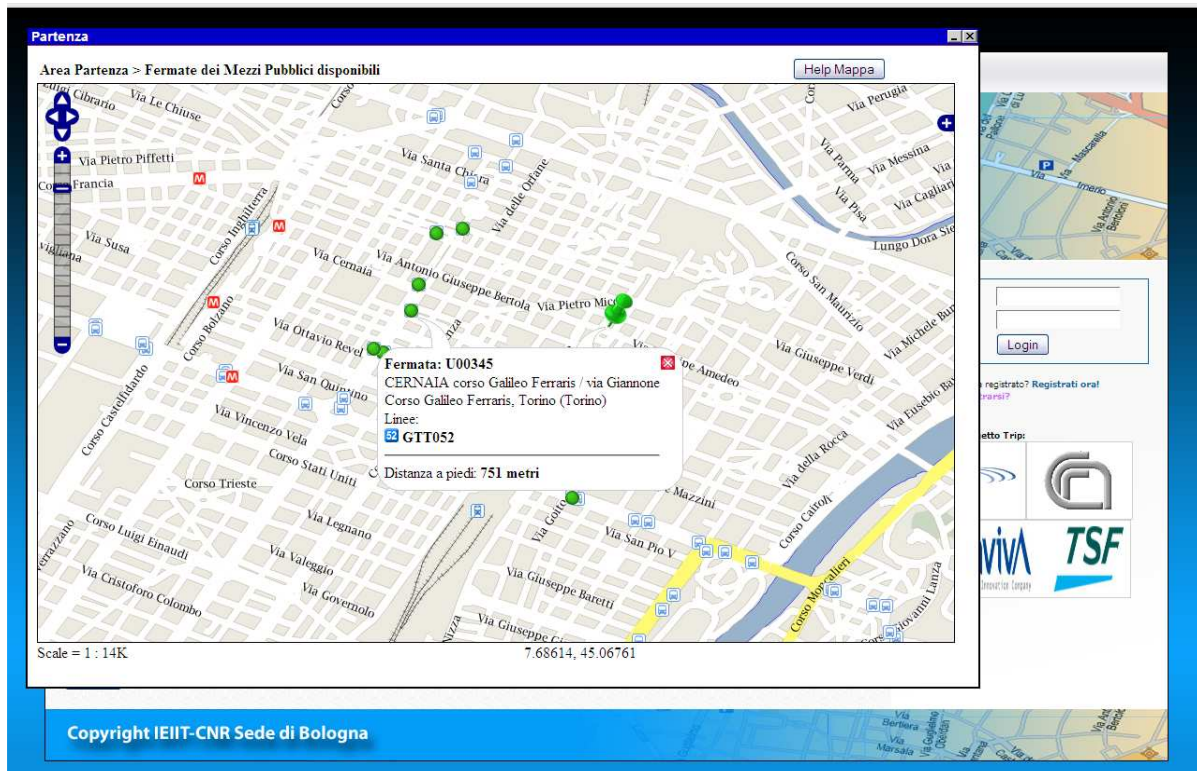


Fig. 107 - Fermate in prossimità del punto di partenza

Un volta inseriti i punti di partenza e arrivo, e scelti i parametri caratterizzanti del viaggio, si può procedere ad inviare la richiesta mediante il tasto “Calcola percorso” che diventerà cliccabile solo dopo che tutti i controlli di correttezza formale sui campi saranno stati verificati.

Portale di Pianificazione Viaggio (versione scalata)

Anche per questo portale è presente la versione scalata per terminali mobili a banda larga con ridotta dimensione del display. In Fig. 108 è riportata la versione a 240 pixel.

The screenshot shows the TRIP web application interface. At the top, there is a header with the text "TRIP" and "Viaggia con noi". Below this, a section titled "PIANIFICA IL VIAGGIO!" contains three main sections: "Partenza", "Arrivo", and "Parametri".

Partenza

Provincia:

Comune:

Inserisci un campo di ricerca:

o scegli un punto sulla mappa:

Arrivo

Provincia:

Comune:

Inserisci un campo di ricerca:

o scegli un punto sulla mappa:

Parametri

Data:

☒ Partenza alle:

☐ Arrivo prima delle:

:

Calcola Percorso!

- TRIP project -
Transport Routing Information Platform

WiLab TSF Gruppo Almaviva

Fig. 108 - Pianificazione del viaggio da portale ridotto

Anche nella versione scalata sono presenti le stesse opzioni e le stesse modalità di funzionamento descritte nel paragrafo precedente.

Calcolo del percorso

Al click sul bottone “Calcola percorso” vengono eseguite una serie di operazioni:

- la ricerca effettuata dall’utente viene memorizzata nella tabella session_dashboard (vedi documentazione su Database MySQL);
- viene creato un file XML di nome “N”, dove N è un numero pari all’indice più alto presente in session_dashboard incrementato di 1e che rappresenta il progressivo delle richieste effettuate
- il file XML viene spedito via socket al motore di calcolo della rotta, il quale risponde con un file XML di nome “RISPOSTA_N” contenente i vari percorsi possibili che permettono di compiere il viaggio soddisfacendo i parametri specificati.

Un volta ricevuto il file “RISPOSTA_N”, tale file viene elaborato e l’utente viene indirizzato ad una pagina che descrive i possibili percorsi, come mostrato in Fig. 109.

Portale TRIP
Viaggia con noi!

RISULTATI DELLA RICERCA
Di seguito puoi trovare i **POSSIBILI PERCORSI** risultanti dalla tua ricerca:

n.	PARTENZA Fermata: U00076, Torino (Torino)	ARRIVO Fermata: M00007, Torino (Torino)	DURATA	N. CAMBI	DETTAGLI	MAPPA	RICHIEDI INFO REALTIME
1	15:49	16:27	38 min	0	dettagli		
2	15:59	16:37	38 min	0	dettagli		
3	16:09	16:47	38 min	0	dettagli		
4	16:19	16:57	38 min	0	dettagli		
5	16:29	17:07	38 min	0	dettagli		
6	16:39	17:17	38 min	0	dettagli		
7	16:49	17:27	38 min	0	dettagli		
8	16:59	17:37	38 min	0	dettagli		
9	17:09	17:47	38 min	0	dettagli		
10	17:19	17:57	38 min	0	dettagli		

Benvenuto gli [Admin_ID:gli]
Area Utente
Logout

Partner Progetto Trip:

Copyright IEIT-CNR Sede di Bologna

Fig. 109 - Risultati di una ricerca

I percorsi sono elencati a partire da quello che impiega meno tempo, e possono essere composti da più tratte che prevedono l’utilizzo di mezzi pubblici (treno, bus, metro, ...) oppure sono pedonali (come per esempio quelle che collegano un indirizzo ad una fermata). Per ogni percorso dell’elenco:

- si può visualizzare la mappa dettagliata del tragitto;
- si possono richiedere informazioni in tempo reale sul tragitto.

Il risultato della ricerca del percorso viene fornita all'utente attraverso la mappa navigabile in cui sono rappresentati i punti di partenza e arrivo e in cui vengono evidenziati i percorsi dei mezzi pubblici da prendere e le relative fermate di scambio. Vengono altresì evidenziati i percorsi pedonali di partenza/arrivo, cioè per raggiungere dal punto di partenza la prima fermata utile e viceversa dall'ultima fermata fino al punto di arrivo (qualora ovviamente non si voglia partire/arrivare ad una fermata). Inoltre in forma di tabella vengono riassunti tutti i passaggi dei mezzi e i cambi necessari ad effettuare il viaggio. Altre opzioni permettono di zoomare sull'area di partenza o su quella di arrivo, di tornare all'elenco dei percorsi e di visualizzare su mappa il percorso precedente o quello successivo. Inoltre, una volta identificato il percorso ottimale è possibile attivare l'opzione di Real Time.

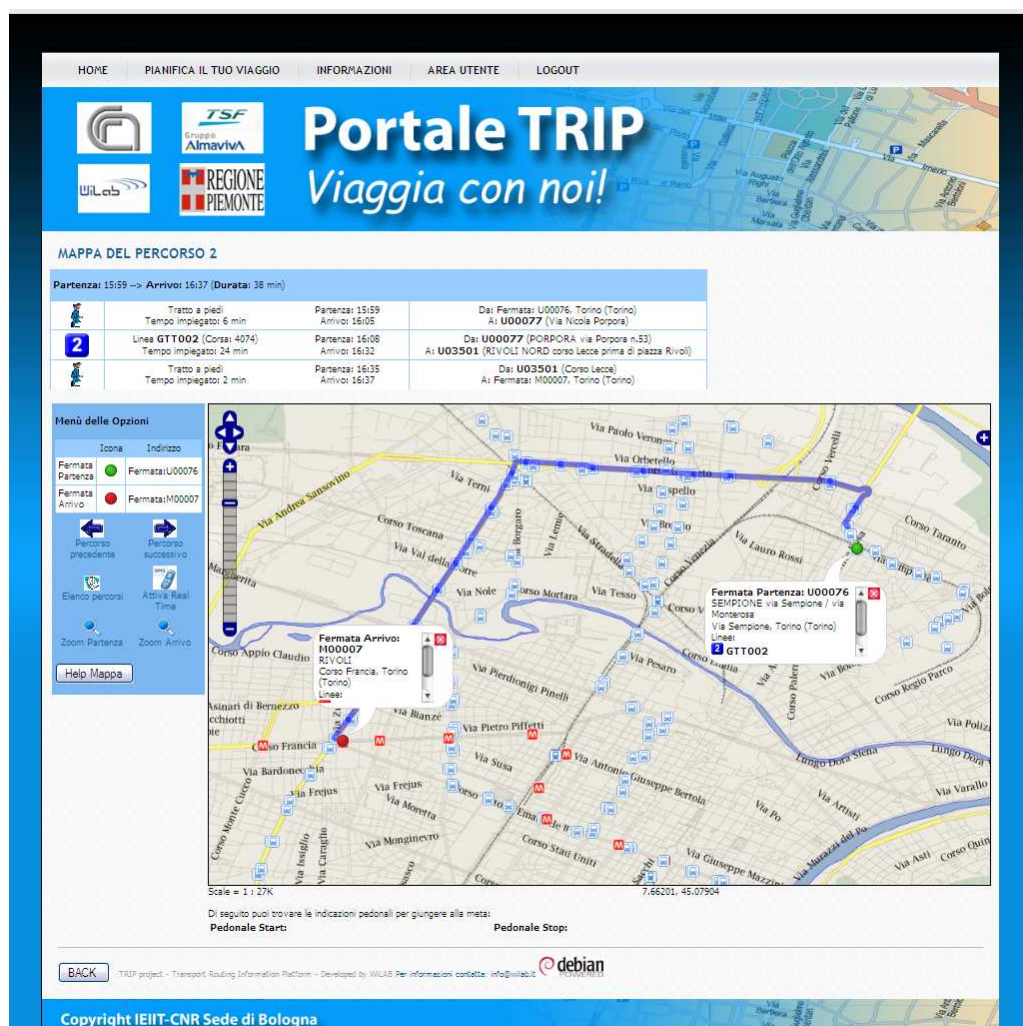


Fig. 110 - Visualizzazione grafica su mappa di un percorso

Scegliendo di richiedere informazioni in tempo reale, si viene portati alla pagina mostrata in Fig. 111.

Se si è regolarmente registrati e si è effettuato il login prima di effettuare il “calcola percorso”, arrivando sul portale di informazioni real-time si può scegliere tra l'opzione:

- Solo nella data indicata: L'utente verrà avvisato (ricevendo un sms sul proprio cellulare) soltanto in caso di eventuali ritardi che dovessero interessare il percorso da lui scelto nella data e nell'orario impostati al momento della ricerca.

- In più date e giornate: L'utente verrà avvisato in caso di ritardi che incidono sul percorso da lui scelto, in ogni giorno della settimana selezionato, per tutto il periodo che comincia dalla data impostata nella ricerca e termina con la data che viene richiesto di inserire. Se non si inserisce la data di "fine del periodo in cui si richiedono le informazioni", viene assunta di default la data corrispondente a una settimana dopo la data di inizio periodo.

Fig. 111 - Prenotazione del servizio di avviso RealTime

Cliccando su “Richiedi informazioni real-time”, il sistema esegue una serie di operazioni:

- Vengono effettuati una serie di controlli sulla richiesta inoltrata, per vedere che le date siano coerenti;
- La richiesta viene processata, e vengono salvati alcuni record relativi alla richiesta nelle tabelle `table_richieste` e `table_date_richieste` (vedi documentazione Database MySQL);
- Se la procedura si conclude con successo, si viene indirizzati alla pagina “Richiesta di informazioni real-time accettata” riportata di seguito.

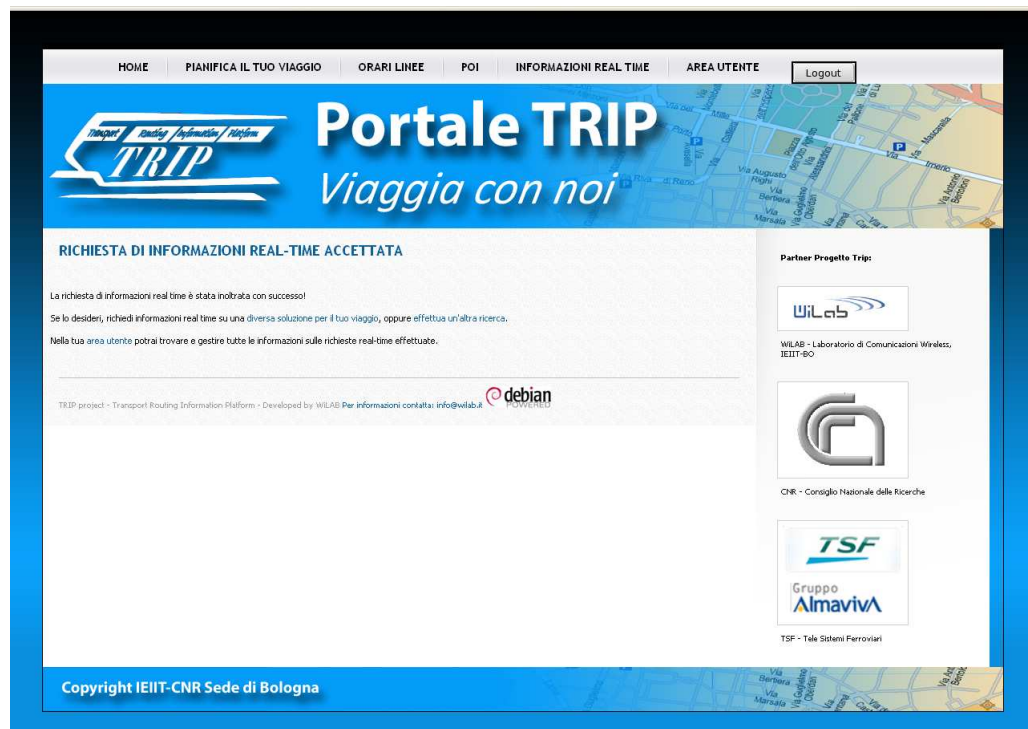


Fig. 112 - Richiesta di informazione real-time accettata

A questo punto, si può scegliere di accedere alla propria area utente e verificare, cliccando sul collegamento alla pagina “richieste real-time”, che la richiesta è stata effettivamente registrata con successo dal sistema.

Portale di Registrazione Utente

Per poter accedere a tutti i servizi personalizzati che il Progetto TRIP mette a disposizione degli utenti attraverso il portale, è necessaria una fase di registrazione. La procedura è molto semplice e inserendo i dati nel form tra cui il numero di cellulare (necessario per l'invio dei messaggi di testo SMS con la notifica degli eventuali aggiornamenti del percorso) e una username e password, necessari per identificarsi e accedere ai servizi. Dopo la registrazione è possibile prevedere l'invio della password via SMS o anche un semplice messaggio di benvenuto. Di seguito è riportata la schermata di registrazione.

Fig. 113 - Form di registrazione utente

Una volta che tutti i campi sono stati inseriti correttamente, cliccando su Registra è possibile completare la procedura di registrazione. Si verrà poi automaticamente rediretti sul portale Area Utente.

Portale Area Utente

Accedendo al portale con le proprie credenziali (username e password) è possibile visualizzare una Area Utente personale con i propri dettagli ed eventualmente modificare e aggiornare i propri dati. Sono inoltre presenti due archivi storici uno con tutte le richieste effettuate e uno con tutte le richieste real time sia scadute che ancora valide, effettuate dall'utente.

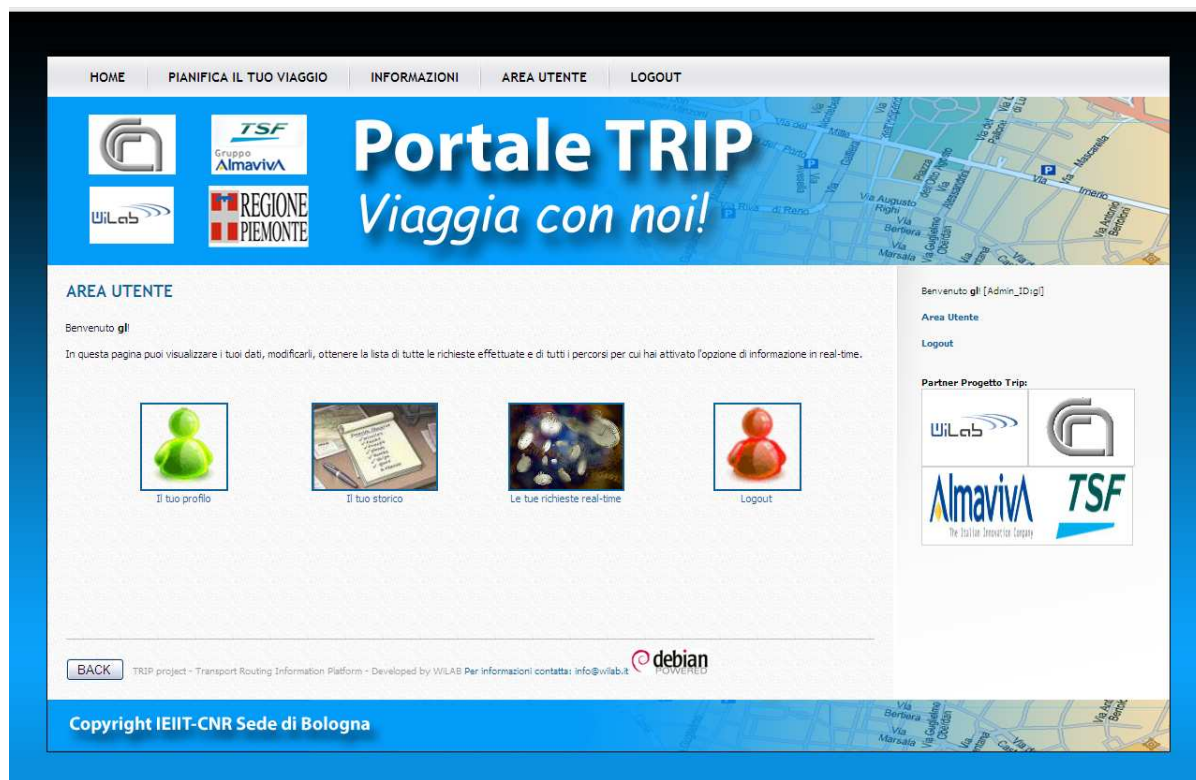


Fig. 114 - Pagina di ingresso area utente

Analizziamo in dettaglio l'archivio storico in cui vengono memorizzate tutte le richieste effettuate dall'utente: per ciascun viaggio vengono memorizzate:

- **Da:** indica il punto di partenza, inteso come indirizzo o fermata o PoI
- **A:** indica il punto di partenza, inteso come indirizzo o fermata o PoI
- **Data:** la data per cui è stata effettuata la ricerca
- **Ora:** l'ora per cui è stata effettuata la ricerca
- **Ricalcola:** con questo pulsante è possibile re-impostare il viaggio passato (stessi punti di partenza e arrivo) per ricalcolarlo nella data odierna o successiva. Cliccando sul pulsante si viene portati sul portale di pianificazione viaggio in cui i campi di partenza e arrivo sono già compilati. La data di default viene impostata pari a quella corrente. E' ovviamente possibile spostare la giornata per cui pianificare il viaggio.

Di seguito un esempio di archivio storico per un singolo utente.

Portale TRIP
Viaggia con noi!

STORICO UTENTE

Di seguito puoi trovare l'elenco dei percorsi per i quali hai effettuato una ricerca.
Cliccando su "Ricalcola" puoi effettuare nuovamente una ricerca con la stessa partenza e destinazione

N.	DA	A	DATA	ORA	RICALCOLA
1	Fermata M00011	Vicino Achille	22/05/2009	14:50	[RICALCOLA]
2	Fermata M00011	Fermata GTT009	22/05/2009	14:50	[RICALCOLA]
3	Fermata M00011	Ristorante Bellavista	22/05/2009	14:50	[RICALCOLA]
4	Fermata U00076	Fermata U00034	10/06/2009	15:41	[RICALCOLA]
5	Via Roma, 53, Torino (Torino)	Autostazione	12/06/2009	14:52	[RICALCOLA]
6	Via Roma, 53, Torino (Torino)	Via Carlo Alberto, 2, Torino (Torino)	12/06/2009	14:52	[RICALCOLA]
7	Fermata M00012	Fermata M00006	07/07/2009	12:50	[RICALCOLA]
8	Fermata M00007	Fermata M00012	07/07/2009	12:59	[RICALCOLA]
9	Fermata U00078	Fermata M00009	29/07/2009	09:47	[RICALCOLA]
10	Via Coazze, 1, Torino (Torino)	Via Pietro Giannone, 1, Torino (Torino)	29/07/2009	09:47	[RICALCOLA]
11	Via Andrea Doria, 4, Torino (Torino)	Via Pietro Giannone, 1, Torino (Torino)	29/07/2009	09:47	[RICALCOLA]
12	Via Antonio Giuseppe Bertola, 4, Torino (Torino)	Via Pietro Giannone, 1, Torino (Torino)	29/07/2009	09:47	[RICALCOLA]
13	Via Antonio Giuseppe Bertola, 1, Torino (Torino)	Via Ormea, 1, Torino (Torino)	29/07/2009	09:47	[RICALCOLA]

[Cancella tutti i percorsi](#)

[BACK](#) TRIP project - Transport Routing Information Platform - Developed by WiLab Per informazioni contatta: info@wilab.it

Copyright IEIIT-CNR Sede di Bologna

Fig. 115 - Storico utente

Per quello che riguarda invece l'elenco dei viaggi per cui è stato richiesto l'aggiornamento in real time per ogni richiesta vengono riportate le informazioni principali: punto di partenza e arrivo, data e ora relativi al percorso scelto, e data/periodo in cui si vogliono ricevere informazioni aggiornate in tempo reale sul percorso scelto. Nel caso in cui la data di fine periodo sia già passata, nella colonna "expired" comparirà 1, mentre se la richiesta è ancora valida comparirà 0. Se si vuole invece cancellare una delle proprie richieste e non venire più avvisati in merito ad eventuali ritardi che riguardano uno specifico percorso, basterà premere il tasto della colonna "cancel" nella riga corrispondente alla richiesta da eliminare. Di seguito è riportata un esempio di schermata di elenco di richieste in real time effettuate da un singolo utente.

[HOME](#)
[PIANIFICA IL TUO VIAGGIO](#)
[INFORMAZIONI](#)
[AREA UTENTE](#)
[LOGOUT](#)






Portale TRIP

Viaggia con noi!



RICHIESTE REAL-TIME

Di seguito puoi trovare l'elenco dei percorsi per i quali hai richiesto l'invio di informazioni in tempo reale.

n.	DA	A	data	ora	DATE RICHIESTE	SCADUTE	CANCELLA
1	Via Beinette, Torino (Torino)	Via Giacomo Puccini, Vercelli (Vercelli)	06-04-2009	7:30	06-04-2009	SI	
2	Via Beinette, Torino (Torino)	Via Pietro Micca, Vercelli (Vercelli)	19-05-2009	14:01	19-05-2009	SI	
3	SS25, Buttigliera Alta (Torino)	Via Goito, Torino (Torino)	19-05-2009	14:01	19-05-2009	SI	
4	Via Torino, Lanzo Torinese (Torino)	Via Goito, Torino (Torino)	19-05-2009	14:01	19-05-2009	SI	
5	Fermata M00001	Fermata U03400	25-03-2009	15:12	Tutti i MAR, MER, VEN, dal 25-03-2009 al 04-12-2009	No	
6	Corso Francesco Ferruccio, Torino (Torino)	Corso Spezia, Torino (Torino)	25-04-2009	15:12	25-04-2009	SI	
7	Via Roma, Torino (Torino)	Via Zecca, Novara (Novara)	25-04-2009	15:12	Tutti i SAB, DOM, dal 25-04-2009 al 31-03-2010	No	
8	SP249, Mezzenile (Torino)	Via Roma, Torino (Torino)	19-05-2009	9:27	19-05-2009	SI	
9	Via Carletto Leonardi, Novara (Novara)	Via Roma, Torino (Torino)	19-05-2009	18:27	19-05-2009	SI	
10	Strada Vasche Tos, San Colombano Belmonte (Torino)	Via Roma, Torino (Torino)	19-05-2009	7:27	19-05-2009	SI	
11	Via Beinette, Torino (Torino)	Via Giacomo Puccini, Vercelli (Vercelli)	06-04-2009	7:30	06-04-2009	SI	
12	Via Beinette, Torino (Torino)	Via Giacomo Puccini, Vercelli (Vercelli)	06-04-2009	6:16	06-04-2009	SI	
13	Via Beinette, Torino (Torino)	Via Giacomo Puccini, Vercelli (Vercelli)	06-04-2009	10:16	06-04-2009	SI	

Benvenuto tpi [Admin_ID:tpi]

[Area Utente](#)

[Logout](#)

Partner Progetto Tripi:






Portale di Informazione

Attraverso il portale di Informazione è possibile accedere ad un elevata quantità di informazioni sui mezzi, le fermate e gli orari dei mezzi pubblici locali, ma anche a dettagli sul traffico, sulle modifiche alla viabilità, ai punti di interesse e al meteo. Il portale è raggiungibile direttamente dal portale di Accesso principale o da qualunque altra pagina cliccando semplicemente Informazioni nella barra dei menù presente in tutte le pagine.

Una cosa fondamentale è che tutte le informazioni vengono erogate comune per comune, quindi prima di accedere ai singoli contenuti è necessario specificare una provincia del Piemonte e un suo comune. Questo fatto permette di limitare la mole di dati che potrebbe essere visualizzata all'utente: in questo modo l'utente accede ai contenuti ed alle mappe solo del comune prescelto.

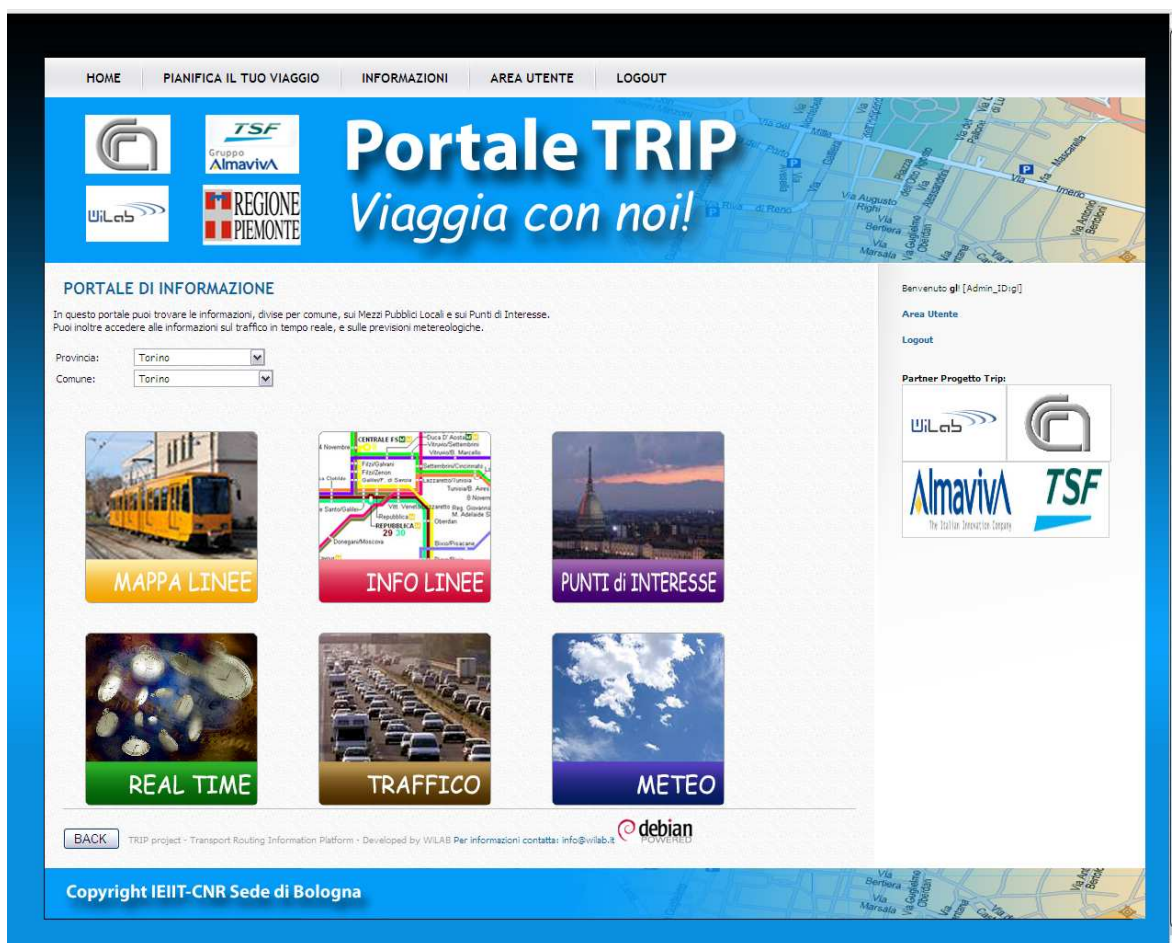


Fig. 116 - Pagina di ingresso al portale di informazione

I principali sotto portali da cui è possibile accedere dal portale di Informazione sono:

- **Mappa Linee:** da qui è possibile consultare direttamente sulla mappa il percorso e la posizione delle fermate dei mezzi di trasporto pubblico di tutte le linee che circolano nel comune

- **Info Linee:** vengono visualizzati percorsi schematizzati, e la sequenza delle fermate per ogni linea
- **Punti di Interesse:** per la visualizzazione e ricerca direttamente sulla mappa dei PoI
- **Real Time:** permette di ottenere informazioni sullo stato del servizio real time e se si è registrati anche di accedere ai servizi in mobilità offerti.
- **Traffico:** da questo portale si ottengono informazioni sullo stato di congestione del traffico che influenza sia la mobilità viaria privata che quella pubblica
- **Meteo:** servizio di utilità con situazione e previsione meteo aggiornate.

Esiste ovviamente anche per questo portale la sua versione scalata adatta ad essere visualizzata sui terminali mobili dotati di un display ridotto.



Fig. 117 - Pagina di ingresso del portale di informazione su dispositivo portatile

Mappa Linee

In questa pagina, riportata in Fig. 118, è possibile consultare direttamente sulla mappa il percorso e la posizione delle fermate dei mezzi di trasporto pubblico di tutte le linee che circolano nel comune.

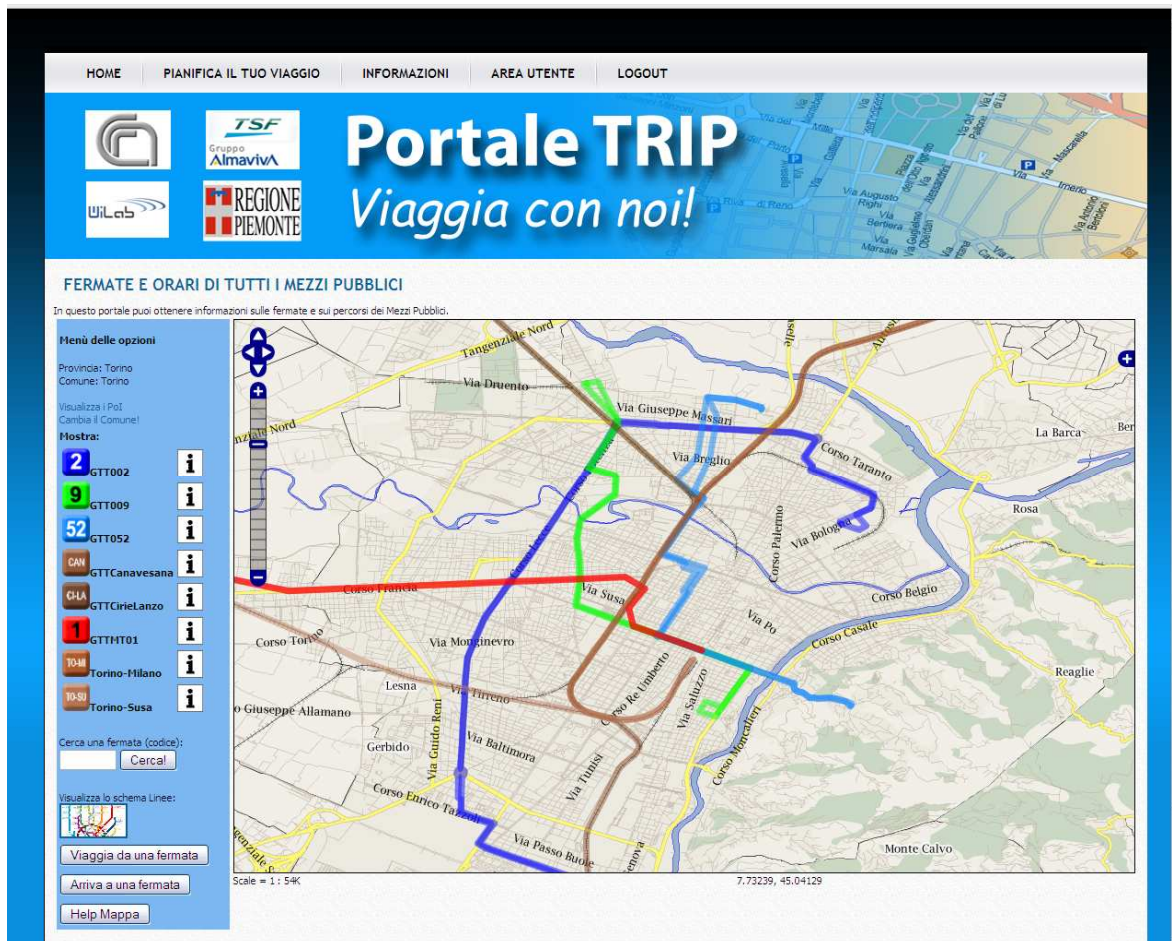


Fig. 118 - Percorsi, fermate e orari dei mezzi pubblici

E' presente anche un elenco sul lato sinistro della mappa con la possibilità di evidenziare o nascondere il percorso della linea di interesse e di ottenere altre informazioni cliccando sul pulsante apposito. Le linee vengono colorate in maniera differente le une dalle altre in maniera automatica per permettere all'utente un più facile riconoscimento.

Info Linee

In questa pagina, riportata in Fig. 119, vengono visualizzati i percorsi schematizzati, e la sequenza delle fermate per ogni linea, con una breve descrizione e il nome del vettore che effettua il servizio. Sono inoltre presenti una serie di strumenti di filtraggio e ricerca sulle linee.

Per ogni linea viene visualizzato uno schema con evidenziato l'elenco delle fermate. Per ogni fermata è possibile ottenere altre informazioni come l'indirizzo esatto, il file pdf con gli orari di passaggio e la possibilità di ottenere i passaggi in real time della prossima ora. Viene inoltre visualizzata una piccola cartina in cui è possibile individuare la fermata e capirne la reale disposizione sul piano stradale.

The screenshot displays the 'Portale TRIP' website interface. At the top, there are navigation links: HOME, PIANIFICA IL TUO VIAGGIO, INFORMAZIONI, AREA UTENTE, and LOGOUT. Below these are logos for Gruppo Almaviva, WiLab, and REGIONE PIEMONTE. The main heading is 'Portale TRIP Viaggia con noi!'. The central section is titled 'MODULO DI INFORMAZIONE DEGLI ORARI, PERCORSI E PDF DELLE LINEE DEI MEZZI PUBBLICI'. It includes a search bar with filters for 'Sei in: Torino > Torino', 'Mezzo di trasporto: Tutti', 'Azienda: Tutte', and 'Linea: Tutte'. A date selector shows '27-08-2009' and a time range from '09' to '10'. A 'Cerca' button is present. Below the search bar is a table of public transport lines:

Clicca sul logo!	Nome Linea	Descrizione	Vettore	Orari PDF
	GT002	Linea 2 - Ponchielli - Corradino Linea 2 - Corradino - Ponchielli	Autobus	
	GT009	Linea GT009 percorso 91 Linea GT009 percorso 92	Tram	
	GT052	Linea GT052 percorso 521 Linea GT052 percorso 522	Autobus	
	GTTCanavesana	Linea Ferroviaria Chieri-Pont	Treno	
	GTTCrieLanzo	Linea Ferroviaria Torino-Ceres	Treno	

Below the table, a map shows the route of the GTTCrieLanzo line, with stops labeled: STAZIONE TORINO (GT007), STAZIONE MADONNA (GT021), STAZIONE VIMARCO (GT023), STAZIONE BORGARO (GT024), STAZIONE CASELLI (GT025), STAZIONE SAN MA (GT027), STAZIONE CRIE (GT028), STAZIONE NOLE (GT029), STAZIONE VILANO (GT030), STAZIONE MATHI (GT031), STAZIONE BALANCO (GT032), STAZIONE LANZO (GT033), STAZIONE ORMAI (GT034), STAZIONE LONIGIO (GT035), STAZIONE TRAFESIO (GT036), STAZIONE LISA (GT037), STAZIONE PESIN (GT038), STAZIONE MEZZANO (GT039), and STAZIONE CERES (GT040). A detailed view of the 'STAZIONE VILANO' stop is shown on the right, including its location on a map and a list of nearby streets: Via Torino, Via Vittorio Veneto, Via Frassineto, and Via Augusto Rogni.

Fig. 119 - Schematiche dei percorsi dei mezzi pubblici con possibilità di visualizzare su mappa la posizione delle fermate

Informazioni sull'orario effettivo di passaggio

Questo messaggio di popup, riportato in Fig. 120, viene visualizzato all'utente ogni volta venga effettuata una richiesta di orario di passaggio ad una fermata specificata. Contiene gli orari di tutti i passaggi a quella fermata, divisi per linea, per una finestra temporale di un'ora.

Effettuando un semplice passaggio del mouse sopra ogni singolo passaggio vengono visualizzate le informazioni sulla direzione e sul capolinea della linea che effettuerà il servizio.

PASSAGGI ALLA FERMATA

Linea GTTMT01	15:47	15:47	15:51	15:51	15:55	15:55
	15:59	15:59	16:03	16:03	16:07	16:07
	16:11	16:11	16:15	16:15	16:19	16:19
	16:23	16:23	16:27	16:27	16:31	16:31
	16:35	16:35	16:39	16:39	16:43	16:43

NOTA: passa con il mouse sopra i passaggi per conoscere i dettagli della corsa.

Chiudi

Fig. 120 - Esempio di risposta ad una richiesta di passaggi alla fermata

PDF della Fermata

Per ogni fermata, grazie agli strumenti messi a disposizione dal portale TRIP, è possibile generare in maniera del tutto automatizzata un file di tipo PDF in cui sono raccolte tutte le informazioni di una singola fermata, in particolare:

- Nome della fermata
- Coordinate GPS
- Indirizzo
- Linee servite
- Cartina con il posizionamento della fermata sul piano stradale
- Schema delle fermate (utile per conoscere fermata precedente e seguente, ma anche per sapere quante fermate ci sono prima del target)
- Orari dei passaggi, divisi tra feriali e festivi.

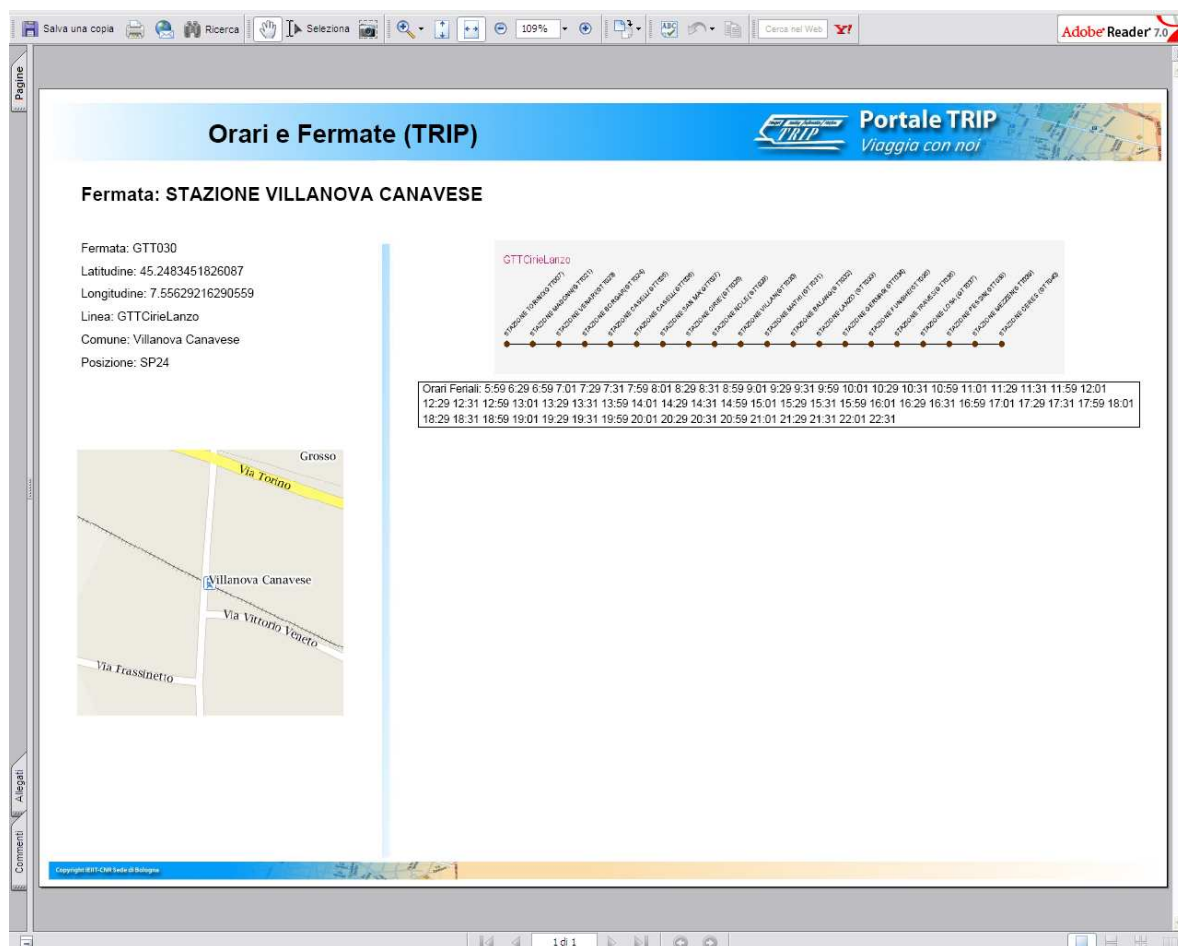


Fig. 121 - PDF con schematica percorso e orari di passaggio alla fermata scelta

I file PDF possono venire generati in maniera automatica o direttamente sulla singola richiesta dell'utente oppure opportunamente generati tutti dall'amministratore e archiviati. Tutti i dati comunque vengono recuperati in maniera automatica direttamente dal database del sistema e a partire da uno schema di impaginazione definito dall'amministratore.

Portale Traffico

All'interno del portale del traffico, sempre limitato ad un singolo comune, sono presenti una serie di link per monitorare lo stato del traffico che interessa sia la viabilità dei mezzi privati che quelli pubblici più una serie di utility molto funzionali per avere informazioni sui parcheggi, i cantieri attivi ed anche sulle webcam direttamente accessibili all'utente.

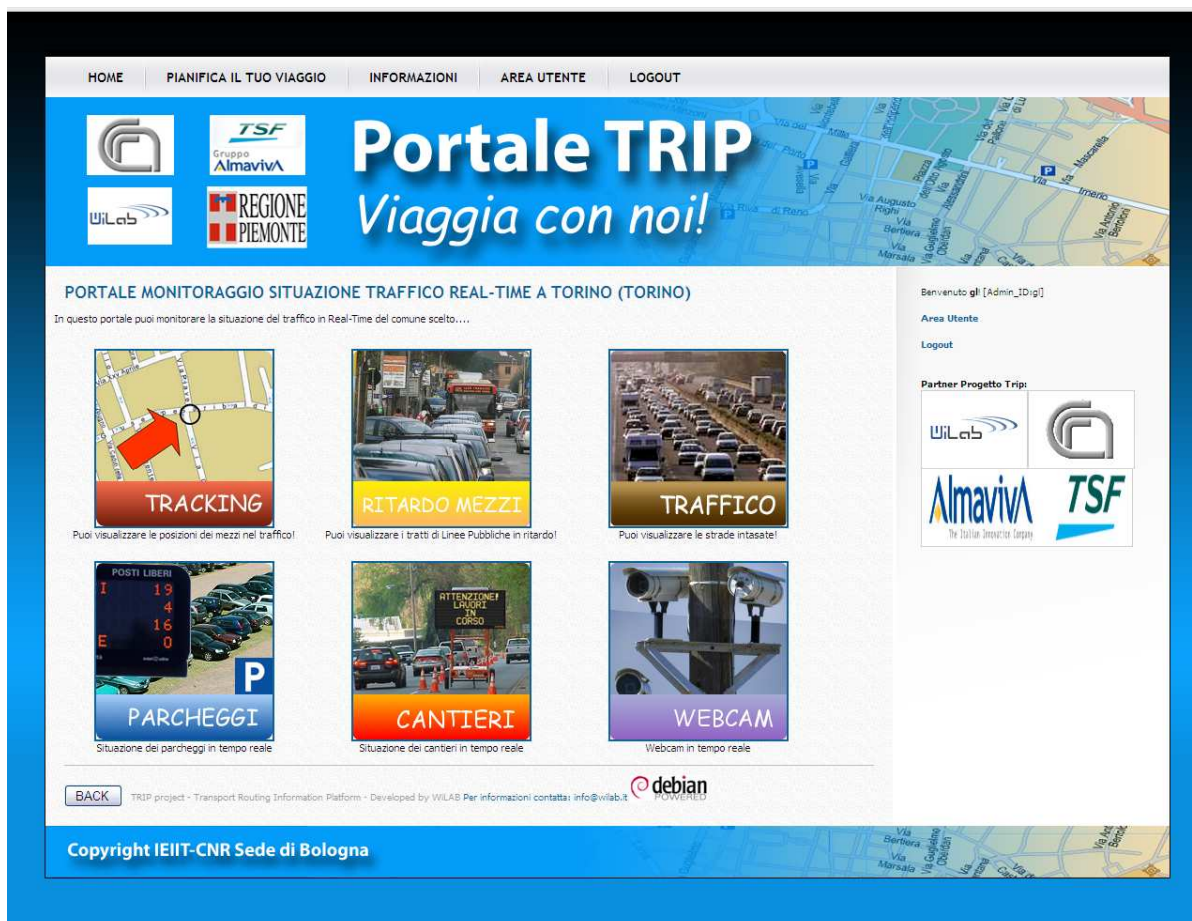


Fig. 122 - Pagina iniziale del portale monitoraggio

I principali sotto portali sono:

- **Tracking:** grazie a questo strumento è possibile monitorare in real time la posizione dei mezzi dotati di un localizzatore GPS e di un sistema di trasmissione verso la centrale di controllo su una cartina navigabile
- **Ritardo Mezzi:** è una utility che permette di visualizzare in maniera facile ed intuitiva direttamente sulla mappa i tratti di strada su cui i mezzi pubblici sperimentano ritardo
- **Traffico:** è una utility che permette di visualizzare in maniera facile ed intuitiva direttamente sulla mappa i tratti di strada su cui i mezzi privati sperimentano ritardo e quindi caratterizzati da cingestioni e incolonnamenti
- **Parcheggi:** permette la possibilità di visualizzare i parcheggi disponibili in zona e se opportunamente equipaggiati, anche il numero di posti attualmente disponibili

- **Cantieri:** grazie all'inserimento in un elenco dei cantieri aperti, è possibile visualizzare direttamente sulla mappa i cantieri aperti, la data di fine cantiere e altre informazioni
- **Webcam:** uno sguardo alla città attraverso gli occhi delle webcam opportunamente riportare sulla mappa navigabile.

GPS tracking

In questa pagina, riportata in Fig. 123, è presente un tool software in grado di monitorare in real time la posizione dei mezzi dotati di un localizzatore GPS e di un sistema di trasmissione verso la centrale di controllo su una cartina navigabile.

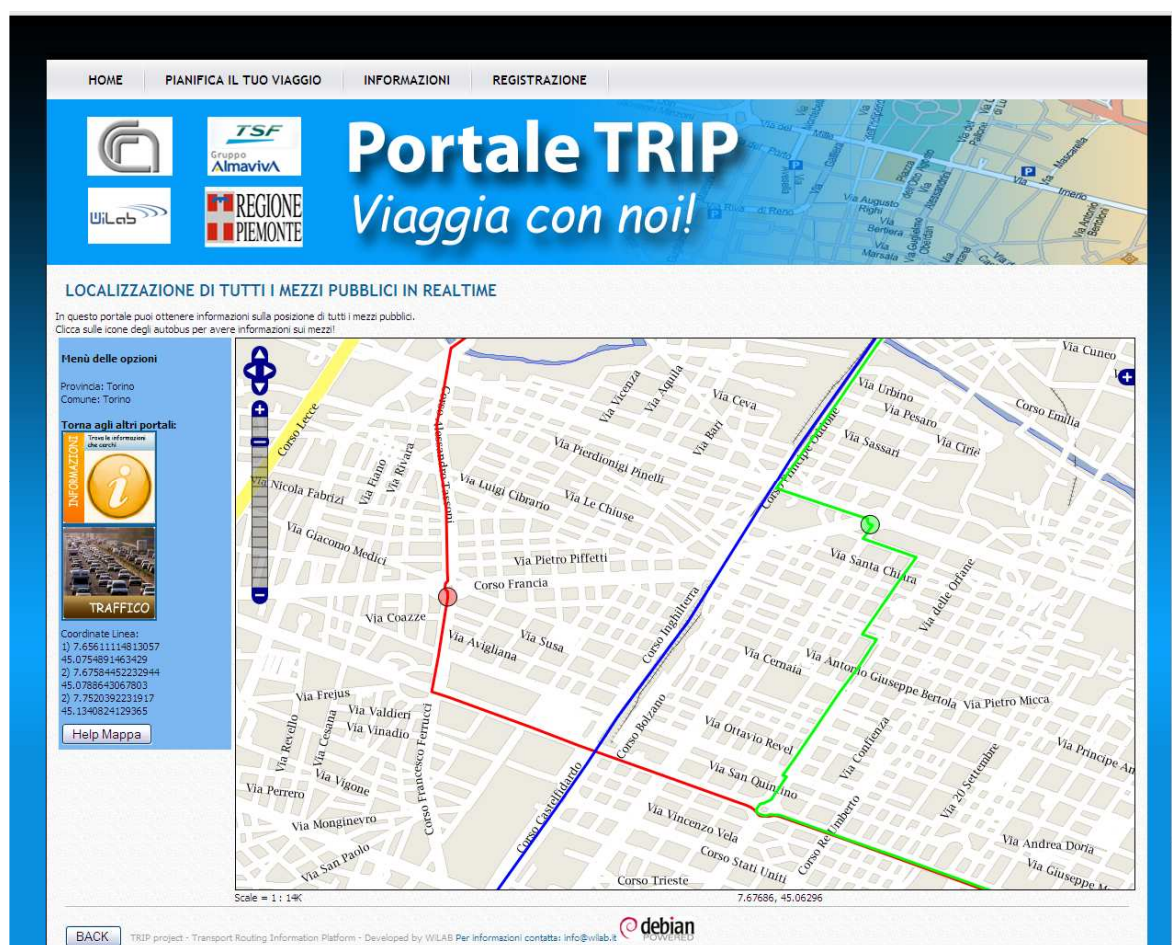


Fig. 123 - Localizzazione dei mezzi in real-time

Sulla mappa navigabile vengono visualizzati dei punti colorati, corrispondenti alle linee che si vogliono monitorare, che si spostano in base ai dati GPS ricevuti direttamente dal mezzo.

Ritardo Mezzi

In questa pagina, riportata in Fig. 124, è presente una utility che permette di visualizzare in maniera facile ed intuitiva direttamente sulla mappa i tratti di strada su cui i mezzi pubblici sperimentano ritardo. In base all'effettivo ritardo dei mezzi il colore delle strade evidenziate può variare dall'arancio, al rosso e infine al nero in caso di blocco totale del traffico.



Fig. 124 - Visualizzazione su mappa della condizione dei mezzi pubblici

Ritardo Autovetture

In questa pagina, riportata in Fig. 125, è presente una utility che permette di visualizzare in maniera facile ed intuitiva direttamente sulla mappa i tratti di strada su cui i mezzi privati (autovetture) sperimentano ritardo. In base all'effettivo ritardo dei mezzi il colore delle strade evidenziate può variare dall'arancio, al rosso e infine al nero in caso di blocco totale del traffico.

Ovviamente per poter ottenere questi dati è necessario sottoscrivere opportuni contratti con i gestori dei dati di traffico privato, in quanto il progetto TRIP si occupa esclusivamente di trasporto pubblico locale.

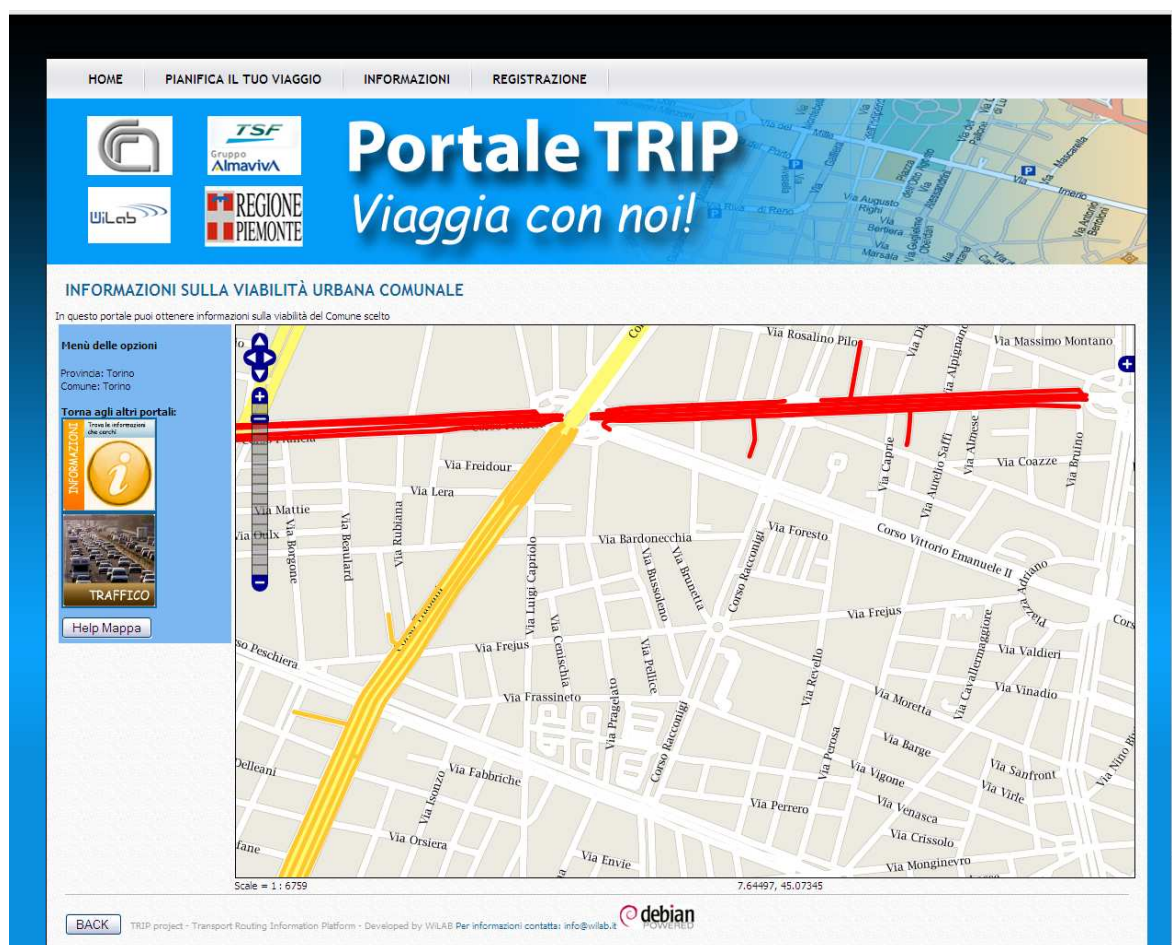


Fig. 125 - Visualizzazione su mappa della condizione del traffico stradale

Parcheggi

In questa pagina, riportata in Fig. 126 , è possibile visualizzare i parcheggi disponibili in zona e se opportunamente equipaggiati, anche il numero di posti attualmente disponibili. Per ogni parcheggio è inoltre possibile cliccare e aprire un popup con tutte le informazioni.

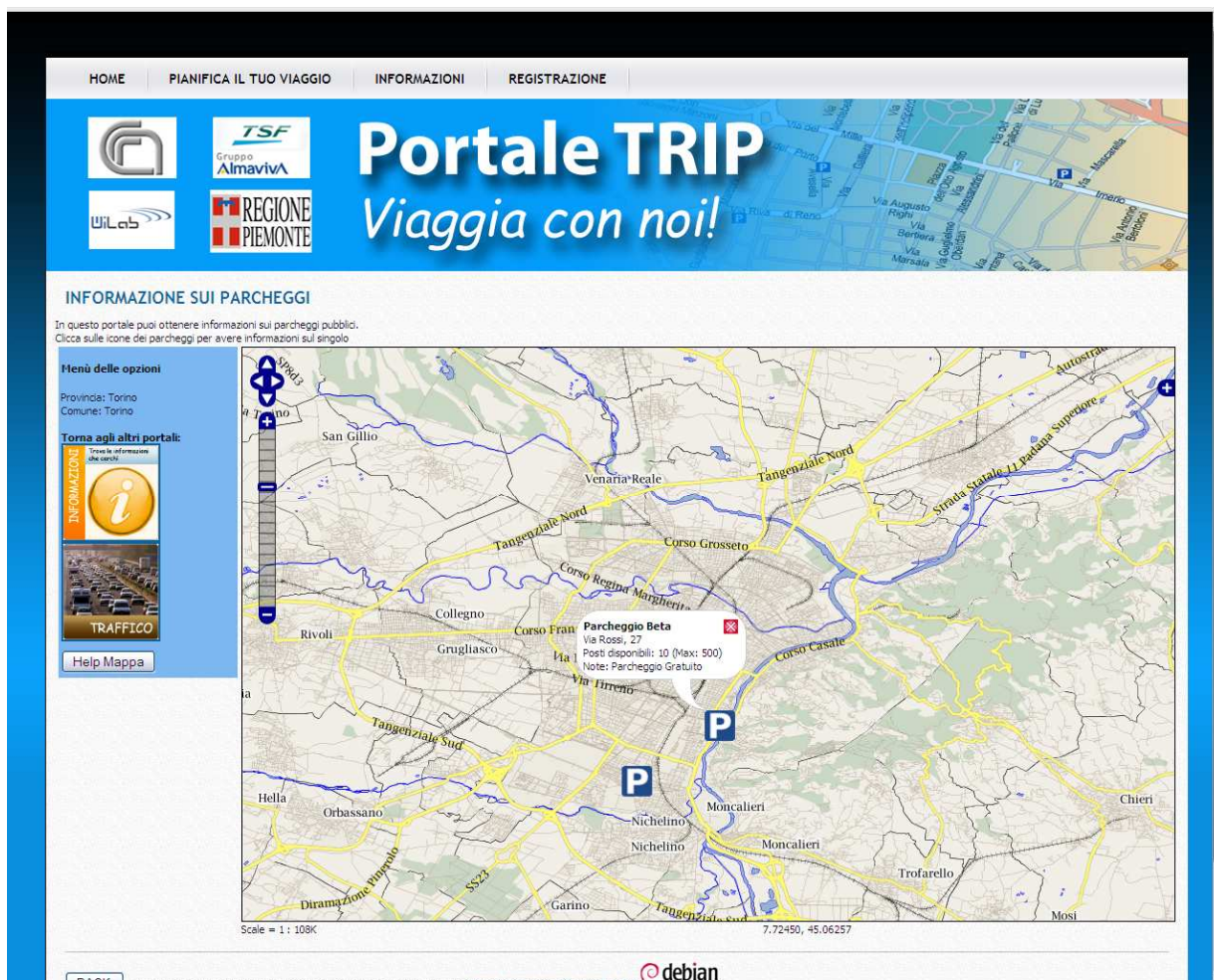


Fig. 126 - Visualizzazione su mappa della posizione e condizione dei parcheggi

Cantieri

In questa pagina, riportata in Fig. 127, è possibile grazie all'inserimento delle segnalazioni dei cantieri aperti, è possibile visualizzare direttamente sulla mappa i cantieri aperti, la data di fine cantiere e altre informazioni. Per ogni cantiere è inoltre possibile cliccare e aprire un popup con tutte le informazioni.

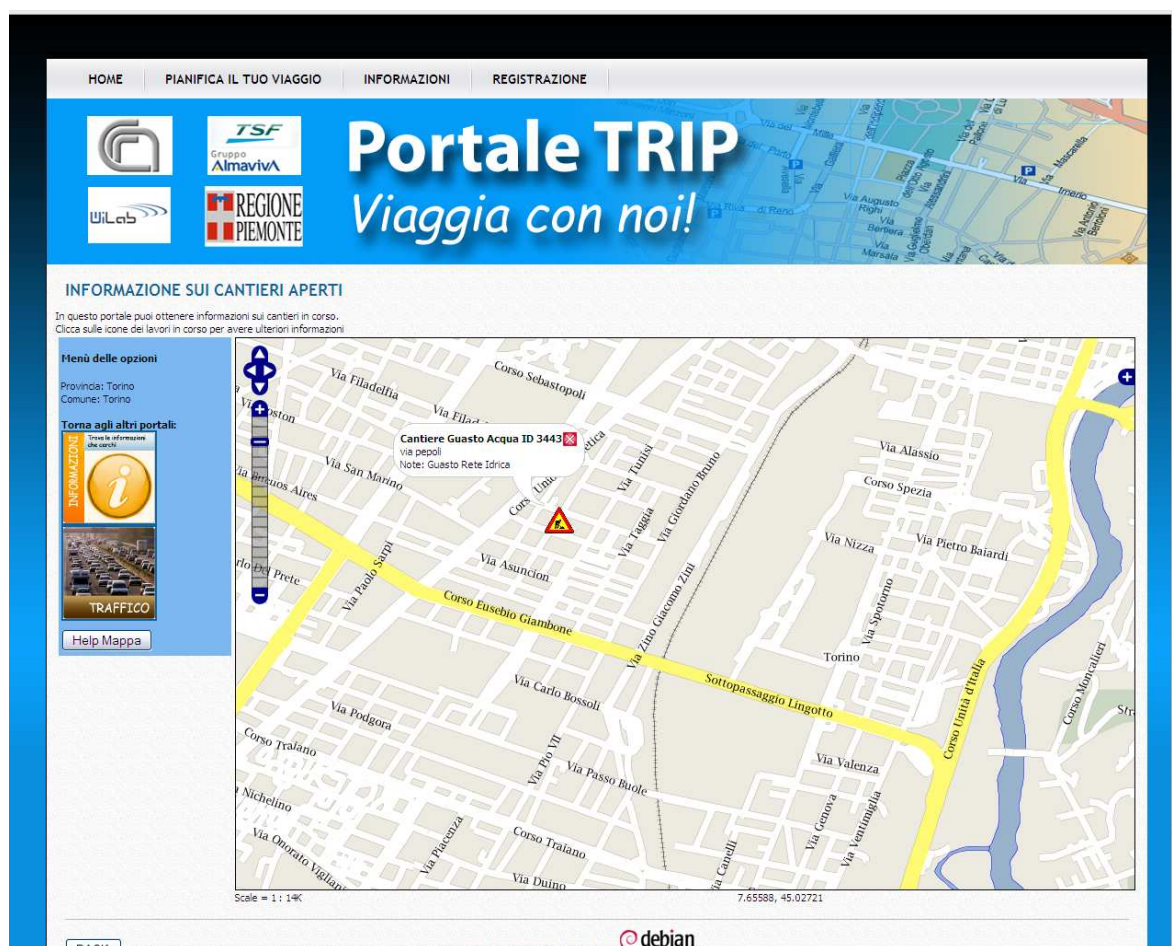


Fig. 127 - Visualizzazione su mappa della presenza di cantieri di lavoro stradale

Webcam

In questa pagina, riportata in Fig. 128, è possibile grazie all'inserimento delle segnalazioni delle webcam attive, è possibile visualizzare direttamente sulla mappa le webcam e dare uno sguardo alla città. Per ogni webcam è inoltre possibile cliccare e aprire un popup con tutte le informazioni.

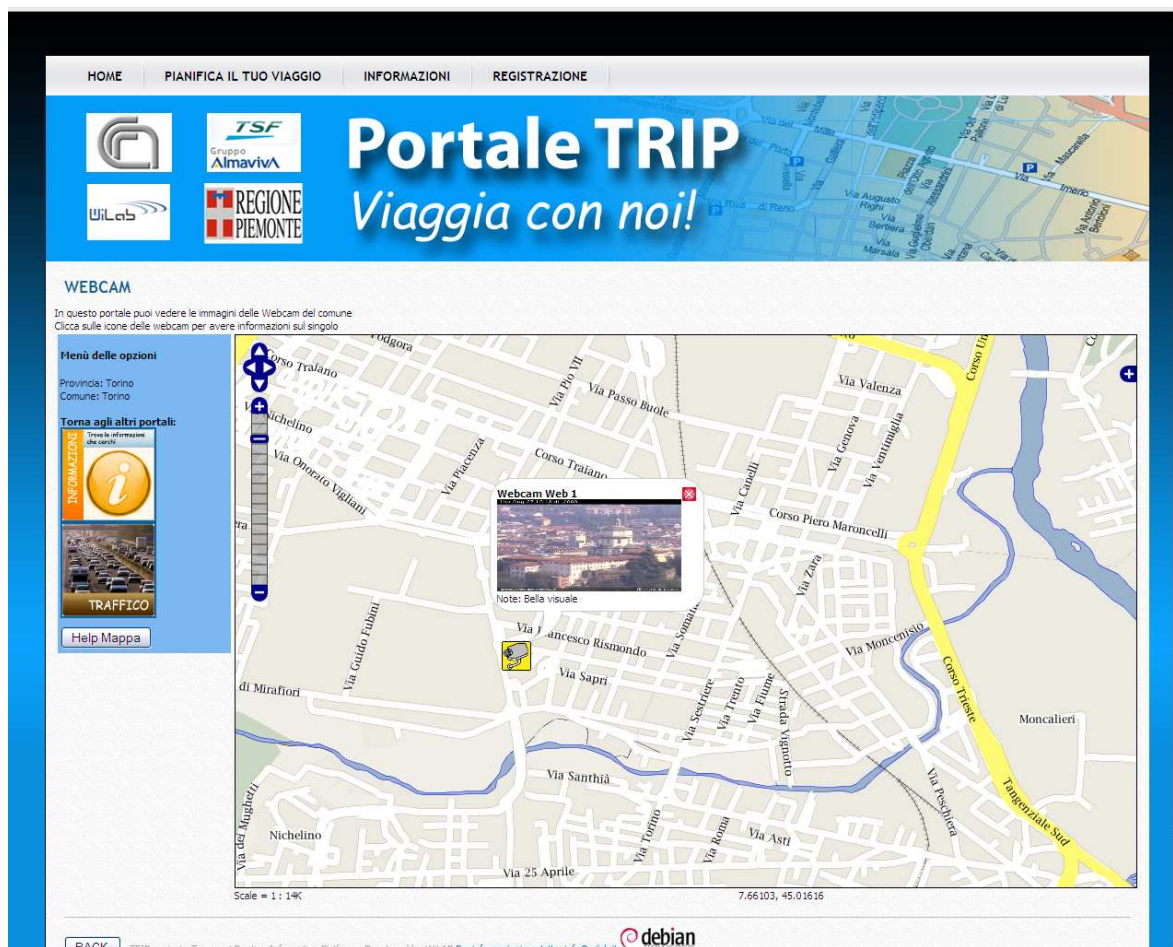


Fig. 128 - Visualizzazione su mappa della presenza di webcam

Portale di Amministrazione

Un portale in cui vengono fornite agli utenti molte informazioni e che vuol fare del real-time il suo punto di forza, deve prevedere la possibilità di aggiornare i contenuti e controllare il funzionamento dell'intero sistema grazie alle diverse funzionalità di amministrazione.

Ovviamente il portale TRIP prevede un portale di amministrazione in cui l'amministratore che vi accede grazie alle proprie credenziali, può avere a disposizione un'ampia serie di tool di amministrazione e controllo.

Il portale di amministrazione prevede quattro distinte aree di management quali:

- La gestione delle linee: per creare i pdf degli orari, le immagini con gli schemi delle fermate e accedere al tool di disegno delle rotte dei mezzi
- Database: per inserire nuovi parcheggi, lavori in corso o webcam o tutti quei servizi che richiedono un aggiornamento costante e dinamico dei contenuti erogati all'utente
- Gestione server: per disporre di strumenti per controllare lo stato dei server e delle socket utilizzate per scambiarsi i dati
- Statistiche: per ottenere informazioni grafiche sull'uso del portale, sulle richieste e gli eventi ritardo.



Fig. 129 - Pagina iniziale del portale di amministrazione

Di seguito verranno visualizzate alcune pagine del portale di amministrazione per evidenziare le potenzialità messe a disposizione degli amministratori. Ovviamente è possibile implementare ancora un'insieme molto ampio di applicazioni per la gestione e il management: questa parte del portale vuole solo essere una dimostrazione delle possibilità che è possibile sfruttare.

Il portale con le statistiche è quello che meglio di tutto esprime il concetto. Selezionando semplicemente una data è possibile visualizzare i dati aggregati sulle richieste effettuate al

portale per confronti tra anni, mesi e giorni in modo da visualizzare i periodi di massimo e minimo accesso e vedere se le richieste di viaggio interessano percorsi all'interno del comune o sono extra-provinciali. Altre informazioni vengono memorizzate come il conteggi degli accessi e il logging degli indirizzi IP dei PC che si sono connessi al sistema.

La possibilità ovviamente di avere disposizione direttamente dei grafici, generati direttamente in maniera automatica dal sistema, rende questo strumento molto funzionale e i contenuti immediatamente fruibili

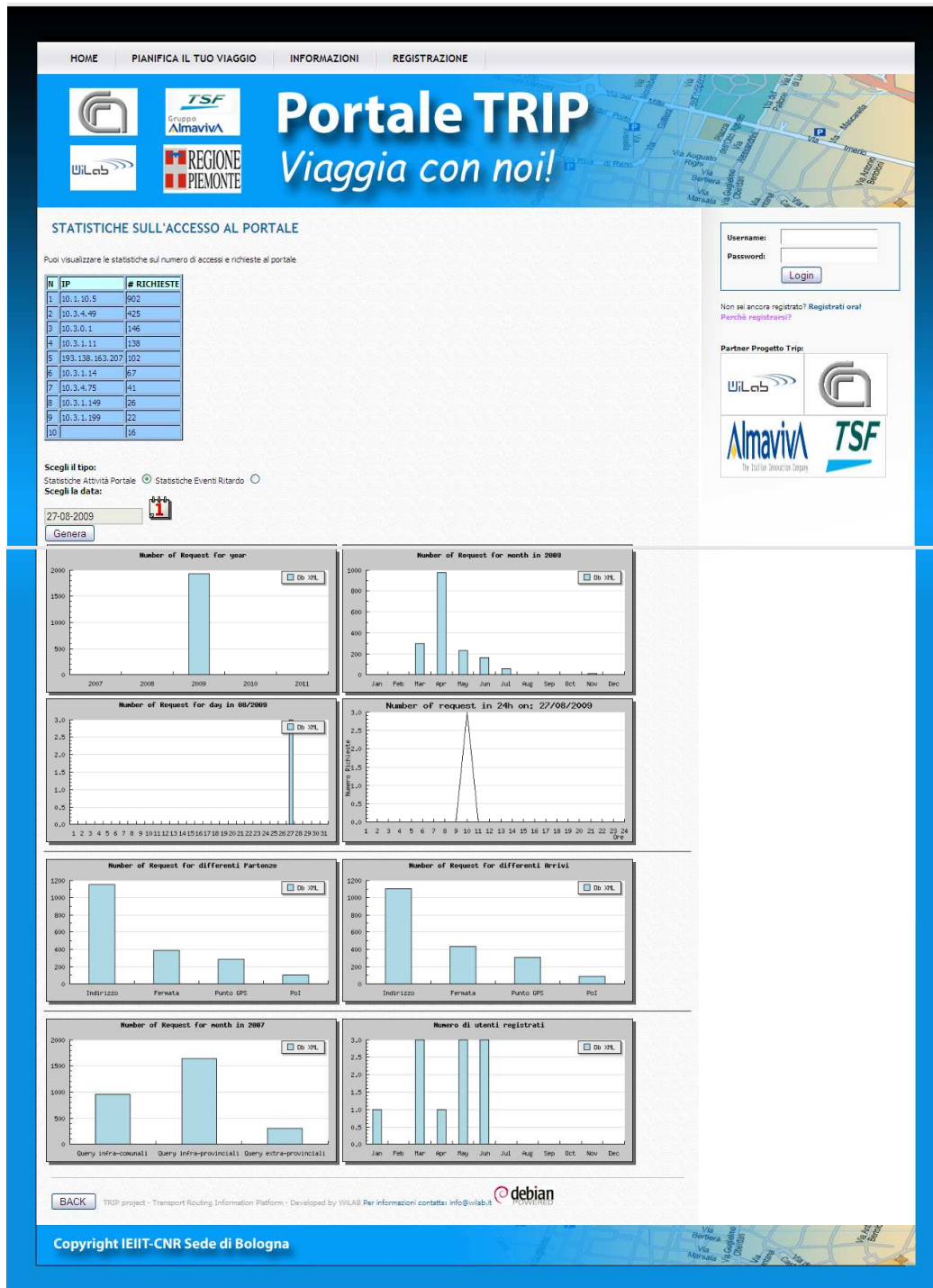


Fig. 130 - Esempi di raccolta di dati statistici

Altri strumenti assolutamente indispensabile per controllare e valutare il funzionamento della piattaforma sono i due portali che gestiscono le socket verso il motore di calcolo del percorso e verso il motore real-time.

Grazie a questi due strumenti, le cui schermate sono riportate di seguito, è possibile attivare o chiudere le socket di cui sopra oppure leggere i log in cui sono riportati tutti i dettagli dei file che vengono costantemente scambiati sulla socket.

HOME PIANIFICA IL TUO VIAGGIO INFORMAZIONI AREA UTENTE LOGOUT

Portale TRIP
Viaggia con noi!

GESTIONE SOCKET MOTORE

SOCKET MOTORE ATTIVA [Chiudi Socket](#)

Benvenuto gli [Admin_IDip]
Area Utente
Logout

Partner Progetto Trip:
WiLab Gruppo Almaviva TSF

```

2009-08-27T15:39:39 Find new XML file: REQ_orario_1261359603.xml
2009-08-27T15:39:39 CODA: 2
2009-08-27T15:39:41 Updated in DB statistic on ID: 1939
2009-08-27T15:39:41 Reconstructed XML file: RISPOSTA_1939.xml - 26398 bytes
2009-08-27T15:39:41 DECREMENTO CODA
2009-08-27T15:39:41 Saved in DB statistic on ID: 1261359603
2009-08-27T15:39:41 INCREMENTO CODA
2009-08-27T15:39:41 Sent XML file: REQ_orario_1261359603.xml
2009-08-27T15:39:41 Find new XML file: REQ_orario_1261115861.xml
2009-08-27T15:39:41 CODA: 2
2009-08-27T15:39:42 Updated in DB statistic on ID: 1938
2009-08-27T15:39:42 Reconstructed XML file: RISPOSTA_1938.xml - 39024 bytes
2009-08-27T15:39:42 DECREMENTO CODA
2009-08-27T15:39:42 Saved in DB statistic on ID: 1261115861
2009-08-27T15:39:42 INCREMENTO CODA
2009-08-27T15:39:42 Sent XML file: REQ_orario_1261115861.xml
2009-08-27T15:39:42 Find new XML file: 1937.xml
2009-08-27T15:39:42 CODA: 2
2009-08-27T15:39:42 Updated in DB statistic on ID: 1261359603
2009-08-27T15:39:42 Reconstructed XML file: CONSULT_1261359603.xml - 6839 bytes
2009-08-27T15:39:42 DECREMENTO CODA
2009-08-27T15:39:42 Saved in DB statistic on ID: 1937
2009-08-27T15:39:42 INCREMENTO CODA
2009-08-27T15:39:42 Sent XML file: 1937.xml
2009-08-27T15:39:42 Updated in DB statistic on ID: 1261115861
2009-08-27T15:39:42 Reconstructed XML file: CONSULT_1261115861.xml - 153 bytes
2009-08-27T15:39:42 DECREMENTO CODA
2009-08-27T15:39:42 Updated in DB statistic on ID: 1937
2009-08-27T15:39:42 Reconstructed XML file: RISPOSTA_1937.xml - 26398 bytes
2009-08-27T15:39:42 DECREMENTO CODA
2009-08-27T15:40:26 Find new XML file: 1940.xml
2009-08-27T15:40:26 CODA: 0
2009-08-27T15:40:26 Updated in DB statistic on ID: 1940
2009-08-27T15:40:26 INCREMENTO CODA
2009-08-27T15:40:26 Sent XML file: 1940.xml
2009-08-27T15:40:26 Updated in DB statistic on ID: 1940
2009-08-27T15:40:26 Reconstructed XML file: RISPOSTA_1940.xml - 27278 bytes
2009-08-27T15:40:26 DECREMENTO CODA
2009-08-27T15:40:39 INCREMENTO CODA
2009-08-27T15:40:39 DECREMENTO CODA
  
```

Fig. 131 - Pagina del portale per la gestione della socket

The screenshot shows the 'Portale TRIP' web application. The top navigation bar includes links for HOME, PIANIFICA IL TUO VIAGGIO, INFORMAZIONI, AREA UTENTE, and LOGOUT. The header features logos for Gruppo Almagiva, TSF, and REGIONE PIEMONTE, along with the text 'Portale TRIP Viaggia con noi!'. A map of a city street is visible in the background.

The main content area is titled 'GESTIONE SCRIPT REALTIME'. It contains a green button labeled 'REAL TIME ATTIVO' and a button labeled 'Ferma Real Time'. Below these, there is a form with a dropdown menu for 'corsa:' (set to 2), a date input field for 'Data:' (set to 27-08-2009), and a 'secondintardo:' field. A 'Genera XML Evento' button is located below the form.

The central part of the page displays a log of real-time events. The log entries are as follows:

```

2009-06-11T12:20:46 Ottengo le tratte dal file: /var/www/sito2/RealT.
2009-06-11T12:20:46 Tratta 0 da S00245 a S00222
2009-06-11T12:20:46 Tratta 0 da S4780 a S7600
2009-06-11T12:20:46 Apro il file 1731.xml di chiamata originale sia
2009-06-11T12:20:46 Nessuna modifica orario partenza
2009-06-11T12:20:46 Scrivo il file RICACOLO_1731.xml nella cartella
2009-06-11T12:20:46 Waiting for /var/www/sito2/XML/received/RISPOSTA_
2009-06-11T12:20:47 Ottengo le tratte dal file: /var/www/sito2/XML/p
2009-06-11T12:20:47 Numero di tratte pre-evento: 3
2009-06-11T12:20:47 Numero di tratte post-evento: 3
2009-06-11T12:20:47 Stesso numero di tratte
2009-06-11T12:20:47 Ora di partenza diversa al tratto0
2009-06-11T12:20:47 Ora di arrivo diversa al tratto0
2009-06-11T12:20:47 Ora di partenza diversa al tratto1
2009-06-11T12:20:47 Ora di arrivo diversa al tratto1
2009-06-11T12:20:47 Ora di partenza diversa al tratto2
2009-06-11T12:20:47 Ora di arrivo diversa al tratto2
2009-06-11T12:20:47 Creazione messaggio nuovo orario....
2009-06-11T12:20:47 Invio SMS....
2009-06-11T12:20:47 Ottenuto numero 1234 per l'utente tp
2009-06-11T12:20:47 Invio SMS a tp sul numero 1234, testo: Causa pro
2009-06-11T12:20:47 Cancello eventi pendenti per la richiesta 92
2009-06-12T14:41:45 Connected to db_TRIP_TO
2009-06-12T14:41:45 Connected to db_TRIP_TO
2009-06-17T14:11:18 Connected to db_TRIP_TO
2009-06-17T14:11:18 Connected to db_geografico
2009-06-17T14:11:18 Connected to db_TRIP_TO
2009-06-17T14:11:18 Connected to db_geografico
2009-07-01T14:59:54 Connected to db_TRIP_TO
2009-07-01T14:59:54 Connected to db_TRIP_TO
2009-07-01T14:59:54 Connected to db_geografico
2009-07-03T14:29:05 Connected to db_TRIP_TO
2009-07-03T14:29:05 Connected to db_TRIP_TO
2009-07-03T14:29:05 Connected to db_geografico
2009-07-06T09:35:49 Connected to db_TRIP_TO
2009-07-06T09:35:49 Connected to db_TRIP_TO
2009-07-06T09:35:49 Connected to db_geografico
2009-07-06T09:35:49 Connected to db_TRIP_TO
2009-07-06T09:35:49 Connected to db_TRIP_TO
2009-07-06T09:35:49 Connected to db_geografico
2009-08-27T10:34:37 Connected to db_TRIP_TO
2009-08-27T10:34:37 Connected to db_TRIP_TO
2009-08-27T10:34:37 Connected to db_geografico
  
```

On the right side of the page, there is a sidebar with the text 'Benvenuto gli [Admin_10gi]', 'Area Utente', 'Logout', and 'Partner Progetto Trip:'. Below this, there are logos for WiLab, Gruppo Almagiva, and TSF.

Fig. 132 - Pagina del portale per la gestione dello script real-time

STUDI DI FATTIBILITA' PER GLI SCENARI DI POSSIBILE EVOLUZIONE

La possibilità di raggiungere l'utente con un'informazione all'utente

- Per raggiungere il maggior numero di utenti si scelto come mezzo di base l'utilizzo del SMS (grande diffusione del cellulare GSM)
- Alternative si stanno aprendo con la diffusione di terminale "smart"
- Scelta di permettere accesso a un portale ridotto da parte di terminali con accesso internet
- Sono possibili altri scenari:
 - Evoluzione del SMS verso MMS con il contributo di informazioni visive (piccole mappe dei punti di cambio) la cui creazione è possibile avendo pieno controllo della gestione mappe (software open)
 - Terminali evoluti e dotati di ricevitore GPS possono installare piccole applicazioni per nuove funzionalità (...)

Informazioni via MMS

Introdurre la capacità di inviare MMS vuol dire predisporre un componente software che gestisca l'invio dei dati multimediali.

MMS

The Multimedia Messaging Service (MMS), as its name implies, is intended to provide a rich set of content to subscribers in a messaging context. It supports both sending and receiving of such messages by properly enabled client devices. An example of such a message is shown in Fig. 133 below.



Fig. 133 - Schema di base della tecnologia MMS

The Multimedia Messaging Service is viewed as a non-real-time delivery system. This is comparable to many messaging systems in use today. Prime examples include traditional email available on the Internet and wireless messaging systems such as paging or SMS.

These services provide a store-and-forward usage paradigm and it is expected that the MMS will be able to interoperate with such systems.³⁰

Integrazione nel sistema

Oltre al software di gestione, tale funzionalità deve essere integrata opportunamente nella piattaforma TRIP. Per questo occorre:

- Dare all'utente in fase di registrazione la possibilità di scegliere se la tipologia di informazioni che vuole ricevere in caso di avvisi real time sia SMS o MMS

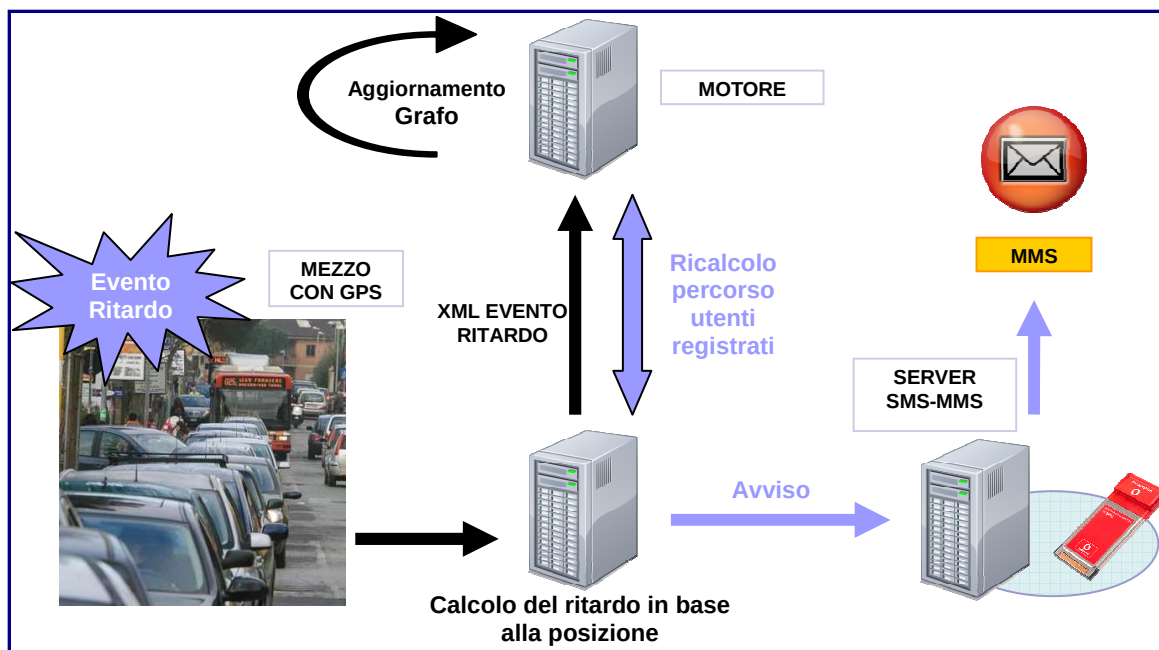


Fig. 134 – Segnalazione realtime di un nuovo percorso attraverso MMS

- Dare all'utente che invia un SMS per effettuare una ricerca di percorso la possibilità di ricevere la risposta via MMS: quindi la richiesta sarà una semplice stringa del tipo DA: ***** A: ***** , mentre la risposta sarà un insieme di descrizione ed immagini sul percorso da seguire;

³⁰ http://www.openmobilealliance.org/Technical/release_program/mms_v1_3.aspx

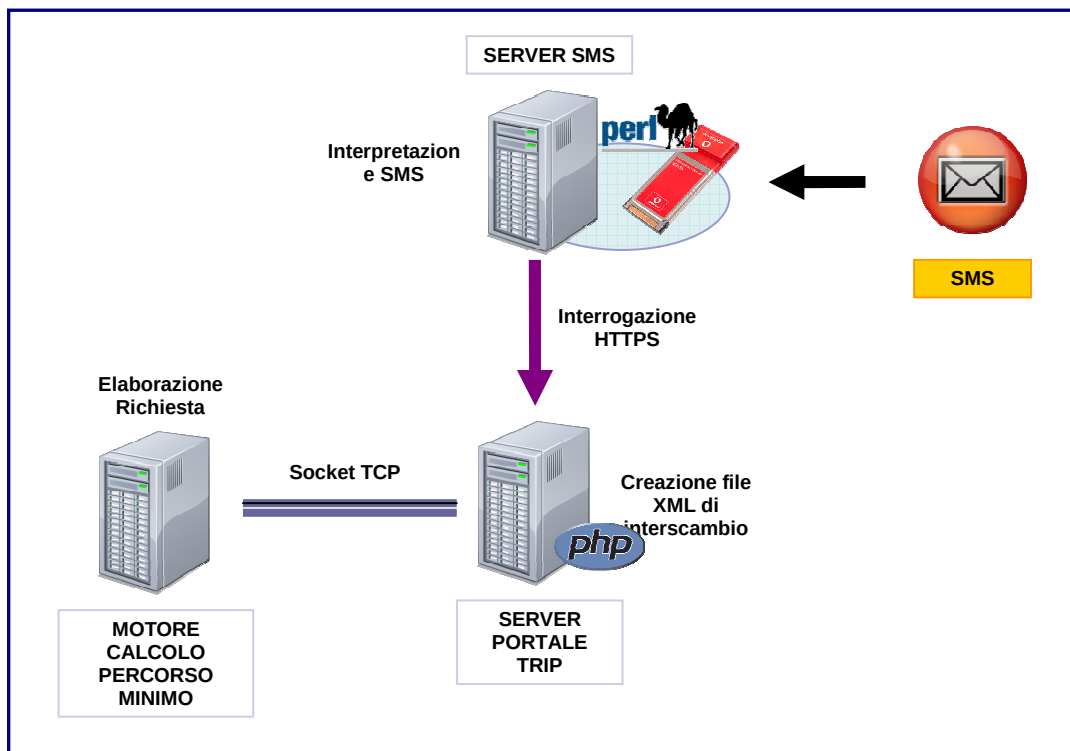


Fig. 135 - Richiesta di informazioni attraverso SMS

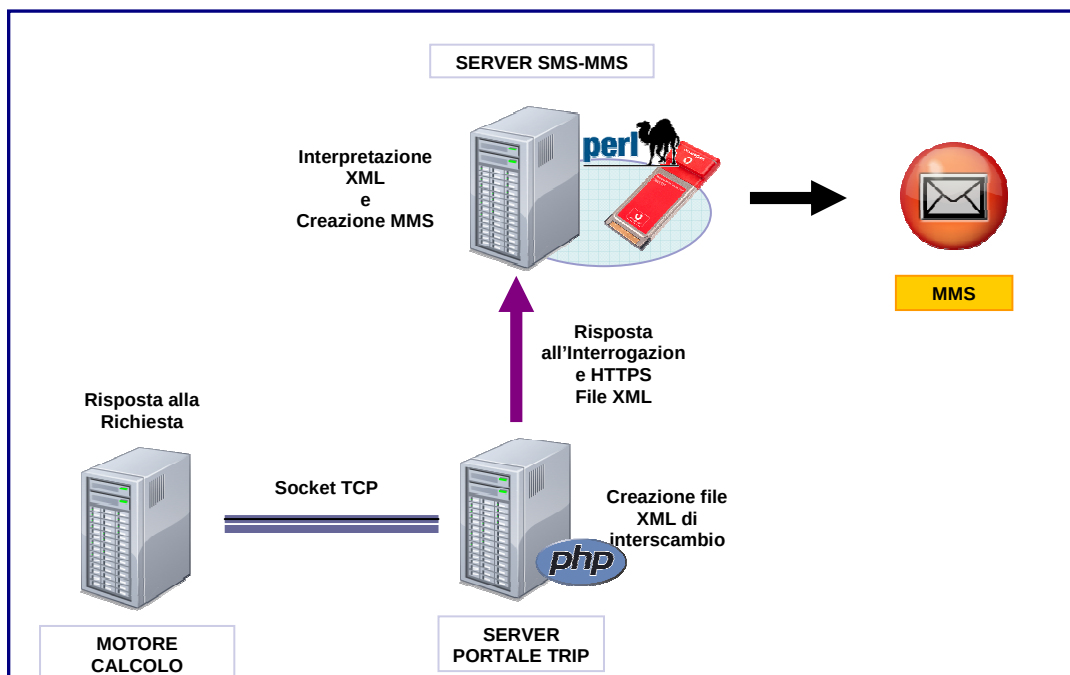


Fig. 136 - Risposta ad una richiesta SMS via MMS

- Prevedere una procedura che in base alle preferenze indicate dall'utente provveda alla costruzione di un oggetto SMS o un oggetto MMS che verrà poi inoltrato all'utente

- Prevedere un interfaccia tra la procedura indicata al punto precedente e il server GIS per la creazione delle immagini con mappa e parti salienti del percorso da integrare nel MMS

Il limite imposto dalla tecnologia MMS per quanto riguarda la mole di dati multimediali, per la maggioranza degli operatori, è fissato a 300kB, mentre per alcuni telefoni il limite si attesta a 100kB. I file con le parti di mappa da inviare per mostrare i punti salienti sono immagini che occupano in media circa 20kB, quindi è chiaro vedere come, in entrambi i casi, sia possibile fornire per un percorso una rappresentazione abbastanza dettagliata.



Fig. 137 - Evoluzione da SMS a MMS

Sviluppo Software

Il software che gestisce l'invio del MMS sarà un'estensione del software di invio SMS. In particolare si estenderà la parte di scrittura su SIM (ossia il processo che gestisce il flusso di dati in uscita verso l'utente mobile remoto) in modo da poter gestire le due tipologie di oggetti, SMS e MMS, e di inviarle correttamente.

- Nel caso di oggetto SMS l'invio procede come descritto nella descrizione della procedura di gestione SMS
- Nel caso di oggetto MMS occorre effettuare le seguenti operazioni sulla SIM:
 - Creazione ed apertura di connessione GPRS-WAP verso portale MMS dell'operatore
 - Invio dell'oggetto MMS opportunamente strutturato attraverso tale connessione
 - Chiusura della connessione GPRS-WAP instaurata.

GPRS AT Commands

La gestione di apertura, gestione, chiusura della connessione GPRS avviene attraverso comandi AT opportuni. Come per il descritto nella documentazione dello Script di Gestione della SIM per SMS, tali comandi vengono inviati direttamente al modem GSM che contiene la SIM attraverso un accesso di tipo seriale. Per gestire la scrittura/lettura di tale stringhe di comando si è scelto il linguaggio di scripting Perl (si faccia riferimento alla documentazione relativa a SCRIPTING PER AUTOMATIZZAZIONE SERVIZI).

Libreria Java jwap

jWAP is a Java implementation of the Wireless Application Protocol (WAP). It includes Wireless Session Protocol (WSP) and Wireless Transaction Protocol (WTP). Additionally, jWAP contains a WAP Binary XML (WBXML) Parser/Generator. jWAP requires Java V >= 1.2.

Class Summary	
<u>GetRequest</u>	Use this class for executing a WSP GET request
<u>PostRequest</u>	Use this class for executing a WSP POST request.
<u>Request</u>	Base class for WSP GET/Post requests.
<u>Response</u>	This class represents a Response on a WSP GET or POST request
<u>WAPClient</u>	This class represents a WSP "User-Agent" which can be used for executing WSP GET and POST methods.

Libreria Java jmmslib

jMmsLib is Java implementation of binary Multimedia Messaging Services (MMS) encoding/decoding. It supports creation and encoding of *send_request* messages and decoding of *send_conf* responses.

Class Summary	
<u>MmsDecoder</u>	Decodes a binary mms object in a MmsMessage class.
<u>MmsEncoder</u>	Encodes an MmsMessage class in a binary MMS object ready to be sent to an MMSC.
<u>MmsMessage</u>	Represents an mms message.
<u>MmsPart</u>	Defines a part of the MMS message.

Exception Summary	
<u>MmsContentException</u>	An error occur reading the content of a mms part.
<u>MmsDecoderException</u>	An error occur decoding the mms message.
<u>MmsEncodingException</u>	An error occur encoding the mms message.
<u>MmsMessageException</u>	An error occurred during mms message creation.

SendMMS

SendMMS is a simple command line tool for creating and sending MMS messages through a WAP connection. It uses *jMmsLib* for creating/encoding the messages and *jWAP* to actually send them to a MMSC. It supports text and images (actually only JPEG images are supported, but adding support for others image formats is trivial).

Usage

```
sendmms [-s ] [-t | -T | -i | ...]
         [-is | -ts | -S | --skip | -h | -v]

-s: defines the subject of the message
-t: adds a mms-part composed by the string
-T: adds a text mms-part composed by the content of
-i: adds an image mms-part composed by the content of
-is: define the max size of an image content
-ts: define the max size of a text content
-S: define the max size of the mms message
--skip: skips problematic contents
-v: verbose mode
-h | --help: this help
```

Exit Status

```
0: ok
1: error in configuration/arguments
2: error in mms contents
3: error during binary mms building
4: error sending mms
5: other errors (I/O)
```

Terminali evoluti dotati di GPS

Per questa tipologia di terminali si può prevedere un'evoluzione del servizio attraverso l'utilizzo dei moduli GPS integrati nei dispositivi.

Infatti la conoscenza in tempo reale dell'esatta posizione dell'utente può essere sfruttata per:

- L'utente effettua una richiesta per un percorso senza indicare il punto di partenza, il punto di partenza è la sua posizione attuale, ottenuta via GPS
- L'utente viene informato dell'avvicinarsi di un punto di cambio mezzo: la posizione rilevata dal GPS si approssima al punto di scambio; sotto una certa soglia di distanza l'utente viene avvertito

Sviluppo

Per realizzare tali servizi occorre installare sull'utente un'applicazione specifica che effettui alcuni processi specifici:

- Leggere dal modulo GPS i dati di posizione
- Poter memorizzare la posizione di alcuni punti di interesse o cambio mezzi
- Poter confrontare la prossimità di due punti di coordinate note
- Poter generare un qualche effetto sonoro/visivo per allertare l'utente di certe condizioni

Si può immaginare di dividere in due macro categorie i dispositivi dotati di modulo GPS

- Sistemi con sistema operativo (Symbian, linux, Windows mobile)
- Sistemi senza sistema operativo

I primi sono apparati più sofisticati che accolgono al loro interno un vero e proprio sistema operativo opportunamente scalato per le capacità di calcolo e visualizzazione del dispositivo.

Tali apparati possono installare una grande varietà di applicazioni sviluppate in un linguaggio di programmazione accettato dal sistema operativo. Questi apparati offrono quindi una maggiore flessibilità nella realizzazione di un'applicazione.

Tuttavia è possibile pensare di realizzare le applicazioni da installare sui dispositivi mobili in linguaggio JAVA (più precisamente linguaggio JAVA conforme al J2ME, ossia la distribuzione per apparati mobili). La scelta del linguaggio JAVA limita in parte la flessibilità nella realizzazione del software, ma permette di scrivere applicazioni compatibili con entrambe le categorie di apparati descritte sopra.

J2ME

Java è un linguaggio di programmazione orientato agli oggetti, derivato dallo Smalltalk (anche se ha una sintassi simile al C++) e creato da James Gosling e altri ingegneri di Sun Microsystems. La piattaforma di programmazione Java è fondata sul linguaggio stesso, sulla Macchina virtuale Java (Java Virtual Machine o JVM) e sulle API Java. Java è un marchio registrato di Sun Microsystems.

Java venne creato per soddisfare quattro scopi:

- essere completamente orientato agli oggetti
- essere indipendente dalla piattaforma
- contenere strumenti e librerie per il networking
- essere progettato per eseguire codice da sorgenti remote in modo sicuro.

Per facilitare il passaggio a Java ai programmatori old-fashioned, legati in particolare a linguaggi come il C, la sintassi di base (strutture di controllo, operatori e così via) è stata mantenuta pressoché identica a quella del C.

Per coprire al meglio le esigenze di diversi scenari, esistono diverse “versioni” della piattaforma Java:

- J2SE (Standard Edition) è la piattaforma di riferimento per lo sviluppo di applicazioni desktop (scenario obsoleto???)
- J2EE (Enterprise Edition) è la piattaforma di riferimento per le applicazioni Web. Da notare che Tomcat, JSP, etc. fanno parte di J2EE
- J2ME (Micro Edition) è la piattaforma di riferimento per applicazioni embedded (es. Cellulari, palmari, ma anche giocattoli ed elettrodomestici).

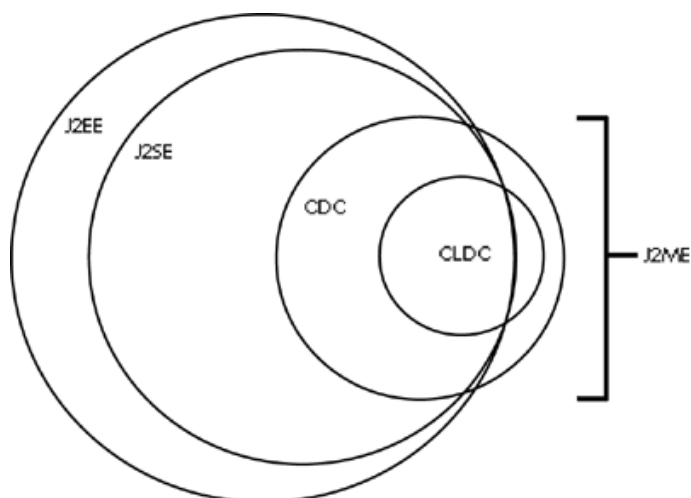
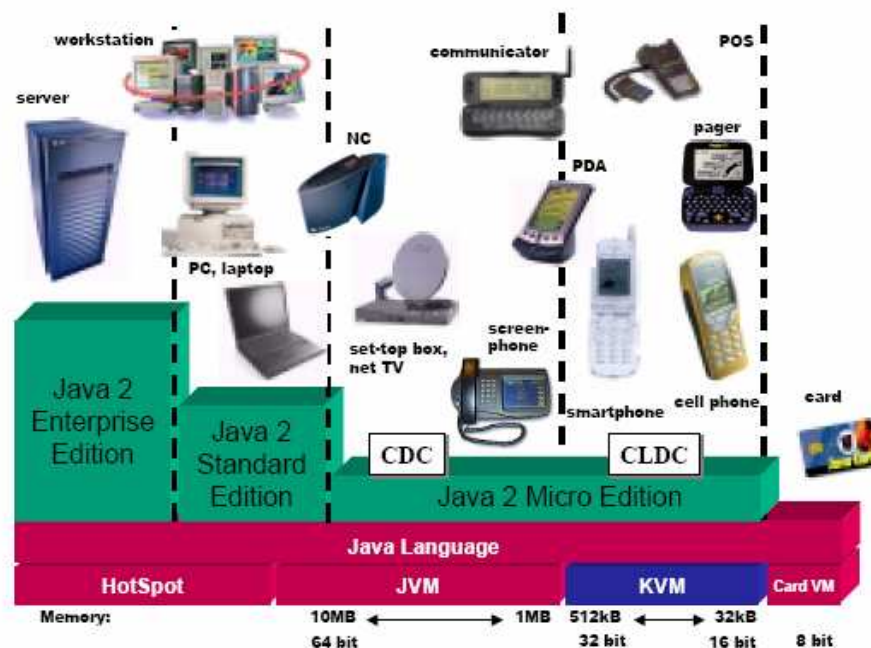


Fig. 138 - Schema riassuntivo della piattaforma JAVA

I vantaggi (e gli svantaggi) di Java per quanto riguarda J2ME si accentuano:

- La portabilità del codice diventa ancora più importante che per le applicazioni desktop e server – perchè aumenta l'eterogeneità dell'hardware.
- Le applicazioni embedded tendono ad essere molto complicate. Quindi le caratteristiche di java come linguaggio di programmazione pulito e potente sono molto utili
- Si sta imponendo come standard (quasi tutti i cellulari offrono J2ME)
- Problemi di performance (sia velocità che occupazione di memoria)

La struttura J2ME nel suo complesso è abbastanza complicata e modulare. Questo perché gli scenari hardware J2ME sono molteplici e molto diversi tra loro (es. da un computer palmare a un giocattolo di ultima generazione).

Per questo motivo J2ME si compone di due parti separate:

- Configurations
- Profiles

Una Configuration è un'infrastruttura per eseguire programmi Java in una certa classe di dispositivi. Fondamentalmente si compone di una Java Virtual Machine ottimizzata per quella classe di dispositivi. Allo stato attuale esistono solo 2 configuration:

- CLDC (Connected Limited Device Configuration) che si basa sulla KVM (Kilo Virtual Machine). Questa è la configurazione presente in quasi tutti i cellulari moderni. Caratteristiche del dispositivo:
 - 128 kB minimi di memoria per Java
 - 32 kB minimo di memoria "runtime"
 - interfaccia utente ridotta
 - bassi consumi, alimentazione a batteria
- CLC (Connected Device Configuration) che si basa sulla CVM (Compact Virtual Machine). Questa è la configurazione presente sui palmari e su smartphone. Caratteristiche del dispositivo:
 - 512 kB minimo di memoria per l'applicazione Java (codice)
 - 256 kB minimo di memoria "runtime", allocata dall'applicazione

I Profiles sono componenti aggiuntivi (delle specie di librerie – package java) che offrono strumenti e funzionalità evolute. Allo stato attuale sono disponibili diversi profiles, ma il più usato in assoluto e montato su tutti i cellulari che offrono CLDC è la MIDP (Mobile Information Device Profile -- 1.0 e 2.0).

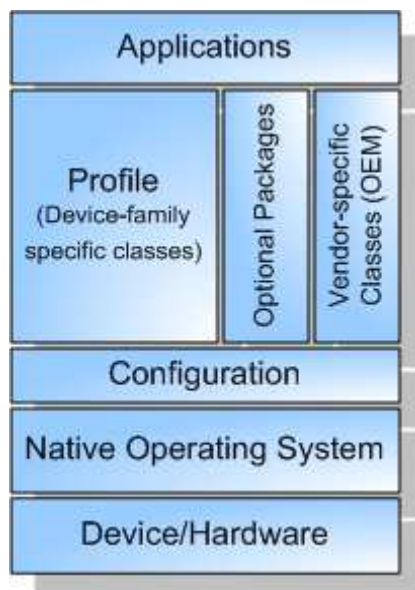


Fig. 139 - Architettura J2ME

Librerie Abilitanti JAVA

Tra le librerie che permettono la realizzazione di applicazioni inerenti localizzazione, coordinate GPS, calcoli di distanza e prossimità sicuramente troviamo la Location API definita dal documento JSR 293 Version 2.0 dal Java Community Process / JSR 293 Expert Group.

Per completezza riportiamo l'introduzione di tale documento

"Location API Contains the basic classes needed to request and get a location result.

The LocationProvider class represents a module that is able to determine the location of the terminal. This may be implemented using any possible location method, for example, satellite based methods like GPS, cellular network based methods, short-range positioning methods like Bluetooth Local Positioning, etc. The implementation may also combine methods in various ways to get the optimal result.

The application can specify criteria for selecting the location provider and obtain a LocationProvider instance that is able to fulfil these criteria as closely as possible. By using the LocationProvider, the application can get Location objects representing the location of the terminal at the time of the measurement. The application can either request a single Location object or be periodically updated with new Location objects via a LocationListener implemented by the application.

The location is represented by the Location object that contains a QualifiedCoordinates object representing the geographical coordinates (latitude, longitude and altitude) and information about their accuracy, a timestamp and possibly information about the speed and course of the terminal. For some location methods, the Location object may also contain an AddressInfo object that includes textual address information, e.g. a street address.

The Location gives the accuracy of the coordinates as the radius of a circular area indicating the 68% confidence level.

This implies that the actual location is within the circle defined by the returned point and radius at a probability of approximately 68%. The actual location may thus also be outside of the circle, which has to be taken into account when using the location information in applications.

*This package also includes a database of landmarks. A landmark is a known physical location that is associated with a name representing that location for the end user. The user can store commonly used locations in the database. Examples of landmarks could be, for example, the user's home, office, etc. The default landmark database **must** be shared*

between all Java applications and **may** be shared with other applications in the terminal, including native applications, so that the end user has a consistent user experience across all location based applications on the terminal.

New features in version 2.0

Importing and exporting landmark and global landmark categories have been added to version 2.0 to enable the exchange of landmarks between different terminals. With the landmark exchange the user is able export landmarks stored in the terminal to another terminal or to a computer and import new landmarks to the terminal or, for example, from a points of interests database. The global landmark categories guarantee interoperability with different terminals. These categories **must** be present in all Location API 2.0 compliant terminals and the global category names **must** be localized to improve the user experience." Location API 2.0 also makes improvements to the **Criteria** object allowing priorities to be set for different criterion. Creating private landmark stores is also one of the new features in Location API 2.0."

e la lista di interfacce e classi messe a disposizione

Interface summary	
AreaListenerExchangeFormatHandler	This interface represents a listener to events associated with crossing the boundary of the given geographical area.
LandmarkStoreListener	This interface provides methods to exchange landmarks. This interface represents a listener that receives events associated with changes in the landmark store.
LocationListener	The LocationListener represents a listener that receives events associated with a particular LocationProvider .
ProximityListener	This interface represents a listener to events associated with detecting proximity to some registered coordinates.
Class summary	
AddressInfo	The AddressInfo class holds textual address information about a location. This class represents a circular geographical area on WGS 84 Ellipsoid (see reference [WGS84]).
CircleGeographicArea Coordinates	The Coordinates class represents coordinates as latitude-longitude-altitude values. The criteria used for the selection of the location provider is defined by the values in this class.
Criteria	This class provides a mechanism for applications to obtain instances of different ExchangeFormatHandler objects.
ExchangeFormatFactory	This interface represents a geographical area on the surface on WGS 84 Ellipsoid (see reference [WGS84]).
GeographicArea Landmark	The Landmark class represents a landmark, such as a known location with a name. The LandmarkStore class provides methods to store, delete and retrieve landmarks from a persistent landmark store.

LandmarkStore Location	<i>The Location class represents the standard set of basic location information. This is the starting point for applications using location information in this API and represents a source of the location information.</i>
LocationProvider Orientation	<i>The Orientation class represents the physical orientation of the terminal.</i>
PolygonGeographicArea	<i>This class represents a polygon geographical area on WGS 84 Ellipsoid (see reference [WGS84]).</i>
QualifiedCoordinates	<i>The QualifiedCoordinates class represents coordinates as latitude-longitude-altitude values that are associated with an accuracy value.</i>
RectangleGeographicArea	<i>This class represents a rectangular geographical area on WGS 84 Ellipsoid (see reference [WGS84]).</i>
Exception summary	
LandmarkException	<i>The LandmarkException is thrown when an error related to handling landmarks has occurred.</i>
LocationException	<i>The LocationException is thrown when a Location API specific error has occurred.</i>

Location e LocationProvider

Esistono varie tecnologie per il calcolo della posizione di un dispositivo radiomobile: ognuna di queste si caratterizza per la precisione che è in grado di fornire, per il tipo di rete alla quale è applicabile, per la richiesta o meno di un qualche supporto (hardware/software) da parte del terminale stesso. Vediamo una breve panoramica delle tecniche più utilizzate.

Il sistema più semplice è il Cell-ID che si basa sulla posizione (nota) della cella per localizzare il dispositivo. Risulta evidente che l'approssimazione della posizione sarà tanto migliore quanto più ristretta sarà la superficie coperta da ogni singola cella. Per questa ragione il Cell-ID risulta molto impreciso nelle aree extraurbane dove le celle coprono superfici dell'ordine di km. Nella versione Timing Advance, oltre all'identificativo della cella viene valutata la distanza radiale del dispositivo dall'antenna della cella (tempo di propagazione del segnale) aumentandone così la precisione.

I sistemi EOTD (Enhanced Observed Time Difference) e TDOA (Time Difference Of Arrival) si basano sulla triangolazione ed utilizzano il ritardo di propagazione del segnale dall'apparato radio-mobilità a quello radio-base per la stima della posizione. Ogni dispositivo mobile riceve contemporaneamente il segnale proveniente da più celle: in base ai ritardi di propagazione misurati o alla differenza tra i ritardi stessi è possibile stimare con buona approssimazione la posizione dell'apparato. Si tratta, più o meno della stessa tecnica utilizzata dal sistema GPS con la differenza che, mentre quest'ultimo utilizza una costellazione di satelliti (24 in 6 diverse orbite) come "punti noti", nell'EOTD e nel TDOA i ritardi vengono calcolati con le celle della rete.

La versione "assistita" del GPS prevede il supporto della rete radio-mobilità in modo da diminuire i tempi di localizzazione. Ogni cella occupa, infatti, una posizione nota in termini di latitudine, longitudine che costituiscono la base di partenza per il calcolo del GPS. Il GPS, in questo caso, conosce già, seppure in modo approssimato, la posizione del dispositivo e questo rende più veloce una localizzazione accurata (riducendo i tempi della cosiddetta "partenza a freddo" del GPS).

Il sistema AOA (Angle Of Arrival) prevede l'utilizzo di "antenne intelligenti" (antenne "settoriali") nelle celle della rete. In questo caso la posizione degli apparati radio-mobili viene stimata valutando l'angolo di arrivo del segnale in almeno tre diverse celle.

Nei sistemi AFLT (Advanced Forward Link Trilateration) e EFLT (Enhanced Forward Link Trilateration), il sistema di calcolo della posizione è implementato direttamente sul dispositivo mobile. I tempi di ritardo del segnale tra stazione-base e l'apparato mobile vengono calcolati direttamente da quest'ultimo. Questo rende necessario un meccanismo di sincronizzazione tra l'apparato e la cella.

La tabella che segue (Fig. 140) riassume quanto detto.

Tipo	Precisione	Tipo di rete	Richiesta supporto da parte del terminale
Cell - ID (Cell - IDentity)	Dipende dalle dimensioni della cella (100 m - 3 km)	Qualsiasi tipo di rete	Nessuno
Cell - ID TA (Cell - IDentity - Timing Advance)	Dipende dalla larghezza di banda (configurabile). Il valore di default è di 500 m	GSM	Nessuno
EFLT (Enhanced Forward Link Trilateration)	250 - 400 m	CDMA	Nessuno
TDOA (Time Difference Of Arrival)	100 - 200 m	Qualsiasi tipo di rete	Nessuno
AOA (Angle of Arrival)	100 - 200 m	Qualsiasi tipo di rete	Nessuno
AFLT (Advanced Forward Link Trilateration)	50 - 200 m	CDMA	Sì
EOTD (Enhanced Observed Time Difference)	50 - 200 m	GSM	Sì
GPS (Global Position System)	5 - 30 m	Qualsiasi tipo di rete	Sì
AGPS (Assisted Global Position System)	5 - 30 m	Qualsiasi tipo di rete	Sì

Fig. 140 - Sistemi di localizzazione radiomobile

Vediamo ora come si può sviluppare un'applicazione location-based in java-midp 2.0. A questo scopo esiste un package opzionale di Java ME denominato Location API. Si tratta di una libreria che si integra con MIDP versione 2.0 mentre non è compatibile con la 1.0 in quanto richiede il supporto dei numeri floating-point, caratteristica non presente nella prima versione MIDP. Cuore dell'API è la classe LocationProvider che permette di ottenere, in modo del tutto trasparente (per quando riguarda il sistema di localizzazione utilizzato, a basso livello, dall'apparato), informazione sulla posizione dell'apparato in un dato momento.

Il riferimento al provider si ottiene con una chiamata al metodo getInstance. Dato che in un apparato possono coesistere più sistemi di localizzazione, si possono avere provider diversi in base al particolare sistema. Per questa ragione il metodo getInstance prevede un parametro di tipo Criteria:

```
public static LocationProvider getInstance(Criteria criteria)
throws LocationException
```

utilizzato per la scelta del tipo del fornitore del servizio di localizzazione che si desidera utilizzare:

```
Criteria criteria = new Criteria();
criteria.setCostAllowed(true);
criteria.setAltitudeRequired(true);
.....
.....
LocationProvider provider = LocationProvider.getInstance(criteria);
```

La scelta del provider può avvenire in base a numerosi criteri (tabella in Fig. 141):

costo del servizio	setCostAllowed (true/false)
precisione	setHorizontalAccuracy, setVerticalAccuracy
informazioni sull'altitudine	setAltitudeRequired (true/false)
informazioni su velocità e direzione	setSpeedAndCourseRequired (true/false)
consumo del dispositivo	setPreferredPowerConsumption (POWER_USAGE_HIGH , POWER_USAGE_MEDIUM , POWER_USAGE_LOW)
tempi di risposta	setPreferredResponseTime (true/false)
informazioni sull'indirizzo associato alla location	setAddressInfoRequired (true/false)

Fig. 141 - Criteri di scelta del LocationProvider

Il metodo di factory `getInstance` restituisce il provider che meglio soddisfa tali criteri, oppure null se nessun sistema presente possiede le caratteristiche richieste. Una volta ottenuto il riferimento al `LocationProvider` lo si può interrogare per ottenere la posizione dell'apparato nell'istante in cui viene effettuata la richiesta:

```
public Location getLocation(int timeout) throws LocationException,
InterruptedException
```

L'oggetto `Location` contiene informazioni relative alle coordinate geografiche del punto in cui ci si trova, che possono essere lette tramite il metodo

```
QualifiedCoordinates getQualifiedCoordinates()
```

`QualifiedCoordinates` è un'estensione della classe `Coordinates` e rappresenta le coordinate geografiche di un punto (longitudine - latitudine - altitudine) e la relativa accuratezza:

```
public QualifiedCoordinates(double latitude, double longitude,
float altitude,
float horizontalAccuracy, float verticalAccuracy)
```

Le coordinate sono espresse nello standard WGS84 (World Geodetic System 1984). La latitudine varia tra +90.0 e -90.0: valori positivi indicano un punto nell'emisfero nord mentre quelli negativi si riferiscono a punti dell'emisfero sud. La longitudine assume valori

compresi tra +180 e -180 man mano che ci si sposta da est verso ovest. L'altitudine è espressa in metri e rappresenta l'altezza dell'ellissoide definito dallo standard WGS84 (Fig. 142).

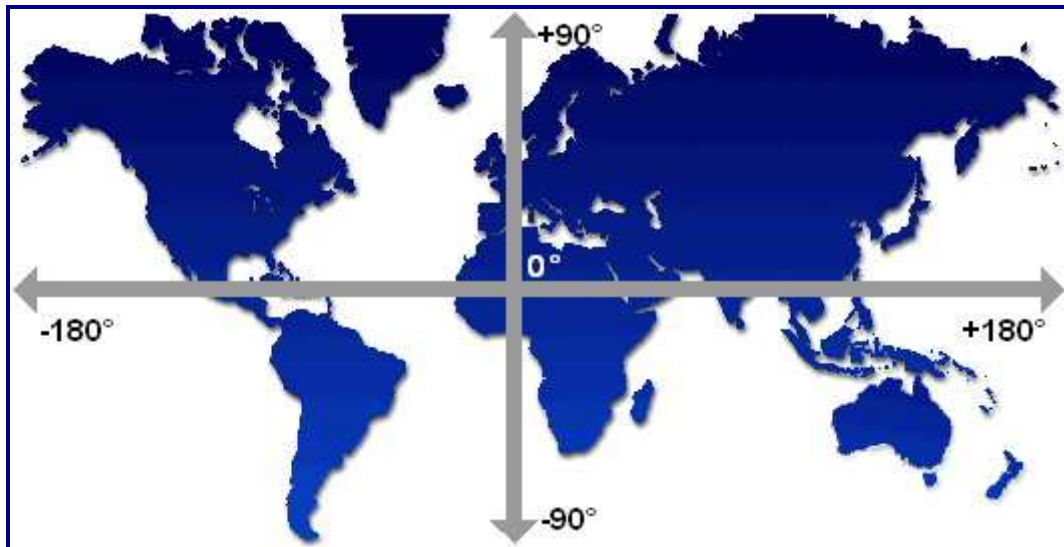


Fig. 142 - Coordinate

Oltre alle informazioni relative alle coordinate spaziali, potrebbe essere utile conoscere la velocità di spostamento dell'apparato radiomobile e il suo orientamento rispetto al nord terrestre. Per ottenere tali dati la classe Location fornisce i metodi:

```
public float getSpeed()  
public float getCourse()
```

Il primo restituisce appunto la velocità con cui l'apparato si sta muovendo espressa in m/s, mentre `getCourse()` ritorna un valore compreso tra 0.0 e 360 gradi riferito al nord terrestre.

Il metodo:

```
public int getLocationMethod()
```

restituisce il metodo di localizzazione utilizzato per stabilire la location. Il valore ritornato da questo metodo è il risultato di un OR bit a bit tra tre diverse componenti:

- tecnologia utilizzata per la localizzazione (MTE_ANGLEOFARRIVAL, MTE_CELLID, MTE_SATELLITE, MTE_SHORTRANGE, MTE_TIMEDIFFERENCE, MTE_TIMEOFARRIVAL);
- metodo utilizzato (MTY_NETWORKBASED, MTY_TERMINALBASED);
- grado di assistenza (MTA_ASSISTED, MTA_UNASSISTED). Nel caso di sistemi assisted la valutazione della locazione viene effettuata con l'ausilio della controparte: assistenza da parte del terminale per i sistemi "network based"; viceversa, nei sistemi di tipo "terminal based" l'aiuto è fornito dalla rete.

Per ottenere le informazioni sulla posizione si può, in alternativa alla lettura diretta, utilizzare il listener `LocationListener` al quale viene notificato il cambiamento di locazione con l'invocazione del metodo:

```
public void locationUpdated(LocationProvider provider, Location
location)
```

Il listener possiede inoltre il metodo:

```
void providerStateChanged(LocationProvider provider, int newState)
per la notifica di eventuali cambiamenti di stato del provider del servizio di localizzazione
(AVAILABLE, OUT_OF_SERVICE, TEMPORARILY_UNAVAILABLE).
```

Il listener deve essere agganciato al location provider con il metodo:

```
public void setLocationListener(LocationListener listener, int
interval,
                                int timeout, int maxAge)
```

dove

- listener è l'ascoltatore
- interval è il lasso di tempo tra le notifiche (-1 indica il valore di default, 0 indica che l'applicazione è interessata solo al monitoring dello stato del provider e non all'"ascolto" della locazione)
- timeout è il tempo massimo entro il quale si desidera ottenere un nuovo valore della locazione (il valore -1 indica il timeout di default). Nel caso il provider non sia in grado di fornire un nuovo valore di locazione entro il valore di timeout impostato verrà ritornata una locazione non valida.
- La validità della locazione è ottenibile con il metodo public boolean isValid() della classe Location.
- maxAge, serve per stabilire il periodo di validità di una locazione. Questo può essere utile quando il sistema di localizzazione risponde con tempi abbastanza lunghi mentre l'applicazione non necessita di aggiornamenti molto frequenti. Allungando i tempi di maxAge si fa in modo che il sistema ritorni un locazione letta in precedenza se non sono disponibili dati aggiornati. Settando maxAge a -1 viene utilizzato il valore (maxAge massimo) di default.

La porzione di codice che segue mostra una semplice implementazione del listener che stampa le informazioni sulla location e sullo stato del provider.

```
.....
// Definisce un criterio per la ricerca del provider
Criteria criteria = new Criteria();
criteria.setCostAllowed(true);
criteria.setSpeedAndCourseRequired(true);

criteria.setPreferredPowerConsumption(Criteria.POWER_USAGE_MEDIUM);
.....
// Recupera un riferimento al provider
LocationProvider provider =
LocationProvider.getInstance(criteria);
// Setta un ascoltatore sul provider (i valori -1 indicano valori
di default)
provider.setLocationListener(new MyLocationListener(), -1, -1, -
1);
.....
.....
// Implementazione di test del listener.
```

```
// Stampa le informazione relative alla Location del dispositivo e
// allo stato del provider.
public class MyLocationListener implements LocationListener {

    public void locationUpdated(LocationProvider provider, Location
location) {
        QualifiedCoordinates qualifiedCoordinate =
location.getQualifiedCoordinates();
        System.out.println( "/***** Location
Event*****/");
        System.out.println("Location - Latitude: " +
qualifiedCoordinate.getLatitude());
        System.out.println("Location - Longitude: " +
qualifiedCoordinate.getLongitude());
        System.out.println("Location - Altitude: " +
qualifiedCoordinate.getAltitude());
        System.out.println("Location - Speed: " + location.getSpeed());
        System.out.println("Location - Course: " + location.getCourse());
        System.out.println("Location - Timestamp: " + new
Date(location.getTimestamp()));
        try {
            Orientation orientation = Orientation.getOrientation();
            System.out.println("Orientation - Compass Azimuth: "
+ orientation.getCompassAzimuth());
            System.out.println("Orientation - Pitch: " +
orientation.getPitch());
            System.out.println("Orientation - Roll: " +
orientation.getRoll());
            System.out.println("Orientation - Orientation Magnetic: "
+orientation.isOrientationMagnetic());
        }
        catch (LocationException e) {
            e.printStackTrace();
            ...
        }
    }

    public void providerStateChanged(LocationProvider provider,
int newState) {
        switch (newState) {
            case LocationProvider.AVAILABLE:
                System.out.println("Location provider available");
                break;
            case LocationProvider.OUT_OF_SERVICE:
                System.out.println("Location provider out of
service");
                break;
            case LocationProvider.TEMPORARILY_UNAVAILABLE:
                System.out.println("Location provider temporarily
unavailable");
                break;
        }
    }
}
```


ProximityListener

Oltre alla notifiche relative al cambiamento di locazione, può essere utile in taluni casi una notifica più "intelligente", qualcosa in grado di avvertirci che siamo giunti in prossimità di un determinato punto (ristorante, parcheggio, ecc.). La classe LocationProvider ha, a questo proposito, un apposito metodo:

```
public static void addProximityListener(ProximityListener listener,
                                         Coordinates coordinates, float
                                         proximityRadius)
```

che permette appunto di ottenere una notifica, tramite il ProximityListener, quando il dispositivo si viene a trovare a una distanza inferiore a proximityRadius dal punto avente coordinate coordinates (Fig. 143).

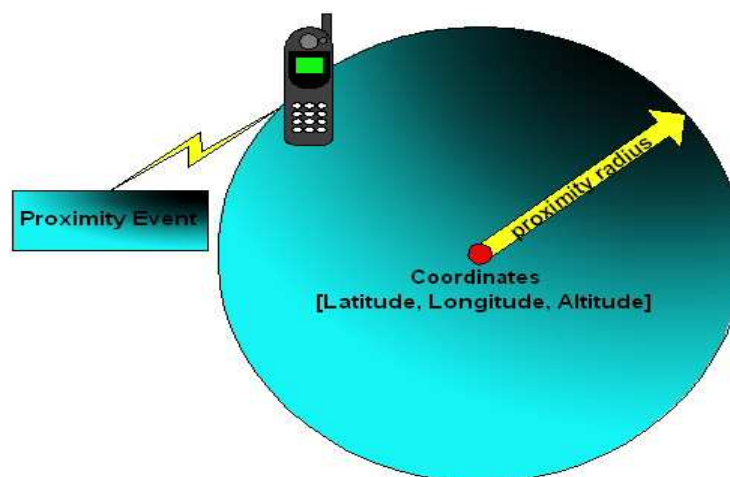


Fig. 143 - Proximity listener

Il codice che segue mostra una semplice implementazione di test del ProximityListener.

```
.....
// Definisce un criterio per la ricerca del provider
Criteria criteria = new Criteria();
criteria.setCostAllowed(true);
LocationProvider provider =
LocationProvider.getInstance(criteria);
// Coordinate di riferimento (sono le coordinate del punto di
interesse)
Coordinates refCoordinates = new Coordinates(43.77916,
11.26621, Float.NaN);
// Aggiunge il listener MyProximityListener. La notifica
avverrà quando il dispositivo
// verrà a trovarsi in un raggio di 100 metri dal punto di
```

```

coordinate refCoordinates
    provider.addProximityListener(new MyProximityListener(),
refCoordinates, 100);
    // Setta il location listener
    provider.setLocationListener(new MyLocationListener(), -1, -1,
-1);
    .....
    .....
    // Implementazione di test del proximity listener.
    // Stampa le informazione relative alla Location del
dispositivo quando questo
    // si viene a trovare in prossimità del punto di coordinate
coordinates
    public class MyProximityListener implements ProximityListener {

        public void monitoringStateChanged(boolean isMonitoringActive){
            System.out.println("Monitoring state changed:
"+isMonitoringActive);
        }

        public void proximityEvent(Coordinates coordinates, Location
location) {
            QualifiedCoordinates qualifiedCoordinate =
location.getQualifiedCoordinates();
            System.out.println("/***** Proximity Event
*****/");
            System.out.println("Location - Latitude: " +
qualifiedCoordinate.getLatitude());
            System.out.println("Location - Longitude: " +
qualifiedCoordinate.getLongitude());
            System.out.println("Location - Altitude: " +
qualifiedCoordinate.getAltitude());
            System.out.println("Location - Speed: " +
location.getSpeed());
            System.out.println("Location - Course: " +
location.getCourse());
            System.out.println("Location - Timestamp: " + new
Date(location.getTimestamp()));
        }
    }
}

```

Possibile Scenario applicativo 1

Utente effettua richieste senza dover indicare il punto di partenza, ma utilizzando il dato GPS.

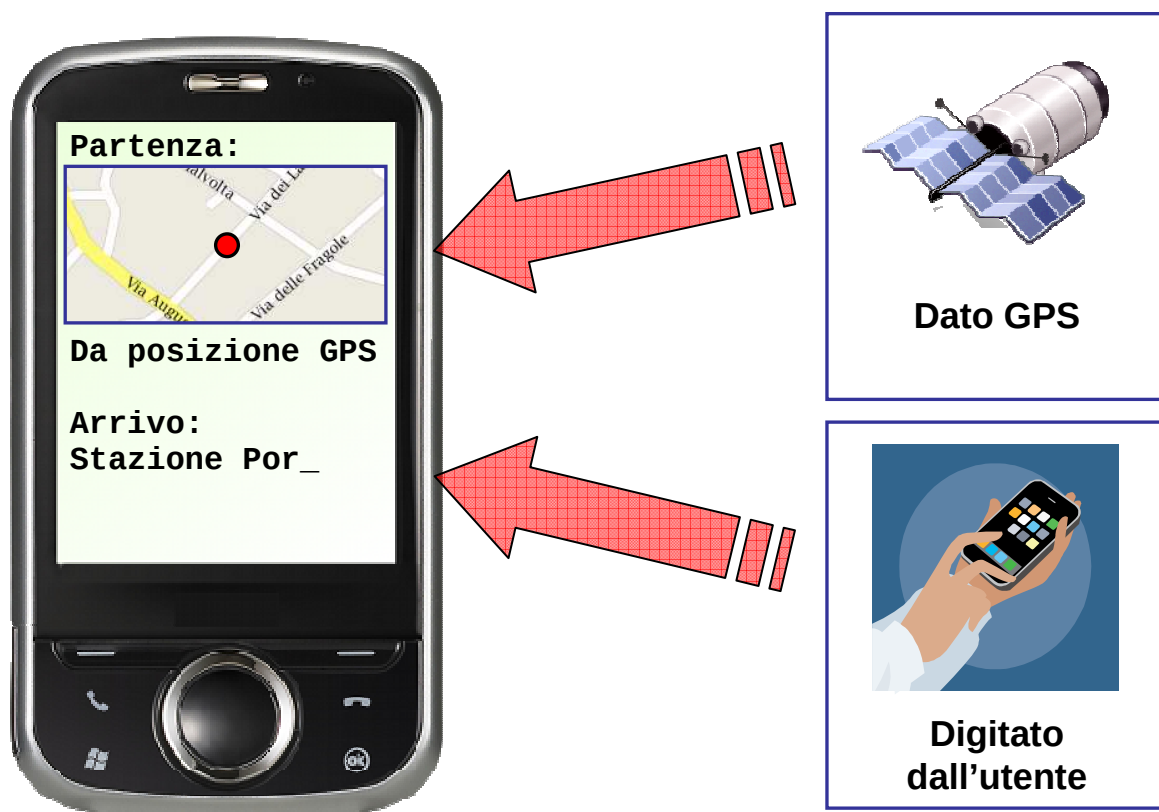


Fig. 144 - utilizzo del dato GPS per effettuare una richiesta

Possibile scenario applicativo 2

L'utente riceve un certo tipo di avviso all'avvicinarsi di un punto importante (punto di arrivo, fermata di cambio mezzo)



Fig. 145 - Avviso ricevuto in prossimità di una fermata di cambio mezzo

Possibile scenario applicativo 3

L'utente vede una mappa in cui vede punto destinazione e la propria posizione che si aggiorna mentre si muove verso il punto di arrivo.



Fig. 146 - Visualizzazione della posizione dell'utente rispetto un punto di destinazione

Integrazione

Lo sviluppo di un'applicazione da installare sui terminali d'utente comporta un attento studio per l'integrazione con il sistema esistente.

Come primo passo l'applicazione deve essere resa disponibile per l'installazione sul dispositivo dell'utente. La soluzione più semplice prevede di aggiungere una pagina al portale TRIP già esistente in cui si possa effettuare il download dell'applicazione su un personal computer per poi effettivamente passare all'installazione attraverso una connessione diretta PC → terminale.

Una soluzione più evoluta permette, sfruttando la struttura standardizzata della piattaforma JAVA-J2ME di predisporre il download diretto dell'applicativo da parte del terminale mobile, con conseguente installazione. Tale procedura elimina di fatto la necessità di una macchina che faccia da tramite e permette inoltre l'accesso remoto mobile.

Occorrerà quindi creare un nuovo servizio sempre attivo che possa dialogare con una pluralità di dispositivi mobili che installano applicazioni per la gestione dei dati GPS

APPENDICE

PORTALE DISEGNO PERCORSI

Descrizione e pianificazione del sistema

Il sistema sviluppato, denominato “Linee Mezzi Pubblici”, è un sistema distribuito che consente l’inserimento e la modifica di rotte urbane, nonché di ricercare e visualizzare le rotte inserite nel sistema. Supporta modifiche concorrenti di rotte da parte di più operatori, gestendo i casi in cui avvengano modifiche concorrenti su una stessa rotta da parte di due o più operatori. Implementa inoltre una serie di algoritmi per rendere rapido ed accurato l’inserimento e la modifica delle rotte e delle fermate associate a una specifica linea.

Requisiti e vincoli operativi

I requisiti utente (semplice oppure operatore) identificati per il progetto sono elencati in seguito:

- possibilità, per un utente non registrato, di visualizzare una rotta di una linea inserita nel sistema, caratterizzata da un nome ed una descrizione, da una sequenza di fermate e da una sequenza di archi stradali. Viene richiesta una visualizzazione grafica della linea.
- possibilità, per un utente non registrato, di visualizzare l'elenco delle linee inserite nel sistema. In particolare, devono poter essere accessibili tutte le linee definite come pubbliche.
- possibilità, per un utente non registrato, di poter effettuare delle ricerche sulle linee, utilizzando i seguenti criteri
- elenco di tutte le linee aventi una parola chiave nel nome
- elenco di tutte le linee aventi una parola chiave nel nome oppure nella descrizione
- elenco di tutte le linee la cui tratta transiti per un determinato indirizzo
- possibilità, per un utente operatore, di inserire nel sistema una nuova linea disegnandola sulla mappa ed aggiungendo le fermate
- possibilità, per un utente operatore, di modificare una linea, modificando il disegno sulla mappa ed aggiungendo/spostando/eliminando le fermate
- possibilità, per un utente operatore, di elencare tutte le linee presenti nel sistema, e di poter discriminare tra linee pubbliche e linee non ancora pubblicate
- possibilità, per un utente operatore, di poter salvare il proprio lavoro su una linea e continuarla in seguito

I requisiti di sistema ed i vincoli operativi sono definiti in seguito:

- **Piattaforma Web:** si richiede che il sistema in discussione sia fruibile tramite browser Web.
- **Produzione output grafico:** si richiede che il sistema produca un output grafico della rotta, comprensivo della localizzazione delle fermate.

- **Capacità di visualizzazione di layer cartografici da basi di dati:** si richiede che il sistema recuperi le informazioni cartografiche da una base dati.
- **Tracciamento grafico della linea:** si richiede che il sistema offra una modalità grafica per l'inserimento e la modifica delle linee, ed offra strumenti efficaci per il tracciamento delle stesse.
- **Definizione di fermate:** si richiede che il sistema offra una modalità grafica per l'inserimento, la modifica e la rimozione di fermate da una rotta.
- **Salvataggio delle linee:** si richiede che il sistema salvi nella base di dati centrale le rotte definite.
- **Salvataggio delle linee come archi stradali tra fermate:** si richiede che il sistema salvi nella base di dati centrale le rotte di ciascuna linea anche come archi stradali di collegamento tra una fermata e la successiva

Pianificazione e scelte di progetto

Per la fase progettuale e di sviluppo, è stato adottato un approccio guidato dal rischio, identificato in prima analisi nel formato dei dati a disposizione (nel caso del presente progetto, file Shapefile forniti da TeleAtlas da convertire e caricare su DBMS con estensioni spaziali abilitate), ed in seconda analisi sulle modalità di visualizzazione tramite web browser dei dati cartografici presenti nel sistema.

Pertanto la pianificazione progettuale è stata così articolata:

- Studio del formato Shapefile e dei principali strumenti software per la manipolazione dei dati cartografici (sistemi GIS). Inoltre in questa fase sono stati analizzati i formati ed i dati utilizzati dalla cartografia di TeleAtlas scelta per il progetto.

Le mappe fornite si dividono in vari gruppi, di cui i più importanti e significativi per il progetto in questione, sono:

- denominazione finale 00-09: suddivisione gerarchica delle strade (autostrade, strade extraurbane principali, strade extraurbane secondarie, strade urbane, strade locali...)
- denominazione finale a0-a9: suddivisione gerarchica delle aree (stato, regione, provincia, città, municipi...)
- denominazione finale jc: JunCtion, ossia i punti di intersezione tra strade
- denominazione finale nw: NetWorks, ossia il grafo stradale.

Per la piattaforma in questione vengono caricati come layer:

1. l'admin boundary a8, che ha l'estensione del comune, con nome <città>_base
 2. il layer NetWorks (nw), con nome <città>_strade
 3. il layer JunCtion (gc), con nome <città>_incroci
- Studio degli standard e delle piattaforme preesistenti per il trasporto delle informazioni cartografiche (server GIS, standard WFS, WMS, WCS, analisi dei

software MapServer e GeoServer). Per la soluzione finali si è adottato il sistema WMS per la generazione di immagini cartografiche. Per la gestione delle linee invece si è deciso di gestire gli inserimenti tramite operazioni codificate ad-hoc e non tramite il servizio WFS-T che, per l'applicazione in esame, si è rivelato poco flessibile.

- Ricerca ed analisi di tecnologie per la visualizzazione lato client dei dati cartografici (scelta tra visualizzazione tramite applet Java oppure tramite libreria Javascript OpenLayers). La scelta è caduta su OpenLayers per il pieno supporto allo standard WMS, e poiché non richiede plugin di visualizzazione ulteriori.
- Ricerca ed analisi di tecnologie per la memorizzazione e la gestione dei dati cartografici, e studio delle funzionalità offerte dall'estensioni spaziale PostGis.
- Pianificazione ad alto livello del sistema, adottando le seguenti tecnologie/piattaforme
 - Tecnologia server-side: Java 2 Enterprise Edition
 - Piattaforma di distribuzione dati cartografici: GeoServer tramite standard WMS
 - Memorizzazione dei dati cartografici: PostgreSQL con estensione PostGis per i dati spaziali
 - Visualizzazione lato client: libreria Javascript OpenLayers su pagine web HTML/CSS
- Progettazione e implementazione iterativa della piattaforma
 - Scheletro del sistema (modulo di configurazione, modulo di gestione e salvataggio dei dati, modulo per la visualizzazione dei dati sulle linee)
 - Funzionalità di autenticazione ed visualizzazione dell'elenco delle linee presenti nel sistema
 - Funzionalità di immissione di una nuova linea con relative fermate
 - Funzionalità di ricerca
 - Funzionalità di modifica di una linea preesistente
 - Implementazione dell'algoritmo di map matching tra linea disegnata e fermate inserite
 - Implementazione ed affinamento dell'algoritmo di adattamento automatico delle linee agli archi stradali
 - Implementazione del sistema di gestione della concorrenza tra modifiche ad una stessa rotta

In particolare, si motivano in seguito le scelte tecnologiche adottate.

Database Management System

La scelta è ricaduta sul DBMS PostgreSQL dal momento sia il componente DBMS che l'estensione per la gestione di dati spaziali PostGis risultano essere ampiamente documentata e testata. Altro aspetto di non secondaria importanza riguarda anche la natura del sistema

PostgreSQL in quanto software Open Source (con codice sorgente disponibile), quindi senza costi di licenza né per uso personale, né in ambito enterprise.

PostGIS è un'estensione spaziale per il Database Management System PostgreSQL distribuito con licenza GPL.

Fornisce i tipi di dati specificati negli standard dell'Open Geospatial Consortium, e quindi include il supporto per tipi geometrici come punti, linee, poligoni, e collezioni geometriche, mettendo a disposizione operatori spaziali di misura per il calcolo di aree, lunghezze, distanze e perimetri, ed utilizzando strutture dati per velocizzare ed ottimizzare query.

Server GIS

Geoserver è un server open source la cui funzione è quella di connettere e fornire informazioni georeferenziate utilizzando standard aperti, implementando le specifiche Web Feature Server (sia 1.0.0 che 1.1.0), Web Map Server (1.1.1) e Web Coverage Server (1.0).

Si è scelto di utilizzare il sistema GeoServer per la linearità di configurazione, l'ampio supporto e documentazione presente, la semplicità di integrazione e deployment in un application server J2EE, l'aderenza agli standard WMS e WFS. Così come per il DBMS, anche il progetto GeoServer è un progetto Open Source. Per questioni legate alle performance, si è deciso di inserire all'interno del sistema GeoServer anche l'estensione GeoWebCache (anch'essa progetto Open Source).

L'estensione GeoWebCache consente di utilizzare un meccanismo di cache lato server, che riduce il tempo di processamento necessario a produrre una immagine della porzione di mappa desiderata. Questo viene ottenuto frapponendo tra il client e il server WMS un servizio proxy trasparente: compito del servizio è quello di verificare l'esistenza dell'immagine da fornire in una cache, prima di inoltrare la richiesta al servizio WMS. Nel caso in cui l'immagine venga trovata in cache, la richiesta non viene inoltrata e al client viene fornita direttamente l'immagine già elaborata.

Web Application

Per uniformità con la scelta del server GIS, e per sfruttarne le caratteristiche di scalabilità, la scelta è caduta su Java 2 Enterprise Edition. Come application server è stato scelto Apache Tomcat.

Apache Tomcat (o semplicemente Tomcat) è un web container open source sviluppato dalla Apache Software Foundation. Implementa le specifiche JSP e Servlet di Sun Microsystems, fornendo quindi una piattaforma per l'esecuzione di applicazioni Web sviluppate nel linguaggio Java. La sua distribuzione standard include anche le funzionalità di web server tradizionale, che corrispondono al prodotto Apache.

Tomcat è rilasciato sotto licenza Apache Software License, ed è scritto interamente in Java; può quindi essere eseguito su qualsiasi architettura su cui sia installata un JVM (Java Virtual Machine).

Visualizzazione grafica lato client

Per l'interazione diretta con il servizio WMS del sistema GeoServer è stata scelta la libreria Javascript OpenLayers. Le ragioni della scelta riguardano soprattutto l'approccio Google Maps – style adottato dalla libreria, la quale tuttavia, così come per i componenti precedenti, è un progetto Open Source.

Componenti architetturali

Il sistema si configura come una vera e propria piattaforma a componenti, come si può notare dal modello in Fig. 147.

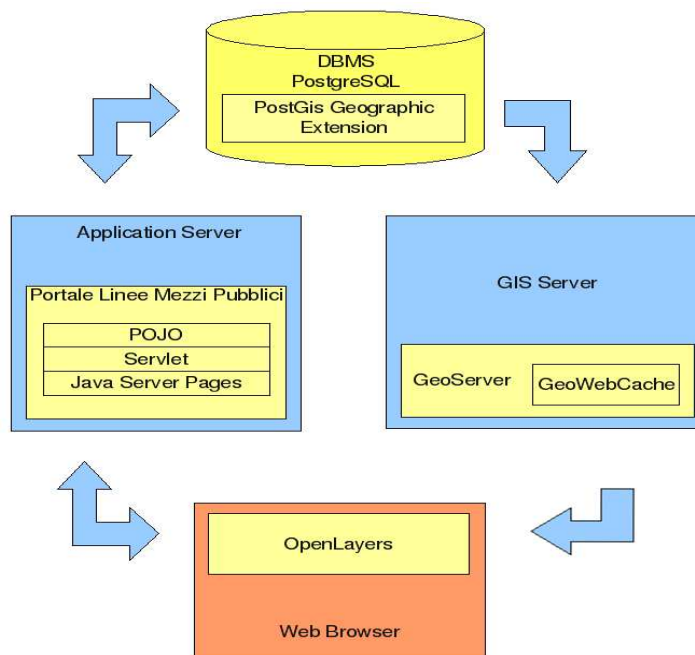


Fig. 147 - Componenti dell'architettura

Le frecce presenti in figura indicano le direzioni in cui si muovono le informazioni all'interno del sistema. Descritte più in dettaglio, le interazioni sono le seguenti:

- Il client (Web Browser) riceve le informazioni formattate come pagine HTML dal Portale, a cui invia a sua volta dati (aggiornamenti di linee, comandi di creazione di nuove linee) sia in modalità sincrona, sia in modalità asincrona tramite richieste di tipo XmlHttpRequest (tecnologia AJAX).
- Il client, tramite la libreria OpenLayers, è in grado di ricevere le informazioni cartografiche e di visualizzarle a schermo direttamente dal server GIS GeoServer.
- Il Portale Linee Mezzi Pubblici, oltre all'elaborazione delle informazioni proveniente dai client, è responsabile del loro salvataggio nella base dati, nonché della loro parziale estrazione in fase di modifica delle linee.
- Il server GIS GeoServer estrae i dati cartografici per offrirli alle richieste dei client tramite servizio WMS. Il sistema di caching dei riquadri (tile) della mappa stradale offerto dall'estensione GeoWebCache consente di non dover rielaborare le informazioni cartografiche per la produzione di immagini.

Progettazione del portale Linee Mezzi Pubblici

Il portale Linee Mezzi Pubblici è stato progettato e realizzato secondo il pattern architetturale MVC (Model View Controller).

Il pattern è basato sulla separazione dei compiti fra i componenti software che interpretano tre ruoli principali:

- il model contiene i dati e fornisce i metodi per accedervi;
- il view visualizza i dati contenuti nel model;
- il controller riceve i comandi dell'utente (in genere attraverso il view) e li attua modificando lo stato degli altri due componenti

Questo schema, fra l'altro, implica anche la tradizionale separazione fra la logica applicativa (in questo contesto spesso chiamata "logica di business"), a carico del model, e l'interfaccia utente, a carico del view e del controller.

L'interazione comune tra i tre componenti è esemplificata in Fig. 148.

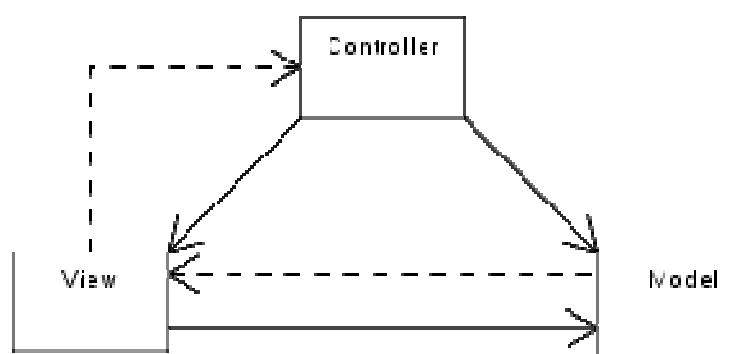


Fig. 148 - Componenti del design patter MVC e relazioni

Model

E' rappresentato dall'insieme dei moduli incaricati della gestione delle informazioni cartografiche e dei meccanismi di autorizzazione:

- **auth** contiene il bean utilizzato per il mapping delle informazioni di autenticazione dell'utente
- **bean** contiene i bean utilizzati per il mapping delle informazioni sulle rotte (quindi linee e fermate)
- **db** contiene le classi di accesso alla base dati, nonché gli oggetti DAO (Data Access Object) con le funzionalità basilari per il salvataggio ed il caricamento delle informazioni relative a linee e fermate
- **util** contiene classi di utilità, soprattutto le classi utilizzate per effettuare le operazioni di map matching tra le fermate e le vie a cui appartengono, e tra gli archi di una rotta ed il grafo stradale

- **config** contiene lo strumento per la configurazione del sistema

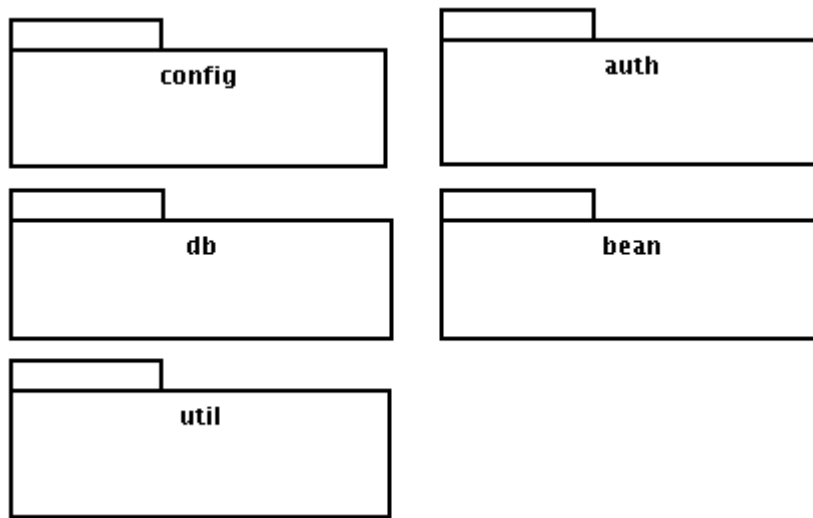


Fig. 149 - Package componenti il modello

Controller

E' rappresentato dall'insieme delle classi servlet, contenute nel package **servlet**. Compito principale delle servlet è quello di intercettare le richieste e gli eventi provenienti dalla vista e tradurle in opportune invocazioni di metodi sugli oggetti componenti il modello. Infine, sono responsabili del forwarding delle informazioni provenienti dal modello verso la vista.

Le servlet implementate si distinguono in servlet in modalità sincrona e servlet in modalità asincrona. Le prime rispondono a richieste effettuate dall'utente, e provocano l'aggiornamento della pagina. Le seconde, riconoscibili dal prefisso *AJAX*, lavorano su richieste asincrone, in genere provenienti da script in linguaggio Javascript presenti nelle pagine, e vengono utilizzate per scambiare informazioni tra la pagina ed il sistema senza causare l'aggiornamento (refresh) della pagina stessa.

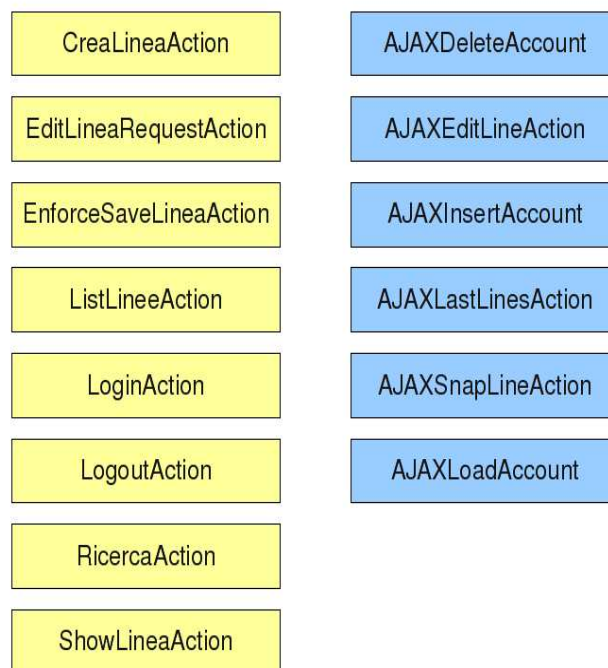


Fig. 150 - Servlet dello strato controller, sincrone (giallo) e AJAX oriented (azzurro)

View

E' l'insieme delle pagine HTML e JSP (Java Server Pages) incaricate di veicolare l'informazione all'utente. In molte di queste pagine è integrata la libreria Javascript OpenLayers (<http://openlayers.org/>) la quale consente di accedere a servizi cartografici compatibili con gli standard dell'OpenGIS Consortium.

La quasi totalità delle pagine JSP viene inoltrata all'utente da specifiche classi servlet, mentre alcune vengono utilizzate per fornire output ad invocazioni asincrone tramite tecnologia AJAX.

Vengono presentate le due possibili tipologie di navigazione della vista, a seconda del tipo di utente (utente semplice oppure operatore).

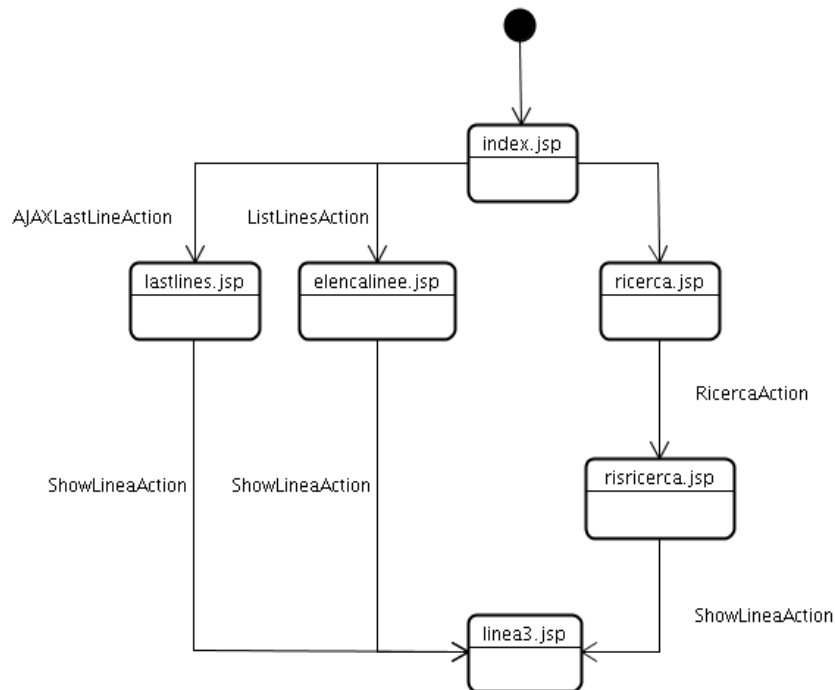


Fig. 151 - Modalità di navigazione per l'utente semplice

Nel caso in cui l'utente sia di tipo semplice, le operazioni messe a disposizione sono:

- la visualizzazione delle ultime 10 linee aggiornate, visibili sulla pagina index.jsp, e caricate tramite chiamata AJAX alla servlet AJAXLastLinesAction
- la visualizzazione dell'elenco di tutte le linee rese disponibili e contrassegnate come “pubbliche”
- la ricerca di una linea per
 - parola chiave presente nel nome
 - parola chiave presente nel nome oppure nella descrizione
 - parola chiave presente nell'indirizzo a cui è situata una fermata
- la visualizzazione della linea scelta

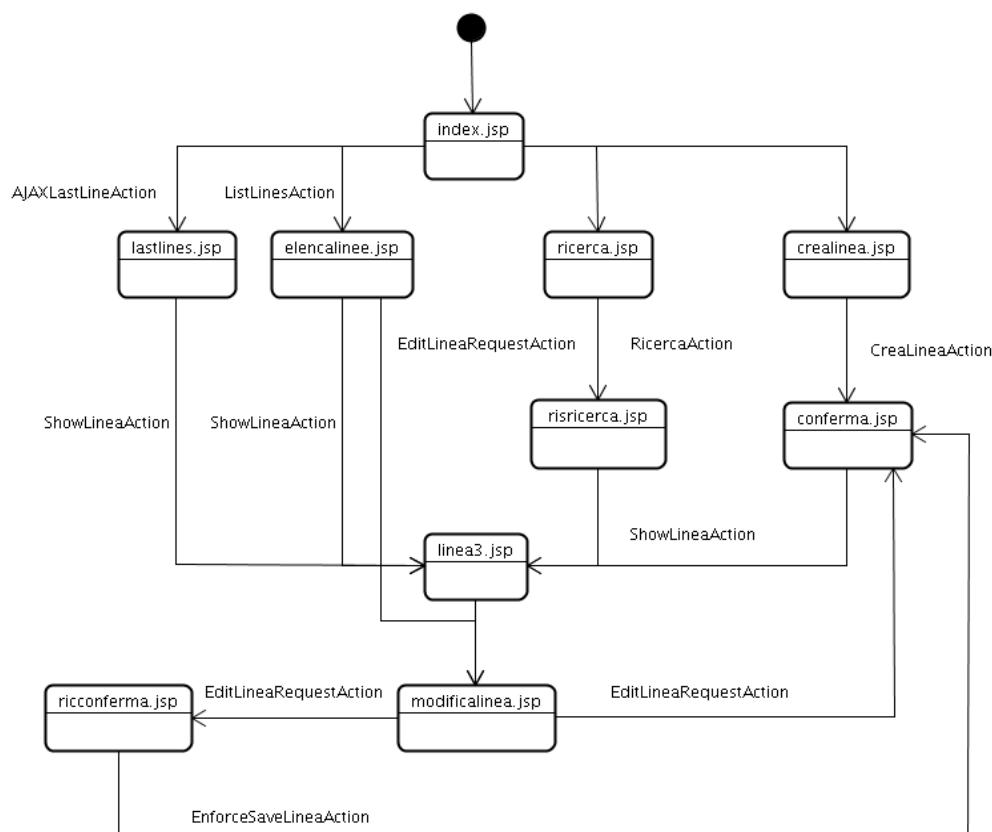


Fig. 152 - Modalità di navigazione per l'utente operatore

Nel caso in cui l'utente sia un operatore, il sistema mette a disposizione, oltre alle operazioni già presenti per l'utente normale, anche le seguenti operazioni:

1. la visualizzazione di tutte le linee presenti nel sistema, comprese le linee non ancora contrassegnate come pubbliche
2. la creazione di una nuova linea
3. la modifica di una linea preesistente

Inoltre, nel caso in cui l'utente sia un amministratore (riconosciuto da un apposito flag all'interno della base dati), il sistema mette a disposizione anche una pagina di gestione degli account tramite la quale l'amministratore potrà listare gli operatori registrati al sistema, cancellarli o aggiungerne di nuovi.

Problematiche principali e relative soluzioni

Nell'affrontare la progettazione del sistema, sono emersi essenzialmente due problemi, nella fase di analisi del meccanismo per la definizione di una nuova rotta e nella modifica di una rotta preesistente:

- i meccanismi per il matching di una rotta sul grafo stradale, sia per quanto riguarda gli archi componenti il percorso, sia per le fermate
- la gestione della concorrenza nella modifica della stessa rotta

Algoritmo di adattamento alle strade

Schema dell'algoritmo

Come si potrà verificare all'interno del capitolo seguente, l'inserimento di una rotta all'interno del sistema è un procedimento manuale, in cui l'utente “disegna” approssimativamente il percorso seguito dal mezzo pubblico, e salva tale percorso all'interno del sistema. Per evitare di dover essere molto precisi in fase di disegno, il sistema mette a disposizione uno strumento di correzione automatica che viene eseguito, nel caso in cui venga attivato da apposito radio button, ad ogni modifica del percorso e in fase di inserimento del percorso stesso.

Il sistema è attivato di default, ma è possibile affinarne il comportamento o disattivarlo tramite un semplice controllo, mostrato in Fig. 153.

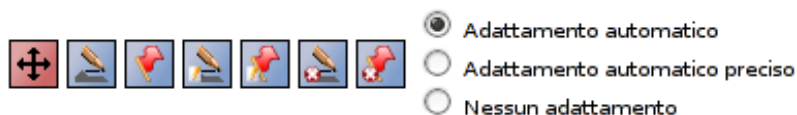


Fig. 153 - Strumenti di modifica e adattamento automatico

L'algoritmo prevede i seguenti passi:

1. Individuazione della sequenza degli incroci posti in prossimità del percorso disegnato dall'operatore
 - per ogni arco componente il percorso disegnato, si calcola un insieme di punti equidistanti tra loro
 - per ogni punto calcolato, si cerca un incrocio in un'area compresa all'interno di un raggio di lunghezza predeterminata
 - nel caso in cui venga individuato un incrocio, il punto su cui cade l'incrocio, viene aggiunto alla sequenza, eccetto nel caso in cui l'ultimo incrocio individuato sia uguale al nuovo incrocio
2. Elaborazione di una sequenza di punti $[0, N]$, composta nel modo seguente:
 - il punto 0, punto di partenza, allineato alla strada più vicina
 - i punti $[1, N-1]$ sono gli incroci individuati al passo 1
 - il punto N, punto di arrivo, allineato alla strada più vicina

3. Tentativo di estrazione degli archi stradali tra un punto ed il successivo, tenendo conto delle seguenti possibilità:
 1. entrambi i punti cadono su incroci, estremi di una stessa strada
 - Dal database viene estratta la sequenza di punti che compone la strada e i punti, eccetto l'ultimo, vengono aggiunti alla lista che compone il percorso
 2. entrambi i punti cadono su incroci, estremi di strade diverse
 - viene aggiunto il primo punto alla lista di punti componenti il percorso
 3. un punto cade su un incrocio, mentre l'altro cade su una strada avente come estremo l'incrocio stesso
 - Dal database viene estratto un sottoinsieme di punti componente la strada, che vengono aggiunti alla lista che compone il percorso, eccetto l'ultimo
 4. un punto cade su un incrocio, mentre l'altro cade su una strada che non ha come estremo l'incrocio stesso
 - viene aggiunto il primo punto alla lista di punti componenti il percorso
 5. entrambi i punti non cadono su un incrocio
 - viene aggiunto il primo punto alla lista di punti componenti il percorso

La differenza tra l'algoritmo di adattamento e l'algoritmo di adattamento preciso riguarda esclusivamente la misura del raggio entro cui viene ricercato un incrocio: nel caso dell'adattamento preciso, il raggio è lungo un decimo rispetto a quello utilizzato nell'adattamento standard. L'utilizzo dell'adattamento preciso è utile per gli ultimi ritocchi al percorso, per risolvere situazioni di ambiguità nelle quali il sistema associ erroneamente ad un punto un incrocio errato.

I risultati dell'algoritmo di adattamento sono mostrati in Fig. 154 e Fig. 155.

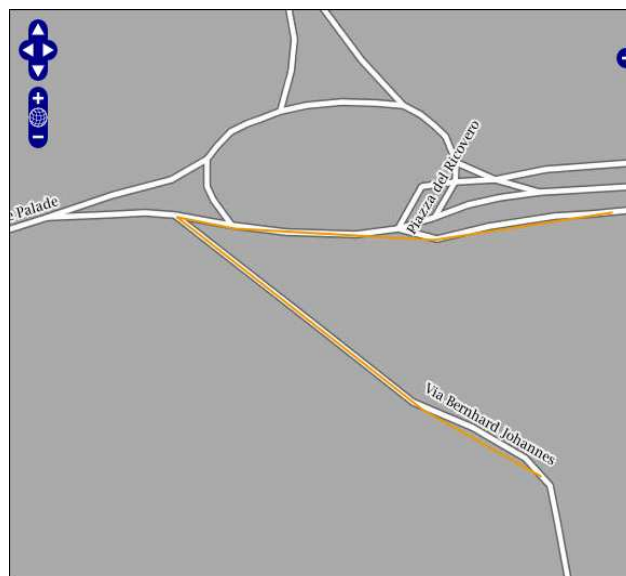


Fig. 154 - Percorso prima dell'adattamento

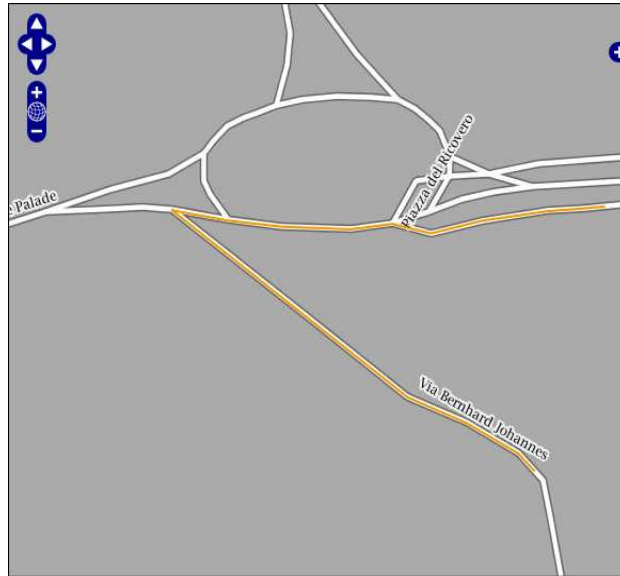


Fig. 155 - Percorso dopo l'adattamento

Un esempio

L'algoritmo viene ben esemplificato in Fig. 156: nella figura in alto a sinistra, l'operatore ha appena terminato di inserire un percorso. L'algoritmo di adattamento, dopo qualche istante, modifica il percorso inserito, portandolo ad assumere la configurazione nella figura in alto a destra. Si può notare come il percorso risulti perfettamente allineato alla strada, eccetto che nella parte bassa: questo comportamento si spiega con il fatto che l'algoritmo, nel processamento di quella sezione di percorso, non è riuscito a trovare un incrocio sufficientemente vicino. L'intervento manuale dell'operatore porta il percorso come nella figura in basso a sinistra. Dopo qualche altro istante, il sistema modifica il percorso nella sua conformazione finale, presente nella figura in basso a destra.

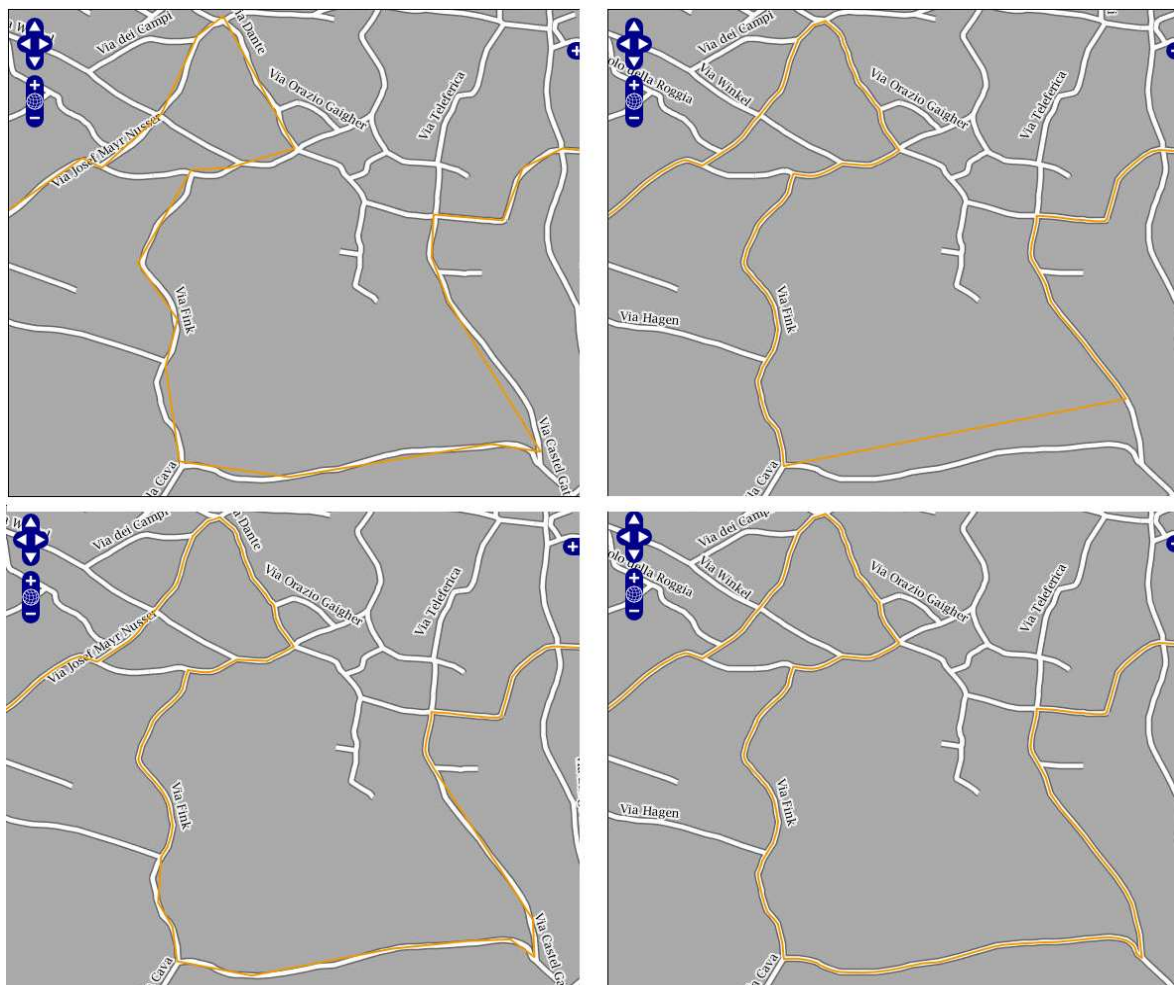


Fig. 156 - Algoritmo di adattamento con intervento manuale dell'operatore

Funzioni PostGis utilizzate

Nella stesura degli algoritmi, particolarmente utili sono risultate alcune funzioni messe a disposizione dall'estensione per dati spaziali PostGis al DBMS PostgreSQL. Tali funzioni sono:

- ST_line_interpolate_point(linestring,location)
 - Permette di ottenere la geometria di un punto, interpolato su una linea
- ST_line_substring(linestring,start,end)
 - Permette di estrarre una sottolinea a partire da una linea e da due punti su questa linea (i punti sono espressi come float tra 0 e 1)
- ST_line_locate_point(linestring,point)
 - Restituisce un float tra 0 e 1, rappresentante il punto interpolato sulla linea più vicino al punto passato come argomento
- ST_distance(geometry,geometry)
 - Restituisce la distanza tra due geometrie in unità di proiezione.

- GeometryFromText(string)
 - Converte una stringa in formato WKT in una entità di tipo *geometry* gestibile direttamente da PostGis

Algoritmo di adattamento delle fermate

L'algoritmo di adattamento delle fermate viene avviato ad ogni salvataggio di fermata sul database. Il procedimento è molto simile al passo 1 dell'algoritmo di adattamento alle strade.

- Per ogni arco stradale componente il percorso della rotta, si calcola un insieme di punti equidistanti, appartenenti all'arco stradale in questione
 - Per ciascun punto, si calcola la distanza con ciascuna fermata
 - Per ogni fermata, si tiene memoria del punto del percorso a distanza minima
- Al termine del procedimento, si allinea la fermata al punto del percorso a distanza minima

Al termine del procedimento, si arricchiscono le fermate con le informazioni riguardanti le vie su cui si trovano con un semplice algoritmo di map matching, ossia cercando la strada a distanza minima.

Un esempio dell'algoritmo può essere osservato in Fig. 157 e Fig. 158.

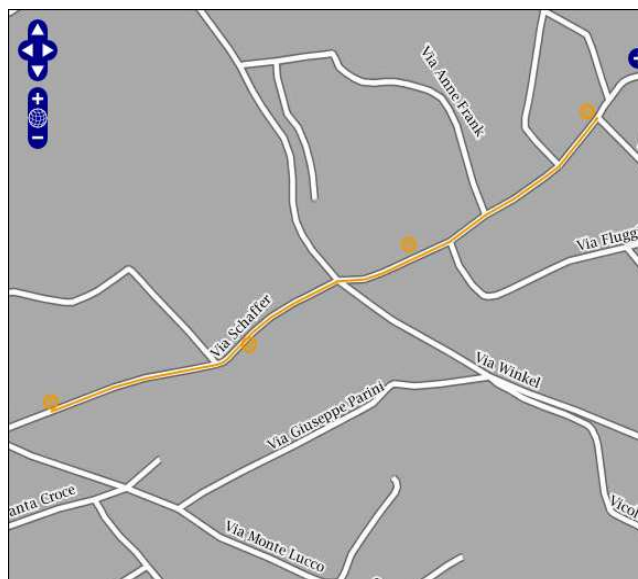


Fig. 157 - Fermate non allineate

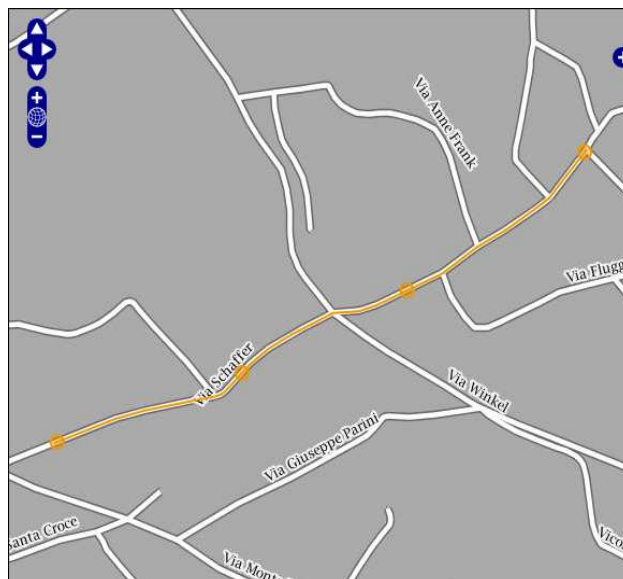


Fig. 158 - Fermate a seguito dell'allineamento

Gestione delle modifiche concorrenti

Durante la modifica di rotte pre-esistenti, può verificarsi l'eventualità di una modifica concorrente di uno stesso percorso da parte di due operatori differenti.

In genere, la gestione di un evento di questo genere può prevedere due soluzioni:

- la prima, più conservativa, impone il blocco della risorsa da parte di uno degli operatori, in modo da impedire agli altri operatori di modificarla fino a quando l'operatore che detiene i diritti di blocco non decide di sbloccare la risorsa occupata
- la seconda, più ottimistica, prevede la gestione di corse critiche avvisando opportunamente l'operatore in fase di salvataggio, se i dati inviati dall'operatore rischiano di sovrascrivere una modifica introdotta e salvata durante il lavoro dell'operatore stesso

In questo contesto, si è deciso di ricorrere alla seconda soluzione in quanto la prima soluzione, sebbene concettualmente più robusta in quanto serializza gli accessi alla risorsa, possa comportare lo scenario in cui il blocco di un percorso si protrae per un tempo molto superiore rispetto a quanto necessario (ad esempio, dimenticando di sbloccare la risorsa non appena terminato il lavoro).

Lo scenario concorrente viene quindi gestito con l'introduzione di un campo "lastUpdate" (timestamp) nella tabella del database contenente le rotte. Ad ogni modifica della rotta, il campo viene aggiornato con la data e l'ora dell'aggiornamento. Le informazioni temporali di aggiornamento, tuttavia, vengono anche trasferite all'operatore quando questo si accinge a modificare un percorso. Al termine della modifica, queste informazioni vengono trasferite nuovamente al sistema: se il timestamp trasferito dal client è identico al timestamp memorizzato sul database, la transazione avviene con successo, la linea modificata viene salvata, ed il timestamp ad essa associato viene aggiornato. In caso contrario, l'utente viene avvisato di un conflitto concorrente, a cui potrà rispondere forzando comunque il salvataggio dei dati, oppure scartando le sue modifiche, visualizzando la rotta aggiornata. La schermata di conferma viene mostrata in Fig. 159.

Nel caso in cui si scegliesse di salvare la rotta in modalità asincrona, verrà mostrato un messaggio di errore durante il salvataggio.



Fig. 159 - Avviso di modifica concorrente

Principali scenari di utilizzo

Vengono ora trattati i principali scenari di caso d'uso del Portale Gestione Linee Trasporto Pubblico, in cui viene descritto sia l'uso del portale da parte di un utente o di un operatore, sia i dettagli tecnici di come determinate operazioni vengano eseguite dal sistema.

Modifica dell'elenco degli operatori

Nel caso in cui il login sia effettuato da un utente avente anche lo status di amministratore, nella barra dei menu comparirà la voce “Gestione operatori”.

Selezionandola, verranno mostrati gli operatori attualmente registrati al sistema e una maschera che permette l'aggiunta di nuovi operatori. A fianco del nome di ciascun operatore è presente un tasto “Elimina” il quale, a seguito di conferma, provvede ad eliminare l'operatore dal sistema.

Visualizzazione di una rotta

All'ingresso nel portale, al visitatore viene mostrata la pagina in Fig. 160.



Fig. 160 - Home Page del Portale

Tramite una richiesta AJAX vengono caricate le ultime linee modificate, le quali sono direttamente accessibili sul lato destro della schermata.

In alternativa, l'utente può selezionare il collegamento “Visualizza Linee”, che lo porterà ad ottenere una schermata simile a quella in Fig. 161.

Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Ricerca

Username Password

Linee presenti nel sistema

Arianna	Prova	Visualizza linea
Prima prova	Descrizione della prima prova effettuata	Visualizza linea
Quarta Prova	Descrizione quarta prova	Visualizza linea
Quinta prova	Descrizione quinta prova	Visualizza linea
Seconda prova	Descrizione	Visualizza linea
Sesta prova	Descrizione numero sei	Visualizza linea
Settima prova	Descrizione settima prova	Visualizza linea
Terza Prova	Descrizione terza prova	Visualizza linea

Copyright IEIIT-CNR Sede di Bologna

Fig. 161 - Elenco delle linee pubbliche presenti nel sistema

L'utente può quindi selezionare la linea a cui è interessato tramite il link “Visualizza linea”. Il sistema provvederà al caricamento dei dati necessari e presenterà all'utente una pagina simile a quella in Fig. 162.

L'utente può visualizzare l'elenco delle fermate eseguite dal mezzo pubblico, comprensive di indirizzo su cui sono localizzate. Selezionando una fermata con un click del tasto sinistro del mouse, la fermata viene selezionata. La mappa provvede a centrarsi sulla fermata selezionata.

In ogni momento l'utente potrà centrare la mappa, selezionando il tasto “Centra la mappa”.

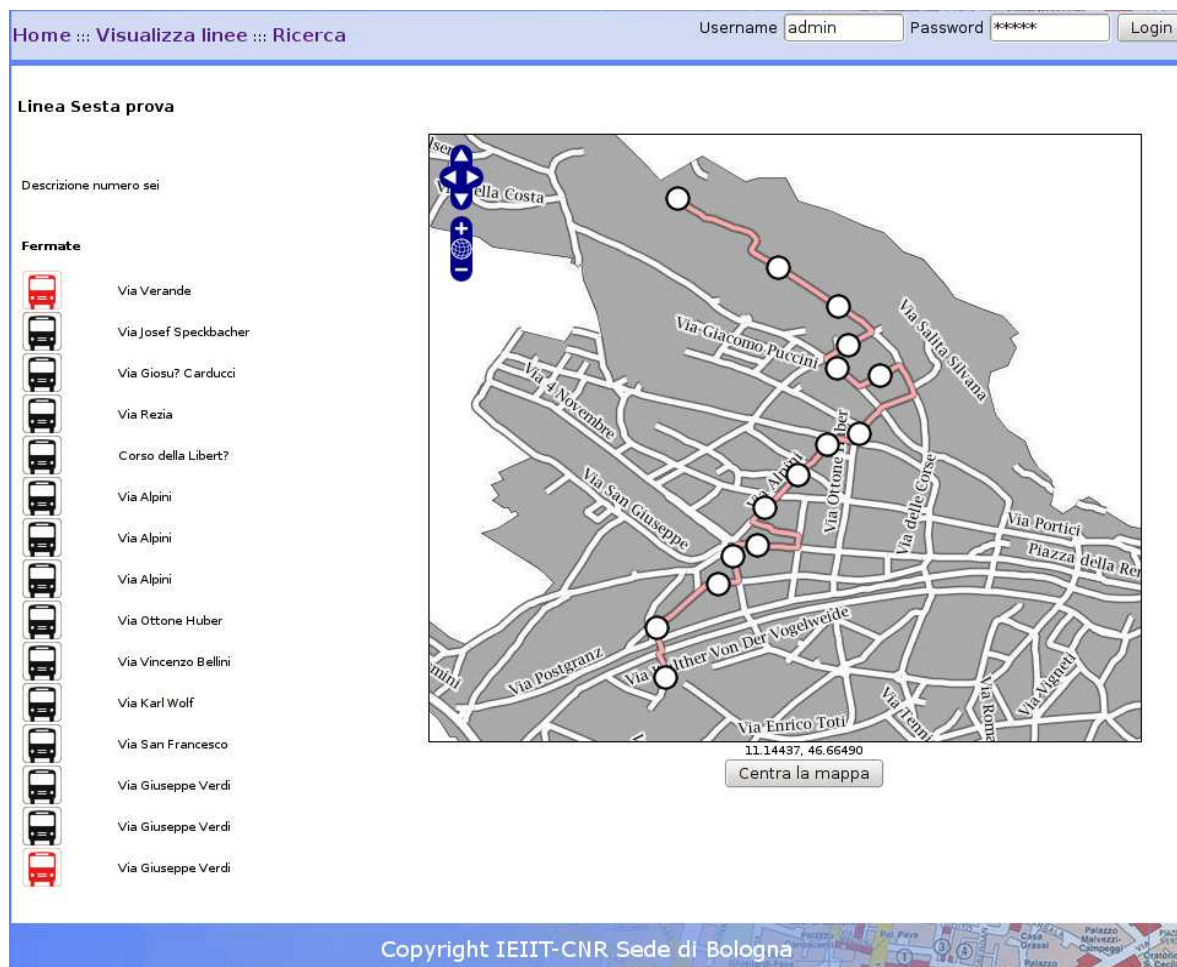


Fig. 162 - Visualizzazione di una rotta

Inserimento di una nuova rotta

A partire da qualunque punto all'interno del sito, un utente può effettuare il login sul sistema, inserendo nome utente e password di accesso nei campi posti in alto a destra. Nel caso in cui le credenziali di accesso fossero sbagliate, il sistema visualizzerà una pagina di errore. In caso di accesso effettuato con successo, il nome utente dell'operatore comparirà al posto dei campi utilizzati per l'accesso, e si verrà automaticamente redirezionati sull'home page. Sulla barra di menu comparirà il link “Crea linea”.

Selezionandolo, l'operatore verrà rediretto alla pagina riportata in Fig. 163 .

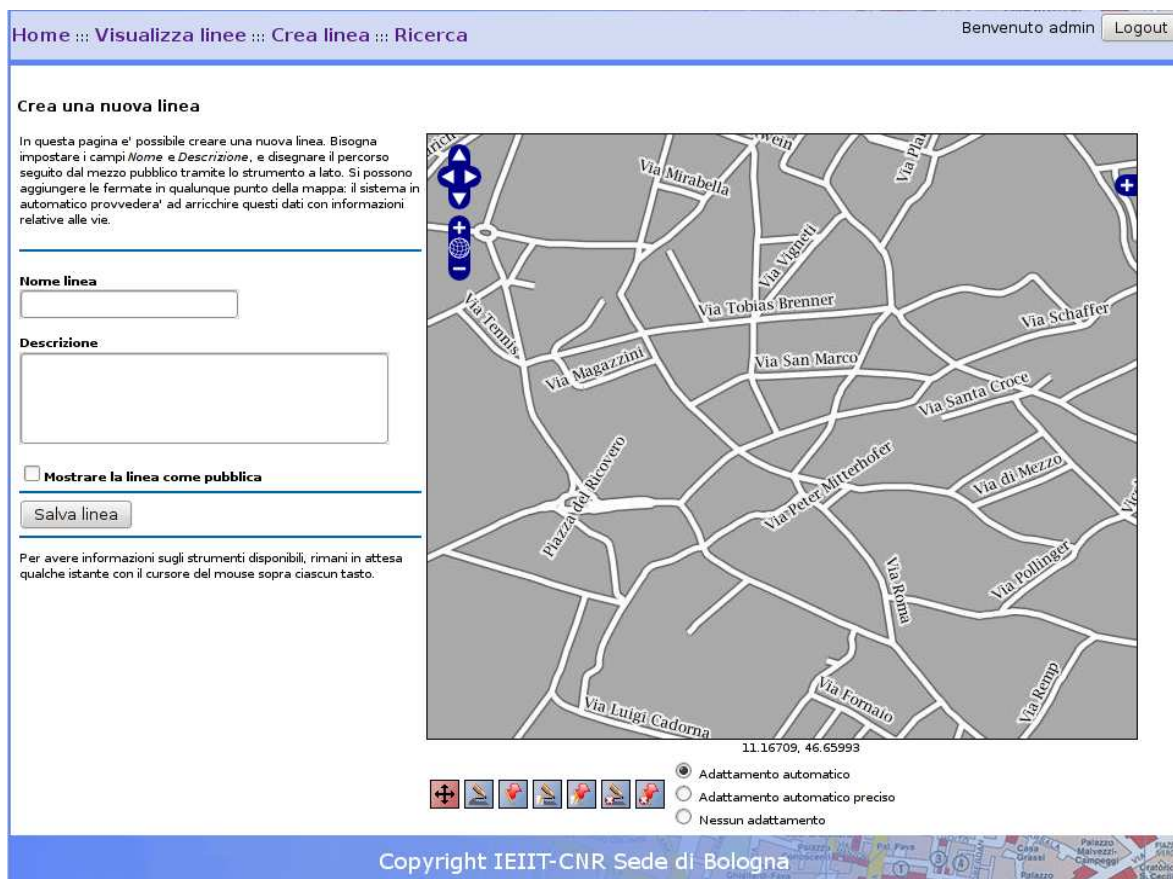


Fig. 163 - Creazione di una nuova linea

L'operatore dovrà inserire il nome della nuova linea e una breve descrizione. Successivamente provvederà a disegnare la rotta seguita dal mezzo pubblico e ad aggiungere le fermate tramite lo strumento di editing riportato in Fig. 164.

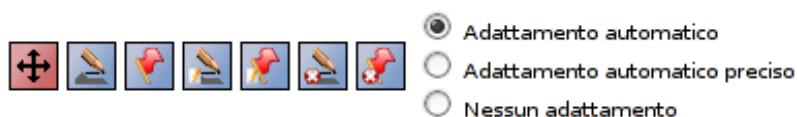


Fig. 164 - Strumento di creazione linee e fermate

Da sinistra verso destra troviamo gli strumenti per:

Navigare la mappa

Tenendo premuto con il tasto sinistro del mouse è possibile trascinare la mappa e caricarne nuove porzioni. La navigazione è anche possibile utilizzando i tasti freccia della tastiera. Tale modalità risulta decisamente utile in fase di inserimento di percorsi, quando si ha necessità di scorrere la mappa.

Disegnare una nuova linea

Si selezioni lo strumento e si faccia click con il tasto sinistro sulla mappa. Spostando il mouse a seguito del click si noterà una linea partire dal punto selezionato e terminare sul cursore del mouse. Selezionare il punto successivo con il click del mouse per creare un arco.

Continuare la creazione del percorso, spostandosi con i tasti freccia per far scorrere la mappa. Terminare la creazione del percorso selezionando l'ultimo punto da creare con un doppio click del mouse. Se la funzione di adattamento automatico è stata attivata, si noterà dopo qualche istante l'adattamento del percorso al grafo stradale, sempre tenendo conto dei limiti riportati nella sezione Algoritmo di adattamento alle strade (Pag. 228).

Aggiungere le fermate

Si seleziona lo strumento e si fa click con il tasto sinistro sulla mappa sul punto in cui aggiungere una fermata. Non è necessario essere particolarmente precisi in quanto il sistema correggerà la posizione immessa, agganciandola al percorso del mezzo, in fase di salvataggio della linea.

Modificare la linea

Si seleziona lo strumento e si fa click con il tasto sinistro del mouse sopra la linea inserita. La linea si colorerà di blu, e verranno evidenziati i punti di controllo (arancione acceso). Verranno inoltre evidenziati i punti mediani di ciascun arco: trascinando uno di questi punti, l'arco verrà suddiviso in due, ed il punto mediano diventerà un nuovo punto di controllo. Al termine della modifica, fare click in un qualsiasi punto della mappa. Se la funzione di adattamento automatico è stata attivata, si noterà dopo qualche istante l'adattamento del percorso al grafo stradale, sempre tenendo conto dei limiti riportati nella sezione Algoritmo di adattamento alle strade (Pag. 228).

Modificare una fermata

Si seleziona lo strumento e si fa click con il tasto sinistro del mouse sopra una fermata. La fermata si colorerà di blu. È possibile quindi trascinare la fermata, tenendo premuto il pulsante sinistro del mouse e spostando la fermata nella nuova posizione. Al termine della modifica, fare click in un qualsiasi punto della mappa.

Cancellare la linea

Si seleziona la linea da eliminare tramite lo strumento di modifica delle linee. Si preme il tasto di cancellazione linea. Il sistema provvederà ad avvisare l'operatore circa l'irreversibilità dell'operazione. In caso di risposta affermativa al messaggio di conferma, la linea verrà eliminata.

Cancellare una fermata

Si seleziona la fermata da eliminare tramite lo strumento di modifica delle fermate. Si preme il tasto di cancellazione linea. Il sistema provvederà ad avvisare l'operatore circa l'irreversibilità dell'operazione. In caso di risposta affermativa al messaggio di conferma, la fermata verrà eliminata.

Attivare o disattivare lo strumento di adattamento automatico

Per una descrizione dell'algoritmo di adattamento automatico, si rimanda alla sezione Algoritmo di adattamento alle strade (Pag. 228). Si raccomanda di tenere attivato inizialmente l'adattamento automatico. Per rifinire il percorso, si consiglia di utilizzare l'adattamento automatico preciso. Nel caso in cui il sistema effettui operazioni errate di adattamento, è possibile disabilitare l'algoritmo selezionando la voce "Nessun adattamento".

Al termine dell'inserimento, è possibile salvare la rotta selezionando il tasto "Salva linea". Il sistema provvederà a compiere dei controlli preliminari. In caso di esito positivo, la linea verrà salvata nel sistema. Se si seleziona il checkbox "Mostrare la linea come pubblica", il

sistema renderà la linea visibile ai visitatori. In caso contrario, la linea sarà accessibile solo agli operatori.

Modifica di una rotta preesistente

Una volta autenticato, l'operatore può selezionare la voce sulla barra dei menu “Visualizza linee”. Verrà mostrata una pagina simile a quella riportata in Fig. 165.

Le linee mostrate su sfondo rosa sono quelle linee non ancora rese pubbliche. In più, rispetto alla Fig. 161, è presente anche un link “Modifica linea” tramite cui si può accedere alla pagina di modifica, come mostrato in Fig. 166.



Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Crea linea :: Ricerca

Benvenuto admin Logout

Linee presenti nel sistema

Linea	Descrizione	Visualizza linea	Linea pubblica	Modifica linea
Arianna	Prova	Visualizza linea	Linea pubblica: true	Modifica linea
Ottava linea	Descrizione ottava linea	Visualizza linea	Linea pubblica: false	Modifica linea
Prima prova	Descrizione della prima prova effettuata	Visualizza linea	Linea pubblica: true	Modifica linea
Prova	Prova	Visualizza linea	Linea pubblica: false	Modifica linea
Quarta Prova	Descrizione quarta prova	Visualizza linea	Linea pubblica: true	Modifica linea
Quinta prova	Descrizione quinta prova	Visualizza linea	Linea pubblica: true	Modifica linea
Seconda prova	Descrizione	Visualizza linea	Linea pubblica: true	Modifica linea
Sesta prova	Descrizione numero sei	Visualizza linea	Linea pubblica: true	Modifica linea
Settima prova	Descrizione settima prova	Visualizza linea	Linea pubblica: true	Modifica linea
Terza Prova	Descrizione terza prova	Visualizza linea	Linea pubblica: true	Modifica linea

Copyright IEIT-CNR Sede di Bologna

Fig. 165 - Elenco delle linee complessive presenti nel sistema

La linea e le fermate vengono caricate in OpenLayers su un layer vettoriale utilizzando la sintassi WKT (Well Known Text). La pagina di modifica è molto simile alla pagina di creazione di una linea. Rispetto a questa, si nota il tasto “Termina inserimento”. In questo caso, infatti, il tasto “Salva linea” provoca una invocazione asincrona tramite AJAX, per salvare la linea senza dover effettuare il refresh della pagina. Nel caso in cui l'operazione di salvataggio vada a buon fine, verrà mostrato un messaggio riportante la data e l'ora dell'ultimo salvataggio. In caso di modifica concorrente, verrà riportato un messaggio di errore. Selezionando il tasto “Termina inserimento” si potrà gestire l'errore di concorrenza, salvando comunque il proprio lavoro oppure annullando le proprie modifiche.

Per gli approfondimenti riguardo la gestione della concorrenza si rimanda alla sezione Gestione delle modifiche concorrenti (pag. 233)

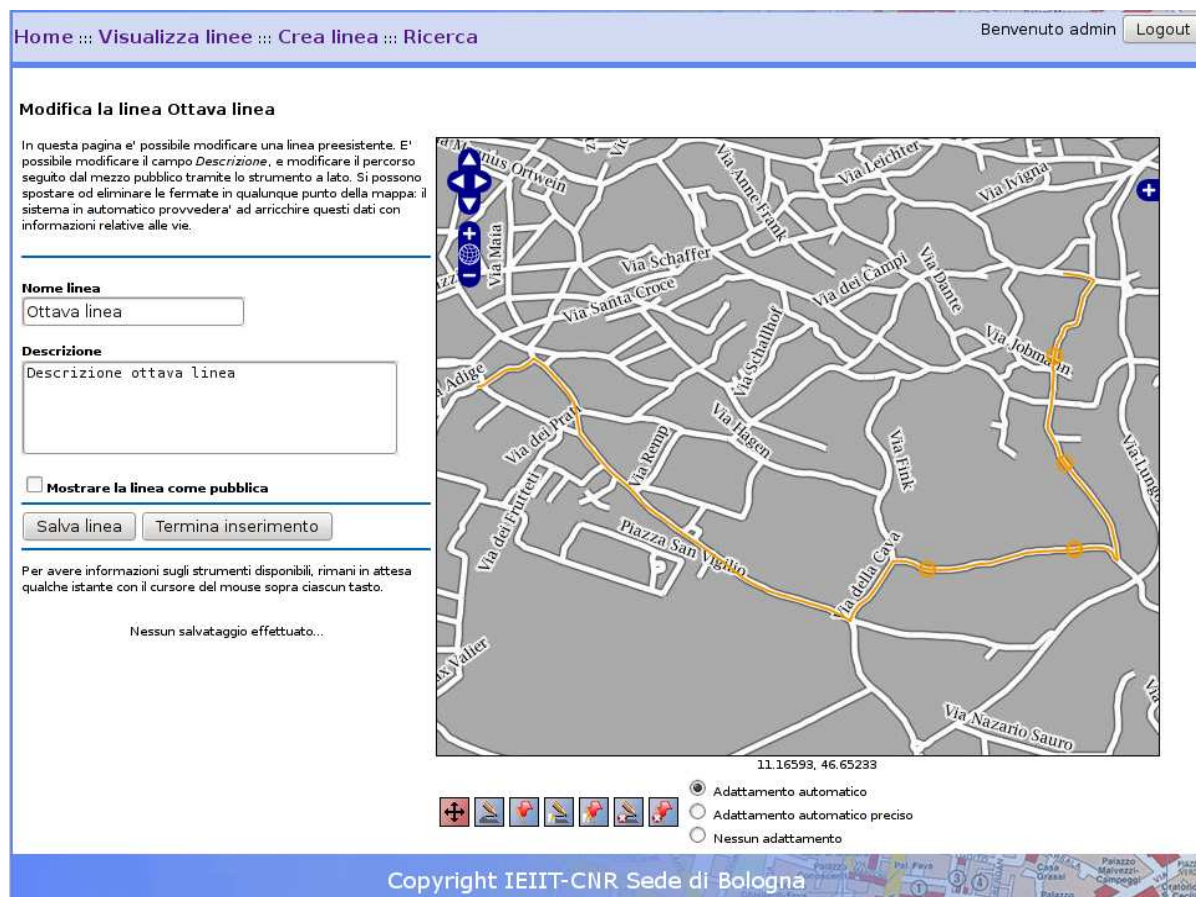


Fig. 166 - Modifica di una linea esistente

Ricerca di una rotta

Per ricercare una rotta, l'utente può selezionare la voce "Ricerca" dalla barra di menu. Il sistema proporrà una schermata simile a quella riportata in Fig. 167.

Tramite la maschera proposta, l'utente può cercare linee:

- tramite una parola che deve essere contenuta nel nome della rotta
- tramite una parola che deve essere contenuta nel nome oppure nella descrizione della rotta
- tramite una parola che deve essere contenuta nell'indirizzo di una fermata della rotta

La ricerca è case sensitive. In seguito all'interrogazione, il sistema inoltra l'elenco delle linee trovate (un esempio è riportato in Fig. 168, in cui è presente un'unica rotta). Nel caso in cui l'utente che avesse sottoposto l'interrogazione fosse un operatore, verrebbero riportate anche le informazioni relativamente alla pubblicazione della linea e alla data di ultima modifica.

Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Ricerca

Username Password

Ricerca linee

Parola chiave

Cerca:

☒ tra le vie della città'

☐ nei nomi delle linee

☐ nei nomi e nelle descrizioni delle linee

Copyright IEIIT-CNR Sede di Bologna

Fig. 167 - Maschera per la ricerca di linee

Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Ricerca

Username Password

Risultato della ricerca

Quinta prova

Descrizione quinta prova

Ultimo aggiornamento: 2008-11-23 17:41:52.979

Copyright IEIIT-CNR Sede di Bologna

Fig. 168 - Risultato della ricerca

MANUALE D'USO PORTALE DISEGNO PERCORSI

Descrizione e pianificazione del sistema

Il sistema sviluppato, denominato “Linee Mezzi Pubblici”, è un sistema distribuito che consente l'inserimento e la modifica di rotte urbane, nonché di ricercare e visualizzare le rotte inserite nel sistema. Supporta modifiche concorrenti di rotte da parte di più operatori, gestendo i casi in cui avvengano modifiche concorrenti su una stessa rotta da parte di due o più operatori. Implementa inoltre una serie di algoritmi per rendere rapido ed accurato l'inserimento e la modifica delle rotte e delle fermate associate a una specifica linea.

Requisiti e vincoli operativi

I requisiti utente (semplice oppure operatore) identificati per il progetto sono elencati in seguito:

1. possibilità, per un utente non registrato, di visualizzare una rotta di una linea inserita nel sistema, caratterizzata da un nome ed una descrizione, da una sequenza di fermate e da una sequenza di archi stradali. Viene richiesta una visualizzazione grafica della linea.
2. possibilità, per un utente non registrato, di visualizzare l'elenco delle linee inserite nel sistema. In particolare, devono poter essere accessibili tutte le linee definite come pubbliche.
3. possibilità, per un utente non registrato, di poter effettuare delle ricerche sulle linee, utilizzando i seguenti criteri
 - elenco di tutte le linee aventi una parola chiave nel nome
 - elenco di tutte le linee aventi una parola chiave nel nome oppure nella descrizione
 - elenco di tutte le linee la cui tratta transiti per un determinato indirizzo
4. possibilità, per un utente operatore, di inserire nel sistema una nuova linea disegnandola sulla mappa ed aggiungendo le fermate
5. possibilità, per un utente operatore, di modificare una linea, modificando il disegno sulla mappa ed aggiungendo/spostando/eliminando le fermate
6. possibilità, per un utente operatore, di elencare tutte le linee presenti nel sistema, e di poter discriminare tra linee pubbliche e linee non ancora pubblicate
7. possibilità, per un utente operatore, di poter salvare il proprio lavoro su una linea e continuarla in seguito

I requisiti di sistema ed i vincoli operativi sono definiti in seguito:

- **Piattaforma Web:** si richiede che il sistema in discussione sia fruibile tramite browser Web.
- **Produzione output grafico:** si richiede che il sistema produca un output grafico della rotta, comprensivo della localizzazione delle fermate.

- **Capacità di visualizzazione di layer cartografici da basi di dati:** si richiede che il sistema recuperi le informazioni cartografiche da una base dati.
- **Tracciamento grafico della linea:** si richiede che il sistema offra una modalità grafica per l'inserimento e la modifica delle linee, ed offra strumenti efficaci per il tracciamento delle stesse.
- **Definizione di fermate:** si richiede che il sistema offra una modalità grafica per l'inserimento, la modifica e la rimozione di fermate da una rotta.
- **Salvataggio delle linee:** si richiede che il sistema salvi nella base di dati centrale le rotte definite.
- **Salvataggio delle linee come archi stradali tra fermate:** si richiede che il sistema salvi nella base di dati centrale le rotte di ciascuna linea anche come archi stradali di collegamento tra una fermata e la successiva

Componenti architetturali

Il sistema si configura come una vera e propria piattaforma a componenti, come si può notare dal modello in Fig. 169.

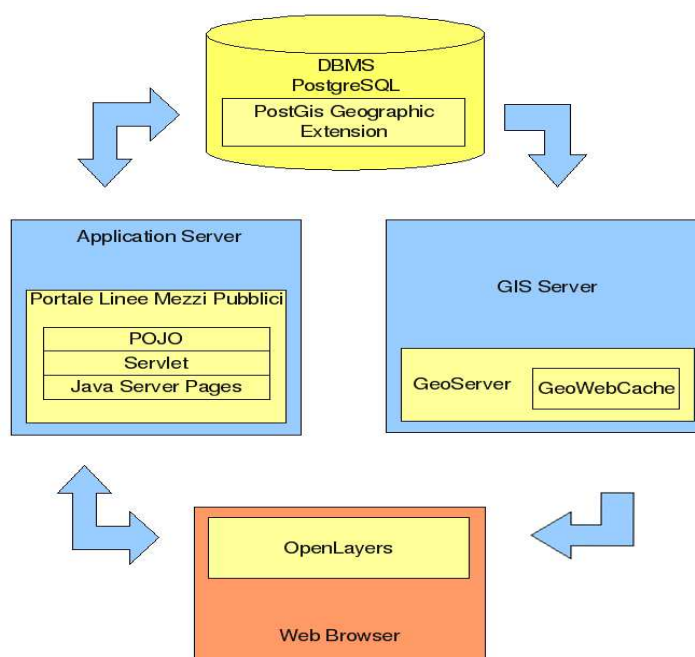


Fig. 169 - Componenti dell'architettura

Le frecce presenti in figura indicano le direzioni in cui si muovono le informazioni all'interno del sistema. Descritte più in dettaglio, le interazioni sono le seguenti:

Il client (Web Browser) riceve le informazioni formattate come pagine HTML dal Portale, a cui invia a sua volta dati (aggiornamenti di linee, comandi di creazione di nuove linee) sia in modalità sincrona, sia in modalità asincrona tramite richieste di tipo XMLHttpRequest (tecnologia AJAX).

Il client, tramite la libreria OpenLayers, è in grado di ricevere le informazioni cartografica e

di visualizzarle a schermo direttamente dal server GIS GeoServer.

Il Portale Linee Mezzi Pubblici, oltre all'elaborazione delle informazioni provenienti dai client, è responsabile del loro salvataggio nella base dati, nonché della loro parziale estrazione in fase di modifica delle linee.

Il server GIS GeoServer estrae i dati cartografici per offrirli alle richieste dei client tramite servizio WMS. Il sistema di caching dei riquadri (tile) della mappa stradale offerto dall'estensione GeoWebCache consente di non dover rielaborare le informazioni cartografiche per la produzione di immagini.

Installazione e configurazione dei componenti

Apache Tomcat

Apache Tomcat (o semplicemente Tomcat) è un web container open source sviluppato dalla Apache Software Foundation. Implementa le specifiche JSP e Servlet di Sun Microsystems, fornendo quindi una piattaforma per l'esecuzione di applicazioni Web sviluppate nel linguaggio Java. La sua distribuzione standard include anche le funzionalità di web server tradizionale, che corrispondono al prodotto Apache.

Tomcat è rilasciato sotto licenza Apache Software License, ed è scritto interamente in Java; può quindi essere eseguito su qualsiasi architettura su cui sia installata un JVM (Java Virtual Machine).

Al momento della pubblicazione di questo documento, la versione disponibile (quella usata dalla piattaforma) è la 6.0.18, e può essere scaricata all'URL

<http://tomcat.apache.org/download-60.cgi>

A seguito dello scaricamento dell'archivio e della sua decompressione, verrà creata una cartella nominata `apache-tomcat-6.0.18`. Affinchè Tomcat possa avviarsi correttamente, è necessario impostare la variabile d'ambiente `JAVA_HOME` affinché contenga il percorso alla cartella di installazione del Java Runtime Environment (JRE). Una volta impostata la variabile d'ambiente, è possibile lanciare l'application server con il comando `bin/startup.[sh|bat]` e fermarlo con il comando `bin/shutdown.[sh|bat]`.

Se si vuole visualizzare a schermo le informazioni di startup e di esecuzione dell'application server, è possibile lanciare `bin/catalina.[sh|bat] run`. Un esempio di output è mostrato in seguito.

```
Using CATALINA_BASE:   /home/dario/Programmi/apache-tomcat-6.0.18
Using CATALINA_HOME:   /home/dario/Programmi/apache-tomcat-6.0.18
Using CATALINA_TMPDIR: /home/dario/Programmi/apache-tomcat-6.0.18/temp
Using JRE_HOME:        /opt/java
```

```

22-nov-2008 8.18.57 org.apache.catalina.core.AprLifecycleListener
init
INFO: The APR based Apache Tomcat Native library which allows
optimal performance in production environments was not found on the
java.library.path:
/opt/java/jre/lib/i386/client:/opt/java/jre/lib/i386:/opt/java/jre/
../lib/i386:/usr/java/packages/lib/i386:/lib:/usr/lib
22-nov-2008 8.18.57 org.apache.coyote.http11.Http11Protocol init
INFO: Initializing Coyote HTTP/1.1 on http-8080
22-nov-2008 8.18.57 org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 465 ms
22-nov-2008 8.18.57 org.apache.catalina.core.StandardService start
INFO: Starting service Catalina
22-nov-2008 8.18.57 org.apache.catalina.core.StandardEngine start
INFO: Starting Servlet Engine: Apache Tomcat/6.0.18
22-nov-2008 8.18.58 org.apache.coyote.http11.Http11Protocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
22-nov-2008 8.18.58 org.apache.jk.common.ChannelSocket init
INFO: JK: ajp13 listening on /0.0.0.0:8009
22-nov-2008 8.18.58 org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/25 config=null
22-nov-2008 8.18.58 org.apache.catalina.startup.Catalina start
INFO: Server startup in 731 ms

```

Per configurare i parametri di autenticazione per l'amministratore dell'application server bisogna impostare il file `conf/tomcat-users.xml`, fornendo le credenziali di accesso desiderate per il ruolo di admin e manager. Un esempio di file di autorizzazione è mostrato in seguito

```

<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="manager"/>
  <role rolename="admin"/>
  <user username="admin" password="admin" roles="admin,manager"/>
</tomcat-users>

```

L'impostazione di un tale utente si rende necessaria per poter effettuare in seguito il deploy delle applicazioni direttamente da interfaccia utente.

Postgis

PostGIS è un'estensione spaziale per il Database Management System PostgreSQL distribuito con licenza GPL.

Fornisce i tipi di dati specificati negli standard dell'Open Geospatial Consortium, e quindi include il supporto per tipi geometrici come punti, linee, poligoni, e collezioni geometriche, mettendo a disposizione operatori spaziali di misura per il calcolo di aree, lunghezze, distanze e perimetri, ed utilizzando strutture dati per velocizzare ed ottimizzare query.

L'ultima versione disponibile all'atto della scrittura di questo documento è la 1.3.3, e può essere scaricata dall'URL <http://www.postgis.org/download/>

Prerequisito per l'installazione consiste nell'avvenuta installazione e configurazione del DBMS PostgreSQL, il cui procedimento va oltre gli scopi di questo testo, e per cui si rimanda a <http://www.postgresql.org/>

L'installazione varia da piattaforma a piattaforma: per piattaforma Windows è disponibile un installer grafico, che si occuperà di installare e configurare l'estensione. Per quanto riguarda i sistemi Linux, PostGis è fornito precompilato sulla maggior parte delle distribuzioni (Ubuntu, Mandriva, Red Hat, Fedora, Arch Linux, Debian, Slackware). Si consiglia pertanto di verificare i meccanismi di installazione del pacchetto della propria distribuzione. In alternativa è sempre possibile scaricare i sorgenti e compilarli usando la procedura standard con automake (`./configure && make && make install`).

Per creare il database per la piattaforma Web, effettuare i seguenti passi:

Creare un database con template template0, nome qualsiasi (ad esempio, lineedb), codifica SQL_ASCII

Impostare il linguaggio usando il comando

```
createlang plpgsql lineedb
```

Importare le funzioni spaziali nel database con il comando

```
psql -d lineedb -f lwpostgis.sql
```

(il file è presente nel pacchetto di distribuzione PostGis)

Importare i sistemi di riferimento cartografici nel database con il comando

```
psql -d lineedb -f spatial_ref_sys.sql
```

(il file è presente nel pacchetto di distribuzione PostGis)

Al termine della procedura, nel database lineedb saranno presenti due tabelle (geometry_columns e spatial_ref_sys) e 663 funzioni spaziali.

La procedura sopra menzionata non è necessaria su piattaforma Windows, in quanto l'installer di PostGis si preoccupa di aggiungere un template nominato postgis_template. E' pertanto sufficiente, all'atto della creazione del database, specificare tale template come template di creazione al posto di template0.

Caricamento Mappe TeleAtlas

I dati cartografici utilizzati per il progetto sono stati forniti da TeleAtlas, una società che produce mappe digitali per sistemi di informazione geografica (GIS), servizi di localizzazione (LBS), e sistemi di navigazione automotiva.

Descrizione delle mappe utilizzate

Le mappe fornite si dividono in vari gruppi, di cui i più importanti e significativi per il progetto in questione, sono:

- denominazione finale 00-09: suddivisione gerarchica delle strade (autostrade, strade extraurbane principali, strade extraurbane secondarie, strade urbane, strade locali...)
- denominazione finale a0-a9: suddivisione gerarchica delle aree (stato, regione, provincia, città, municipi...)
- denominazione finale jc: JunCtion, ossia i punti di intersezione tra strade
- denominazione finale nw: NetWorks, ossia il grafo stradale.

Per la piattaforma in questione vengono caricati come layer:

- l'admin boundary a8, che ha l'estensione del comune, con nome <città>_base
- il layer NetWorks (nw), con nome <città>_strade
- il layer JunCtion (gc), con nome <città>_incroci

Conversione e caricamento

Per caricare un layer sul database lineedb si procede nel modo seguente (nell'esempio si importano le mappe della città di Merano, pertanto si è deciso di chiamare le tabelle con il prefisso *merano*):

- si eseguono i seguenti comandi di conversione (l'applicazione shp2pgsql è fornita nella distribuzione PostGis)

```
shp2pgsql ita021051_____jc.shp merano_incroci >
incroci.sql
```

```
shp2pgsql ita021051_____nw.shp merano_strade >
strade.sql
```

```
shp2pgsql ita021051_____a8.shp merano_base > base.sql
```

- si carica ciascuno dei file sql prodotti sul database lineedb (si suppone che il proprietario del database sia l'utente *postgres*)

```
psql -U postgres -d lineedb -f base.sql
```

```
psql -U postgres -d lineedb -f incroci.sql
```

```
psql -U postgres -d lineedb -f strade.sql
```

Si verifichi quindi la presenza di tre nuove tabelle nel database lineedb, e la creazione di tre sequence, una per ciascuna tabella.

Per il funzionamento corretto dell'applicazione Web, è **necessario** chiamare le tabelle con i nomi sopra menzionati, ossia con un nome generico seguito da *_base* per l'admin boundary layer, *_strade* per il layer NetWorks, *_incroci* per il layer JunCtion.

Nel caso in cui avvenga un errore in fase di importazione causato da errata codifica dei caratteri, bisogna anteporre il comando

```
SET CLIENT_ENCODING TO <encoding>;
```

all'inizio di ciascun file SQL. Nel caso di piattaforma Windows, si provi con <encoding> = Latin1.

Infine, bisogna caricare le tabelle per il contenimento delle informazioni sulle linee e sulle fermate, una tabella per il contenimento delle tratte di ogni singola linea, tra una fermata e la successiva, nonché una tabella di utenti per tracciare e registrare gli utenti autorizzati ad operare modifiche sulle rotte.

Lo script di caricamento si ottiene avviando il programma InitScript.jar con il seguente comand, passando come parametro il prefisso utilizzato nella creazione delle tabelle precedenti.

```
java -jar InitScript.jar merano > init.sql
```

Il file init.sql deve essere quindi caricato così come mostrato in precedenza per altri file SQL.

```
psql -U postgres -d linedb -f init.sql
```

Il suo caricamento provocherà la creazione di tre ulteriori tabelle geometriche, una tabella *utenti* e quattro ulteriori sequence.

L'inserimento di un primo account come amministratore nella tabella utenti deve essere fatto manualmente.

Geoserver

Geoserver è un server open source la cui funzione è quella di connettere e fornire informazioni georeferenziate utilizzando standard aperti, implementando le specifiche Web Feature Server (sia 1.0.0 che 1.1.0), Web Map Server (1.1.1) e Web Coverage Server (1.0).

All'atto della scrittura del documento, la versione più recente è la 1.7.0. Tuttavia, dal momento che il progetto è iniziato ad inizio settembre, in questo testo si fa riferimento alla versione 1.6.5, comunque scaricabile all'indirizzo <http://geoserver.org/display/GEOS/Stable>

Tra le versioni scaricabili, in questo testo si fa riferimento all'installazione e alla configurazione di Geoserver come applicazione Web integrabile in un application server. Pertanto si consiglia di scaricare il pacchetto in formato Web Archive (WAR).

Una volta scaricato ed estratto il file war, bisogna effettuare il deploy in Tomcat. Il deploy si

effettua avviando Tomcat e recandosi sulla pagina <http://localhost:8080/>



Apache
Tomcat



The **Apache Software Foundation**
<http://www.apache.org/>

**If you're seeing this page via a web browser, it means you've setup Tomcat successfully.
Congratulations!**

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:

`$CATALINA_HOME/webapps/ROOT/index.html`

where "\$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the [Tomcat Documentation](#) for more detailed setup and administration information than is found in the INSTALL file.

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in `$CATALINA_HOME/conf/tomcat-users.xml`.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation, and an introductory guide to developing web applications.

Tomcat mailing lists are available at the Tomcat project web site:

- users@tomcat.apache.org for general questions related to configuring and using Tomcat
- dev@tomcat.apache.org for developers working on Tomcat

Thanks for using Tomcat!

Powered by



TOMCAT

Copyright © 1999-2008 Apache Software Foundation
All Rights Reserved

Administration
Status Tomcat Manager
Documentation
Release Notes Change Log Tomcat Documentation
Tomcat Online
Home Page FAQ Bug Database Open Bugs Users Mailing List Developers Mailing List IRC
Miscellaneous
Servlets Examples JSP Examples Sun's Java Server Pages Site Sun's Servlet Site

Fig. 170 - Schermata iniziale dell'application server Tomcat

A questo punto bisogna selezionare il link “Tomcat Manager”, inserire le credenziali di accesso (file tomcat-users.xml, ruolo “manager”).

Path	Display Name	Running	Sessions	Commands
/	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy

Deploy directory or WAR file located on server

Context Path (optional):
XML Configuration file URL:
WAR or Directory URL:

WAR file to deploy

Select WAR file to upload

Fig. 171 - Tomcat Application Manager

Si deve quindi selezionare, tramite il tasto “Browse”, il file WAR geoserver.war e selezionare infine il tasto “Deploy”. Geoserver verrà quindi installato in modo automatico dal gestore applicazioni di Tomcat e comparirà nell'elenco delle applicazioni avviate. Per verificare l'avvenuta installazione, si punti il browser all'indirizzo <http://localhost:8080/geoserver>



Fig. 172 - Schermata iniziale del servizio Geoserver

Se si presenta la schermata come in Fig. 172, l'applicazione risulta essere correttamente installata.

Bisogna ora configurare il sistema GIS per riconoscere e fornire i dati inseriti nel database. Per fare questo bisogna per prima cosa aggiungere una sorgente dati, ed in seguito aggiungere i FeatureType serviti, ossia la lista di layers da veicolare, prelevati dalla sorgente dati. Infine bisogna impostare degli stili di visualizzazione per i vari dati.

Aggiunta di una sorgente dati



Fig. 173 - Schermata iniziale delle configurazioni

Per aggiungere una sorgente dati, si selezioni il link Config, si immettano il nome utente e la password di default di Geoserver (si consiglia fortemente di cambiare la password di default una volta terminata l'installazione della piattaforma) e si selezioni il link Data. Se i passi eseguiti saranno stati corretti, ci si troverà davanti alla pagina come in Fig. 173.

Si selezioni il link “DataStores”, e si crei una nuova sorgente dati selezionando il link “New”. Si selezioni “Postgis” come Feature Data Set Description e si inserisca un nome (nel nostro caso, “merano”) come ID.

Dopo aver selezionato il tasto “New”, si inseriscano i dati di connessione al DBMS PostgreSQL, e si lasci il resto invariato. Alla selezione del tasto “Submit”, se l'inserimento è stato corretto, si verrà reindirizzati alla pagina contenente la lista dei Feature Data Set (presentata in Fig. 174). In caso contrario, sulla parte sinistra dello schermo comparirà un avvertimento.



Fig. 174 - Inserimento di un nuovo Feature Data Set

Si noti l'asterisco (*) sulla sinistra a fianco della voce Configuration. Tale simbolo indica che è avvenuta una modifica nella configurazione del server, non ancora applicata. Per applicare la modifica, premere su “Apply”. L'asterisco a questo punto si sposterà a fianco della voce GeoServer, indicando che la configurazione corrente è diversa da quella salvata. Premere il tasto “Save” per salvare la configurazione corrente.

Aggiunta degli stili di visualizzazione

Il passo successivo è quello di aggiungere gli stili da utilizzare per la visualizzazione dei layers caricati nel database. Gli stili sono forniti insieme con l'applicazione Web Linee Mezzi Pubblici e sono stati pensati per permettere una migliore fruizione dei contenuti della piattaforma.

Per caricare gli stili, si selezioni, in GeoServer, il link “Data”, ed in seguito il link “Style”. Si crei un nuovo stile, selezionando il link “New” e inserendo come ID un nome esemplificativo (nell'esempio si utilizzerà il nome “strade_style”).

Nella pagina successiva si potrà effettuare l'upload del file “strade.sld”. Al termine dell'upload, si preme il tasto “Submit” per salvare lo stile.

Si prosegue aggiungendo gli stili successivi. Al termine dell'operazione, si ricordi di applicare e salvare la configurazione del server.

Aggiunta dei layer

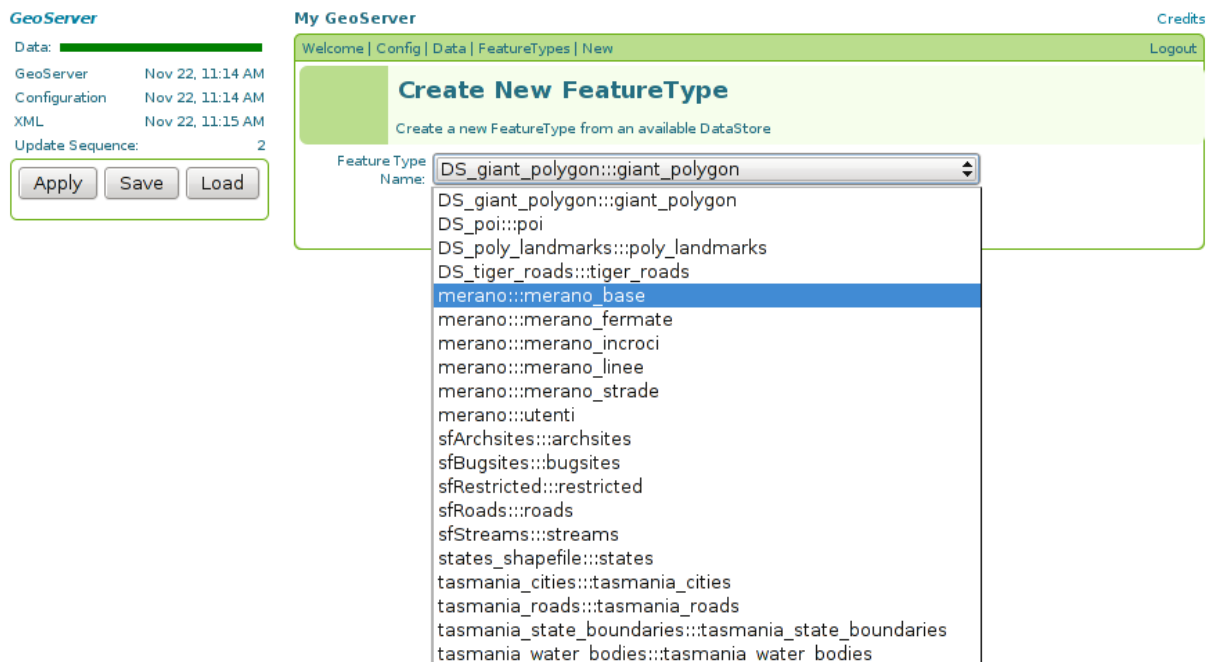


Fig. 175 - Aggiunta di un nuovo layer

Il passo ulteriore consiste nell'aggiungere i singoli layer per la loro gestione ed invio a client remoti. Per fare questo, si selezioni il link “Data” a partire dalla pagina principale di configurazione, ed in seguito il link “FeatureType”. Si crei un nuovo FeatureType selezionando il link “New”. Nella pagina successiva, sarà possibile vedere nell'elenco dei FeatureType disponibili anche quelli presenti nel database lineedb, così come mostrato in Fig. 175.

- Si selezioni uno dei layer da aggiungere (ad esempio merano_strade), e si selezioni il tasto “New”. Nella schermata che segue, i parametri da impostare sono i seguenti:
- si imposti lo stile
- nel campo SRS si scriva: 4326 (sistema di riferimento WGS84)
- si selezioni il tasto “Generate” per la generazione della bounding box
- Nel caso in cui la bounding box non venga generata (ad esempio con le tabelle _linee e _fermate, essendo tali layer ancora vuoti) si inseriscano le coordinate limite -180,180 per la longitudine e -90,90 per la latitudine
- si selezioni il tasto “Submit”
- Si ripeta il procedimento anche per i layer _base (per questo layer si selezioni lo stile “polygon”), _fermate, _incroci (per questo layer si selezioni lo stile “point”), _line.
- Al termine dell'inserimento ci si ricordi di applicare e salvare la configurazione.

Aggiunta di un layer come risultato della fusione di due layer

Per minimizzare il quantitativo di dati scaricati nell'atto di visualizzazione di una mappa,

l'applicazione Web è pensata per richiedere un unico layer di base, comprendente sia il layer di admin boundary, che il layer stradale.

Geoserver deve quindi essere configurato per poter fornire tale layer. Per fare questo, si selezioni il link “WMS” ed in seguito il link “Contents”.

In basso alla pagina, si selezioni il tasto “Add New Layer-Group”. Verrà creato un nuovo aggregato. Si forniscano quindi le seguenti informazioni:

- il nome del Layer-group (**attenzione:** questo nome deve coincidere con il nome posto come prefisso alle tabelle nella base di dati. Nel caso qui descritto sarà quindi *merano*)
- i nomi dei layer da “fondere”, separati da virgole e forniti di namespace
- le informazioni relative alla bounding box (ottenibili con il tasto “Generate”).

In Fig. 176 si ha un esempio di inserimento relativo al caso in questione. Al termine dell'inserimento, si sottomettano le informazioni con il tasto “Submit”.

Layer-group Name: merano Remove

Base Map Layers: topp:merano_base,topp:merano_strade

Base Map Styles:

SRS: EPSG:4326 SRS Help - SRS List

Generate

Envelope: Min X: 11.1364622116089 Min Y: 46.6162910461426

Max X: 11.2396869659424 Max Y: 46.6943054199219

Submit Reset Add New Layer-Group

Fig. 176 - Aggiunta di un layer WMS composito

Si applichi e si salvino le impostazioni. La configurazione di Geoserver è terminata. Per verificare le corrette impostazioni, Geoserver dispone di un client OpenLayers. Si selezioni il link “Welcome”, e si navighi sul link “Demo”. Selezionando il link “Map Preview” si potranno visualizzare tutte le mappe presenti e fornite dal server. Si cerchi la mappa “merano” e la si selezioni. Il risultato dovrebbe essere lo stesso presentato in Fig. 177.

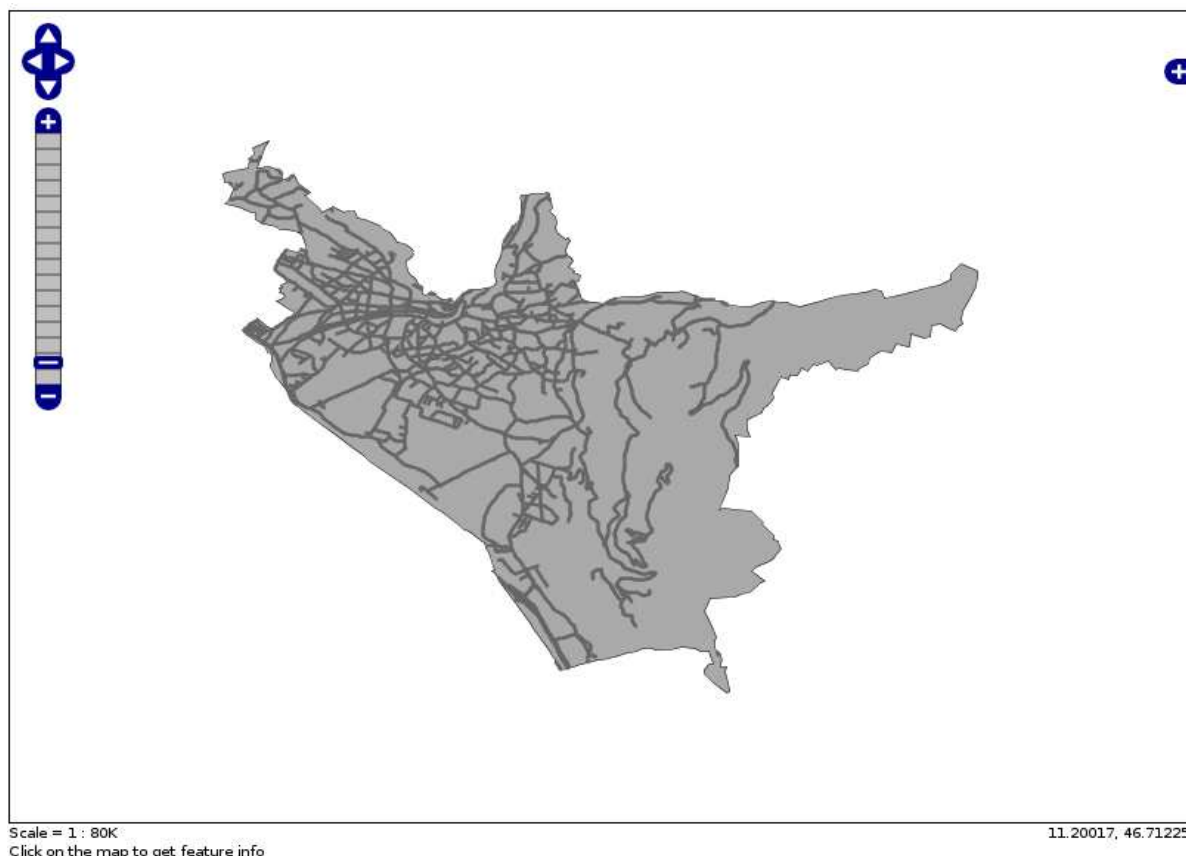


Fig. 177 - Visualizzazione con OpenLayers

GeoWebCache

L'estensione GeoWebCache consente di utilizzare un meccanismo di cache lato server, che riduce il tempo di processamento necessario a produrre una immagine della porzione di mappa desiderata. Questo viene ottenuto frapponendo tra il client e il server WMS un servizio proxy trasparente: compito del servizio è quello di verificare l'esistenza dell'immagine da fornire in una cache, prima di inoltrare la richiesta al servizio WMS. Nel caso in cui l'immagine venga trovata in cache, la richiesta non viene inoltrata e al client viene fornita direttamente l'immagine già elaborata.

Al momento della scrittura di questo testo, la versione ultima disponibile è la 1.7.0. Tuttavia, dal momento che il progetto è iniziato ad inizio settembre, in questo testo si fa riferimento alla versione 1.6.5, comunque scaricabile all'indirizzo <http://geoserver.org/display/GEOS/Stable>

All'interno dell'archivio sono presenti le istruzioni di installazione e configurazione dell'estensione. E' sufficiente copiare tutti i file contenuti nel pacchetto nella cartella:

```
<tomcat-dir>/webapps/geoserver/WEB-INF/lib
```

Il servizio di caching sarà a questo punto raggiungibile tramite l'URL <http://localhost:8080/geoserver/gwc/service/wms>

Linee Mezzi Pubblici Web Application

L'applicazione Web Portale inee Mezzi Pubblici è una applicazione Web sviluppata con tecnologia J2EE. L'applicazione consente l'accesso e la modifica ai dati cartografici messi a disposizione dal server Geoserver.

L'installazione dell'applicazione è decisamente semplice, dal momento che avviene nello stesso modo dell'applicazione Geoserver, ossia tramite il manager di configurazione di Tomcat.

Al termine del deploy, l'applicazione può essere raggiunta all'URL

<http://localhost:8080/LineeMezziPubblici/>

Tuttavia, prima di accedere all'indirizzo, l'applicazione necessita di un minimo di configurazione, possibile tramite il file:

```
<tomcat-dir>/webapps/LineeMezziPubblici/WEB-INF/  
classes/it/ieiit/mezzipubblici/config/config.properties
```

Le voci da configurare sono le seguenti:

- `ieiit.db.schema`: nome del database (nel caso in esame, `lineedb`)
- `ieiit.db.city.name`: nome del prefisso usato (nel caso in esame, `merano`)
- `ieiit.db.url`: url JDBC di accesso al DBMS
- `ieiit.db.username`: nome utente
- `ieiit.db.password`: password
- `ieiit.db.pool.size`: grandezza del pool di connessioni
- `ieiit.wfs.url`: URL **pubblico** al servizio WFS
- `ieiit.wms.url`: URL **pubblico** al servizio WMS
- `ieiit.cache.url`: URL **pubblico** al servizio di caching WMS

Al termine della configurazione, basta rilanciare l'applicazione tramite il manager di configurazione di Tomcat.

Gestione della definizione di una rotta

Introduzione

Nella definizione di una nuova rotta e nella modifica di una rotta preesistente, sono stati affrontati alcuni problemi, riguardanti essenzialmente due aree:

la gestione della concorrenza nella modifica della stessa rotta

i meccanismi per il matching di una rotta sul grafo stradale, sia per quanto riguarda gli archi componenti il percorso, sia per le fermate

Algoritmo di adattamento alle strade

Schema dell'algoritmo

Come si potrà verificare all'interno del capitolo seguente, l'inserimento di una rotta

all'interno del sistema è un procedimento manuale, in cui l'utente “disegna” approssimativamente il percorso seguito dal mezzo pubblico, e salva tale percorso all'interno del sistema. Per evitare di dover essere molto precisi in fase di disegno, il sistema mette a disposizione uno strumento di correzione automatica che viene eseguito, nel caso in cui venga attivato da apposito radio button, ad ogni modifica del percorso e in fase di inserimento del percorso stesso.

Il sistema è attivato di default, ma è possibile affinarne il comportamento o disattivarlo tramite un semplice controllo, mostrato in Fig. 178.

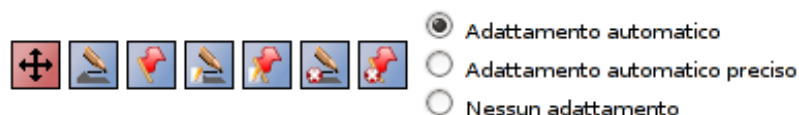


Fig. 178 - Strumenti di modifica e adattamento automatico

L'algoritmo prevede i seguenti passi:

Individuazione della sequenza degli incroci posti in prossimità del percorso disegnato dall'operatore

1. per ogni arco componente il percorso disegnato, si calcola un insieme di punti equidistanti tra loro
2. per ogni punto calcolato, si cerca un incrocio in un'area compresa all'interno di un raggio di lunghezza predeterminata
3. nel caso in cui venga individuato un incrocio, il punto su cui cade l'incrocio, viene aggiunto alla sequenza, eccetto nel caso in cui l'ultimo incrocio individuato sia uguale al nuovo incrocio

Elaborazione di una sequenza di punti $[0, N]$, composta nel modo seguente:

1. il punto 0, punto di partenza, allineato alla strada più vicina
2. i punti $[1, N-1]$ sono gli incroci individuati al passo 1
3. il punto N, punto di arrivo, allineato alla strada più vicina

Tentativo di estrazione degli archi stradali tra un punto ed il successivo, tenendo conto delle seguenti possibilità:

1. entrambi i punti cadono su incroci, estremi di una stessa strada
 Dal database viene estratta la sequenza di punti che compone la strada e i punti, eccetto l'ultimo, vengono aggiunti alla lista che compone il percorso
2. entrambi i punti cadono su incroci, estremi di strade diverse
 viene aggiunto il primo punto alla lista di punti componenti il percorso
3. un punto cade su un incrocio, mentre l'altro cade su una strada avente come estremo l'incrocio stesso
 Dal database viene estratto un sottoinsieme di punti componente la strada, che vengono aggiunti alla lista che compone il percorso, eccetto l'ultimo
4. un punto cade su un incrocio, mentre l'altro cade su una strada che non ha come

estremo l'incrocio stesso

viene aggiunto il primo punto alla lista di punti componenti il percorso

5. entrambi i punti non cadono su un incrocio

viene aggiunto il primo punto alla lista di punti componenti il percorso

La differenza tra l'algoritmo di adattamento e l'algoritmo di adattamento preciso riguarda esclusivamente la misura del raggio entro cui viene ricercato un incrocio: nel caso dell'adattamento preciso, il raggio è lungo un decimo rispetto a quello utilizzato nell'adattamento standard. L'utilizzo dell'adattamento preciso è utile per gli ultimi ritocchi al percorso, per risolvere situazioni di ambiguità nelle quali il sistema associ erroneamente ad un punto un incrocio errato.

I risultati dell'algoritmo di adattamento sono mostrati in Fig. 179 e Fig. 180.

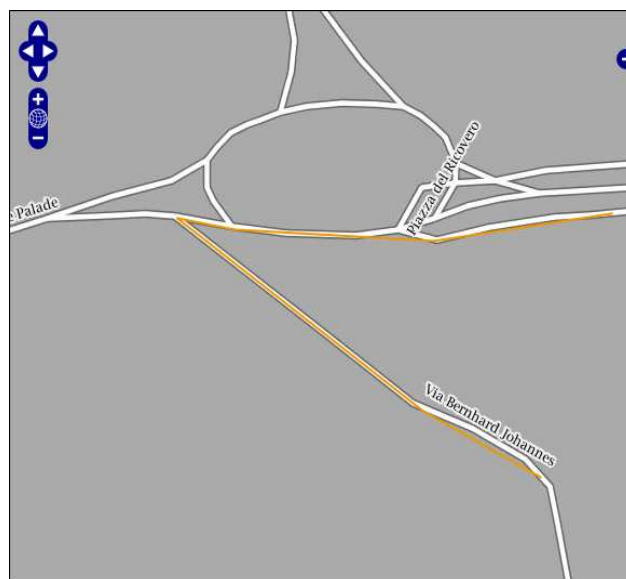


Fig. 179 - Percorso prima dell'adattamento

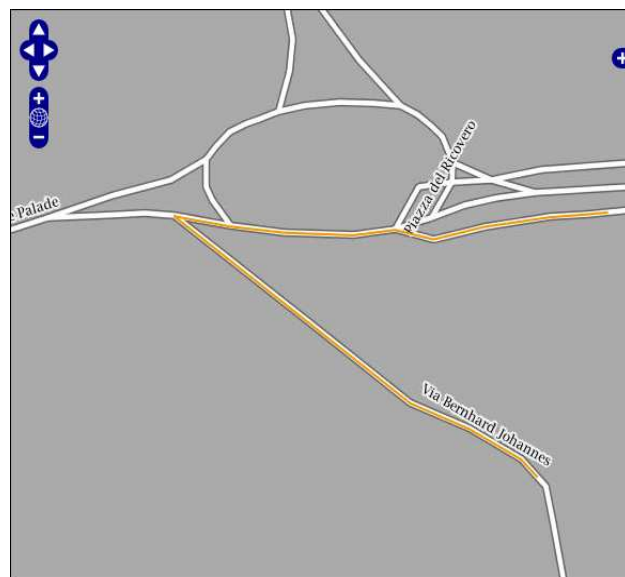


Fig. 180 - Percorso dopo l'adattamento

Un esempio

L'algoritmo viene ben esemplificato in Fig. 181: nella figura in alto a sinistra, l'operatore ha appena terminato di inserire un percorso. L'algoritmo di adattamento, dopo qualche istante, modifica il percorso inserito, portandolo ad assumere la configurazione nella figura in alto a destra. Si può notare come il percorso risulti perfettamente allineato alla strada, eccetto che nella parte bassa: questo comportamento si spiega con il fatto che l'algoritmo, nel processamento di quella sezione di percorso, non è riuscito a trovare un incrocio sufficientemente vicino. L'intervento manuale dell'operatore porta il percorso come nella figura in basso a sinistra. Dopo qualche altro istante, il sistema modifica il percorso nella sua conformazione finale, presente nella figura in basso a destra.

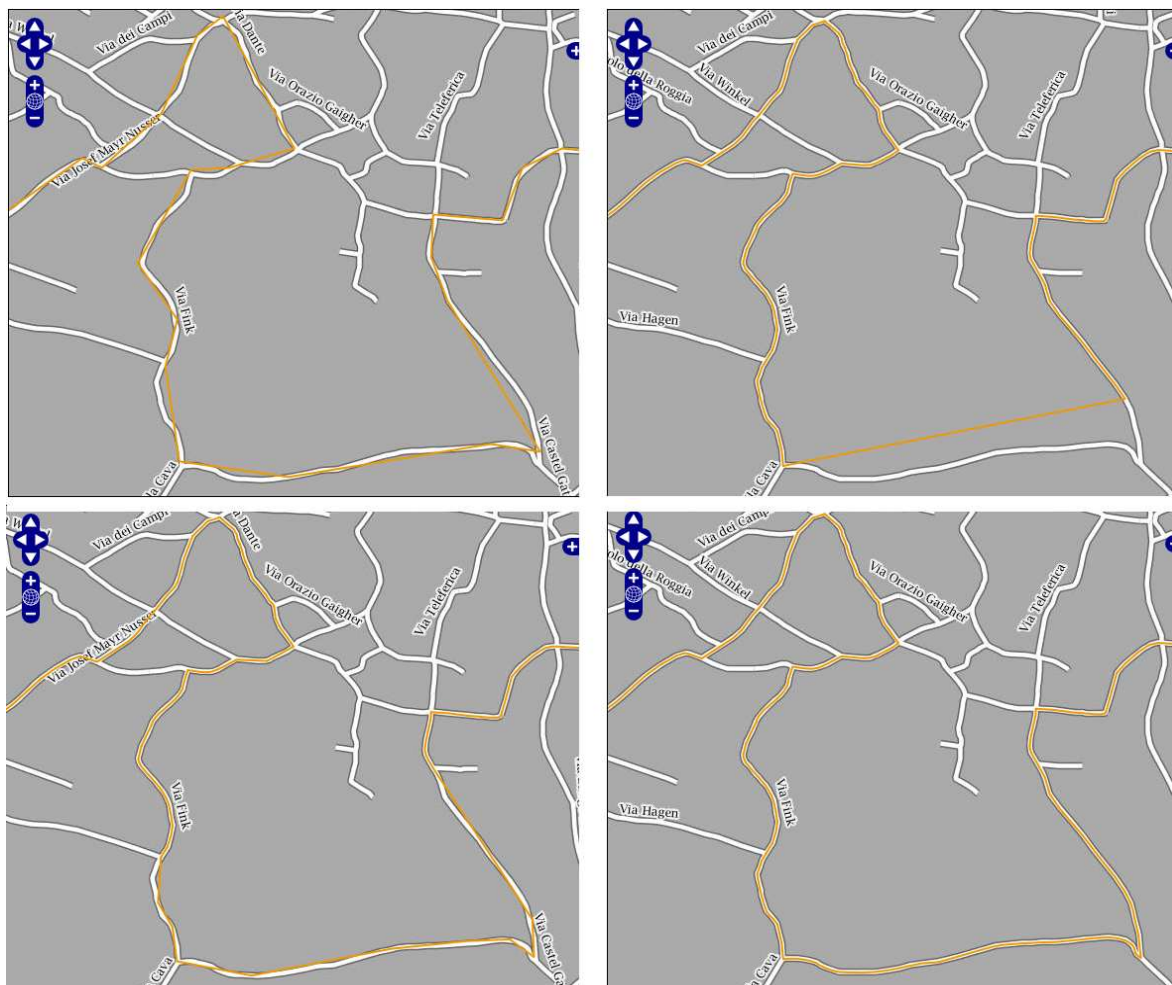


Fig. 181 - Algoritmo di adattamento con intervento manuale dell'operatore

Funzioni PostGis utilizzate

Nella stesura degli algoritmi, particolarmente utili sono risultate alcune funzioni messe a disposizione dall'estensione per dati spaziali PostGis al DBMS PostgreSQL. Tali funzioni sono:

`ST_line_interpolate_point(linestring,location)`

Permette di ottenere la geometria di un punto, interpolato su una linea

`ST_line_substring(linestring,start,end)`

Permette di estrarre una sottolinea a partire da una linea e da due punti su questa linea (i punti sono espressi come float tra 0 e 1)

`ST_line_locate_point(linestring,point)`

Restituisce un float tra 0 e 1, rappresentante il punto interpolato sulla linea più vicino al punto passato come argomento

`ST_distance(geometry,geometry)`

Restituisce la distanza tra due geometrie in unità di proiezione.

`GeometryFromText(string)`

Converte una stringa in formato WKT in una entità di tipo *geometry* gestibile direttamente da PostGis

Algoritmo di adattamento delle fermate

L'algoritmo di adattamento delle fermate viene avviato ad ogni salvataggio di fermata sul database. Il procedimento è molto simile al passo 1 dell'algoritmo di adattamento alle strade.

Per ogni arco stradale componente il percorso della rotta, si calcola un insieme di punti equidistanti, appartenenti all'arco stradale in questione

Per ciascun punto, si calcola la distanza con ciascuna fermata

Per ogni fermata, si tiene memoria del punto del percorso a distanza minima

Al termine del procedimento, si allinea la fermata al punto del percorso a distanza minima

Al termine del procedimento, si arricchiscono le fermate con le informazioni riguardanti le vie su cui si trovano con un semplice algoritmo di map matching, ossia cercando la strada a distanza minima.

Un esempio dell'algoritmo può essere osservato in Fig. 182 e Fig. 183

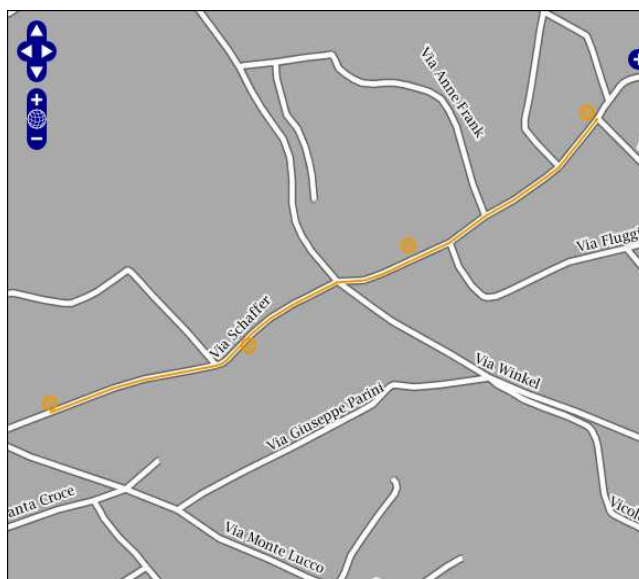


Fig. 182 - Fermate non allineate

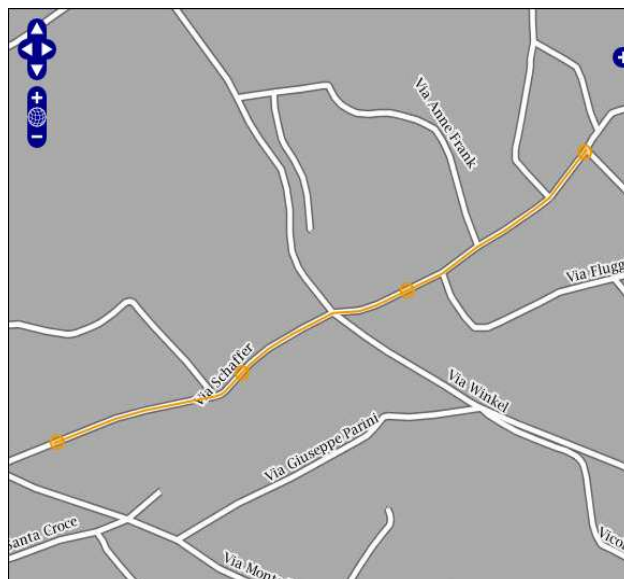


Fig. 183 - Fermate a seguito dell'allineamento

Gestione delle modifiche concorrenti

Durante la modifica di rotte pre-esistenti, può verificarsi l'eventualità di una modifica concorrente di uno stesso percorso da parte di due operatori differenti.

In genere, la gestione di un evento di questo genere può prevedere due soluzioni:

la prima, più conservativa, impone il blocco della risorsa da parte di uno degli operatori, in modo da impedire agli altri operatori di modificarla fino a quando l'operatore che detiene i diritti di blocco non decide di sbloccare la risorsa occupata

la seconda, più ottimistica, prevede la gestione di corse critiche avvisando opportunamente l'operatore in fase di salvataggio, se i dati inviati dall'operatore rischiano di sovrascrivere una modifica introdotta e salvata durante il lavoro dell'operatore stesso

In questo contesto, si è deciso di ricorrere alla seconda soluzione in quanto la prima soluzione, sebbene concettualmente più robusta in quanto serializza gli accessi alla risorsa, possa comportare lo scenario in cui il blocco di un percorso si protrae per un tempo molto superiore rispetto a quanto necessario (ad esempio, dimenticando di sbloccare la risorsa non appena terminato il lavoro).

Lo scenario concorrente viene quindi gestito con l'introduzione di un campo "lastUpdate" (timestamp) nella tabella del database contenente le rotte. Ad ogni modifica della rotta, il campo viene aggiornato con la data e l'ora dell'aggiornamento. Le informazioni temporali di aggiornamento, tuttavia, vengono anche trasferite all'operatore quando questo si accinge a modificare un percorso. Al termine della modifica, queste informazioni vengono trasferite nuovamente al sistema: se il timestamp trasferito dal client è identico al timestamp memorizzato sul database, la transazione avviene con successo, la linea modificata viene salvata, ed il timestamp ad essa associato viene aggiornato. In caso contrario, l'utente viene avvisato di un conflitto concorrente, a cui potrà rispondere forzando comunque il salvataggio dei dati, oppure scartando le sue modifiche, visualizzando la rotta aggiornata. La schermata di conferma viene mostrata in Fig. 184.

Nel caso in cui si scegliesse di salvare la rotta in modalità asincrona, verrà mostrato un

messaggio di errore durante il salvataggio.



Fig. 184 - Avviso di modifica concorrente

Principali scenari di utilizzo

Vengono ora trattati i principali scenari di caso d'uso del Portale Gestione Linee Trasporto Pubblico, in cui viene descritto sia l'uso del portale da parte di un utente o di un operatore, sia i dettagli tecnici di come determinate operazioni vengano eseguite dal sistema.

Modifica dell'elenco degli operatori

Nel caso in cui il login sia effettuato da un utente avente anche lo status di amministratore, nella barra dei menu comparirà la voce “Gestione operatori”.

Selezionandola, verranno mostrati gli operatori attualmente registrati al sistema e una maschera che permette l'aggiunta di nuovi operatori. A fianco del nome di ciascun operatore è presente un tasto “Elimina” il quale, a seguito di conferma, provvede ad eliminare l'operatore dal sistema.

Visualizzazione di una rotta

All'ingresso nel portale, al visitatore viene mostrata la pagina in Fig. 185. Tramite una richiesta AJAX vengono caricate le ultime linee modificate, le quali sono direttamente accessibili sul lato destro della schermata.

In alternativa, l'utente può selezionare il collegamento “Visualizza Linee”, che lo porterà ad ottenere una schermata simile a quella in Fig. 186.



Fig. 185 - Home Page del Portale

Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Ricerca

Username Password

Linee presenti nel sistema

Arianna	Prova	Visualizza linea
Prima prova	Descrizione della prima prova effettuata	Visualizza linea
Quarta Prova	Descrizione quarta prova	Visualizza linea
Quinta prova	Descrizione quinta prova	Visualizza linea
Seconda prova	Descrizione	Visualizza linea
Sesta prova	Descrizione numero sei	Visualizza linea
Settima prova	Descrizione settima prova	Visualizza linea
Terza Prova	Descrizione terza prova	Visualizza linea

Copyright IEIIT-CNR Sede di Bologna

Fig. 186 - Elenco delle linee pubbliche presenti nel sistema

L'utente può quindi selezionare la linea a cui è interessato tramite il link “Visualizza linea”. Il sistema provvederà al caricamento dei dati necessari e presenterà all'utente una pagina simile a quella in Fig. 187.

L'utente può visualizzare l'elenco delle fermate eseguite dal mezzo pubblico, comprensive di indirizzo su cui sono localizzate. Selezionando una fermata con un click del tasto sinistro del mouse, la fermata viene selezionata. La mappa provvede a centrarsi sulla fermata selezionata.

In ogni momento l'utente potrà centrare la mappa, selezionando il tasto “Centra la mappa”.

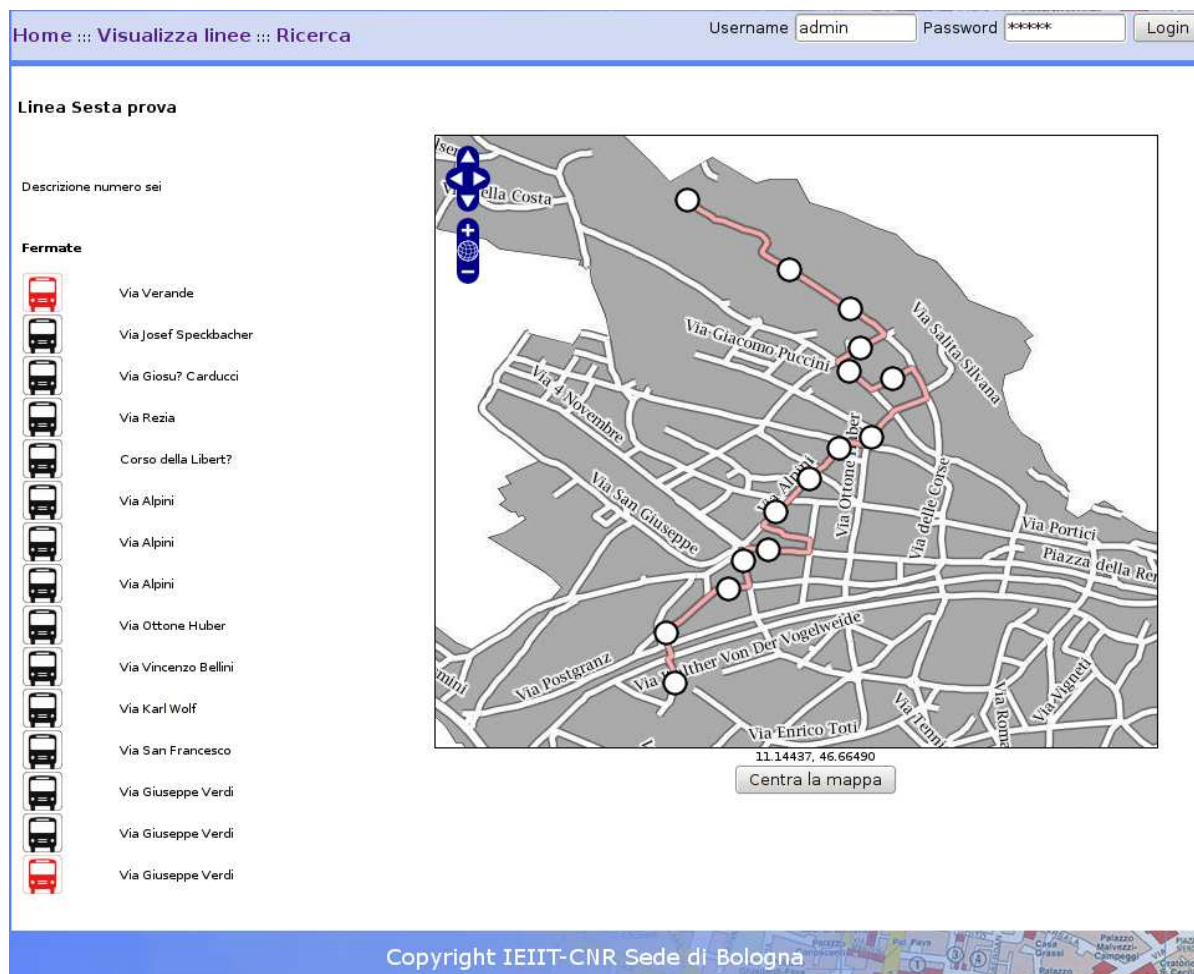


Fig. 187 - Visualizzazione di una rotta

Inserimento di una nuova rotta

A partire da qualunque punto all'interno del sito, un utente può effettuare il login sul sistema, inserendo nome utente e password di accesso nei campi posti in alto a destra. Nel caso in cui le credenziali di accesso fossero sbagliate, il sistema visualizzerà una pagina di errore. In caso di accesso effettuato con successo, il nome utente dell'operatore comparirà al posto dei campi utilizzati per l'accesso, e si verrà automaticamente redirezionati sull'home page. Sulla barra di menu comparirà il link "Crea linea".

Selezionandolo, l'operatore verrà rediretto alla pagina riportata in Fig. 188.

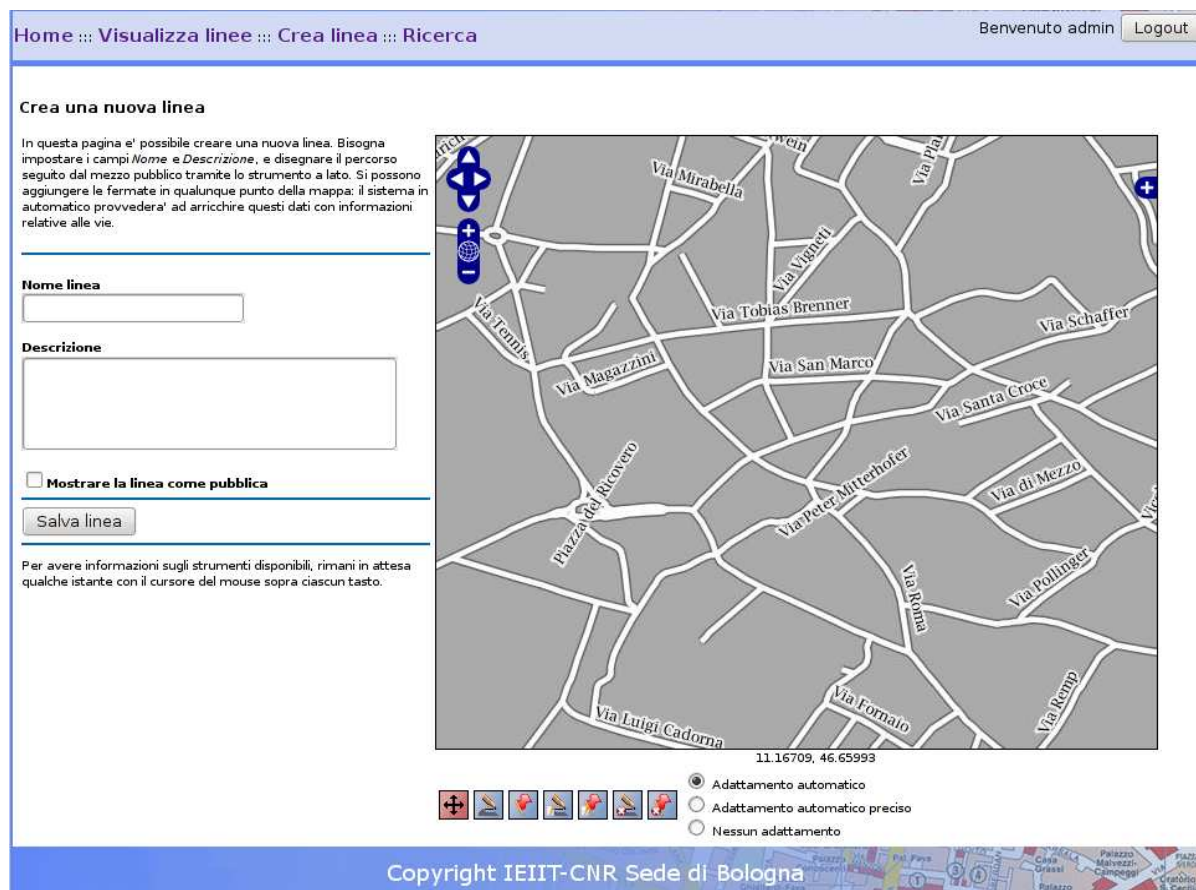


Fig. 188 - Creazione di una nuova linea

L'operatore dovrà inserire il nome della nuova linea e una breve descrizione. Successivamente provvederà a disegnare la rotta seguita dal mezzo pubblico e ad aggiungere le fermate tramite lo strumento di editing riportato in Fig. 189.

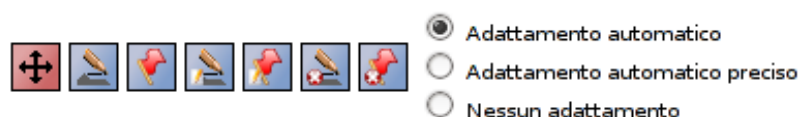


Fig. 189 - Strumento di creazione linee e fermate

Da sinistra verso destra troviamo gli strumenti per:

Navigare la mappa

Tenendo premuto con il tasto sinistro del mouse è possibile trascinare la mappa e caricarne nuove porzioni. La navigazione è anche possibile utilizzando i tasti freccia della tastiera. Tale modalità risulta decisamente utile in fase di inserimento di percorsi, quando si ha necessità di scorrere la mappa.

Disegnare una nuova linea

Si selezioni lo strumento e si faccia click con il tasto sinistro sulla mappa. Spostando il mouse a seguito del click si noterà una linea partire dal punto selezionato e terminare sul cursore del mouse. Selezionare il punto successivo con il click del mouse per creare un arco. Continuare la creazione del percorso, spostandosi con i tasti freccia per far scorrere la

mappa. Terminare la creazione del percorso selezionando l'ultimo punto da creare con un doppio click del mouse. Se la funzione di adattamento automatico è stata attivata, si noterà dopo qualche istante l'adattamento del percorso al grafo stradale, sempre tenendo conto dei limiti riportati nella sezione Algoritmo di adattamento alle strade (pag. 257).

Aggiungere le fermate

Si seleziona lo strumento e si fa click con il tasto sinistro sulla mappa sul punto in cui aggiungere una fermata. Non è necessario essere particolarmente precisi in quanto il sistema correggerà la posizione immessa, agganciandola al percorso del mezzo, in fase di salvataggio della linea

Modificare la linea

Si seleziona lo strumento e si fa click con il tasto sinistro del mouse sopra la linea inserita. La linea si colorerà di blu, e verranno evidenziati i punti di controllo (arancione acceso). Verranno inoltre evidenziati i punti mediani di ciascun arco: trascinando uno di questi punti, l'arco verrà suddiviso in due, ed il punto mediano diventerà un nuovo punto di controllo. Al termine della modifica, fare click in un qualsiasi punto della mappa. Se la funzione di adattamento automatico è stata attivata, si noterà dopo qualche istante l'adattamento del percorso al grafo stradale, sempre tenendo conto dei limiti riportati nella sezione Algoritmo di adattamento alle strade (Pag. 257).

Modificare una fermata

Si seleziona lo strumento e si fa click con il tasto sinistro del mouse sopra una fermata. La fermata si colorerà di blu. E' possibile quindi trascinare la fermata, tenendo premuto il pulsante sinistro del mouse e spostando la fermata nella nuova posizione. Al termine della modifica, fare click in un qualsiasi punto della mappa.

Cancellare la linea

Si seleziona la linea da eliminare tramite lo strumento di modifica delle linee. Si preme il tasto di cancellazione linea. Il sistema provvederà ad avvisare l'operatore circa l'irreversibilità dell'operazione. In caso di risposta affermativa al messaggio di conferma, la linea verrà eliminata.

Cancellare una fermata

Si seleziona la fermata da eliminare tramite lo strumento di modifica delle fermate. Si preme il tasto di cancellazione linea. Il sistema provvederà ad avvisare l'operatore circa l'irreversibilità dell'operazione. In caso di risposta affermativa al messaggio di conferma, la fermata verrà eliminata.

Attivare o disattivare lo strumento di adattamento automatico

Per una descrizione dell'algoritmo di adattamento automatico, si rimanda alla sezione Algoritmo di adattamento alle strade (pag. 257). Si raccomanda di tenere attivato inizialmente l'adattamento automatico. Per rifinire il percorso, si consiglia di utilizzare l'adattamento automatico preciso. Nel caso in cui il sistema effettui operazioni errate di adattamento, è possibile disabilitare l'algoritmo selezionando la voce "Nessun adattamento".

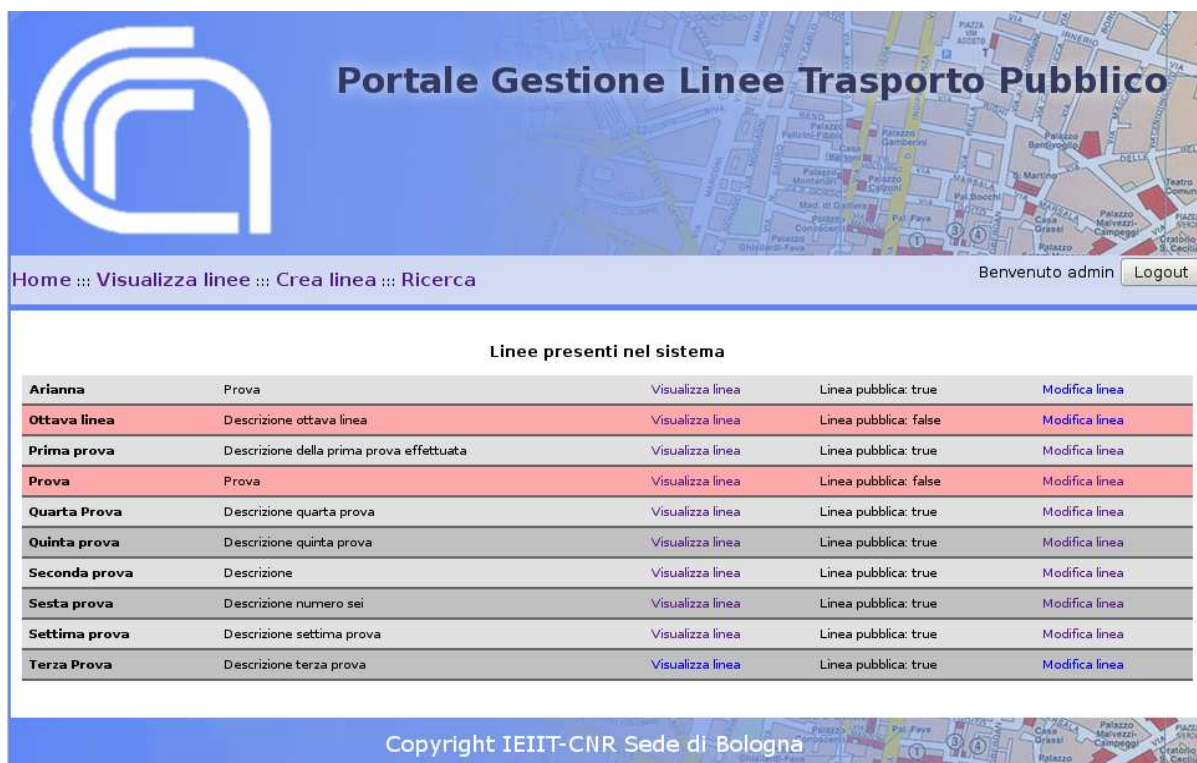
Al termine dell'inserimento, è possibile salvare la rotta selezionando il tasto "Salva linea". Il sistema provvederà a compiere dei controlli preliminari. In caso di esito positivo, la linea verrà salvata nel sistema. Se si seleziona il checkbox "Mostrare la linea come pubblica", il sistema renderà la linea visibile ai visitatori. In caso contrario, la linea sarà accessibile solo

agli operatori.

Modifica di una rotta preesistente

Una volta autenticato, l'operatore può selezionare la voce sulla barra dei menu “Visualizza linee”. Verrà mostrata una pagina simile a quella riportata in Fig. 190.

Le linee mostrate su sfondo rosa sono quelle linee non ancora rese pubbliche. In più, rispetto alla Fig. 186, è presente anche un link “Modifica linea” tramite cui si può accedere alla pagina di modifica, come mostrato in Fig. 191.



Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Crea linea :: Ricerca Benvenuto admin Logout

Linee presenti nel sistema				
Arianna	Prova	Visualizza linea	Linea pubblica: true	Modifica linea
Ottava linea	Descrizione ottava linea	Visualizza linea	Linea pubblica: false	Modifica linea
Prima prova	Descrizione della prima prova effettuata	Visualizza linea	Linea pubblica: true	Modifica linea
Prova	Prova	Visualizza linea	Linea pubblica: false	Modifica linea
Quarta Prova	Descrizione quarta prova	Visualizza linea	Linea pubblica: true	Modifica linea
Quinta prova	Descrizione quinta prova	Visualizza linea	Linea pubblica: true	Modifica linea
Seconda prova	Descrizione	Visualizza linea	Linea pubblica: true	Modifica linea
Sesta prova	Descrizione numero sei	Visualizza linea	Linea pubblica: true	Modifica linea
Settima prova	Descrizione settima prova	Visualizza linea	Linea pubblica: true	Modifica linea
Terza Prova	Descrizione terza prova	Visualizza linea	Linea pubblica: true	Modifica linea

Copyright IEIIT-CNR Sede di Bologna

Fig. 190 - Elenco delle linee complessive presenti nel sistema

La linea e le fermate vengono caricate in OpenLayers su un layer vettoriale utilizzando la sintassi WKT (Well Known Text). La pagina di modifica è molto simile alla pagina di creazione di una linea. Rispetto a questa, si nota il tasto “Termina inserimento”. In questo caso, infatti, il tasto “Salva linea” provoca una invocazione asincrona tramite AJAX, per salvare la linea senza dover effettuare il refresh della pagina. Nel caso in cui l'operazione di salvataggio vada a buon fine, verrà mostrato un messaggio riportante la data e l'ora dell'ultimo salvataggio. In caso di modifica concorrente, verrà riportato un messaggio di errore. Selezionando il tasto “Termina inserimento” si potrà gestire l'errore di concorrenza, salvando comunque il proprio lavoro oppure annullando le proprie modifiche.

Per gli approfondimenti riguardo la gestione della concorrenza si rimanda alla sezione Gestione delle modifiche concorrenti (pag. 262).

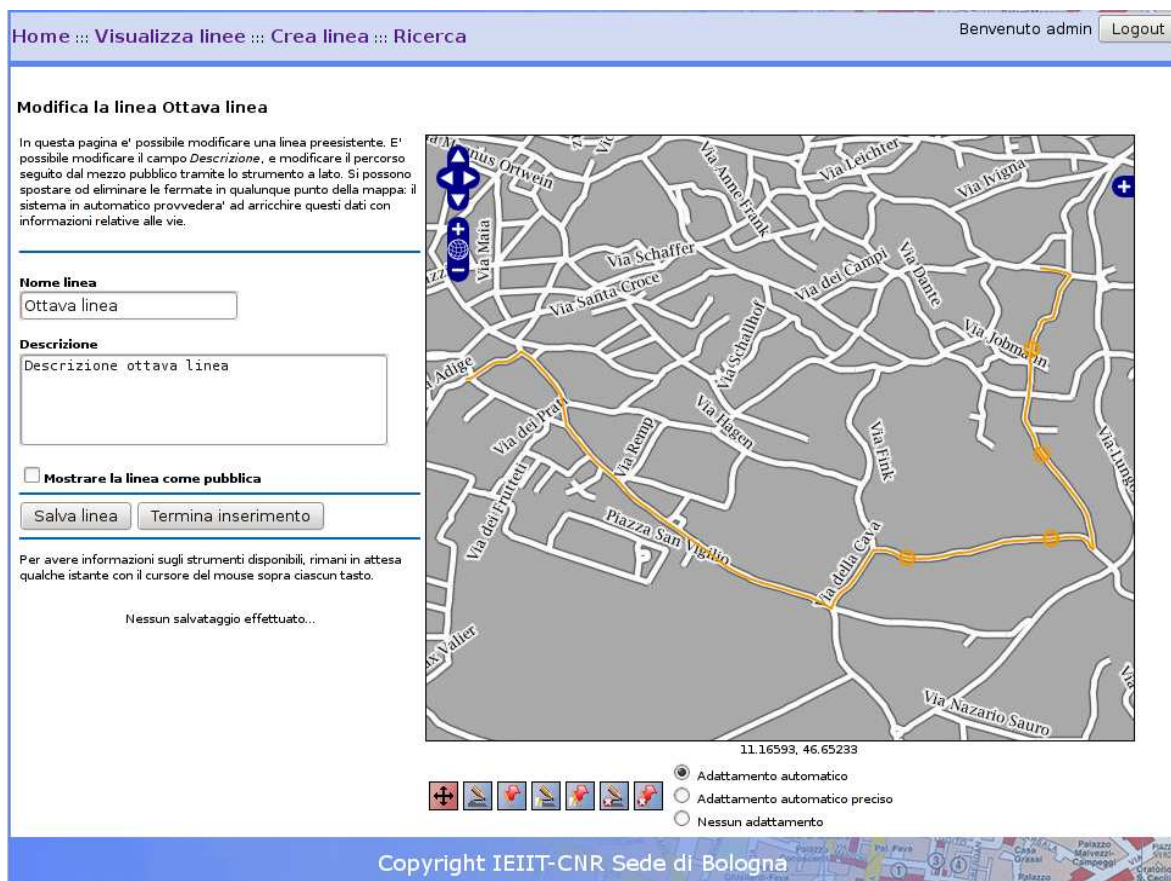


Fig. 191 - Modifica di una linea esistente

Ricerca di una rotta

Per ricercare una rotta, l'utente può selezionare la voce "Ricerca" dalla barra di menu. Il sistema proporrà una schermata simile a quella riportata in Fig. 192.

Tramite la maschera proposta, l'utente può cercare linee:

- tramite una parola che deve essere contenuta nel nome della rotta
- tramite una parola che deve essere contenuta nel nome oppure nella descrizione della rotta
- tramite una parola che deve essere contenuta nell'indirizzo di una fermata della rotta

La ricerca è case sensitive. In seguito all'interrogazione, il sistema inoltra l'elenco delle linee trovate (un esempio è riportato in Fig. 193, in cui è presente un'unica rotta). Nel caso in cui l'utente che avesse sottoposto l'interrogazione fosse un operatore, verrebbero riportate anche le informazioni relativamente alla pubblicazione della linea e alla data di ultima modifica.

Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Ricerca

Username Password

Ricerca linee

Parola chiave:

Cerca:

☒ tra le vie della città'

☐ nei nomi delle linee

☐ nei nomi e nelle descrizioni delle linee

Copyright IEIIT-CNR Sede di Bologna

Fig. 192 - Maschera per la ricerca di linee

Portale Gestione Linee Trasporto Pubblico

Home :: Visualizza linee :: Ricerca

Username Password

Risultato della ricerca

Quinta prova

Descrizione quinta prova

Ultimo aggiornamento: 2008-11-23 17:41:52.979

Copyright IEIIT-CNR Sede di Bologna

Fig. 193 - Risultato della ricerca

CASO DI STUDIO PER LA LOCALIZZAZIONE REMOTA

In questo capitolo viene mostrato l'integrazione di un sistema che permetta di conoscere da remoto la posizione GPS di un'entità mobile. Tale sistema prevede quindi raccolta del dato GPS da parte dell'entità mobile e sua condivisione con un centro di raccolta remoto.

Questa esperienza viene presentata all'interno di questo documento come parte del background formativo nella gestione di dati GPS e nella comprensione delle problematiche relative alle tecnologie impegnate nella reliazione di sistemi di Automatic Vehicle Location.

Cenni sul Global Positioning System

Il *Global Positioning System* è un sistema di posizionamento su base satellitare, a copertura globale e continua, gestito dal dipartimento della difesa statunitense. Il sistema di posizionamento GPS è basato sulla ricezione di segnali radio emessi da una costellazione di satelliti artificiali in orbita attorno alla Terra. La denominazione completa del sistema è NAVSTAR GPS che significa "*NAVigation Satellite Timing And Ranging Global Positioning System*".



Fig. 194 - Costellazione di satelliti

Cenni storici

La progettazione del GPS parte ufficialmente nel 1963. Il progetto, classificato top secret per 15 anni, prende il via durante la guerra fredda per scopi esclusivamente militari. La sola fase di progettazione del sistema GPS ha richiesto quasi 16 anni principalmente a causa del fatto che all'epoca i progettisti avevano individuato soluzioni tecnologiche che non erano ancora state in realtà sviluppate o affinate. Lo sviluppo del sistema si è in parte basato anche sulle esperienze e sulle conoscenze acquisite con sistemi simili o precursori (es. Landmark, Omega, Satnav, Loran).

Con il completamento della seconda versione del GPS (denominata "*Block two*" 1989-1994), il governo USA decide di rendere pubblici i parametri di accesso per sfruttare il sistema. Il 1994 segna l'inizio ufficiale della grande esperienza del GPS civile, con ogni probabilità il più vasto, sofisticato e complesso sistema tecnologico permanente mai creato

dal genere umano e punto di svolta epocale nella sicurezza aerea e marittima. Fino al maggio 2000, il segnale per uso civile veniva degradato per ridurre la precisione nell'ordine di 100-150 m. In seguito, è stata disabilitata la degradazione del segnale, consentendo la precisione attuale di circa 10-20 m. Nei modelli per uso civile è presente un dispositivo che inibisce il funzionamento ad altezze e velocità superiori a certi valori. Questa restrizione è stata adottata per impedire il montaggio di missili improvvisati.

La costellazione di satelliti è il risultato di invii successivi; il primo gruppo di satelliti (Blocco 1) fu lanciato dalla California tra il 1978 e il 1985 e non sono attualmente più operativi. Nel 1998, infatti, è iniziata la messa in orbita dei satelliti del Blocco 2 delle serie II, HA e IIR, di dimensioni estremamente contenute (meno di 6 metri di larghezza con i pannelli solari completamente estesi e un peso di circa 1.500 kg), funzionanti a energia solare e dotati di piccoli propulsori autonomi per la correzione degli errori di rotta. Gli orologi atomici montati a bordo sono quattro, due al cesio e due al rubidio, per garantire la massima precisione e l'adeguata ridondanza.

Una delle caratteristiche peculiari del sistema GPS è quella di essere costantemente al passo con l'evoluzione tecnologica grazie al periodico rinnovo della costellazione (la vita media di un satellite GPS è di circa 7 anni e mezzo).

La costellazione

La "mappatura" della terra è ottenuta grazie ad una costellazione di satelliti in orbita circolare intorno alla Terra a circa 20.200 km di altezza. Attualmente vi sono 24 satelliti operativi e tre satelliti latenti (vere e proprie "ruote di scorta") disposti su 6 differenti piani orbitali con un piano di inclinazione di 55°. Questo sistema di orbite ed inclinazioni consente di visualizzare la superficie terrestre da perlomeno 4 satelliti contemporaneamente. Questi satelliti hanno un periodo di rotazione di 12 ore, o meglio "quasi" 12 ore; infatti ogni giorno anticipano di circa 4 minuti. Questo scarto, unito ad altre considerazioni (tra cui le irregolarità della terra e l'attrazione gravitazionale della luna), rendono indispensabile un controllo a terra, rappresentato da 4 postazioni disposte in prossimità dell'equatore: Isola di Ascensione, Diego Garcia, Hawaii e atollo di Kwajalein. Queste postazioni supportano i satelliti, correggendone gli orologi atomici. I dati raccolti vengono inviati alla stazione master di Colorado Springs, dove vengono elaborati per valutare possibili errori. Il dato aggiornato viene ritrasmesso al satellite che, a sua volta, lo ritrasmette ai vari ricevitori degli utenti, consentendo agli stessi la precisa individuazione della costellazione.

L'opera delle basi a terra è fondamentale per la funzionalità del sistema satellitare, senza queste i satelliti diverrebbero presto inutilizzabili.

Funzionamento

Il sistema di funzionamento del GPS si basa sul metodo della *triangolazione* (Fig. 195), un sistema utilizzato per secoli nella navigazione marittima. L'interfaccia dell'utente riceve impulsi dai satelliti della costellazione GPS e, attraverso un sistema di equazioni, desume la propria posizione triangolando i segnali. Utilizzando la rilevazione della posizione di almeno tre punti fissi (con coordinate note) si calcola la propria posizione, data dall'incontro delle rette passanti per detti punti. Questa prima considerazione fa dedurre, quindi, che il dispositivo di navigazione al suolo (in mare o nel cielo) non comunica la propria posizione ai satelliti, ma riceve segnali univoci e continui da quest'ultimo che quindi li invia in maniera unidirezionale. La ricezione dei segnali di tre distinti satelliti fornisce un'indicazione abbastanza precisa della posizione, ma non assoluta. Per questo motivo occorre ricevere l'impulso da un quarto satellite per ottenere la maggiore precisione possibile.

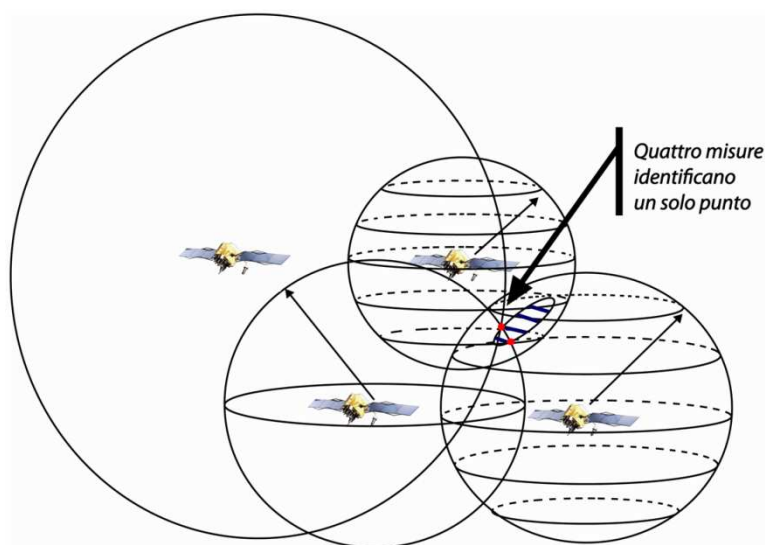


Fig. 195 - Metodo della triangolazione

Vediamo ora nei dettagli il motivo per il quale occorrono quattro satelliti per una localizzazione precisa.

Ogni satellite della costellazione genera un segnale contenente tre informazioni: il proprio identificativo, la posizione sull'orbita in cui si trova e un segnale temporale la cui precisione è garantita dall'orologio atomico montato a bordo. Attraverso queste informazioni il dispositivo ricevente è in grado di conoscere la distanza esatta dal satellite, applicando una delle leggi di Newton ($distanza = velocità \times tempo$) e quindi moltiplicando il tempo di percorrenza del segnale per la velocità della luce (circa 300.000 km al secondo). Chiaramente, però, il ricevitore può trovarsi in un qualsiasi punto di un'ipotetica sfera il cui raggio è rappresentato dalla distanza ricevitore/satellite. La ricezione di un secondo segnale (da un secondo satellite), analogo al primo, genera una seconda sfera che s'intersecherà con la prima in due punti generando un'ellisse entro la quale si troverà il punto ricercato. Basterà quindi ricevere il segnale da un terzo satellite per limitare le possibilità a due punti molto vicini (uno dei quali potrà essere automaticamente eliminato per via di considerazioni matematiche e cinematiche) e quindi ottenere la corretta posizione dell'apparato ricevente.

I problemi essenziali a questo punto sono due: calcolare con massima precisione il tempo di percorrenza del segnale e potere garantire la massima sincronizzazione degli orologi (quello sul satellite e quello dell'apparato di ricezione). Il sistema di codici utilizzato per ottenere la massima precisione del tempo di percorrenza è di tipo "pseudocasuale" (con questo termine s'intende un sistema di codici estremamente complesso, tale da apparire pressoché casuale. In realtà si tratta di un codice ripetuto mille volte al secondo). Il confronto tra la lettura del codice generato dal satellite e quella del codice generato localmente dal dispositivo ricevente (teoricamente nello stesso istante) causa una discrepanza e quindi una grandezza misurabile che fornisce il valore della distanza.

Il satellite emette il proprio segnale, e invia il proprio codice, su due portanti: L1 (1.575,42 MHz) e L2 (1.227,60 MHz). Sulla prima portante viaggia un codice, detto C/A (*Coarse Acquisition* o acquisizione grossolana), ripetuto ogni 1.023 bit, che occupa una banda pari a 1 MHz. Esso definisce un'acquisizione grossolana in quanto lo sfasamento di un microsecondo (pari a poco meno di un ciclo), moltiplicato per l'enorme distanza, può generare errori fino a 300 metri. Il secondo codice P (*Precise*) è modulato invece a 10 MHz,

quindi con una frequenza al secondo dieci volte maggiore. Questo è il codice che, combinato con il primo, garantisce la precisione richiesta. Il codice P è anche quello che l'amministrazione americana si è riservata di potere degradare in caso di assoluta necessità e di minaccia alla sicurezza nazionale (come è avvenuto nelle ore successive agli attacchi alle Twin Towers dell'11 settembre 2001). Risolto il problema dei codici la questione è riconducibile ad un problema di precisione degli orologi. I quattro montati sul satellite sono precisi al milionesimo di secondo, ma non altrettanto precisi possono essere quelli montati sull'apparecchiatura ricevente (il costo e le dimensioni sarebbero proibitivi). Entra in gioco allora il segnale ricevuto dal quarto satellite.

Supponendo che tutti i dispositivi (spazio-terra) siano perfettamente sincronizzati, la triangolazione di quattro segnali che s'intersecano dovrebbe fornire l'indicazione accurata di un punto. Poiché questo nella realtà non accade, si deve assumere come unica fonte di errore il dispositivo ricevente. Ragionando al contrario è possibile calcolare quali aggiustamenti possano riportare l'intersezione dei segnali a un punto singolo e in base a tali valori ricalcolare le coordinate e la posizione. Ma i problemi non sono finiti, a causa dell'attrazione gravitazionale terrestre e lunare, nonché delle radiazioni solari, le orbite dei satelliti subiscono scostamenti che vanno corretti con il supporto delle postazioni terrestri indicate prima; inoltre, durante l'attraversamento della ionosfera e della troposfera i segnali, a causa delle particelle cariche elettricamente che interferiscono con le onde elettromagnetiche, subiscono rallentamenti che vanno compensati. Per risolvere questo problema viene utilizzato il metodo di confronto tra le due portanti (L1 e L2), che vengono rallentate in maniera differente dalle particelle. Confrontando i ritardi, è possibile calcolare gli effetti dell'atmosfera sul segnale. Ultimo, ma non per importanza, è l'effetto dovuto al "multipath" (multipercorso), ovvero la riflessione dei segnali causata dalla presenza di ostacoli a terra, nonché l'effetto Doppler causato dal movimento del ricevitore. Queste potenziali fonti di errore sono corrette attraverso elaborazioni numeriche dei segnali operate direttamente dai dispositivi di ricezione.

Per quanto riguarda il ricevitore, invece, nel momento in cui esso viene acceso si pone in ascolto del segnale proveniente dai satelliti "visibili" da quella determinata posizione. Se il ricevitore viene acceso dopo un lungo tempo dall'ultimo utilizzo, si parla di "*cold start*" (avvio a freddo). In questo caso il ricevitore deve creare un "almanacco" della posizione presunta di almeno un satellite e ricevere un segnale orario sufficientemente preciso da cui desumere la posizione degli altri satelliti della costellazione. A questo punto, se le condizioni di ricezione sono favorevoli, il satellite è in grado di creare le "effemeridi", ovvero il posizionamento preciso degli altri tre satelliti che gli sono necessari per calcolare il posizionamento (*fix*), e può cominciare la navigazione. Il tempo per compiere questi calcoli ed effettuare il *fix* è definito con l'acronimo *Ttff* (*Time To First Fix*) e si aggira normalmente attorno a un minuto, sempre che non siano presenti troppi ostacoli e il ricevitore non sia in movimento. Nel caso di una riaccensione del dispositivo dopo poco tempo dall'ultimo spegnimento, si parla invece di "*warm start*" (avvio a caldo). In questo caso il *Ttff* scende in maniera drastica a tempi che vanno da dieci a venti secondi, questo perché le effemeridi e l'almanacco registrato nella precedente sessione di lavoro possono essere ancora ritenuti validi e molte operazioni non devono così essere ripetute.

Il numero e la posizione dei satelliti visibili dal dispositivo influiscono naturalmente sulle operazioni sopradescritte. I dispositivi in commercio sono in grado di "ascoltare" un numero variabile di canali tra 12 e 16 (nessun dispositivo è in grado di visualizzarne 24). Per la determinazione efficace della posizione il dispositivo dovrà essere in grado di operare scelte selettive tra i vari segnali selezionando i migliori e i meglio distribuiti.

Altri sistemi di posizionamento satellitare

La continua evoluzione del sistema, l'innovazione tecnologica dei ricevitori, le nuove metodologie, l'ingresso del rilievo GPS nel Catasto Italiano, hanno contribuito a far crescere l'interesse verso questa tecnologia, facendo entrare questo acronimo (GPS) nel frasario comune. Tuttavia il sistema GPS non è l'unica costellazione di satelliti esistente; vediamo altri sistemi di localizzazione satellitare attualmente presenti sulla Terra.

GLONASS

Esiste un sistema simile al GPS denominato *Glonass* e messo in funzione dalla Russia. Ad una prima analisi Glonass sembra strutturalmente e tecnicamente più limitato del GPS, ma per qualche anno ha retto bene il confronto. Dal 1995 anche la Russia consente l'accesso civile gratuito al proprio sistema di localizzazione satellitare, al punto che esistono appositi ricevitori combinati GPS/Glonass in grado di fornire maggiore affidabilità, tempi di aggancio e di rilevazione della posizione più ridotti e maggiore precisione.

Mentre i satelliti GPS ruotano su 6 piani, 4 per ogni piano, quelli GLONASS sono posizionati su 3 piani con 8 satelliti ognuno ad un'altitudine di circa 19.140 Km. GLONASS non ha nessun degrado della precisione né crittografia dei segnali.

GALILEO

Il sistema di posizionamento *Galileo* è un sistema satellitare globale di navigazione civile sviluppato in Europa come alternativa al Global Positioning System (GPS).

Galileo comprende una costellazione di 30 satelliti suddivisi su 3 piani ad una quota di circa 23.222 Km. E' già in orbita, ma la sua entrata in servizio è prevista solo per il 2013, i principali scopi sono:

- una maggior *precisione* nella geo-localizzazione degli utenti rispetto a quella attualmente fornita dal GPS;
- un'alta *disponibilità* del segnale nelle aree urbane;
- una certificata *affidabilità*, supportata anche dall'invio del messaggio di integrità che avverte immediatamente l'utente di eventuali perdite di integrità nel segnale della costellazione o, viceversa, ne conferma l'esattezza del segnale ricevuto;
- un'elevata *continuità* di servizio che, essendo indipendente dagli USA, potrà funzionare sempre e non verrà disattivato senza preavviso (come successo in tempi di guerra per il GPS).

Sebbene il sistema Galileo sia stato pensato per essere completamente indipendente ed autosufficiente, sarà compatibile ed interoperabile con il sistema GPS; cioè, le caratteristiche di Galileo saranno tali da non interferire con il funzionamento del GPS (principio di *compatibilità*) ed i due sistemi potranno anche essere usati congiuntamente (principio di *interoperabilità*).

BEIDOU

Anche la Cina, a partire dall'anno 2000, ha messo in orbita un proprio sistema di localizzazione satellitare denominato Beidou (*Beidou 1*). Questo sistema opera con principi tecnici molto diversi rispetto a GPS, Glonass e Galileo. I satelliti sono appena quattro e sono geostazionari, quindi le aree coperte dal servizio sono limitate, inoltre, a differenza di un ricevitore GPS che non trasmette nulla, un ricevitore Beidou deve anche trasmettere un potente segnale radio da terra verso i satelliti. Ne conseguono limiti tecnici, pratici ed operativi tali da escludere impieghi civili di massa anche nelle aree coperte dal servizio. Nel 2003 la Cina ha annunciato ulteriori "mosse": da un lato la partecipazione al progetto europeo Galileo con un investimento modesto, dall'altro lato il lancio futuro della costellazione Beidou 2, che ricalcherà i principi tecnico/operativi di GPS, Glonass e Galileo grazie a 35 satelliti.

I limiti del GPS

Ciò che influenza una buona ricezione di un segnale satellitare è in primo luogo l'ambiente circostante. Plastica, gomma, vetro, stoffa, legno ad esempio non schermano in maniera importante il flusso delle onde radio GPS, mentre metallo, cemento, roccia e terreno costituiscono un ostacolo importante. Una via di mezzo è rappresentata dalla vegetazione, che solitamente non costituisce un ostacolo rilevante purché non sia troppo fitta e non sia presente un forte tasso di umidità sul fogliame. Una strada di città stretta e circondata da abitazioni può costituire un ostacolo per il GPS (questa situazione si verifica frequentemente nei centri storici di molte città italiane). Analogamente la ricezione potrebbe risultare difficile in una zona montana dove il punto risulti attorniato da elevate pareti rocciose. Un'altra limitazione è dovuta alle onde radio satellitari riflesse contro gli ostacoli, anche se l'errore dovuto alla ricezione riflessa di uno o due satelliti viene in buona parte compensato qualora l'aggancio complessivo sia di almeno 6 o 7 unità.

Il GPS civile può operare ad una altitudine massima di 18000 metri slm³¹ e su mezzi che non superino i 1999Km/h.

WGS84

La posizione determinata dal GPS è in generale legata a:

- sistema di coordinate terrestri scelto (latitudine/longitudine o cartesiano)
- forma ideale della terra.

Ma qual è la forma della terra ?

Il nostro pianeta ha la geometria approssimativa di un ellissoide. In realtà però ha una forma irregolare, costellata di protuberanze e depressioni, detta **geoide**. Questa figura richiama la forma di una pera, con una prominente al polo Nord e un leggero schiacciamento al polo Sud. L'esatta forma della Terra si è potuta conoscere grazie ai rilevamenti dei satelliti geodetici, che sono in grado di determinare le irregolarità della superficie terrestre con un errore inferiore al metro.

³¹ s.l.m.: sul livello del mare.

Si possono scegliere varie dimensioni per l'ellissoide, valide globalmente o localmente, oppure definire un geoide fornendo le correzioni rispetto ad un modello. Si sceglie così il cosiddetto *datum*.

Si definisce **datum geodetico** un sistema di riferimento che permette di esprimere in termini matematici la posizione di punti della superficie fisica della Terra o prossimi ad essa. In generale è possibile definire un datum in diversi modi; va però osservato che la definizione ha carattere convenzionale, e nella pratica è legata a una serie di punti materializzati sulla superficie terrestre, ai quali vengono attribuiti determinati valori delle coordinate (tale operazione costituisce la cosiddetta *realizzazione* del datum). Nella geodesia satellitare (attualmente basata principalmente sul sistema GPS, e in misura minore sul GLONASS) si utilizzano datum geodetici di tipo *globale*, validi cioè per tutto il mondo.

La definizione di un datum globale è basata su una terna d'assi *OXYZ* geocentrica, avente cioè l'origine coincidente con il centro di massa della Terra. L'asse *Z* coincide con l'asse polare (asse di rotazione della Terra); gli assi *X* ed *Y* giacciono sul piano equatoriale, con l'asse *X* diretto secondo il meridiano fondamentale (quello di Greenwich) e *Y* diretto in modo da completare una terna destrorsa. La terna geocentrica è solidale alla Terra, cioè la segue rigidamente nel suo moto: per questo, tali sistemi vengono denominati ECEF (*Earth Centered Earth Fixed*). Si tratta anche in questo caso di una definizione convenzionale, dato che la posizione del geocentro e la direzione dell'asse polare (quest'ultima variabile nel tempo) devono essere stabilite convenzionalmente.

La definizione adottata attualmente per le applicazioni del GPS è detta **WGS84** (*World Geodetic System 1984*, sistema geodetico mondiale riferito al 1984). Per analogia con i sistemi classici e per rendere più agevole la georeferenziazione di punti (per mezzo delle usuali coordinate geografiche), alla terna cartesiana è associato un ellissoide geocentrico, avente centro coincidente con quello della terna ed assi orientati secondo le direzioni *XYZ*.

In Italia, la rete geodetica che realizza il datum WGS84 è la IGM95, determinata dall'Istituto Geografico Militare con misure GPS eseguite intorno alla metà degli anni '90.

NMEA 0183

La *National Marine Electronics Association* (NMEA) è una associazione di produttori, distributori, rivenditori e scuole che si occupano dell'elettronica in campo nautico. NMEA definisce una serie di standard, fra cui NMEA 0183, uno standard di interfacciamento tra apparecchiature digitali. Il sistema ha origini e finalità prevalentemente nautiche e viene utilizzato anche per sistemi di autopilota di imbarcazioni. I formati di dati del sistema NMEA sono molto numerosi e solo una parte molto limitata ha rilevanza nell'ambito del GPS, dove viene prevalentemente impiegato per trasmettere dati da un ricevitore (GPS) verso un computer.

I dati (detti "*sentences*" ovvero "frasi") operano su un bus dati seriale (tipicamente RS232) a 4800 bps.

Struttura delle frasi

Tutte le frasi NMEA sono costituite da una sequenza di caratteri ASCII. Sono formate da un prefisso, una serie di campi ed un checksum finale (è posto dopo un asterisco ed è calcolato come XOR ad 8 bit di tutti i caratteri che compongono la frase). L'inizio della frase è delimitato da un carattere "\$", la fine da una sequenza CR LF ed i singoli campi sono separati da virgole.

Esempio:

\$PREFISSO, campo, campo, ..., campo, *checksumCRLF

Una frase NMEA può contenere fino a 82 caratteri compresi "\$" e CR LF.

Se i dati per un campo non sono disponibili, il campo viene omissso, ma le virgole che lo delimiterebbero sono comunque aggiunte, senza spazi tra di loro.

Nell'ambito quindi del GPS, tutte le frasi iniziano con "\$GP", dove "GP" denota che la frase è generata da un dispositivo GPS. I successivi tre caratteri del prefisso indicano il tipo di frase. Per esempio una frase che inizia con \$GPGLL è inviata da un dispositivo di tipo GPS (GP) ed è del tipo Geographic position, Longitude and Latitude (GLL).

Si vogliono ora elencare le principali frasi NMEA, in particolare quelle utilizzate per la realizzazione del progetto di tesi.

\$GPRMC - Raccomanded Minimum Specific GPS/TRANSIT Data

Contiene dati essenziali relativi a data/ora, posizione, velocità e un minimo relativo alla qualità della rilevazione.

Campo	Formato	Esempio	Descrizione
1	hhmmss.ss	125642.57	Ora UTC-GPS
2	A	A	Stato: A=Active, Attivo; V=Void, Nullo (per ricordarne il significato: è come una freccia in su/giù)
3	llll.ll	4512.72	Latitudine della posizione attuale (numero variabile di decimali)
4	A	N	Emisfero della posizione attuale: N=Nord; S=Sud
5	yyyy.yy	00934.88	Longitudine della posizione attuale (numero variabile di decimali)
6	A	E	Verso della posizione attuale: E=Est; W=Ovest
7	x.x	023.45	Velocità al suolo, in nodi
8	x.x	358.20	Track Made Good - Direzione di movimento, in gradi reali
9	ddmmyy	291105	Data
10	x.x	1.2	Variazione / declinazione magnetica, in gradi
11	A	E	Verso della variazione / declinazione magnetica
12	A	D	Tipo di rilevazione: A=Autonomous, D=Differential, E=Estimated, N=Non valid data (dalla versione 2.3)

Tab. 5 - NMEA con prefisso "GPRMC"

\$GPGGA - Global Positioning System fix data

Frase che comprende dati relativi al fix (3D) e all'uso della correzione differenziale.

Campo	Formato	Esempio	Descrizione
1	hhmmss.ss	125642.57	Ora UTC - GPS
2	llll.ll	4512.72	Latitudine della posizione attuale (numero di decimali variabile)
3	A	N	Emisfero della posizione attuale: N=Nord; S=Sud
4	yyyy.yy	00934.88	Longitudine della posizione attuale (numero di decimali variabile)
5	A	E	Verso della posizione attuale: E=Est; W=Ovest
6	n	1	Qualità del rilevamento GPS: 0=non valido; 1=GPS; 2=DGPS
7	x	8	Numero di satelliti in vista (teorica, non necessariamente in uso o ricevuti)
8	x.x	1.9	HDOP - Horizontal Dilution Of Precision
9	x.x	540.4	Altitudine dell'antenna GPS relativa al livello medio del mare (geoide)
10	A	M	Indicatore dell'unità di misura dell'altitudine M=metri (ma è mai diverso da M?)
11	x.x	45.8	Altezza (separazione) del geoide (livello medio del mare) rispetto all'elissioide WGS84
12	A	M	Indicatore dell'unità di misura della separazione del geoide M=metri (ma è mai diverso da M?)
13	x.x	8.0	Tempo in secondi dall'ultimo aggiornamento DGPS
14	mmnn	1023	Identificatore della stazione DGPS (0000-1023)

Tab. 6 - NMEA con prefisso "GPGGA"

\$GPGSA - GPS DOP and Satellites Active

Frase che comprende dati relativi alla qualità del fix e l'indicazione dei satelliti ricevuti (o di cui i vari canali stanno tentando la ricezione).

Il DOP (*Dilution Of Precision*) è un indice (stimato) di diluizione della precisione che serve a capire quale potrà essere l'accuratezza della misura in presenza di una ben determinata configurazione satellitare. L'indice varia da 0 (ottimo) a 10 (pessimo).

Campo	Formato	Esempio	Descrizione
1	A	M	Tipo di selezione del tipo di rilevamento (2D o 3D): M=Manuale; A=Automatico
2	n	3	Tipo di rilevamento: 2=2D (bidimensionale); 3=3D (tridimensionale)
3	nn	03	Identificatore (SV = PRN) di satellite ricevuto
4	nn	12	Identificatore (SV = PRN) di satellite ricevuto
...			
...			
14	nn	01	Identificatore (SV = PRN) di satellite ricevuto
15	x.x	1.9	PDOP - Position Dilution Of Precision
16	x.x	1.2	HDOP - Horizontal Dilution Of Precision
17	x.x	2.2	VDOP - Vertical Dilution Of Precision

Tab. 7 - NMEA con prefisso "GPGSA"

\$GPGLL - Geographic position - Latitude/Longitude

Frase minima, contiene solo dati posizionali, ora e validità del fix.

Campo	Formato	Esempio	Descrizione
1	llll.l1	4512.72	Latitudine della posizione attuale (numero di decimali variabile)
2	A	N	Emisfero della posizione attuale: N=Nord; S=Sud
3	yyyyy.yy	00934.88	Longitudine della posizione attuale (numero di decimali variabile)
4	A	E	Verso della posizione attuale: E=Est; W=Ovest
5	hhmmss.ss	125642.57	Ora UTC GPS
6	A	A	Stato: A=Active, Attivo; V=Void, Nullo (freccia in su/giù)

Tab. 8 - NMEA con prefisso "GPGLL"

\$GPGSV - Satellites in View

Indica dati relativi a tutti i satelliti che il ricevitore, in base ai propri dati di almanacco, ritiene di poter ricevere. Di ciascun satellite vengono indicati: identificatore, azimuth, elevazione e livello di segnale ricevuto.

Campo	Formato	Esempio	Descrizione
1	n	3	Numero di frasi del gruppo
2	n	1	Numero progressivo della frase nel gruppo (1 di 3 nell'esempio)
3	nn	09	Numero totale di satelliti teoricamente ricevibili
4	nn	03	Identificatore (SV = PRN) di satellite ricevuto del primo dell'elenco
5	nn	85	Elevazione in gradi sull'orizzonte del primo satellite dell'elenco
6	nnn	224	Azimuth in gradi del primo satellite dell'elenco
7	nn	37	Livello del segnale ricevuto dal primo satellite dell'elenco
8	nn	09	Identificatore (SV = PRN) di satellite ricevuto del secondo dell'elenco
9	nn	125	Elevazione in gradi sull'orizzonte del secondo satellite dell'elenco
10	nnn	34	Azimuth in gradi del secondo satellite dell'elenco
11	nn	24	Livello del segnale ricevuto dal secondo satellite dell'elenco
...			
16	nn	31	Identificatore (SV = PRN) di satellite ricevuto del quarto dell'elenco
17	nn	17	Elevazione in gradi sull'orizzonte del quarto satellite dell'elenco
18	nnn	120	Azimuth in gradi del quarto satellite dell'elenco
19	nn	42	Livello del segnale ricevuto dal quarto satellite dell'elenco

Tab. 9 - NMEA con prefisso "GPGSV"

Vi sono inoltre altre frasi che sono però proprietarie del costruttore. Ad esempio *Garmin* (dispositivo utilizzato per questa tesi) è proprietaria delle seguenti:

- \$PGRME - Estimated Position Error
- \$PGRMF - Position Fix Sentence
- \$PGRMM - Map Datum
- \$PGRMV - Velocity Sentence
- \$PGRMZ - Altitude Information
- \$PSLIB - Differential Control

In particolare, è stata utilizzata la frase "\$PGRME" per il calcolo dell'errore di posizione. Questa frase contiene i campi: HPE (*Horizontal Position Error*) e VPE (*Vertical Position Error*), che rappresentano una stima sull'errore espressa in metri.

Nei capitoli che seguono verrà descritto come utilizzare tali stringhe al fine di poterne estrarre le informazioni che servono per il raggiungimento degli obiettivi di questa tesi.

Conversione di dati seriali per trasmissione su rete TCP/IP

L'obiettivo di questa tesi è quello di realizzare un sistema di localizzazione attraverso l'utilizzo di un'interfaccia GPS. In generale, occorre elaborare i dati forniti da quest'ultima in modo da renderli disponibili in rete. Come descritto nel capitolo 2, i dati in uscita dal GPS sono nel formato NMEA (operanti su bus dati seriali), quindi per essere inviati su rete TCP/IP dovranno subire una conversione. LS100 è il dispositivo utilizzato in questa tesi.

HelloDevice LS100

Prodotto dalla SENA, LS100 (Fig. 196) è un convertitore Seriale/Ethernet basato sul protocollo TCP/IP che permette di collegare dispositivi seriali attraverso Ethernet o Internet. E' costituito da un connettore DB9 che permette il collegamento del dispositivo seriale da interfacciare e da un connettore RJ45 per la connessione Ethernet.



Fig. 196 - Convertitore LS100

Il dispositivo permette di configurare, monitorare e diagnosticare a basso costo le apparecchiature seriali (Fig. 197). Può lavorare nelle seguenti modalità:

- TCP server
- TCP client
- TCP server/client.

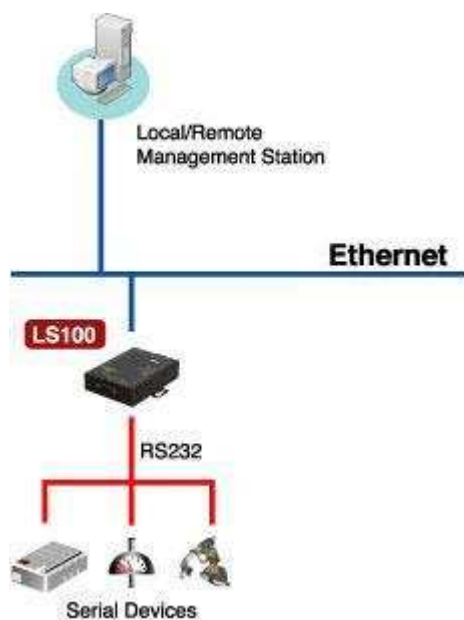


Fig. 197 - Funzionalità LS100

Caratteristiche tecniche

Di seguito sono riportate le caratteristiche del convertitore fornite dal costruttore:

Interfaccia seriale	1-porta RS232 Maschio DB9 Velocità di trasmissione seriale: da 1200bps a 115Kbps Controllo di flusso: Hardware RTS/CTS Segnali: Rx, Tx, RTS, CTS, DTR, DSR, GND
Interfaccia di rete	10Base-T Ethernet con connettore Ethernet RJ45 Supporto di indirizzi IP statici e dinamici
Protocolli	ARP, IP/ICMP, TCP/IP, Telnet, DHCP client, PPPoE
Sicurezza	User ID & Password
Gestione	Telnet, porta console seriale, HelloDevice Manager
Led di diagnostica	Power Ready 10 Base-T Link, Act Serial Rx/Tx
Alimentazione	Tensione = 7.5V ~ 15V DC Corrente = 300mA @ 7.5V DC
Ambiente	Temperatura di esercizio : 0° ~ 50 °C Temperatura di stoccaggio: -20° ~ 66 °C
Dimensioni e Peso	Dimensione: 100 mm L x 72 mm W x 25 mm H Peso: 240g
Approvazioni	FCC(A), CE(A), MIC

Tab. 10 - Caratteristiche tecniche LS100

La gestione delle porte è la seguente:

Port number	Protocol	TCP/UDP
21	FTP (File Transfer Protocol)	TCP
22	SSH (Secure SHell)	TCP
23	<u>Telnet</u>	TCP
25	SMTP (Simple Mail Transfer Protocol)	TCP
37	Time	TCP, UDP
39	RLP (Resource Location Protocol)	UDP
49	TACACS, TACACS+	UDP
53	DNS	UDP
67	BOOTP server	UDP
68	BOOTP client	UDP
69	TFTP	UDP
70	Gopher	TCP
79	Finger	TCP
80	HTTP	TCP
110	POP3	TCP
119	NNTP (Network News Transfer Protocol)	TCP
161/162	SNMP	UDP

Tab. 11 - Gestione porte LS100

Configurazione

Ci sono due modi per accedere ai parametri di configurazione del convertitore LS100:

- connessione seriale (attraverso porta COM)
- connessione remota (attraverso porta TCP).

Nel caso in cui non si conosca l'indirizzo IP della periferica è possibile effettuare una connessione di tipo seriale in modo da impostare un indirizzo IP utilizzabile per la connessione remota.

Configurazione mediante connessione seriale

La configurazione seriale si esplica nei seguenti passi:

1. collegare, tramite cavo seriale, il PC al convertitore
2. impostare lo switch Data/Console su Console (Fig. 198)

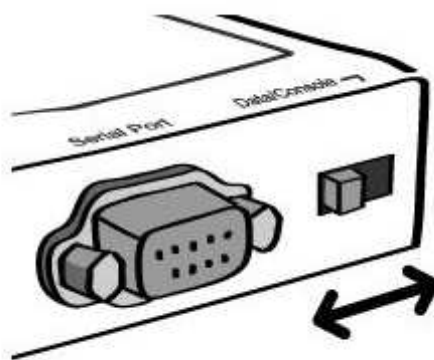


Fig. 198 - Data/Console switch LS100

3. aprire un emulatore di terminale (es. HyperTerminal)
4. impostare i parametri come in Fig. 199
5. inserire username e password (di default entrambi *admin*).



Fig. 199 - Parametri di accesso al convertitore LS100

Effettuato il login sarà possibile accedere ai parametri del convertitore e modificarne i valori.

Configurazione mediante connessione remota

Gli utenti possono accedere in remoto (attraverso una rete TCP/IP) al convertitore LS100. Utilizzando HyperTerminal, o qualunque altro client TELNET, è possibile stabilire una connessione (attraverso la porta 23, indicata in “Tabella 3.4”) con il convertitore³² ed impostare, come nel caso seriale, i parametri di configurazione. Dopo aver effettuato il login è possibile digitare il comando help (Fig. 200) per visionare l’aiuto alla configurazione.

```
login: admin
password: *****
type 'help' to get command usages
> help
set group par1 [par2 ...] + <CR>
- group = 'ip', 'host', 'serial', 'locatingsvr' or 'admin'
- par1 ... = configuration parameters. Use * to keep a parameter's value
get [group] + <CR>
- group = 'ip', 'host', 'serial', 'locatingsvr', 'admin' or 'status'
- If group is specified, shows settings of the group.
- If group is omitted, shows settings of all groups.
help [group] + <CR>
- If group is omitted, shows this screen.
- If group is specified, shows 'set' command usage of the group.
factorydefault [option] + <CR>
- if option is omitted, all parameters are set with factory default values.
- if option = '-ip',
  all parameters except IP settings are set with factory default values.
save + <CR>
- Save changes
exit + <CR>
- Exit without rebooting the device
reboot + <CR>
- Exit and reboot the device

>
```

Fig. 200 - Login LS100

Parametri di configurazione per il collegamento di un'interfaccia GPS

Collegiamo ora al convertitore il dispositivo seriale (in questo caso un'interfaccia GPS) come in Fig. 201.

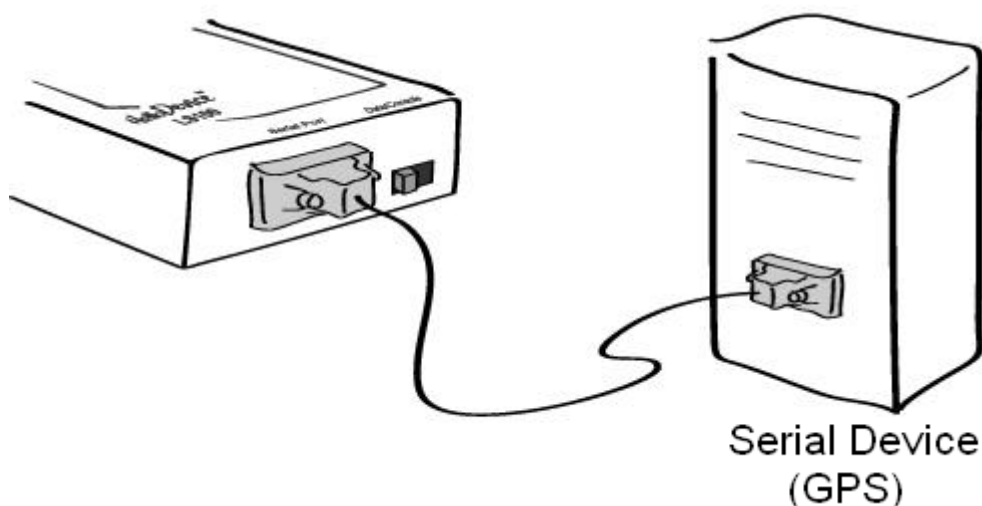


Fig. 201 - Connessione della periferica seriale al convertitore LS100

³² Come figura 3.11 sostituendo il valore 6001 con 23 (porta di configurazione).

La configurazione necessaria per l'acquisizione dati dall'interfaccia GPS è la seguente:

→ Configurazione indirizzo IP LS100:

```
> set ip static 192.168.1.100 255.255.255.0 192.168.1.1
```

Si è così configurato:

- Indirizzo del convertitore: 192.168.1.100 (di tipo statico)
- Subnet mask: 255.255.255.0
- Gateway: 192.168.1.1

Per visualizzare le impostazioni relative all'ip:

```
> get ip (Fig. 203)
```

→ Configurazione HOST:

```
> set host tcpc 192.168.1.50 6001 10 300
```

Si è così configurato:

- Host mode: TCP client (Fig. 202)
- IP di destinazione: 192.168.1.50 (l'indirizzo del server)
- Porta di destinazione: 6001
- Cyclic connection interval (min): 10
- Inactivity timeout (sec): 300

Per visualizzare le impostazioni relative all'host:

```
> get host (Fig. 203)
```

L'interfaccia GPS assume il ruolo di client, il server dovrà mettersi in ascolto sulla porta (in questo caso è stata scelta la 6001) e accettare le eventuali richieste di comunicazione.

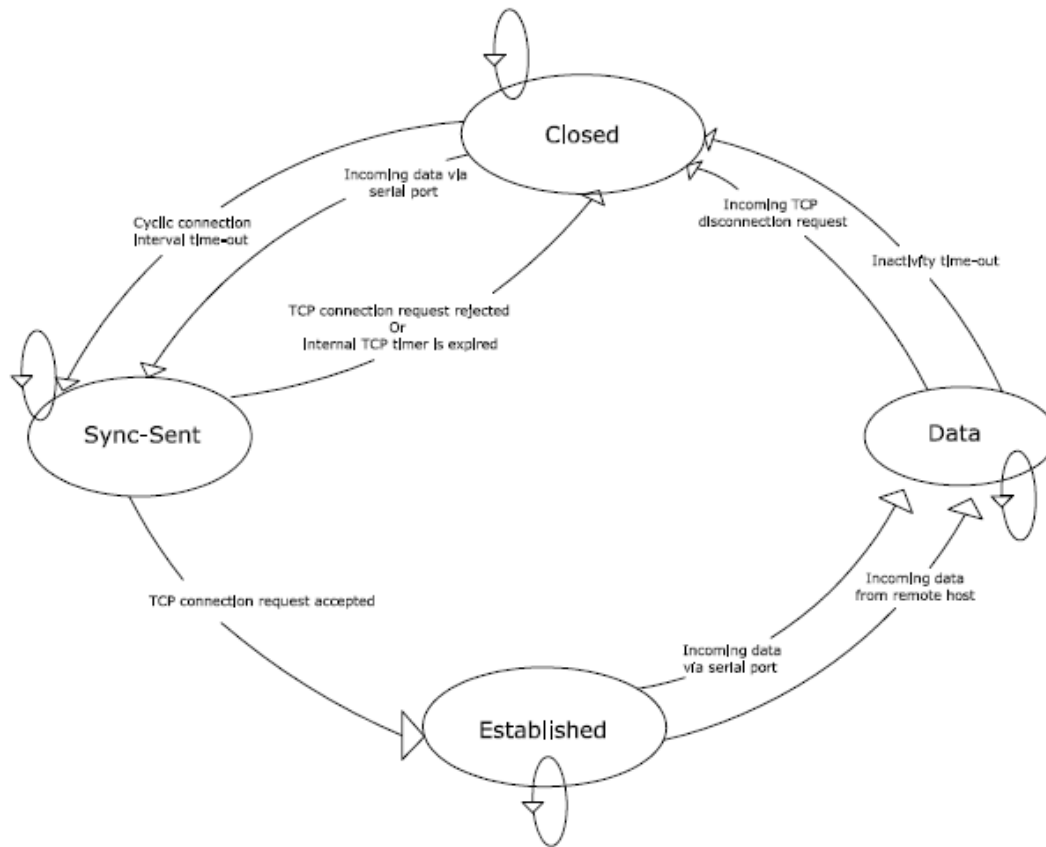


Fig. 202 - Diagramma degli stati relativo alla modalità TCP client

→ Configurazione porta SERIALE:

```
> set serial 4800 n h n 10
```

Si è così configurato:

- Baudrate: 4800 (pari a quella di lavoro dell'interfaccia GPS)
- Flow control: None
- DTR: sempre alto
- DSR: None
- Interchart timeout (ms): 10

Per visualizzare le impostazioni relative alla porta seriale:

```
> get seriale
```

→ Salvare e uscire:

```
> save
```

OK

```
> reboot
```

In generale è possibile utilizzare il comando `get` per visualizzare tutti i parametri impostati.

```
> get ip
IP_mode: static
IP_address: 192.168.1.100
Subnet_mask: 255.255.255.0
Gateway: 192.168.1.1
> get host
Host_mode: tcp
Destination_IP: 192.168.1.50
Destination_port: 6001
Cyclic_connection_interval(min): 10
Inactivity_timeout(sec): 300
>
```

Fig. 203 - get ip & get host

Conversione RS232/Ethernet

Si vuole ora testare il corretto funzionamento del collegamento utilizzando i parametri visti nel Paragrafo 3.1.3. Effettuando l'accesso con un client Telnet (es. HyperTerminal) è possibile instaurare una connessione TCP/IP con l'interfaccia GPS (Fig. 204).

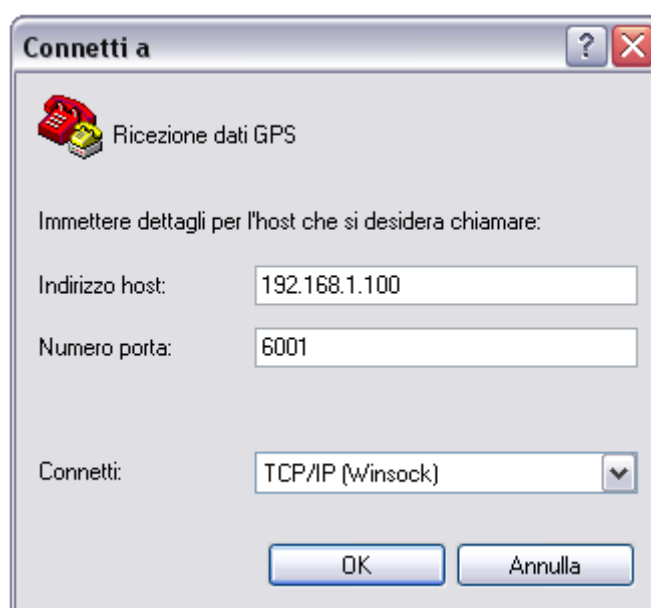


Fig. 204 - Connessione remota all'interfaccia GPS

Per accedere al convertitore LS100 ed effettuare la lettura dati occorre impostare come indirizzo IP 192.168.1.100 (quello del convertitore) e come porta la 6001. Occorre ora attendere il collegamento del client selezionando l'opzione *Attendi chiamata* (Fig. 205). Instaurata la connessione, i dati saranno ricevuti dal Server e stampati a video (Fig. 206).

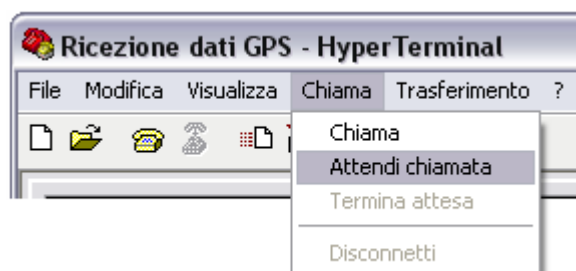


Fig. 205 - Il server attende la chiamata del client

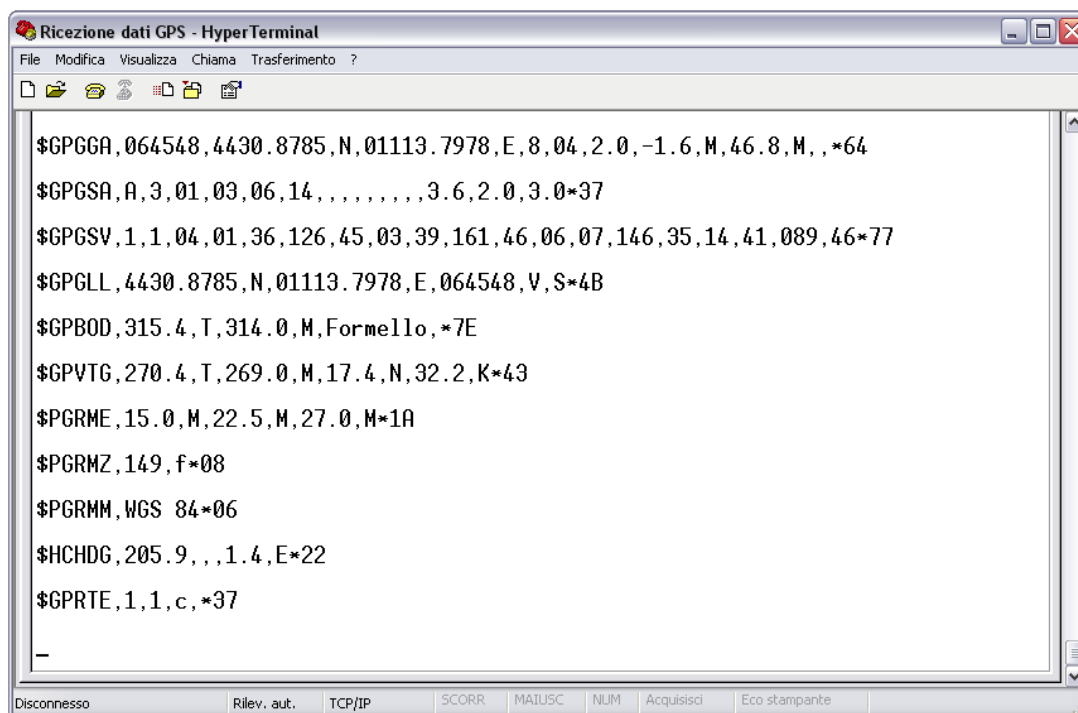


Fig. 206 - Ricezione dati GPS da client Telnet (HyperTerminal)

In Fig. 206 si possono osservare le frasi NMEA in uscita dall'interfaccia GPS. Queste sono state convertite in pacchetti IP dal dispositivo LS100 e inviate al server (che in questo caso è un pc collegato attraverso un cavo di rete alla porta ethernet del convertitore).

Acquisizione dati da interfaccia GPS e conversione mediante LS100

Verificato il corretto funzionamento della trasmissione è possibile procedere all'elaborazione dei dati ottenuti.

La porta seriale del convertitore LS100 è collegata all'interfaccia GPS, i dati generati da quest'ultimo (NMEA sentences) vengono così convertiti in pacchetti IP, inviati al server ed infine elaborati dal software. Questo esegue uno scan dei dati ricevuti selezionando esclusivamente le stringhe che dovranno essere inserite in un database. Attraverso lo stesso sarà poi possibile utilizzare i dati salvati per operazioni che saranno analizzate più avanti.

Fig. 207 - Banco di lavoro

L'ambiente di sviluppo utilizzato per l'implementazione del software è *Eclipse*, progetto open source disponibile per le piattaforme Linux, HP-UX, AIX, Mac OS X e Windows.

Implementazione del software

Si vuole ora analizzare il software, realizzato in linguaggio di programmazione Java, relativo all'acquisizione dati dall'interfaccia seriale. In primo luogo, vengono descritti i procedimenti necessari per creare una connessione tra client e server atta a realizzare lo scambio dati. Successivamente verranno invece descritte le istruzioni necessarie per memorizzare i dati ricevuti nel database.

Il linguaggio Java, tramite l'utilizzo di particolari classi incluse nel suo package nativo, permette la gestione del flusso di dati su reti basate sul protocollo TCP/IP (e UDP/IP).

Le socket

Con il termine *socket* (che letteralmente vuol dire *presa*) si definisce una rappresentazione a livello software utilizzata per interfacciare due terminali (endpoint) in gioco in una connessione. Esse forniscono quindi un'astrazione, rispetto a meccanismi di livello più basso, per la comunicazione in rete. Attraverso una socket un'applicazione può inviare dati sulla rete e leggere dati in ingresso.

Esistono due modi principali per comunicare in rete:

- **Connectionless:** non viene garantita l'effettiva consegna del singolo pacchetto, né l'ordine temporale di arrivo.
- **Connection oriented:** assicura la corretta ricezione di ogni singolo pacchetto.

In corrispondenza di tali modi si hanno due tipi di socket:

- **Datagram socket:** trasferiscono i dati senza garantire ordine o arrivo dei pacchetti. Non instaurano una connessione tra client e server e non effettuano il re-invio dei dati nel caso in cui quest'ultimi non siano giunti a destinazione. Hanno il vantaggio di trasferire velocemente i dati. Sono supportate nel dominio internet dal protocollo UDP (*User Datagram Protocol*).
- **Stream socket:** forniscono stream di dati affidabili e ordinati. Nel dominio internet sono supportati dal protocollo TCP (*Transmission Control Protocol*) (Fig. 208).

Una socket mantiene due flussi di dati: input e output. In particolare, un processo invia dei dati ad un altro processo attraverso la rete scrivendo sullo stream di output associato ad una socket; un altro processo accede ai dati scritti in precedenza leggendo dallo stream di input della socket stessa.

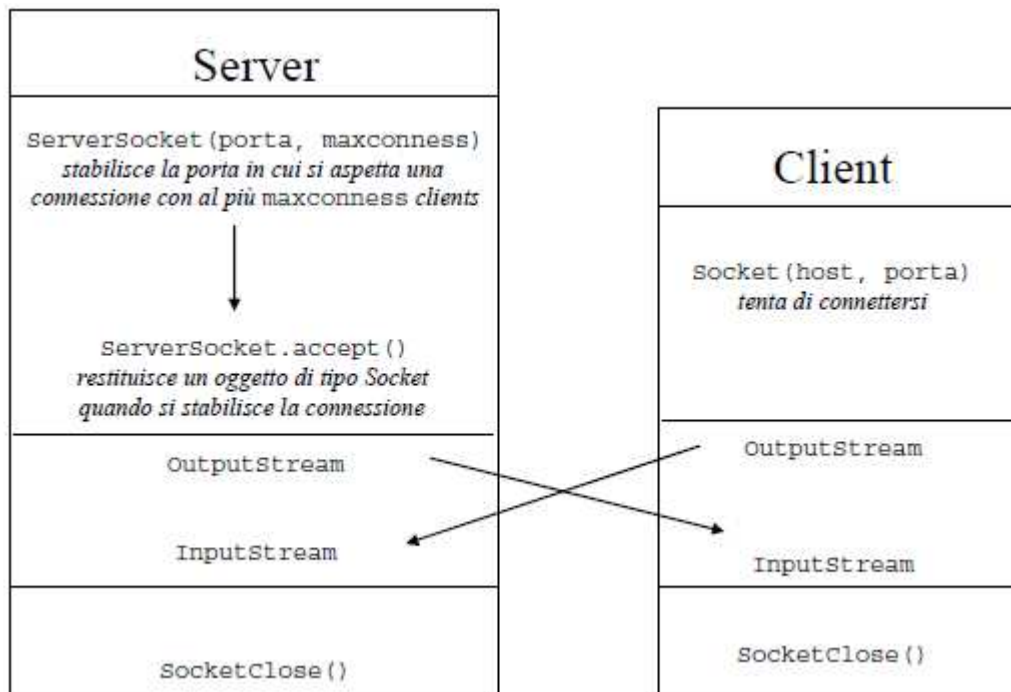


Fig. 208 - Schema generale di una stream socket server-client

Le classi utilizzate

Il concetto di classe è alla base della programmazione orientata agli oggetti, tipologia di sviluppo di cui il linguaggio Java fa parte. Una classe rappresenta una descrizione di una categoria di oggetti, individuati da comportamenti e caratteristiche simili. I comportamenti vengono definiti "metodi" mentre le caratteristiche vengono definite "proprietà". Le classi in Java sono quindi utilizzate per definire la struttura ed il comportamento di un determinato oggetto.

Per poter effettuare la lettura dati dall'interfaccia GPS occorre quindi definire degli oggetti che permettano accedere allo stream di una socket.

Le principali classi necessarie per la realizzazione del progetto sono:

- InetAddress
- ServerSocket.
- Socket.
- BufferedReader.

La classe **InetAddress** fornisce informazioni relative agli indirizzi IP in gioco nella comunicazione, in particolare permette di ricavare l'indirizzo alla quale la socket è connessa. Nel caso di più client, ad esempio, risulterà utile conoscere la provenienza dei pacchetti.

Per monitorare il trasporto pubblico occorre poter tracciare su mappa tutti i punti rappresentanti gli autobus che circolano in un determinato momento nella città. Quindi occorrerà creare un database in cui, oltre l'informazione "posizione", sarà presente anche l'informazione "IP", la quale permetterà di associare ad ogni mezzo un particolare flusso dati, rappresentante appunto il percorso effettuato.

Le due classi **Socket** e **ServerSocket** sono quelle che intervengono nell'implementazione di una comunicazione client/server basata sulle socket. La prima si occupa della gestione del client che effettua la chiamata mentre la seconda fornisce le funzionalità necessarie a creare un server che rimanga in ascolto su una particolare porta.

La classe **BufferedReader** ci consente di andare a leggere il contenuto dello stream ricavato semplicemente utilizzando una serie di chiamate successive al metodo `readLine()`, ovvero riga per riga.

Per realizzare un canale di comunicazione client/server occorre far girare un programma che resti in ascolto su una determinata porta e che accetti una connessione non appena il client ne farà richiesta.

Quando il client si connette a tale porta, il server ottiene una socket per mezzo della quale i due terminali possono comunicare. Java mette a disposizione la classe **ServerSocket**, tramite cui il server può accettare connessioni dai client attraverso la rete. I passi tipici di un server saranno quindi:

- creare un oggetto di classe **ServerSocket** specificando un numero di porta locale, cioè la porta in cui il server rimarrà in ascolto di richieste di connessioni; attendere (tramite il metodo `accept()` di suddetta classe) connessioni dai client;
- usare una socket ottenuta ad ogni connessione per comunicare con il client.

Il metodo `accept()` della classe **ServerSocket** crea infatti un oggetto **Socket** per ogni connessione. Il server potrà poi comunicare estraendo gli stream di input ed output dalla socket.

Per utilizzare le classi elencate occorre importare i package `java.io` e `java.net`.

Le istruzioni

Occorre ora applicare quanto enunciato nei paragrafi precedenti ad uno scenario in cui il client sia un'interfaccia GPS che trasmette dati al SERVER (Fig. 209), il quale dovrà gestire un software in grado di:

- acquisire i dati in uscita dal client
- elaborarli
- inserirli in un database.

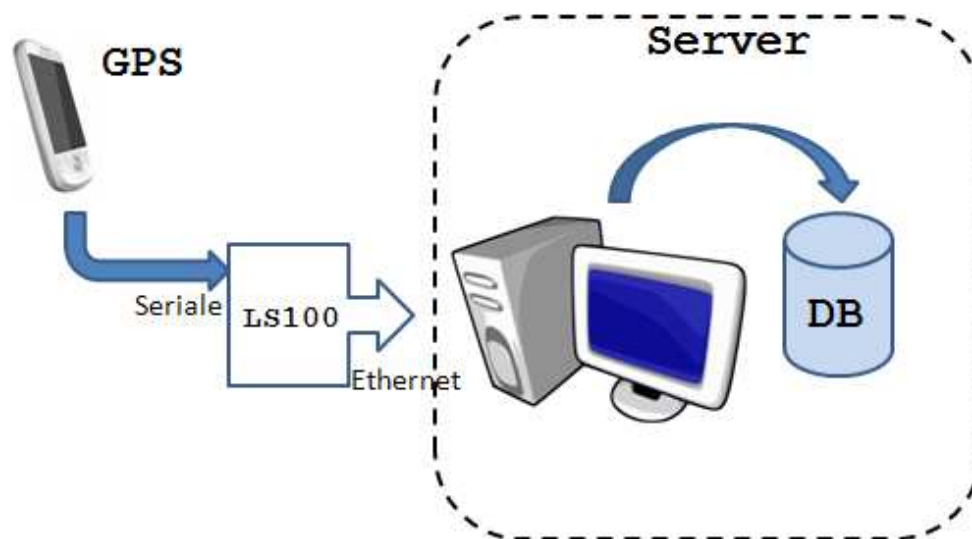


Fig. 209 - Schema di collegamento

Connessione e acquisizione dati

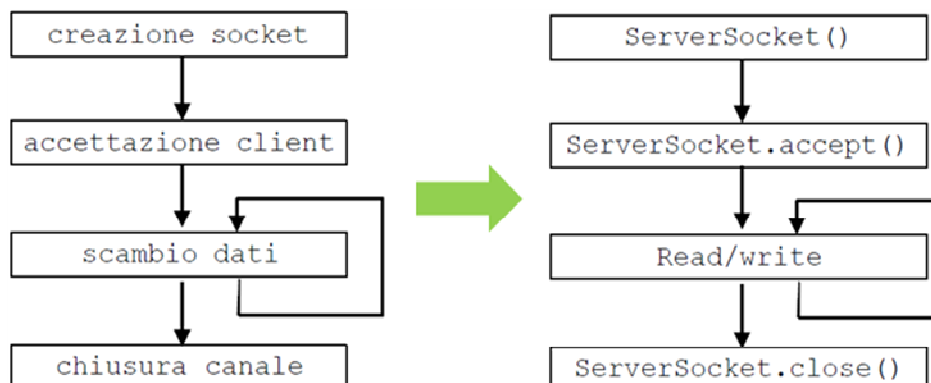


Fig. 210 - Schema di connessione

La realizzazione del programma che instaura una connessione tra il client (GPS) ed il server può essere articolata nei seguenti punti (Fig. 210):

- Creare un oggetto `ServerSocket`, utilizzando il costruttore

`ServerSocket(int port)` che si aspetta come parametro la porta del Server in ascolto per il servizio da attivare:

```
int porta = 6001;
ServerSocket server = new ServerSocket(porta);
```

- Utilizzare il metodo `accept()` dell'oggetto `ServerSocket` per poter accettare le chiamate client su un oggetto `Socket`:

```
Socket connection = server.accept();
```

Il metodo che viene invocato ritorna un oggetto `Socket` che potrà essere utilizzato per le comunicazioni con il client.

- Creare l'oggetto `InputStreamReader` per leggere le richieste del client:

```
BufferedReader is = new BufferedReader(new  
    InputStreamReader(connection.getInputStream()));
```

- Richiamare il metodo `close()` sugli oggetti dello stream, `Socket` e `ServerSocket` per chiudere i canali di comunicazione:

```
is.close();  
connection.close();  
server.close();
```

Server e connection vanno definite a inizio programma nel seguente modo:

```
ServerSocket server;  
Socket connection;
```

Le istruzioni di connessione potrebbero generare delle eccezioni, devono quindi essere opportunamente gestite tramite le `try` e `catch`:

```
try{  
    ServerSocket server = new ServerSocket(porta);  
    .  
    .  
    .  
}  
  
catch (IOException e){  
    System.out.println("Errore :"+e.getMessage());  
}
```

Le istruzioni che ritornano l'indirizzo IP alla quale la socket è connessa sono:

```
InetAddress ip_inet=connection.getInetAddress();  
String ip = ip_inet.getHostAddress();
```

Il software deve inoltre contenere un ciclo, infinito o condizionato dalla chiusura del programma, in cui viene atteso il collegamento del client.

Elaborazione dati

Stabilita la connessione, il server sarà in grado di recuperare il flusso dati in uscita dall'interfaccia GPS. L'istruzione che permette di leggere una linea dallo stream della socket è:

```
String stringa = is.readLine();
```


In Fig. 206 è visualizzato il flusso dati ricevuto dal server. Il software dovrà effettuare un ciclo di lettura (infinito o che termini alla chiusura del programma) che permetta di selezionare esclusivamente i dati desiderati e di inserirli nel database.

L'istruzione necessaria per leggere i dati dall'interfaccia GPS è:

```
StringTokenizer stringa;  
stringa = new StringTokenizer(is.readLine(), ",");
```

Si vuole ora analizzare una linea in cui sono presenti informazioni “utili” per capire come selezionare ed interpretare tali dati. Supponiamo che la stringa sia la seguente:

\$GPGGA,065756,4430.9363,N,01103.3131,E,8,04,2.0,-1.8,M,47.0,M,,*62

Per selezionare solo alcuni campi della stringa occorre suddividere le informazioni ricevute. La classe Java che permette la suddivisione di stringhe in più *token* è la `StringTokenizer` contenuta nel package `java.util`. Il costruttore accetta due argomenti: il primo è la stringa da analizzare, il secondo è il carattere delimitatore. La `StringTokenizer` suddivide così l'intera linea in più token, dove ogni stringa compresa tra due “,” ne rappresenta uno.

Obiettivo di questa tesi è estrarre dalla frase “\$GPGGA” le informazioni relative a³³:

- UTC
- Latitudine (NORD o SUD)
- Longitudine (OVEST o EST)
- Numero di satelliti collegati
- Altitudine

A tal fine occorre implementare il seguente codice:

```
while (true)
{
    stringa = new StringTokenizer(is.readLine(), ",");

    if(stringa.nextToken().equals("$GPGGA"))
    {
        utc=stringa.nextToken(",");           //utc
        lat=stringa.nextToken(",");           //latitudine
        sn=stringa.nextToken(",");           //SUD o NORD
        lon=stringa.nextToken(",");           //longitudine
        we=stringa.nextToken(",");           //OVEST o EST
        stringa.nextToken();
        nsatc=stringa.nextToken(",");         //num.sat. collegati
    }
}
```

³³ Vedi CAPITOLO 2 per dettagli sulla posizione e la descrizione dei singoli campi.

```
stringa.nextToken();  
alt=stringa.nextToken(",");          //altitudine  
  
do{  
    stringa = new StringTokenizer(is.readLine(),",");  
    }while(stringa.nextToken().equals("$GPGSA")==false);  
    .  
    .  
    .  
}
```

Il ciclo finale permette di passare alla linea contenente il dato utile successivo (in questo caso è quella con prefisso “GPGSA”).

Di notevole importanza risulta essere l’ordine secondo il quale si vanno ad analizzare le stringhe ricevute. Gli “if” devono essere gestiti in maniera tale da poter inserire in ogni record del database i dati relativi allo stesso pacchetto. Non bisogna cioè creare confusione tra token di pacchetti differenti.

L’obiettivo finale è quindi quello di analizzare ogni pacchetto ricevuto dall’interfaccia GPS e selezionare esclusivamente i dati che si desidera inserire nel database.

Interfacciamento al database

La finalità del progetto è quella di acquisire la posizione (latitudine e longitudine) di un mezzo pubblico che si muove su strada, memorizzarla e renderla disponibile per un tracking relativo al percorso del mezzo stesso. Occorre allora creare una struttura in grado di ospitare i dati precedentemente elaborati e renderli disponibili per una successiva lettura in rete.

In rapporto ad uno scenario reale avremo una situazione così articolata:

un GPS presente su un mezzo pubblico deve fornire i dati relativi alla sua posizione alla stazione di controllo; a tal fine occorre inoltrare quest’ultimi su una rete TCP/IP che collega via wireless stazione e mezzo.

Il dispositivo LS100 si occupa dell’immissione dei pacchetti sulla rete; poiché questi non possono essere direttamente inoltrati nel formato NMEA il suo impiego è indispensabile (come si è potuto constatare nei paragrafi precedenti). I dati, una volta ricevuti dal server (in questo caso la stazione di controllo), dovranno subire l’elaborazione enunciata nel Paragrafo 4.1.3.2 e successivamente essere memorizzati in un database. Il sito web del progetto TRIP a questo punto potrà prelevare le informazioni disponibili ed utilizzarle per il tracciamento del mezzo sulla mappa o per fornire all’utente informazioni relative alla viabilità.

I database sono delle strutture nelle quali è possibile memorizzare grandi quantità di informazioni, che possono poi essere lette attraverso linguaggi di scripting come il PHP. Il punto di forza di un database sta nella velocità con cui le informazioni vengono trovate. Dato il loro largo impiego ne esistono diversi tipi a seconda delle esigenze, fra questi quello attualmente più diffuso è sicuramente il *modello relazionale*, che consente di memorizzare i dati all’interno di tabelle.

WampServer

La realizzazione di un database che permetta quanto sopra enunciato necessita l’installazione di vari software che operino tra di loro: un webserver, PHP e un database MySQL. Una soluzione semplice ma efficace a questo problema è rappresentata da WampServer.

WampServer è un acronimo con cui si indica una piattaforma di sviluppo web/database che prende il nome dalle iniziali dei componenti software con cui è realizzata.

Deriva da Wamp, di cui ne è l'evoluzione, è gratuito e libero ed è rilasciato sotto la GNU (*General Public License*).

Prerequisito:

- ❖ **W:** Windows

Componenti installati:

- ❖ **A:** Apache
- ❖ **M:** MySQL
- ❖ **P:** PHP

Wamp e WampServer sono la versione Windows della piattaforma AMP, così come LAMP è quella adatta per la GNU/Linux.

WampServer offre le interfacce di gestione PhpMyAdmin e SQLiteManager ed è quindi possibile accedere ai file/pagine web che sono ospitati sul server WAMP digitando: *http://localhost/* o *http://127.0.0.1/* nella barra degli indirizzi del browser web. Wamp deve essere in esecuzione per poter effettuare l'accesso.

Creazione del database

Al fine di memorizzare in un database le coordinate dei mezzi pubblici in movimento, si descrive in questo paragrafo la procedura necessaria per la creazione di una tabella finalizzata al progetto.

La procedura per la realizzazione di un nuovo database risulta semplificata grazie all'interfaccia grafica fornita dall'applicazione *phpMyAdmin* (Fig. 211), che ne permette la creazione in modalità guidata.

PhpMyAdmin è un'applicazione libera che permette di amministrare in modo semplificato un database MySQL tramite un qualsiasi browser.

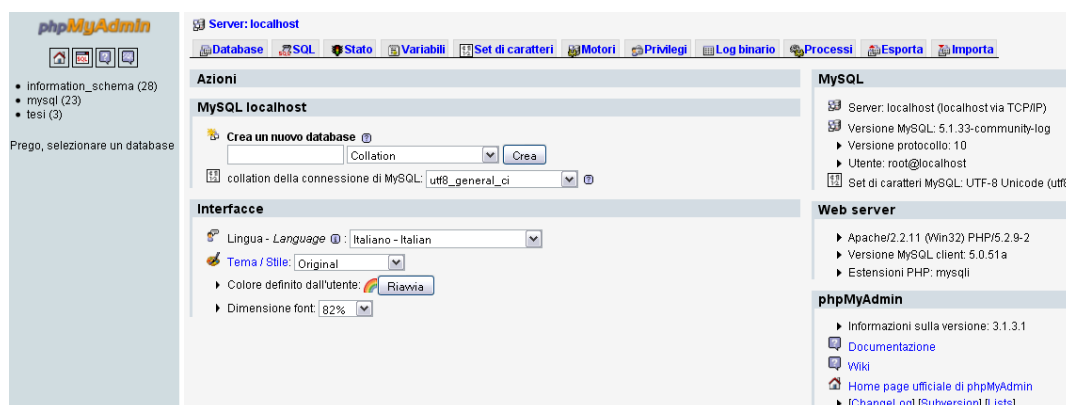


Fig. 211 - Creazione database con phpMyAdmin

Il database in Fig. 212 contiene i record relativi ai dati acquisiti dall'interfaccia GPS, in particolare quelli utilizzati per questo lavoro di tesi. Occorre quindi creare un software che permetta di inserire, nel database creato, le informazioni elaborate dalle istruzioni Java descritte in precedenza.

	Data	UTC	IP	Latitudine	S_N	Longitudine	W_E	Altitudine	num_sat_vista	num_sat_uso	SNR	HPE	VPE	Velocità	PDOP	HDOP	VDOP
	2009-07-12 16:45:08	144522	192.168.1.100	44.4931	N	11.3055	E	62	12	6	42	9.3	9	0	2.7	2	1.2
	2009-07-12 16:45:09	144524	192.168.1.100	44.4931	N	11.3055	E	62.1	12	6	43	9.3	9	0	2.8	2.1	1.2
	2009-07-12 16:45:11	144526	192.168.1.100	44.4931	N	11.3055	E	61.9	12	6	0	9.3	9	0	2.8	2.1	1.2
	2009-07-12 16:45:13	144528	192.168.1.100	44.4931	N	11.3055	E	61.4	12	6	42	9.3	9	0	2.8	2.1	1.4
	2009-07-12 16:45:15	144530	192.168.1.100	44.4931	N	11.3055	E	60.9	12	6	43	9.3	9	0	2.8	2.1	1.4
	2009-07-12 16:45:17	144532	192.168.1.100	44.4931	N	11.3055	E	60.2	12	6	0	9.3	9	0	2.8	2.1	1.4
	2009-07-12 16:45:19	144534	192.168.1.100	44.4931	N	11.3055	E	59.4	12	6	42	9.3	9	0	2.8	2.1	1.5
	2009-07-12 16:45:22	144536	192.168.1.100	44.4931	N	11.3055	E	58.8	12	6	44	9.5	9	0	2.8	2.1	1.5
	2009-07-12 16:45:23	144538	192.168.1.100	44.4931	N	11.3055	E	58.4	12	6	0	9.5	9	0	2.8	2.1	1.5
	2009-07-12 16:45:25	144540	192.168.1.100	44.4931	N	11.3055	E	58.3	12	6	42	9.5	9	0	2.8	2.1	1.5
	2009-07-12 16:45:27	144542	192.168.1.100	44.4931	N	11.3055	E	58.6	12	6	44	9.5	9	0	2.8	2.1	1.5
	2009-07-12 16:45:29	144544	192.168.1.100	44.4931	N	11.3055	E	58.8	12	6	0	9.5	9	0	2.8	2.1	1.6
	2009-07-12 16:45:31	144546	192.168.1.100	44.4931	N	11.3055	E	59	12	5	42	9.6	9	0	2.8	2.1	1.6
	2009-07-12 16:45:33	144548	192.168.1.100	44.4931	N	11.3055	E	59	12	5	42	9.6	9	0	2.8	2.1	1.6

Fig. 212 - Screenshot database, tabella gps_data

I vari campi sono riportati qui di seguito³⁴:

- Data: data e ora relative alla scrittura del record nel database
- UTC
- IP: indirizzo IP del client
- Latitudine (specificando NORD/SUD)
- Longitudine (specificando OVEST/EST)
- Altitudine in metri
- num_sat_vista: numeri di satelliti in vista
- num_sat_uso: numeri di satelliti collegati (in uso)
- SNR
- HPE (Horizontal Position Error) in metri
- VPE (Vertical Position Error) in metri
- Velocità (Km/h)
- PDOP (Position Dilution Of Precision)
- HDOP (Horizontal Dilution Of Precision)

³⁴ Vedi Capitolo 2 paragrafo 2.7.1 "Struttura delle frasi" .

- VDOP (Vertical Dilution Of Precision)

Connessione al database e inserimento dati

Per interfacciare Java con il database SQL occorre la libreria *JDBC*.

JDBC (Java DataBase Connectivity) è un connettore per database che consente l'accesso alle basi di dati da qualsiasi programma scritto in linguaggio di programmazione Java. Essa fornisce i metodi per l'accesso e la manipolazione di basi di dati di tipo relazionale.

Mantenendo la filosofia di Java, le API (*Application Programming Interface*) *JDBC* sono state progettate per ottenere l'accesso ai dati indipendentemente dalla piattaforma e dal particolare tipo di database utilizzato. *JDBC* offre infatti delle interfacce standard e uniche per qualunque database. Per ciascuno di questi deve poi esistere un driver che si occupi di tradurre le chiamate *JDBC*, effettuate dall'applicazione, in opportune istruzioni necessarie per accedere e manipolare i dati di uno specifico database.

Quando l'applicazione richiede l'accesso ad un database, un particolare componente detto *Driver Manager* ne controlla il tipo e carica il driver appropriato. In questo modo è possibile cambiare sia piattaforma che database semplicemente cambiando driver.

Le classi e le interfacce *JDBC* si trovano nel package `java.sql`.

La prima operazione da eseguire per iniziare a lavorare sul database è caricare il driver adatto. Per farlo è necessario forzare il caricamento della classe che rappresenta il driver utilizzando il metodo `forName` della classe `Class`.

L'istruzione è:

```
Class.forName("com.mysql.jdbc.Driver").newInstance();
```

A questo punto si stabilisce la connessione tra l'applicazione Java ed il database chiamando il metodo `getConnection` della classe `DriverManager` nel seguente modo:

```
Connection db = DriverManager.getConnection("jdbc:mysql://" +  
      "localhost:3306/tesi?user=root&password=");35
```

Per eseguire delle interrogazioni o modifiche sulla base di dati si crea un oggetto di tipo `PreparedStatement` utilizzando il metodo `prepareStatement` dell'oggetto di tipo `Connection` ottenuto in precedenza.

L'istruzione è:

```
PreparedStatement stmt=db.prepareStatement("insert into " +  
      "gps_data(UTC, IP, Latitudine, S_N, Longitudine, W_E, " +  
      "Altitudine, num_sat_vista, num_sat_uso, SNR, HPE, VPE, " +  
      "Velocità, PDOP, HDOP, VDOP)" +  
      " values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
```

³⁵ I parametri di accesso, in questo caso, sono user = "root" senza password e nome db = "tesi".

Per inserire i valori mancanti si utilizzano i metodi `setX` (dove `X` è un tipo di dato) a cui vanno passati due parametri: il primo è l'indice del parametro della query che si vuole specificare, il secondo è il valore che gli si vuole dare.

```
stmt.setString(1,iutc);  
stmt.setString(2,iip);  
stmt.setString(3,ilat);  
stmt.setString(4,isn);  
stmt.setString(5,ilon);
```

```
.  
.  
.
```

I parametri `iutc`, `iip`, `ilat`, `isn`, `ilon` sono le variabili `utc`, `ip`, `lat`, `sn`, `lon` calcolate nel paragrafo 4.1.3.2 e passate al metodo `Inserisci` che gestisce l'inserimento dei dati nel database.

L'istruzione

```
stmt.executeUpdate();
```

esegue l'interrogazione sul database con i parametri specificati.

Anche per le istruzioni di interfacciamento con il database, come fatto per le istruzioni di socket, occorre gestire le eccezioni tramite le `try` e `catch`.

Attraverso le istruzioni viste i dati vengono così memorizzati nel database.

Query sul database e mapping real-time

SQL è il linguaggio standard per la creazione e manipolazione di database relazionali. Esso non è però un linguaggio completo in quanto, oltre al fatto di non essere in grado di eseguire alcune interrogazioni a causa della mancanza di costrutti tipici della programmazione imperativa (per cui un programma viene inteso come un insieme di istruzioni, ciascuna delle quali può essere pensata come un "ordine" che viene impartito alla macchina virtuale del linguaggio di programmazione utilizzato), non è adatto alla realizzazione di vere e proprie applicazioni. A causa di queste problematiche risulta necessario utilizzarlo insieme ad un altro linguaggio di programmazione.

Molti linguaggi incorporano SQL (*embedded-SQL*). In questi casi il programmatore, nel punto in cui deve accedere al database, smette di scrivere nel linguaggio ospite e inserisce le istruzioni da eseguire in SQL direttamente nel codice.

Per ottenere il programma eseguibile il codice deve essere prima esaminato da un particolare traduttore che traduca SQL nel linguaggio ospite; solo in seguito è possibile passare al compilatore e quindi al linker.

Lettura dati

I dati sono a questo punto in memoria nel database, occorre quindi realizzare un'interfaccia web che permetta di effettuare query su quest'ultimo così da recuperare le informazioni dalla tabella creata (informazioni di Latitudine e Longitudine).

A questo punto il sito del progetto TRIP avrà a disposizione i dati necessari per rappresentare sulla mappa la posizione dell'interfaccia GPS.

Per accedere al database occorrono le informazioni relative a:

- Host: *localhost*
- Username: *root*
- Password (in questo caso non è presente)
- Nome del database: *tesi*

Il file che permette la connessione al database (*GPS.php*) è strutturato nel seguente modo:

```
<?php
//Parametri del database
$db_host = "localhost";
$db_user = "root";
$db_password = "";
$db_name = "tesi";

//Connessione al database
mysql_connect($db_host,$db_user,$db_password) or
die("ERRORE DI CONNESSIONE" . mysql_error());
mysql_select_db($db_name);

//Recupero delle coordinate "latitudine e longitudine"
$query = "SELECT Latitudine,Longitudine FROM
gps_data
ORDER BY data desc";
$resultato = mysql_query($query) or
die("<br> ATTENZIONE: ".$query." Fallita!");
$LST = mysql_fetch_array($resultato);
mysql_free_result($resultato);
echo $LST["Longitudine"]." ".$LST["Latitudine"];

?>
```

Le funzioni presenti nel codice hanno il seguente significato:

- La funzione **mysql_connect** permette il collegamento al database. Dopo aver effettuato la connessione è possibile agire direttamente sul database utilizzando insieme i linguaggi PHP ed SQL.
- La variabile **\$query** rappresenta l'operazione da effettuare sul database e viene comunicata attraverso la funzione **mysql_query**. Questa funzione, come la *mysql_connect*, restituisce FALSE in caso di errore.
- La funzione **mysql_fetch_array** crea un array che ha come indice i nomi dei campi della tabella e come argomento i record della stessa.
- La **echo** permette infine di stampare le due coordinate (Fig. 213).

DG4ARCHEOS

In questo capitolo viene descritto il funzionamento di particolari librerie per la gestione del dato GPS e comunicazione mobile realizzata nell'ambito del progetto DG4ARCHEOS.

Tale descrizione viene allegata alla documentazione del progetto TRIP in quanto si tratta di un background formativo relativo alle funzionalità di acquisizione e gestione delle informazioni di localizzazione derivate dalla tecnologia GPS, sia del loro invio attraverso tecnologie di comunicazioni wireless in regime di bassa stabilità di connessione (utenti mobili).

Sintesi del progetto DG4ARCHEOS

DG4ARCHEOS - Digital Guide for Art & Cultural Heritage Outdoor Scenario progetto per la fruizione in mobilità dei beni culturali realizzato da Al mavivA - CNR, l'azienda del Gruppo Al mavivA in compartecipazione con il CNR per la ricerca e lo sviluppo di soluzioni tecnologiche innovative

DG4ARCHEOS è caratterizzato da una forte multidisciplinarietà che ha coinvolto competenze archeologiche per le ricostruzioni storiche e i contenuti, competenze di localizzazione via GPS e telecomunicazioni wireless oltre ad un software innovativo per la ricostruzione degli edifici. L'obiettivo è la realizzazione di una guida virtuale, utilizzabile su un palmare di mercato, che non solo descriva il sito archeologico, ma mostri sullo schermo la fedele ricostruzione tridimensionale dei palazzi e degli ambienti, con un forte impatto emotivo per il visitatore.

Attualmente è stato ricostruito tutto il complesso voluto da Domiziano, il più imponente e suggestivo del Palatino: partendo dalle aerofotogrammetrie, con il supporto di piantine, fotografie e testi descrittivi, si sono ricreate la reale pianta del Palazzo imperiale, la divisione in ambienti in 3D e, infine, la decorazione delle stanze interne.

Il visitatore potrà scegliere la lingua tra quelle disponibili, il profilo, junior o senior, in base all'approfondimento dei contenuti che preferisce, la visita libera o il percorso predefinito.

Il GPS guida il visitatore nei suoi spostamenti, rileva costantemente e visualizza sullo schermo il luogo in cui si trova. Se sceglie un nuovo punto di interesse da visitare, il sistema DG4ARCHEOS calcola la distanza in metri e indica la via da seguire. Sullo schermo del palmare, un video e una mappa presentano le principali caratteristiche del sito archeologico e forniscono informazioni generali: strade, monumenti significativi e servizi.

Un tasto help è presente in ogni sezione della guida per fornire tutti i chiarimenti necessari. Quando si entra in un'area di interesse, si viene avvertiti automaticamente della disponibilità di approfondimenti: descrizioni, audio e video, immagini, modelli 3D. Al termine della visita, il visitatore può scegliere i contenuti di maggiore interesse e acquistare un catalogo multimediale personalizzato.

Descrizione del Funzionamento di GPSRefiner

In breve

La classe mette a disposizione un metodo pubblico che accetta come argomento la posizione GPS fornita dal ricevitore GPS. Questa posizione viene confrontata con i valori definiti non-validi dal gestore della mappa. Se tale posizione non è valida, a partire dalla posizione precedente e tenendo conto della posizione fornita dal ricevitore GPS viene calcolata una posizione valida per l'utente.

Costruttore

Il costruttore della classe prevede due argomenti: il primo è un oggetto `PositionInformationSource` che gestisce le informazioni riguardanti la mappa (dal punto di vista del GPSRefiner serve per definire le posizioni non valide in essa), il secondo è un oggetto `GPSPosition` che definisce una posizione di riferimento nel caso di prima posizione non valida.

Metodo principale: refine

Una volta istanziato l'oggetto GPSRefiner è possibile utilizzare il metodo

public `GPSPosition refine(GPSPosition position)`

il metodo accetta come argomento un oggetto `GPSPosition` e restituisce un secondo oggetto `GPSPosition`.

L'argomento è la posizione fornita dal ricevitore GPS, mentre la posizione restituita è la posizione valida ottenuta dall'algoritmo.

Il primo passaggio dell'algoritmo è quello di valutare se la posizione fornita dal ricevitore GPS è una posizione valida nella mappa utilizzata. Per fare questo utilizza il metodo `PositionInformationSource.isValidPixel()`.

Se la posizione fornita dal GPS è valida non si operano altre operazioni e tale posizione viene restituita come posizione attuale per l'utente. La posizione viene inoltre memorizzata come posizione passata per la prossimo utilizzo di refine.

Nel caso la posizione acquisita dal ricevitore non sia valida occorre determinarne una valida.

Per fare questo si costruisce una griglia di punti su cui cercare una posizione valida e si attribuiscono dei pesi a ciascuno di questi punti in base alla probabilità di essere raggiunti dall'utente.

Creazione della griglia

La griglia viene creata in base alle due posizioni GPS disponibili all'algoritmo: la posizione passata dell'utente (sempre valida) e la posizione letta dal ricevitore (sempre non valida in questo caso).

La griglia è un insieme di posizioni che formano una matrice quadrata. Il numero di punti che forma un lato della matrice è determinato dal valore `StepDensity`.

L'orientamento della griglia è definito dalla direzione del presunto spostamento dell'utente. Fissato che la griglia si sviluppa per righe, identificate dal valore `i` (`i=0` indica la riga che contiene la posizione passata dell'utente), e da colonne identificate dal valore `j`, allora la direzione delle colonne è la direzione che collega la posizione passata a quella rilevata dal ricevitore GPS.

La direzione delle righe è perciò perpendicolare. Alcuni esempi grafici illustrano le griglie con valori pari e dispari di punti per lato. La posizione passata viene posta al centro della riga $i=0$, quindi in caso di un numero dispari di colonne la posizione passata sarà il primo elemento della colonna centrale (ad esempio in una matrice 3×3 sarà il punto $i=0, j=1$), in caso di un numero di colonne pari, la posizione passata sarà a centro delle due colonne centrali, senza farne parte (ad esempio in una matrice 4×4 si troverà tra i punti ($i=0, j=1$) e ($i=0, j=2$)).

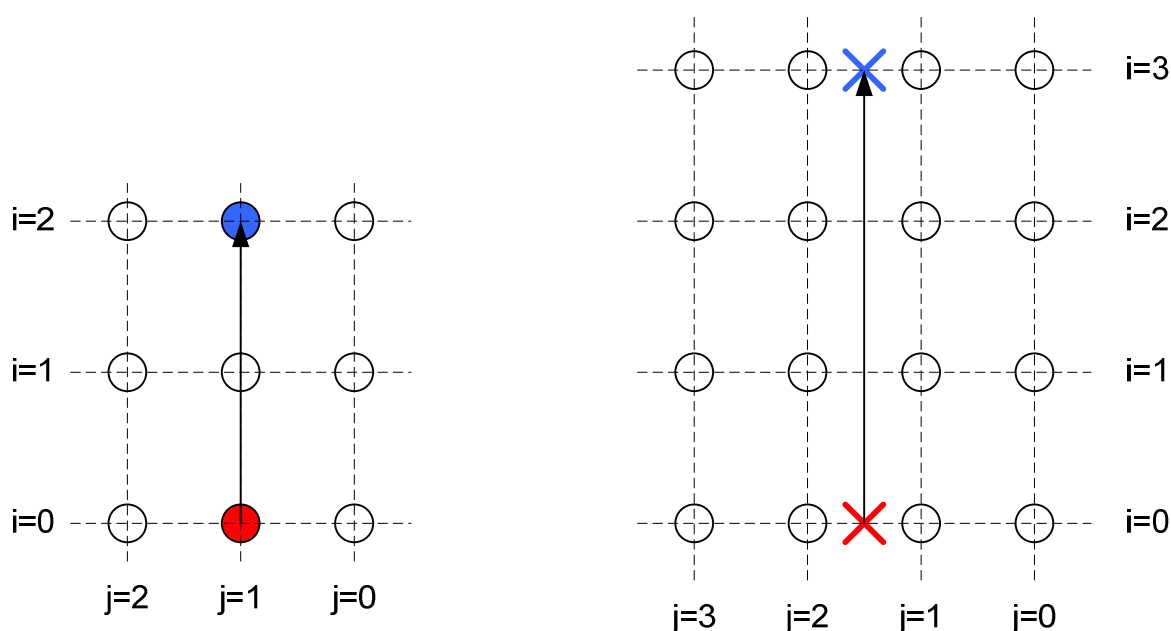


Fig. 215 - Griglia con posizioni discrete

La distanza tra le righe della griglia è definita dalla distanza (misurata in secondi di latitudine e longitudine) che esiste tra la posizione passata e la posizione rilevata dal GPS. Il valore viene memorizzato nella variabile `MainDistance` ed è misurata in metri; viene calcolata attraverso la funzione `calculatedDistance(latShift, lonShift)` che accetta in ingresso lo scarto di latitudine e longitudine tra le posizioni di cui si vuole avere la distanza (il calcolo è descritto a seguire in Trasformazione delle Distanze).

Tale distanza viene divisa dal valore `StepDensity-1` in modo da ottenere lo scarto tra ogni riga.

Per quanto riguarda la distanza tra le colonne si utilizza un valore che è una frazione della `MainDistance`. Tale valore viene memorizzato nella variabile `trasverseDistance` che viene limitata tra due soglie in modo da garantire uno scostamento laterale minimo anche in caso di piccoli movimenti, e allo stesso tempo viene fissato un massimo per non scostarsi troppo dalla direzione presa dall'utente.

Trasformazione delle distanze

Per il calcolo dei pesi dei punti della griglia al fine di ottenere i valori più probabili occorre avere le distanze in metri dei punti rispetto alle posizioni passate e quella ottenuta dal ricevitore GPS. Per effettuare tale trasformazione si utilizza una semplice conversione di scala ottenuta linearizzando il geodide GPS nella zona relativa al sito in esame.

Viste infatti le dimensioni della griglia (legate grossomodo al movimento di un utente a piedi in un secondo), gli errori introdotti da questa linearizzazione sono trascurabili ed al contempo evitano di operare una mole considerevole di calcoli.

I fattori di conversione sono semplicemente (validi nei pressi della zona in esame):

latMetersPerSec=30.844

lonMetersPerSec=23.048

Il calcolo deve tener conto del non allineamento della griglia rispetto alla direzione cardinale di latitudine e longitudine.

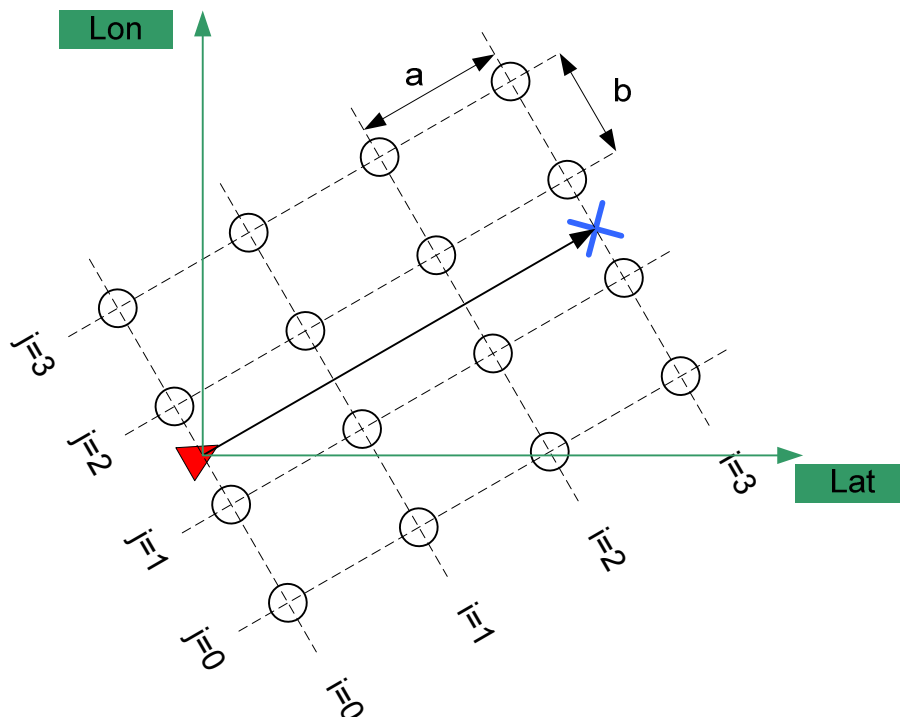
Come specificato in precedenza il primo passo è il calcolo della distanza tra oldPostion e readPosition. Il calcolo sarà semplicemente:

$$\text{mainDistance} = \sqrt{[(LAT_{read} - LAT_{old}) \cdot \text{latMetersPerSec}]^2 + [(LON_{read} - LON_{old}) \cdot \text{lonMetersPerSec}]^2}$$

da questo si ottiene `trasverseDistance`

$$\text{trasverseDistance} = \frac{\text{mainDistance}}{7.5}$$

Il valore ottenuto viene limitato tra 0.3 e 1.5 metri per i motivi descritti in precedenza.



Quindi per il calcolo della distanza dei punti occorre sfruttare formule trigonometriche. Fissate

$$a = \frac{\text{mainDistance}}{\text{StepDensity} - 1}$$

$b = \text{trasverseDistance}$

In particolare per la distanza dalla vecchia posizione (triangolo rosso) vale:

$$d^2 = (i \cdot a)^2 + \left(j - \frac{\text{StepDensity} - 1}{2} \cdot b \right)^2$$

Per la distanza dalla posizione rilevata dal ricevitore GPS (croce azzurra) vale:

$$d^2 = ((i - \text{StepDensity} + 1) \cdot a)^2 + \left(j - \frac{\text{StepDensity} - 1}{2} \cdot b \right)^2$$

Calcolo dei pesi

Per scegliere tra le posizioni della griglia quale sarà la posizione da assegnare all'utente (visto che quella rilevata dal GPS non è valida) occorre effettuare alcune ipotesi.

La posizione rilevata dal GPS è non valida e quindi errata in quanto l'utente non può entrare fisicamente nelle zone definite non valide.

Lo spostamento dell'utente dalla posizione passata non è arbitrario ma è legato al movimento fisico di una persona (in questo caso a piedi).

Generalmente un utente a piedi si muove ad una velocità attorno ai 1,5m/s o meno. La scelta per definire quale probabilità ha un dato spostamento è una distribuzione Gaussiana centrata sulla posizione passata dall'utente. Per definire la Gaussiana occorre stabilire il valore della varianza che in questo caso viene chiamata varianza di movimento. Il valore di varianza di movimento viene definito scegliendo con quale probabilità gli spostamenti devono superare un certo valore. Considerando il movimento di un utente a piedi si è scelto di porre che solo il 5% degli spostamenti in un secondo possa essere oltre i 2 metri.

Dalla funzione di ripartizione della distribuzione Gaussiana si ottiene una varianza pari a:

$$\frac{\text{raggio di movimento}}{2} = \frac{2}{2} = 1$$

Questo significa che la probabilità di essersi spostati in un punto ad una distanza di d metri dalla posizione passata è legata al valore:

$$e^{-\frac{d^2}{\sigma_m^2}}$$

Lo stesso risultato si può pensare per la posizione fornita dal GPS. Tale posizione non è esatta, ma è affetta da un errore dovuto alla non perfetta sincronizzazione del segnale dei satelliti e dalla incertezza intrinseca del chip di acquisizione del ricevitore GPS. La prima forma di errore viene considerato dallo stesso sistema GPS e viene fornito al ricevitore sotto forma del valore HDOP (Horizontal Dilution Of Precision). Il secondo è definito dall'incertezza indicato come errore in metri. L'errore totale all'uscita del ricevitore è HDOP x incertezza. Imponendo anche in questo caso una distribuzione Gaussiana in cui il 95% dei casi sia entro l'errore ottenuto si ottiene una varianza, denominata varianza di localizzazione, pari a:

$$\frac{HDOP \cdot \text{incertezza}}{2} = \frac{HDOP \cdot 1,5}{2} = 0,75 \cdot HDOP$$

Quindi la probabilità di essere in una posizione che dista d metri da quella rilevata è legata al valore:

$$e^{-\frac{d^2}{\sigma^2}}$$

Considerando indipendenti i due fenomeni, la probabilità totale del processo sarà legata al loro prodotto.

Come passaggio per l'ottimizzazione del software si passa a considerare il logaritmo naturale dei valori individuati sopra. Si ottengono diversi vantaggi:

- Non è più necessario calcolare il valore dell'esponenziale, basta il calcolo del valore

$$\frac{d^2}{\sigma^2}$$

riducendo di molto il peso computazionale

- I valori ottenuti in questo modo sono valori più facilmente trattabili da un calcolatore (con l'esponenziale si possono avere valori molto prossimi allo zero, difficilmente gestibili con un numero limitato di byte da parte del calcolatore)
- Il prodotto dei coefficienti diviene una semplice somma, riducendo inoltre i calcoli necessari.

Il risultato finale è che i punti più probabili hanno pesi più bassi.

I pesi calcolati per i punti della griglia vengono organizzati in lista ed ordinati. Procedendo in ordine di peso si valuta se il punto è una posizione valida. Il primo punto che risulta valido è la posizione che viene restituita come posizione dell'utente.

Data la simmetria nella costruzione della griglia e la dipendenza dei pesi dalla distanza tra i punti in studio. Può accadere che punti diversi abbiano lo stesso peso (generalmente in maniera simmetrica rispetto la colonna centrale della griglia).

Nel tentativo di risparmiare un accesso alla verifica `IsValidPosition()` che si tratta di un processo oneroso, l'algoritmo mantiene in memoria quale se il punto considerato valido era a destra o sinistra della colonna centrale. Questo perché con buona probabilità lo stesso caso si presenterà alla posizione successiva (le condizioni al contorno dell'utente variano lentamente visto che l'utente si muove a bassa velocità). Applicando per prima la stessa scelta si ha buona probabilità di evitare almeno una verifica di una posizione.

Lista dei pesi

Per tenere memoria del peso dei punti della griglia si è scelto di memorizzare l'associazione (punto, peso) in una HashTable. Questo consente di accedere facilmente ai dati memorizzati ed operare su di essi.

In particolare la HashTable viene costruita utilizzando i valore dei pesi come Chiavi e la coppia (i,j) che individua il punto come Elemento.

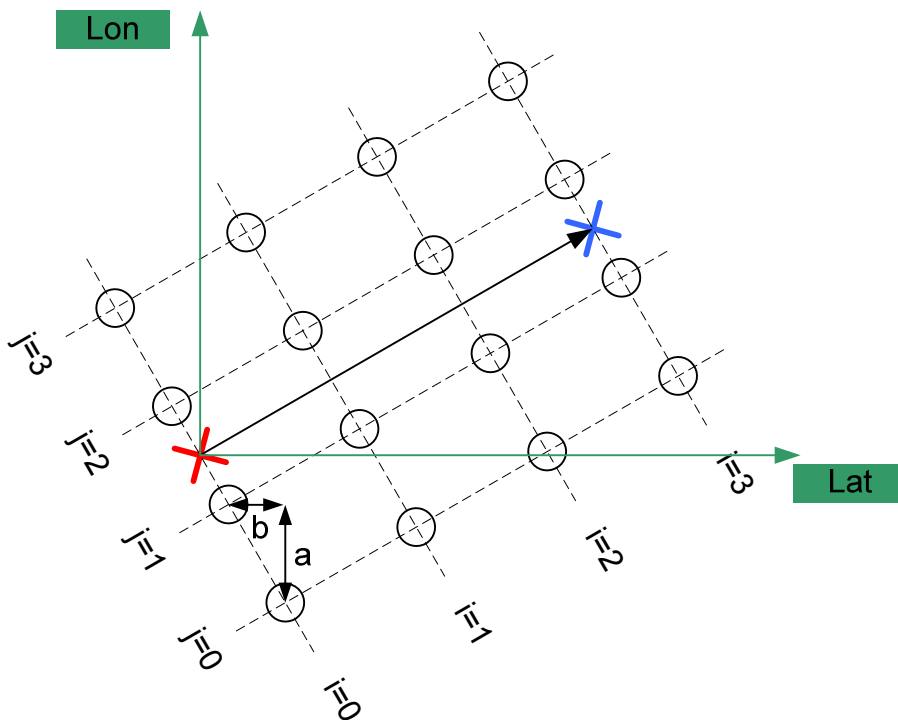
I pesi vengo copiati in una lista e quindi ordinati. Refiner scorre la lista in ordine di peso. Per ogni punto viene valutata la condizione `IsValidPosition()`. In caso affermativo quella posizione viene restituita come posizione da considerare per l'utente.

Conversione da griglia a `GPSPosition`

La conversione di un punto della griglia in una GPSPosition definita come latitudine e longitudine deve tener conto del non allineamento della griglia con le direzioni cardinali. La conversione passa attraverso funzioni trigonometriche.

Per prima cosa si determina l'angolo rispetto la direzione Est (direzione della Longitudine)

$$\text{angle} = \arctan \frac{LAT_{read} - LAT_{old}}{LON_{read} - LON_{old}}$$



Detti:

$$a = \frac{\text{trasverseDistance} \cdot \cos(\text{angle})}{\text{latMetersPerSec}}$$

$$b = -\frac{\text{trasverseDistance} \cdot \sin(\text{angle})}{\text{lonMetersPerSec}}$$

si ha

$$\text{Latitude} = LAT_{old} + i \cdot \frac{LAT_{read} - LAT_{old}}{\text{StepDensity} - 1} + \left(j - \frac{\text{StepDensity} - 1}{2} \right) \cdot a$$

$$\text{Longitude} = LON_{old} + i \cdot \frac{LON_{read} - LON_{old}}{\text{StepDensity} - 1} + \left(j - \frac{\text{StepDensity} - 1}{2} \right) \cdot b$$

Lista dei Metodi

public **GPSPositionRefiner**(PositionInformationSource info, GPSPosition defPos)

Costruttore della classe; accetta come argomenti un oggetto PositionInformationSource che gestisce le informazioni della mappa e un oggetto GPSPosition da utilizzare come valore di default in caso di prima posizione rilevata.

public GPSPosition **refine**(GPSPosition position)

Metodo principale della classe (descritto in precedenza); accetta come argomento un oggetto GPSPosition fornito dal ricevitore GPS, restituisce un oggetto GPSPosition che individua una posizione valida che verrà utilizzata come posizione attuale dell'utente.

private **double** **calculateDistance**(**double** latShift, **double** lonShift)

Dato differenze di latitudine e longitudine calcola la distanza in metri.

private Hashtable **generateWeightTable**()

Metodo in cui viene costruita la Hashable che contiene la relazione tra pesi e punti della griglia di studio.

private **double** **calcWeight**(**int** i, **int** j)

Dato un punto della griglia calcola il peso relativo a quel punto griglia.

private List<**double**> **orderList**(Hashtable weightTable)

Data una Hashtable restituisce una lista con le chiavi della tabella ordinate in ordine crescente (in questo caso la lista ha i pesi ordinati in ordine crescente)

private GPSPosition **obtainValidPosition**(List<**double**> weightList, Hashtable weightTable)

Il metodo accetta come argomenti la lista ordinata dei pesi e la Hashtable con la relazione peso-punto; il metodo scorre la lista e valuta se la posizione corrispondente al peso risulta essere valida oppure no. Restituisce la prima posizione considerata valida.

private GPSPosition **obtainPosition**(**int**[] p)

Dato un punto della griglia calcola il valore di longitudine e latitudine corrispondente. Restituisce un oggetto GPSPosition

private **double** **degreeToSec**(Degree d)

Dato un oggetto Degree restituisce un double che rappresenta lo stesso angolo in secondi decimali.

private Degree **secToDegree**(**double** s)

Dato un oggetto double che rappresenta un angolo in secondi decimali restituisce lo stesso angolo come oggetto Degree.

public **void** **reset**()

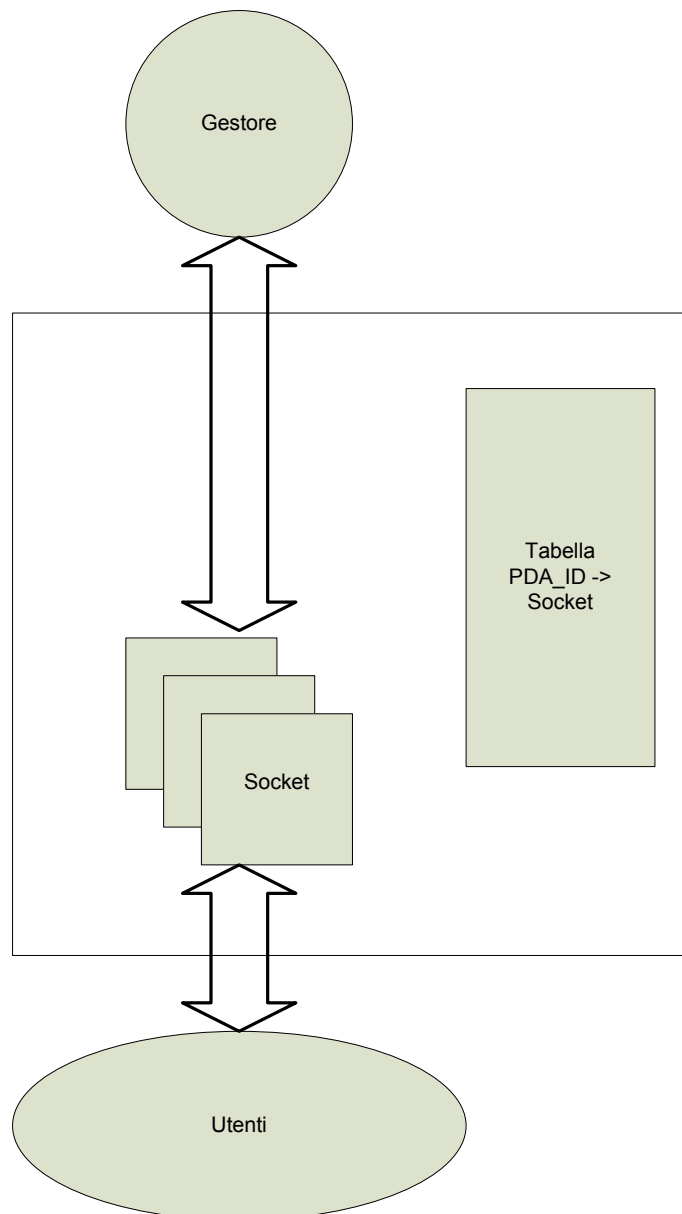
Metodo che riporta la posizione passata dell'utente alla posizione definita come default (oggetto GPSPosition passato al costruttore).

Libreria di gestione della comunicazione

La libreria si frappona tra l'insieme degli utenti e il server che gestisce il sistema.

I compiti della libreria sono:

- la creazione di canali di comunicazione indipendenti tra utenti e server, garantendo accesso multiplo ed asincrono;
- fornire una comunicazione trasparente verso gli utenti resolvendo internamente questioni di disconnessione, cambio di indirizzo, etc.



Il primo obiettivo viene raggiunto attraverso la creazione di socket di comunicazione biunivoci tra server ed utente gestiti da thread separati; la seconda attraverso la compilazione e l'aggiornamento di una tabella con associazioni tra identificativi d'utente e socket di comunicazione.

Formato dei Messaggi

Il server deve gestire degli oggetti che rappresentano lo scambio di informazioni tra le diverse entità all'interno del sistema. Denominiamo questi oggetti MESSAGGIO (si tratta di stringhe formate da uno o più campi divisi dal carattere #).

Dal punto di vista della libreria ServerGPRSCommunication ci sono due tipi di MESSAGGIO: il primo tipo è il MESSAGGIO PING, mentre il secondo tipo è il MESSAGGIO NON PING (gruppo formato dall'insieme dei rimanenti MESSAGGI). Tale divisione deriva dal fatto che il comportamento della libreria è diverso nei confronti di MESSAGGI PING o NON PING.

Il MESSAGGIO PING prevede infatti l'aggiornamento della tabella di riferimento della libreria, mentre il MESSAGGIO NON PING prevede solo l'inoltro verso il server.

La libreria deve quindi fare il parse di ogni MESSAGGIO in ingresso per leggerne il CODICE OPERATIVO e capire se si tratta o meno di un PING.

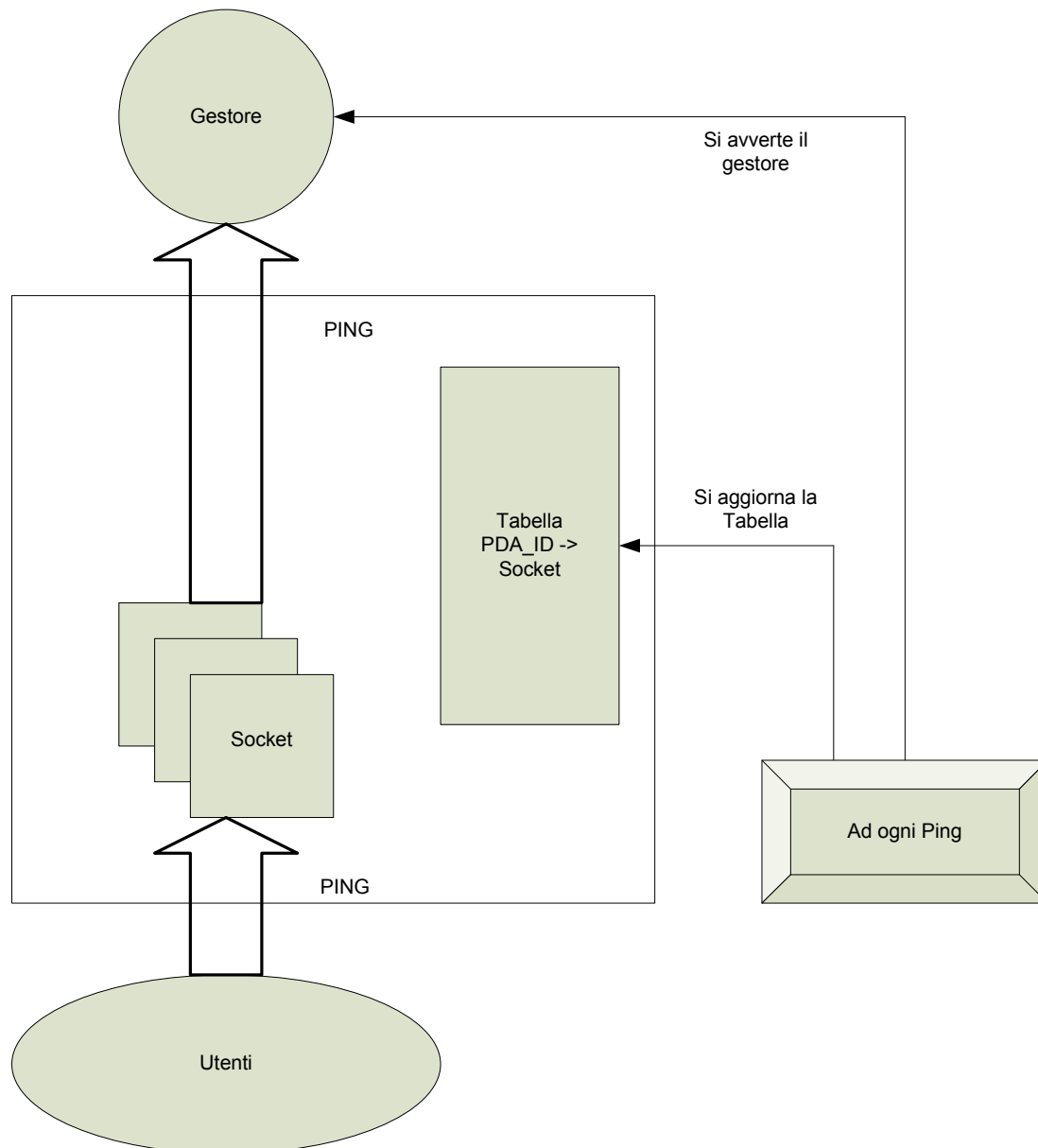
Connessione

Gli utenti all'ingresso nel sistema si connettono al server. Tali connessioni sono gestite dalla libreria server che crea socket attraverso le quali avviene lo scambio di informazioni tra server e entità client.

Messaggio Ping

Alla creazione della socket il server non conosce alcuna informazione sull'entità client che si è connessa, ne conosce solo l'indirizzo IP. Quindi l'entità client deve mandare come primo MESSAGGIO un MESSAGGIO PING con tutti i dati necessari per le future identificazioni e comunicazioni. (La tabella è indicizzata sulle ID delle PDA, si può pensare che all'affitto della PDA venga generato una voce il cui campo verrà riempito alla connessione della PDA stessa).

A seguito del PING la libreria costruisce una tabella di riferimento in cui sono associate identificativi, indirizzi IP e dati delle entità connesse.



La tabella (da definire il formato, per ora hash-table) ha la struttura

KEY	VALUE
PDA_ID #	Oggetto #
PDA_ID #	Oggetto #

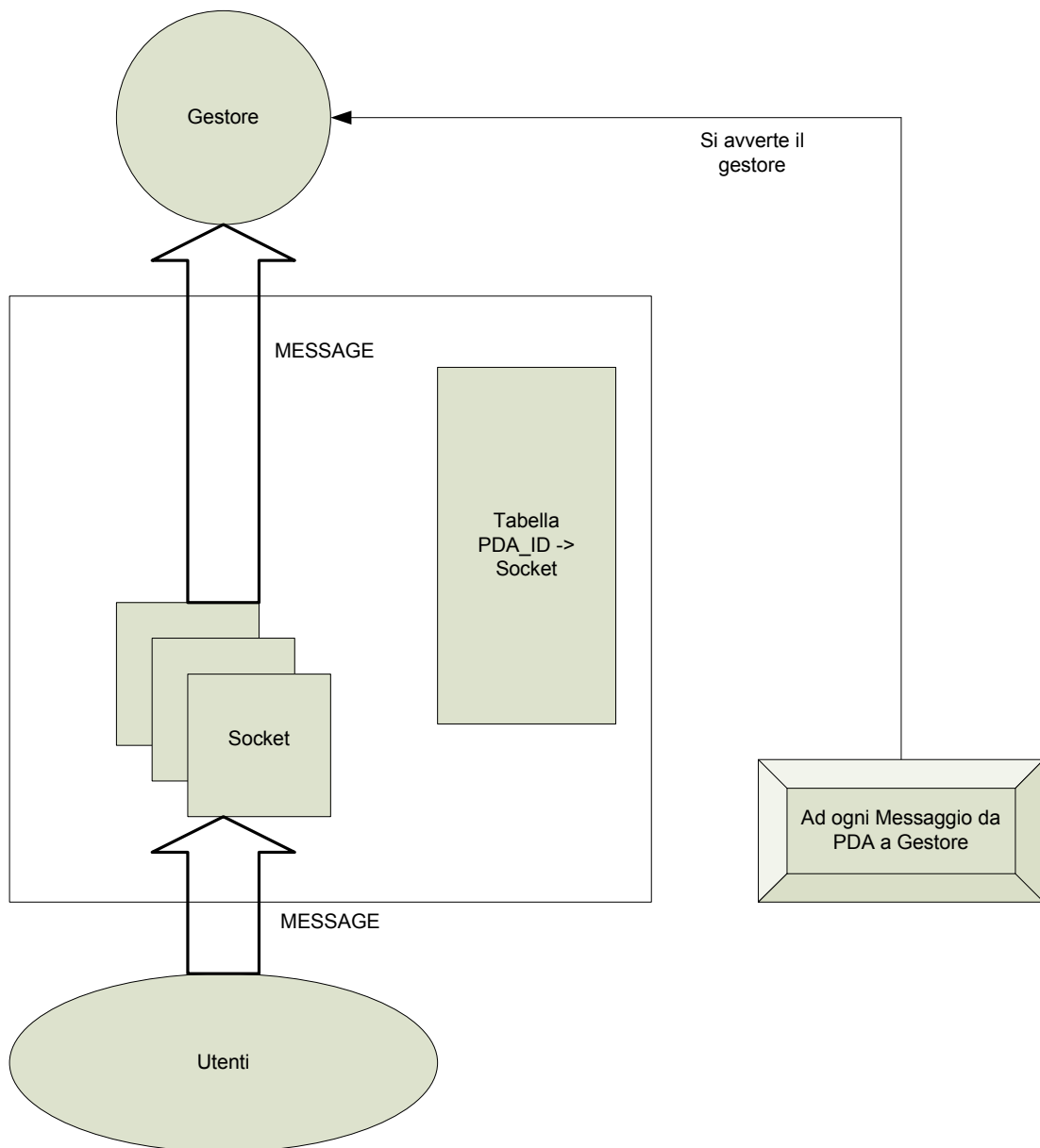
Dove con oggetto si intende una struttura che contenga a sua volta l'oggetto socket che permette la connessione tra utente e server (dall'oggetto socket si ricava in maniera semplice anche l'indirizzo IP se necessario)

Messaggio NON-Ping

La PDA invia un MESSAGGIO NON PING al server. Effettua un unico tentativo sul socket che è stato creato alla connessione con il server. La libreria, controllato il fatto che non si tratta di un MESSAGGIO PING cerca a quale PDA_ID corrisponde il mittente.

Se trova il mittente avverte il server dell'arrivo del nuovo messaggio e glielo inoltra accompagnato dal PDA_ID del mittente.

Se non lo trova scarta il MESSAGGIO e non segnala nulla al server.



Messaggio da Server a Utente

Il server vuole inviare un MESSAGGIO alla PDA identificata con PDA_ID, non sapendo se questa è connessa o meno. Si prefigge quindi un numero di tentativi entro il quale l'invio deve avvenire.

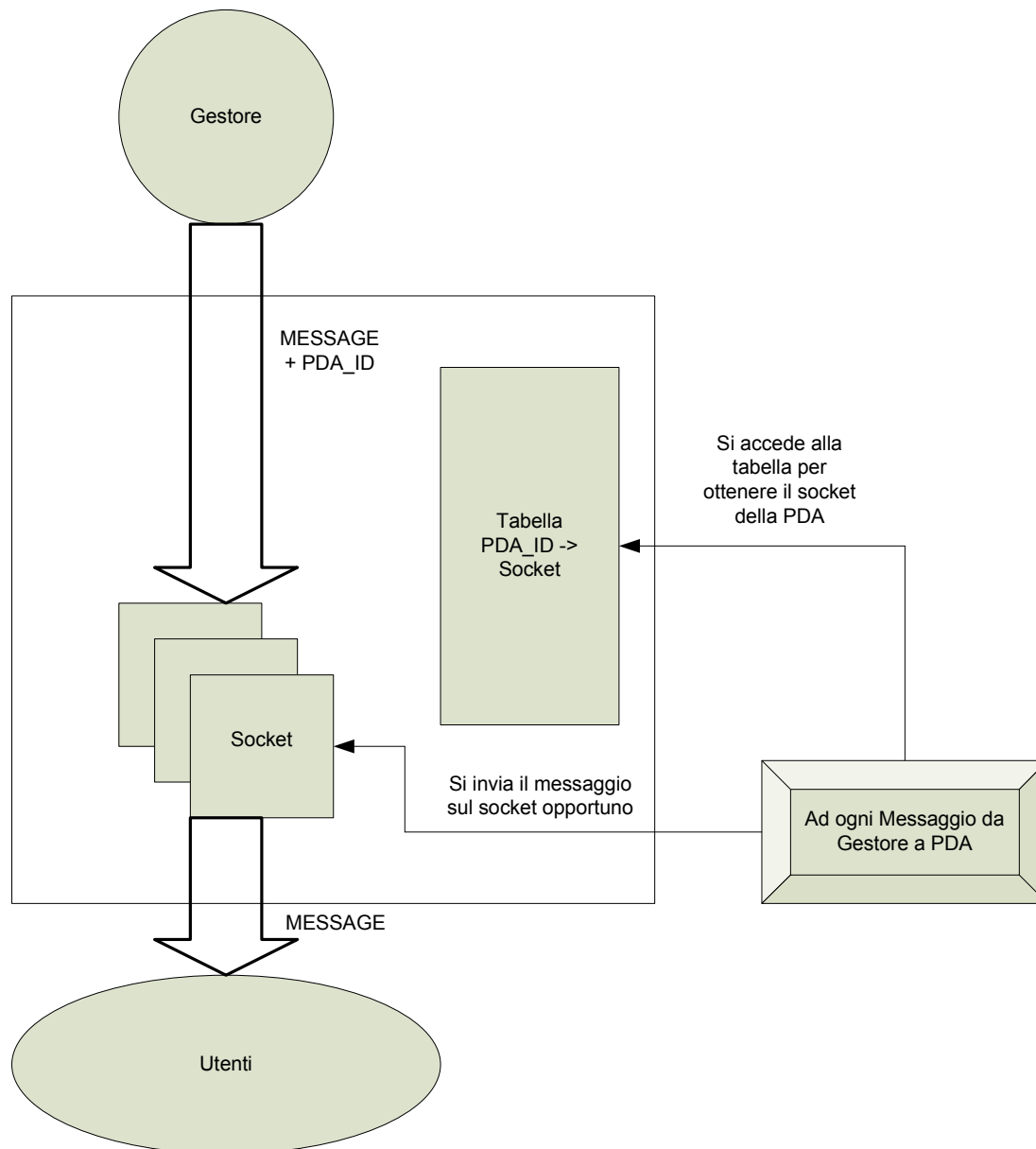
Ad ogni tentativo affida alla libreria il MESSAGGIO e la PDA_ID destinazione.

La libreria cerca all'interno della tabella il socket associato a tale PDA_ID; se non lo trova avverte il server che continuerà nella serie dei tentativi.

Se il socket viene trovato, la libreria inoltra il MESSAGGIO attraverso di esso per raggiungere la PDA.

Nel caso di errore nell'invio del messaggio il server viene avvertito; anche in questo caso continuerà con la serie di tentativi per l'invio del messaggio.

Per l'invio multiplo a diverse PDA la libreria accetta anche liste di PDA_ID destinatari. Il procedimento è lo stesso, ma replicato per ogni destinatario.



Disconnessione

Se un utente si disconnette (e questo può avvenire senza preavviso, ossia senza alcun messaggio di avviso) la voce viene cancellata e sarà ricreata alla creazione del nuovo socket quando la PDA si riconnette al server.

TESTBED

Il dimostratore TRIP è stato realizzato per rendere fruibili alcuni dei servizi che la piattaforma può garantire alla messa in opera.

Un primo aspetto prevede l'accesso da parte dell'utente al sistema TRIP attraverso le diverse interfacce. L'utente è in grado di ottenere informazioni dal sistema attraverso:

- Portale web
- Portale web scalato per terminali mobili
- Richieste info via SMS

Questa parte del test-bed mette in evidenza la capacità del sistema di gestire diverse forme di accesso e di rendere disponibile in maniera scala le informazioni che l'utente richiede.

Un secondo aspetto riguarda la simulazione di un evento di traffico all'interno della rete controllata dal sistema. Tale evento viene preso in carica dal sistema che:

- Aggiorna il database degli orari
- Avvisa l'utenza interessata

Questa parte del test-bed mostra la funzione di controllo e gestione di eventi in real-time da parte del sistema TRIP, che quindi si dimostra essere un sistema "attivo" che non è solo in grado di rispondere alle richieste dell'utente, ma è capace di suggerire informazioni utili all'utente quando questi ne può beneficiare.

Architettura del test-bed

Il testbed è organizzato in modo da fornire tre tipologie di accesso al sistema. L'architettura prevede quindi un insieme di server (che eventualmente possono coesistere su una singola macchina) che gestiscono rispettivamente:

- Il portale web, informazioni GIS e dati d'utenza
- il database degli orari e il motore di calcolo dei percorsi minimi
- gestione della connessione SMS

Gli altri apparati del sistema sono quindi i dispositivi a disposizione dell'utente:

- personal computer connesso via LAN o WLAN
- palamre o smartphone connesso via WLAN o UMTS
- normale telefono cellulare per comunicazione SMS

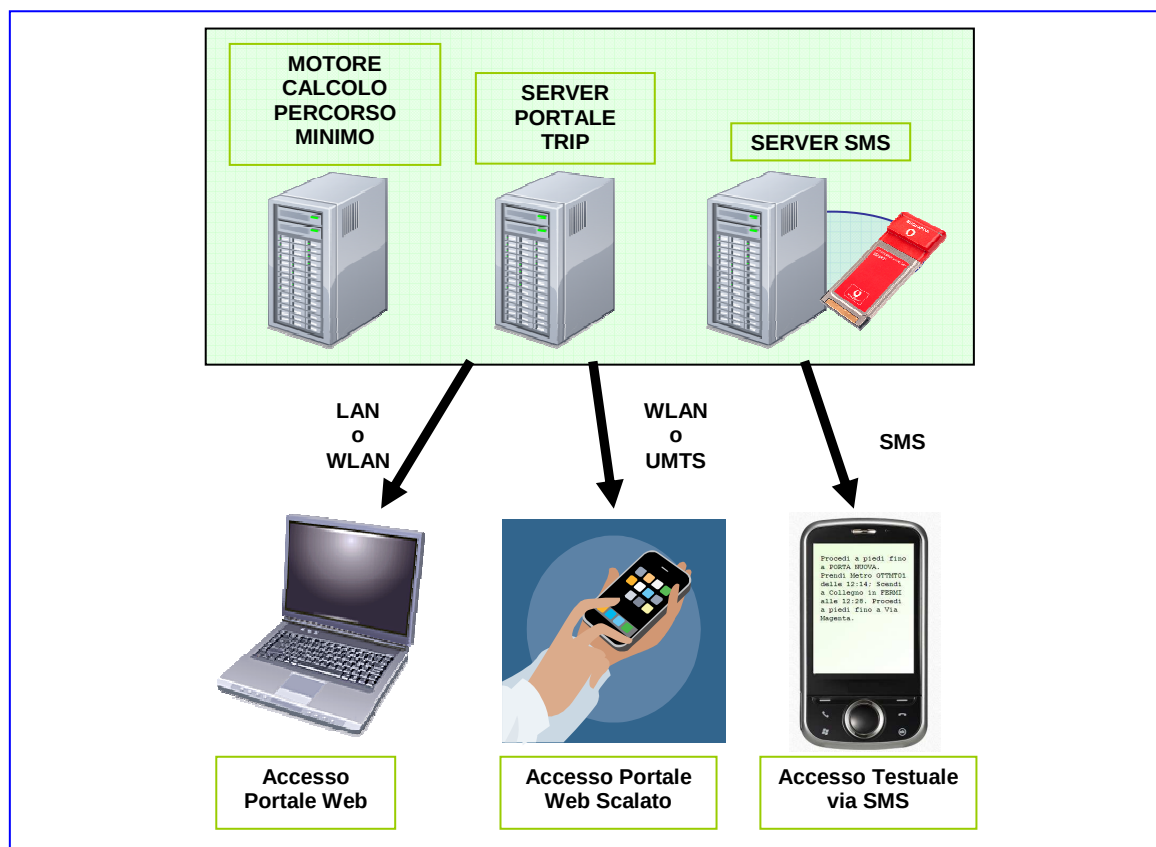


Fig. 216 - Architettura del testbed

Prima fase

Portale Web

L'utente accede ai servizi di TRIP utilizzando un PC connesso al server TRIP attraverso rete cablata o rete LAN wireless ad alta capacità (es. hotspot wi-fi).

1. L'utente accede quindi al portale TRIP con la veste grafica estesa ed effettua una richiesta di percorso.
Ottendrá diverse soluzioni che potrà visualizzare in forma grafica completa: percorso dei mezzi visualizzato su mappe geografiche con
 - Fermate di partenza, arrivo e cambio mezzi
 - Possibilità di info aggiuntive attraverso iterazione (popup e simili)
2. L'utente accede alle informazioni relative alle singole corse dei mezzi (disegno del percorso e orari dei passaggi) coinvolti nel progetto (bus, tram, metro, treni); può effettuare richieste relative ad orari tabellari di una linea o richieste dei passaggi ad una specifica fermata.
3. L'utente accede ad informazioni di contorno come accesso ai parcheggi, presenza di cantieri sulle strade, informazioni sul traffico stradale
4. L'utente accede utilizzando username e password create come utente dimostrativo ed è in grado di visualizzare nell'area utente richieste precedenti con possibilità di ricalcolare allo stato attuale

Portale Web Scalato

L'utente accede ai servizi di TRIP utilizzando un terminale mobile tipo smartphone capace di visualizzare attraverso browser pagine web. La connessione con il server TRIP avviene attraverso rete LAN wireless ad alta capacità (es. hotspot wi-fi) oppure attraverso rete 3G (UMTS).

1. L'utente accede quindi al portale TRIP con la veste grafica ridotta (risoluzione diminuita) e scalata (minor numero di comandi più facilmente accessibili) ed effettua una richiesta di percorso.
Otterrà diverse soluzioni che potrà visualizzare in forma grafica completa: percorso dei mezzi visualizzato su mappe geografiche con
 - Fermate di partenza, arrivo e cambio mezzi
 - Possibilità di info aggiuntive attraverso iterazione (popup e simili)
2. L'utente accede alle informazioni relative alle singole corse dei mezzi (disegno del percorso e orari dei passaggi) coinvolti nel progetto (bus, tram, metro, treni); può effettuare richieste relative ad orari tabellari di una linea o richieste dei passaggi ad una specifica fermata.
3. L'utente accede ad informazioni di contorno come accesso ai parcheggi, presenza di cantieri sulle strade, informazioni sul traffico stradale
4. L'utente accede utilizzando username e password create come utente dimostrativo ed è in grado di visualizzare nell'area utente richieste precedenti con possibilità di ricalcolare allo stato attuale

Accesso via SMS

L'utente accede ai servizi di TRIP attraverso singole richieste testuali inviate sottoforma di SMS. Il SMS viene scritto ed inviato attraverso un comune telefono GSM.

Ci sono due casi di esempio:

1. L'utente effettua la richiesta di percorso, quindi invia informazioni come punto di partenza e punto di arrivo; l'utente riceve un SMS di risposta con le istruzioni per effettuare il viaggio
2. L'utente richiede gli orari di passaggio ad una fermata/stazione inviando un SMS con il codice identificativo della fermata/stazione di interesse; l'utente riceve un SMS di risposta con gli orari di passaggio delle linee che effettuano fermata in quella fermata/stazione.

Seconda fase

In questa seconda fase si mostrano le capacità del sistema TRIP di analizzare le informazioni ricevute dai mezzi per determinare eventi di congestione e quindi avvertire l'utenza interessata.

Registrazione di un percorso

Come primo passo l'utente accede al sistema attraverso il portale web e si registra lasciando tra gli altri dati il proprio numero di cellulare.

Quindi effettua una richiesta di percorso. Oltre a visualizzarne il risultato, l'utente registra tale percorso per avere informazioni in tempo reale in caso di eventi che impattano sui mezzi utilizzati.

Generazione evento ritardo

Attraverso il portale di gestione viene introdotto nel sistema un evento ritardo simulando il fatto che un mezzo abbia accumulato del ritardo.

Tale evento viene processato dal sistema che aggiorna le tabelle degli orari e si prepara ad inviare eventuali messaggi di allerta agli utenti interessati.

Ricezione SMS di avviso

L'evento introdotto nel sistema ha un impatto sul percorso scelto dall'utente e quindi lo avverte con un messaggio SMS sul numero di cellulare lasciato in fase di registrazione. Il messaggio conterrà informazioni su eventuale ritardo oppure sul nuovo percorso da scegliere per cercare di compensare l'evento di ritardo.

Manuale del Simulatore WiMAX LLS per piattaforma SHINE

Introduzione

Il seguente documento ha l'intento di descrivere nel dettaglio il simulatore di rete d'accesso WiMAX basato sullo standard IEEE802.16e-2005 realizzato nel laboratorio WiLab, IEIIT-BO/CNR della sezione di Bologna dall'Ing. Giacomo Leonardi.

Il simulatore dei livelli inferiori (Lower Layer Simulator) con tecnologia IEEE802.16e WirelessMAN-OFDMA nelle versioni FDD e TDD, è compatibile con la piattaforma di simulazione SHINE (Simulation platform for Heterogeneous Interworking Networks) sviluppata dal WiLab di Bologna. Nel simulatore è implementato con attenzione il livello fisico, con modello semi-analitico. La codifica convoluzionale è solo di tipo hard; mancano invece i turbo codici. Sono implementati MAC ed RLC. Per gestire la frammentazione, ARQ, allocazione delle risorse con modalità PUSC, FUSC, ecc. Call Admission Control e Scheduling sono stati implementati con attenzione a traffico UGS (deterministico) e Best Effort; devono essere gestiti gli altri tipi di traffico. Sono stati implementati algoritmi di Link Adaptation.

Nel tempo sono state introdotte su tale simulatore numerose funzionalità ed attualmente è in grado di simulare con relativa efficienza dei tempi di simulazione e precisione reti WiMAX rispondenti allo standard attualmente più significativo: IEEE802.16e-2005.

Le numerose funzionalità verranno nel seguito descritte e illustrate in modo approfondito ma è bene precisare in questa parte introduttiva che l'area applicativa di tale simulatore fa riferimento a reti di accesso di tipo fisso e mobile. E' valida l'ipotesi che tutte le BaseStation sono collegate direttamente alla Core Network.

L'utilizzo del simulatore WiMAX richiede la cooperazione con la piattaforma SHINE (Simulation platform for Heterogeneous Interworking Networks) per la simulazione dei livelli superiori al MAC (Rete, Trasporto e Applicazione).

Sono disponibili back-up di tutte le versioni significative, ma si fa qui riferimento all'ultima versione prodotta e testata.

Assunzioni

Gestione dell'asse temporale

La modalità di funzionamento implementata per il livello fisico è l'accesso multiplo a divisione di frequenza OFDMA.

I parametri che caratterizzano i simboli OFDMA sono principalmente:

- N_{OFDMA} : numero di sotto-portanti utilizzate per la tecnica OFDM
- BW : valore nominale della larghezza di banda del canale
- n : fattore di campionamento, viene fissato pari a 8/7

- $G = (T_g / T_b)$: rapporto tra il tempo di CP e quello utile (valori di G: 1/4, 1/8, 1/16, 1/32)
- N_{FFT} : numero di portanti complessive, può variare da 128 a 2048

poiché da questi è possibile ricavare tutti gli altri:

- F_s : frequenza di campionamento, $F_s = \text{floor}(n \cdot BW / 8000) \cdot 8000$
- $\Delta f = F_s / N_{FFT}$: spaziatura tra le portanti,
- $T_b = 1 / \Delta f$: tempo utile,
- $T_g = G \cdot T_b$: tempo di CP,
- $T_s = T_b + T_g$: durata di un simbolo OFDM

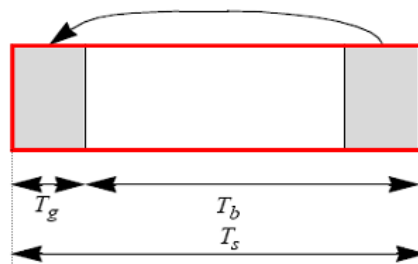
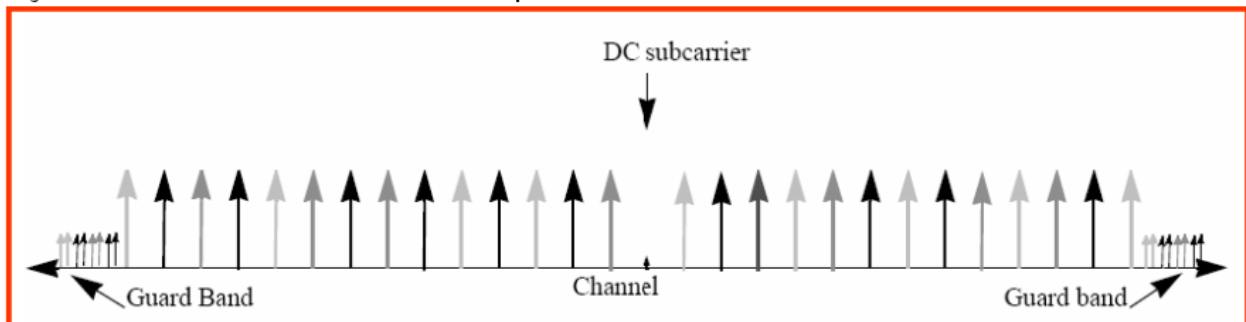


Fig 1. Simbolo OFDMA:
descrizione nel dominio del tempo

Fig 2. Simbolo OFDMA: descrizione nel dominio delle frequenze

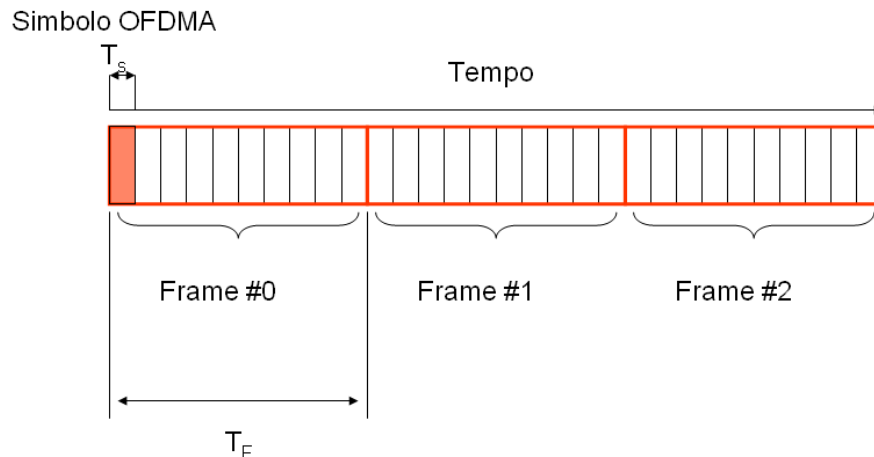


I simboli OFDMA sono raggruppati in Frame, le cui durate possibili (T_F) sono: 2.5, 4, 5, 10, 15, 20 ms.

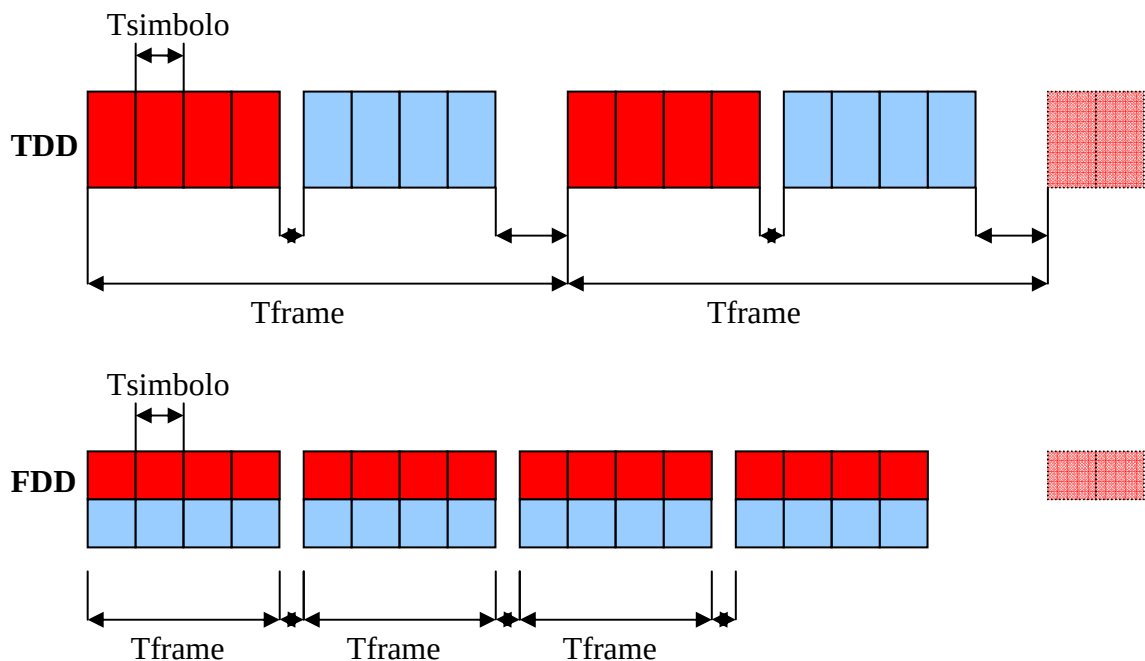
Il numero di simboli OFDMA all'interno del frame ($N_{OFDMA} = \text{floor}(T_F / T_s)$) varia con:

- la durata del frame (arbitraria)
- la durata del simbolo ($f(BW, G, N_{FFT})$)

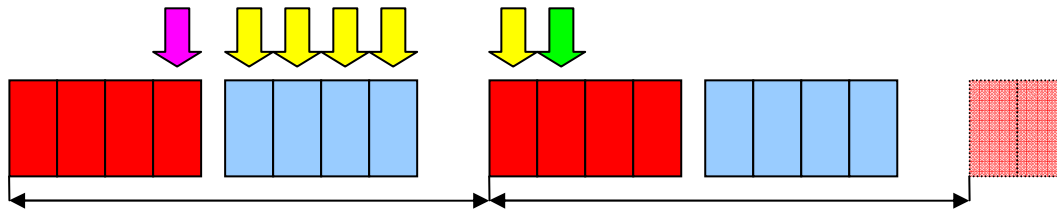
Per cui N_{OFDMA} può variare tra 10 e 150 simboli.



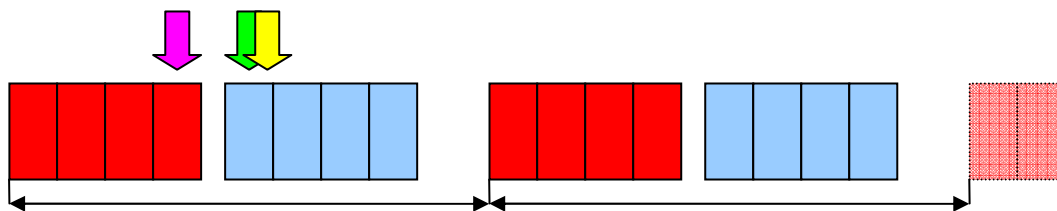
Il simulatore si basa sul livello fisico OFDMA definito dallo standard IEEE802.16e-2005. L'unità fondamentale nella scala dei tempi è il Tempo di simbolo OFDMA. Nel caso di duplexing di tipo TDD un certo numero di simboli riservati al Downlink e all'Uplink formano una struttura chiamata Frame di durata T_{frame} . Nel caso FDD il numero di simboli in Downlink e Uplink è lo stesso e ciascuno di esso rappresenta per intero un Frame.



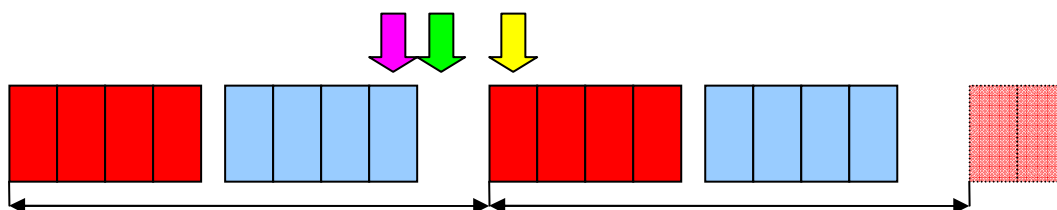
Per la gestione della corretta esecuzione temporale degli eventi interni al LLS (p.es. trasmissione dei singoli bit) ed esterni (p.es. trasmissione di un pacchetto IP) si è adottata la logica delle “bandierine” o eventi così come sono state definite nella piattaforma di simulazione SHINE. Di seguito vengono riportati alcuni esempi di eventi di livello Rete (interni) e di livello Trasporto (esterni).



CASO A) A partire dall'istante attuale (VIOLA), devo eseguire tutti gli eventi di livello Rete (GIALLO) (trasmissione simboli e/o mappatura) fino al prossimo evento di livello Trasporto (VERDE)



CASO B) A partire dall'istante attuale (VIOLA), devo eseguire l'evento di livello Trasporto (VERDE) prima di tutti gli eventi di livello Rete (GIALLO)



CASO C) Caso particolare: anche in questo caso a partire dall'istante attuale (VIOLA), devo eseguire l'evento di livello Trasporto (VERDE) prima di tutti gli eventi di livello Rete (GIALLO)

Nel simulatore WiMAX a livello PHY simuliamo e valutiamo l'esito di tutti i bit ricevuti, per ciascun ricevitore, associati a ciascun simbolo della modulazione, per ciascuna sottoportante, per ogni simbolo OFDM della trasmissione.

Un esempio numerico:

- Tempo Simulato: 5 min [Nframe=30000, Tf=10ms]
- Ogni Frame: 30 simboli OFDM [Nsimboli =30]
- Ogni simbolo OFDM: 1536 sotto-portanti dati [Ndata=1536]
- Numero di bit valutati = Nutenti x Nframe x Nsimboli x Ndata = Nutenti x 1.382Gbit

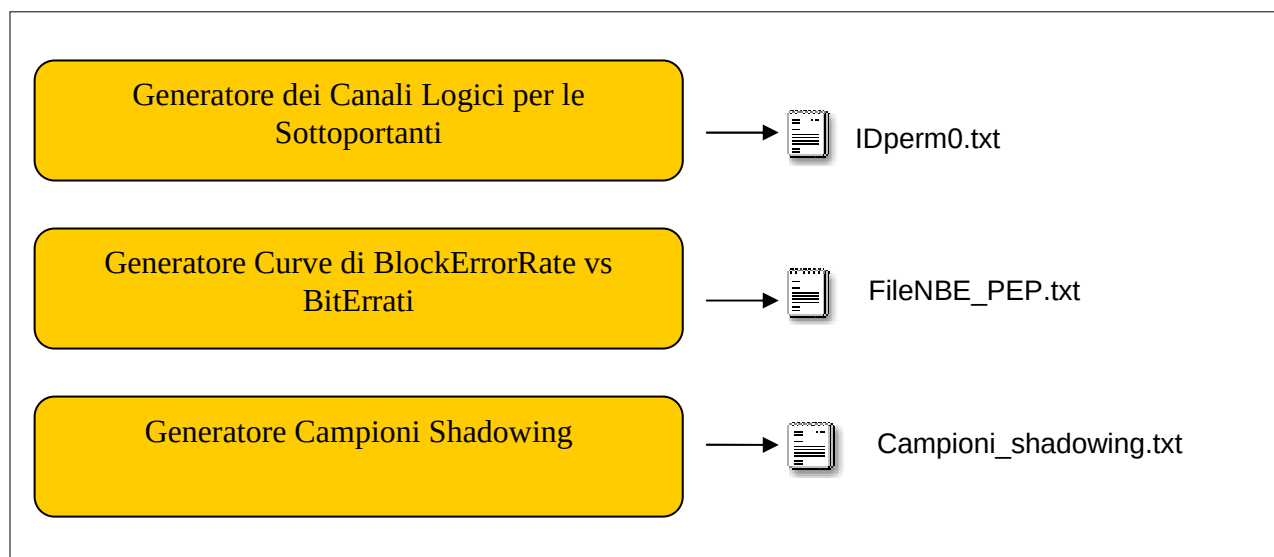
Per simulare cioè 5 minuti di traffico wimax per ciascun utente, simuliamo la corretta ricezione e trasmissione di 1.4 miliardi di bit.

Prima di simulare

In questo paragrafo verrà fornito l'elenco delle relazioni con altri simulatori/programmi utility da eseguire off-line. A seguire verrà illustrata una descrizione sommaria di quali file vanno generati prima di poter iniziare la simulazione. La descrizione puntuale dei file sarà fatta in un capitolo successivo.

Relazioni con altri software

Elenco delle relazioni con altri simulatori/programmi utility da eseguire off-line:



Per la descrizione dei campi dei file che verranno forniti al simulatore come input, si faccia riferimento al capitolo con la descrizione dei file di input. Il funzionamento dei software necessari per generare questi file non è al momento disponibile.

Lanciare il simulatore da riga di comando

Tutti i parametri sono definiti all'interno dell'obbligatorio file di configurazione descritto nella sezione che segue, però possono essere modificati quando si lancia l'eseguibile da riga di comando. Possono essere modificati tramite riga di comando tutti i parametri del file di configurazione, però la modifica non avrà influenza sul file di configurazione, che resterà invariato, ma solo sulla simulazione corrente. La sintassi per modificare i valori del file di configurazione principale è la seguente:

```
>WiMAX.exe Path_Relativo\MainConfig.txt ID_Parametro ValParametro
```


Il primo argomento, come si può vedere, rappresenta il file di configurazione di cui si vogliono modificare alcuni campi. L'ID parametro rappresenta univocamente l'elemento del file di configurazione che si vuole inserire tramite riga di comando. Il valore del parametro deve essere sensato e rispettare le caratteristiche sintattiche descritte nella sezione che segue.

Per avere una lista completa dei nomi validi per questa operazione si digiti da finestra comandi DOS:

```
>wiMAX.exe ?
```

questo l'output:

```
C:>wimax ?
Help del programma wiMAX.exe

Uso del programma:
Eseguibile      [FileCfg      [Nome_Param_1      Valore_Param_1      [Nome_Param_2
Valore_Param_2...]]]

Nome del file di configurazione di default: MainConfig.txt

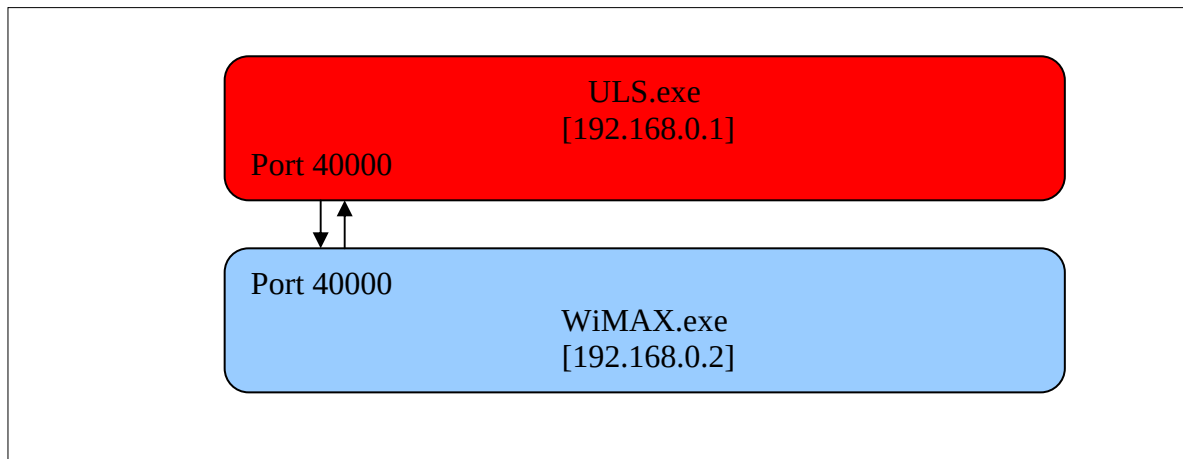
Parametri validi (e tipi):
      Nome Server      stringa
      Porta            int
      Tipo Rete        stringa
      Seed             int
      Primo X          long
      Primo Y          long
      Secondo X        long
      Secondo Y        long
      Cartella CONFIG  stringa
      Cartella FUSC    stringa
      Cartella PUSC    stringa
      Cartella UL_PUSC stringa
      Cartella USCITE  stringa
      File Parametri   stringa
      File Opzioni     stringa
      File AntenneCfg  stringa
      File Stazioni BS stringa
      File Pathloss    stringa
      File Canale      stringa
      File Soglia NBE_PEP stringa
      File Shadowing   stringa
      File Channelling stringa
      File TrafficMap  stringa
      File LA Algorithm stringa
      File CAC Algorithm stringa
      Version Number   stringa
      Release Data     stringa
      Developer        stringa
      Key              stringa
```

Premere un tasto...

Questo modo di attribuire i valori dei parametri è estremamente utile quando si vogliono lanciare in automatico molte simulazioni, attraverso file BATCH, modificando alcuni parametri da una simulazione alla successiva.

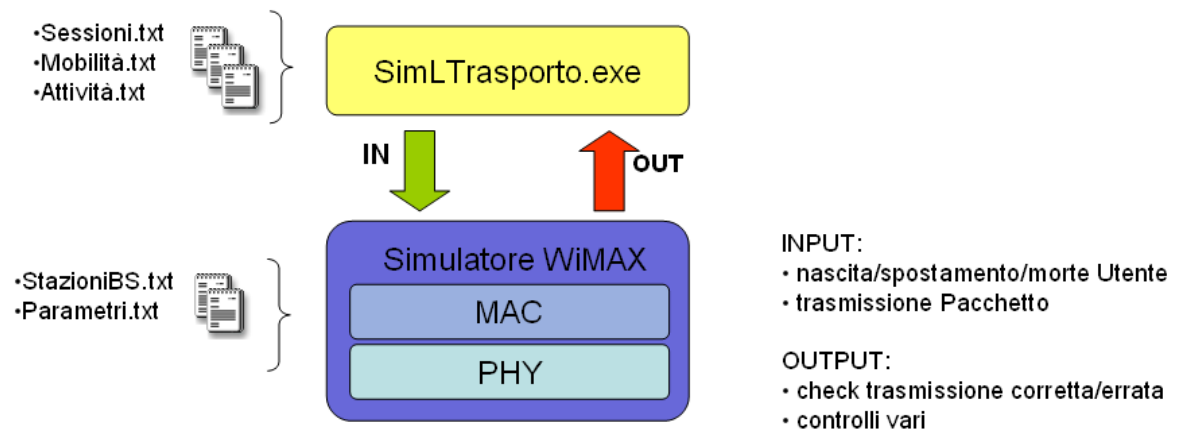
Schema a blocchi del simulatore

Il simulatore WiMAX IEEE802.16 è realizzato su un unico eseguibile che permette di simulare entrambi gli strati OSI MAC e PHY, ma il funzionamento richiede una piattaforma SHINE, operante come server, mentre il simulatore WiMAX agirà come client. Prima di far partire l'eseguibile WiMAX.exe è necessario aprire sulla stessa porta e con l'opportuno indirizzo IP l'eseguibile del simulatore dei livelli superiori (Upper Layer Simulator, ULS.exe), come illustrato in figura.



Il Simulatore WiMAX è stato quindi pensato e implementato per integrarsi con il Simulatore di Rete (con logica ad eventi):

- utilizza le librerie di comunicazione con il Simulatore di Livello Trasporto
- sfrutta le conoscenze già maturate in questo campo



Il Simulatore WiMAX si occupa quindi:

- di gestire la nascita/spostamento/morte di un utente nello scenario
- di ricevere i pacchetti da trasmettere (DL/UL) dal SLT
- di provvedere opportunamente a tutte le operazioni necessarie al trasferimento fisico del pacchetto
- di notificare quando un pacchetto è stato correttamente trasmesso (o è andato perduto)

A partire da una distribuzione opportunamente fissata delle stazioni radiobase [BS] sul territorio, vengono collocate dal Simulatore di Livello Trasporto due tipologie di utenti

(domestici [SS] e mobili [MS]) con diversa caratterizzazione. Di seguito è riportata una tabella con le grandezze tipiche utilizzate per le simulazioni, per i diversi elementi attivi del simulatore BS, SS e MS-

Tipologia	Mobilità	Numero	Altezza	Antenna Gain	ERP
Stazioni radiobase [BS]	NO	Fissato (File)	10..22 m	15 dB (Ampiezza: 120°)	40 dBmW
Utenti domestici [SS]	NO	Runtime (SLT)	10 m	15 dB (Ampiezza: 45°)	27 dBmW
Utenti mobili [MS]	SI	Runtime (SLT)	2 m	5 dB (Omni)	20 dBmW

Per ogni installazione [x,y] di una **BS** nello scenario posso prevedere:

- altezza variabile tra 10 e 22 metri
- N antenne settoriali, con ampiezza a piacere
- orientamento configurabile (fissando direzione massimo)
- frequenza configurabile
- lettura da file, grado per grado, del valore del guadagno dell'antenna.
- 15 dBi nella direzione di massimo (esempio)
- ERP 40 dBmW (esempio)

Gli utenti vengono fatti nascere nelle posizioni [x,y] dello scenario indicate dal file di Mobilità, e passate attraverso la socket con SHINE.

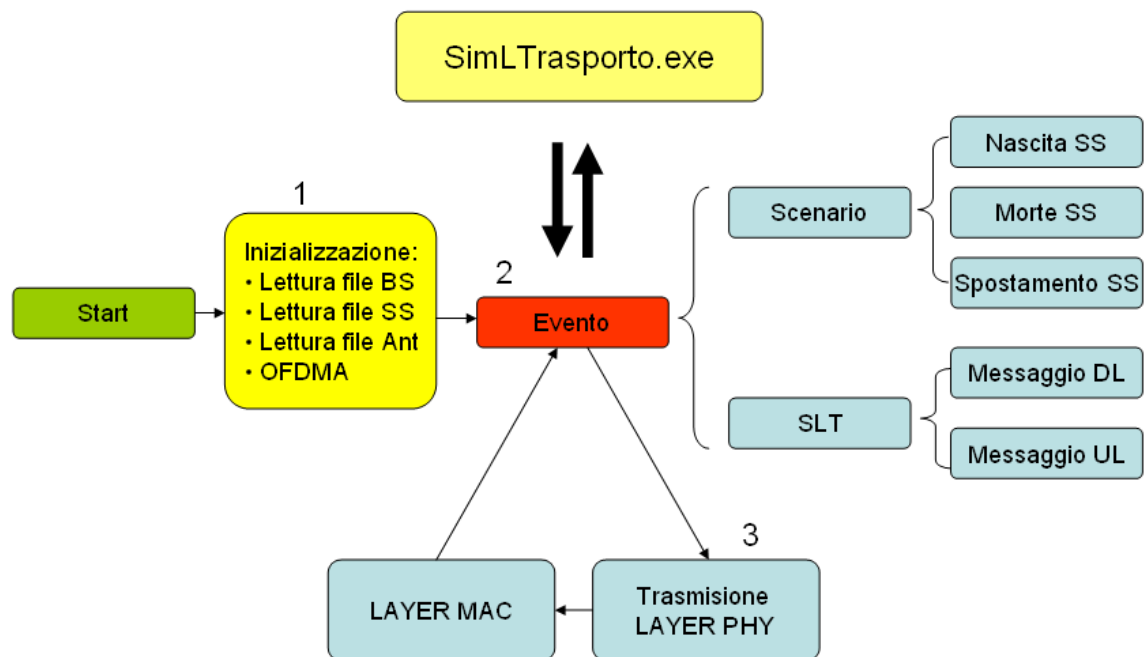
Le stazioni **SS** sono caratterizzate da:

- antenna direttiva: 45° ampiezza
- orientata verso la BS, Gain=15dB
- lettura da file, grado per grado, del valore del guadagno dell'antenna
- frequenza configurabile con aggancio alla BS
- ERP 27 dBmW

Le stazioni **MS** sono caratterizzate da:

- antenna omnidirezionale: 360° ampiezza
- Gain=5dB
- frequenza configurabile con aggancio alla BS
- ERP 20 dBmW

Lo schema a blocchi principale del simulatore è il seguente:

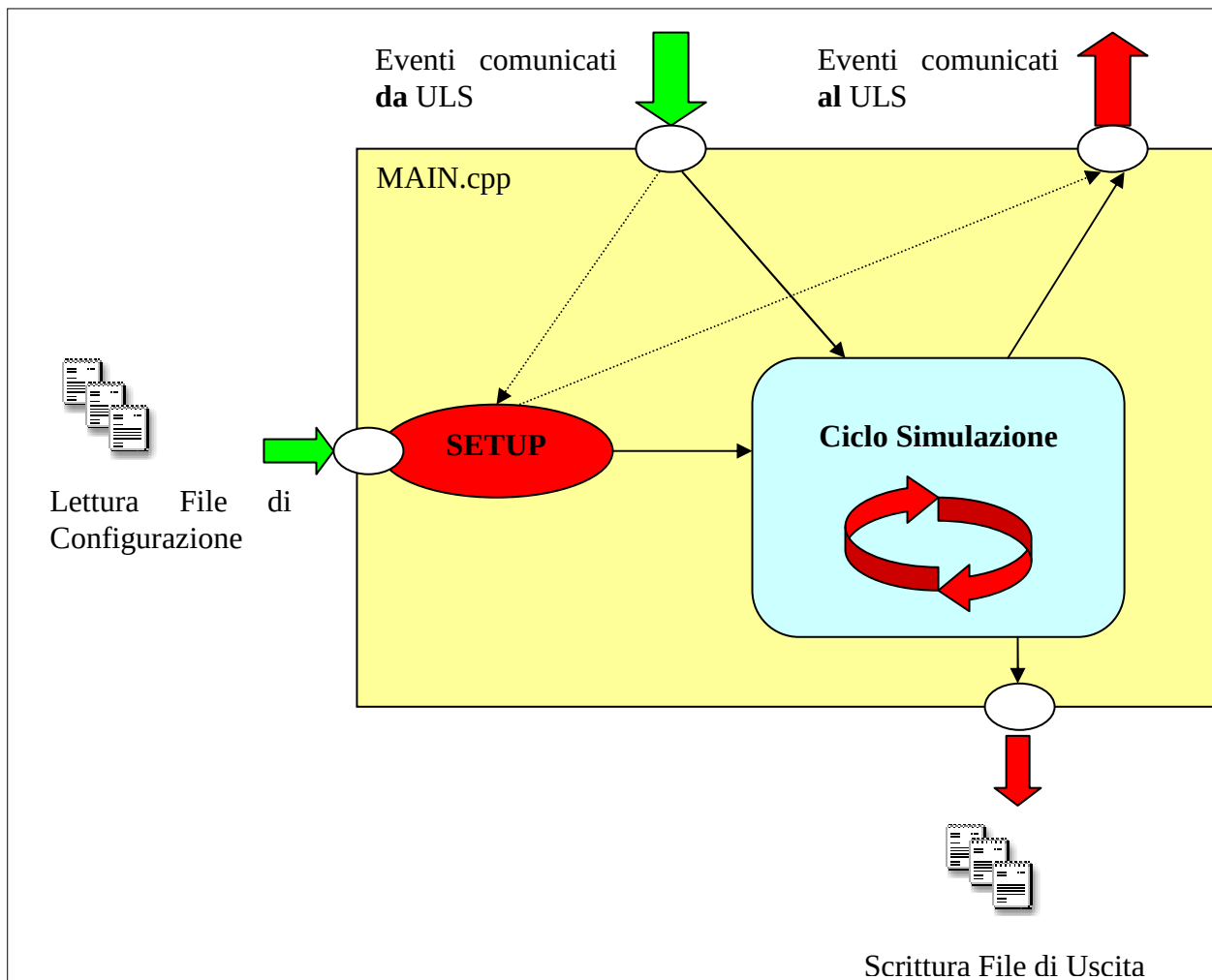


Nello schema si può vedere che una volta fatto partire il simulatore (start) c'è una fase di inizializzazione in cui vengono letti e caricati in memoria tutti i file con le impostazioni delle BS, SS e MS, le antenne e i parametri di livello fisico per l'OFDMA.

Dopodiché il simulatore resta in attesa di un evento dal SLT altrimenti prosegue il processamento consecutivo di tutti i simboli OFDMA uno dopo l'altro, calcolando la correttezza dei bit trasmessi nel simbolo e cercando di capire se è possibile prendere decisioni anche sulla correttezza delle PDU mappate sulle data region che terminano in quel particolare simbolo. Vengono ovviamente effettuate anche tutte le operazioni connesse al livello MAC che interessano le PDU effettivamente trasmesse/ricevute.

Per la gestione dei diversi tipi di eventi si faccia riferimento al manuale di SHINE.

E' possibile quindi individuare nello schema a blocchi una fase di setup e una di simulazione vera e propria che viene eseguita simbolo OFDMA per simbolo, così come riportato nella figura di seguito.



Le operazioni che vengono eseguite durante il ciclo di simulazione vero e proprio sono:

- mappatura: all'inizio di ogni nuovo frame vengono mappate tutte le PDU, in attesa negli appositi buffer di essere trasmesse sulla matrice OFDMA tempo-frequenza
- trasmissione: all'inizio di ogni nuovo simbolo OFDMA vengono valutati i bit e le PDU che risultano essere state trasmesse per intero nel simbolo in esame
- ARQ: tutte le PDU che vengono perse scatenano la necessità di essere riaccodate per essere trasmesse appena il ricevitore si accorgerà di un buco nella sequenza delle PDU ricevute
- Frammentazione: tutti i pacchetti IP passati dal SLT vengono frammentati in ARQ e viceversa dopo la corretta ricezione di tutti gli ARQ facenti parte di un singolo IP viene acknowledged al SLT

Livello Fisico

Il livello fisico concerne la:

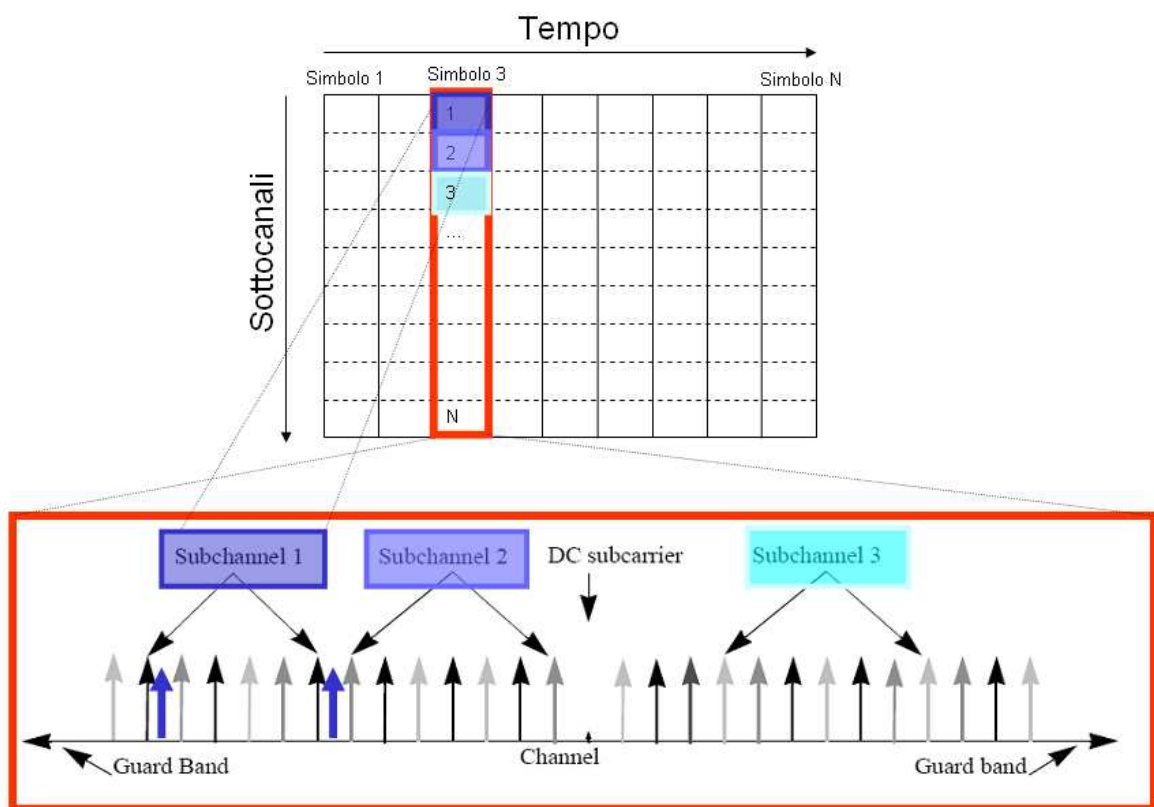
- caratterizzazione del livello fisico trasmissivo (OFDMA, sottocanali ecc)
- caratterizzazione degli apparati di trasmissione e ricezione
- caratterizzazione del canale wireless
- elaborazione del flusso binario (codifica e modulazione)

Livello Fisico Trasmissivo

La modalità di funzionamento adottata è l'accesso multiplo a divisione di frequenza OFDMA. I parametri che caratterizzano i simboli OFDMA che è possibile ricavare da tutti gli altri sono:

- F_s : frequenza di campionamento, $F_s = \text{floor}(n \cdot BW / 8000) \cdot 8000$
- $\Delta f = F_s / N_{FFT}$: spaziatura tra le portanti,
- $T_b = 1 / \Delta f$: tempo utile,
- $T_g = G \cdot T_b$: tempo di CP,
- $T_s = T_b + T_g$: durata di un simbolo OFDM

La suddivisione in sottocanali è effettuata mediante il raggruppamento delle sottoportanti, secondo schemi fissati. A seconda della modalità di allocazione delle sottoportanti il numero di sottocanali può variare:



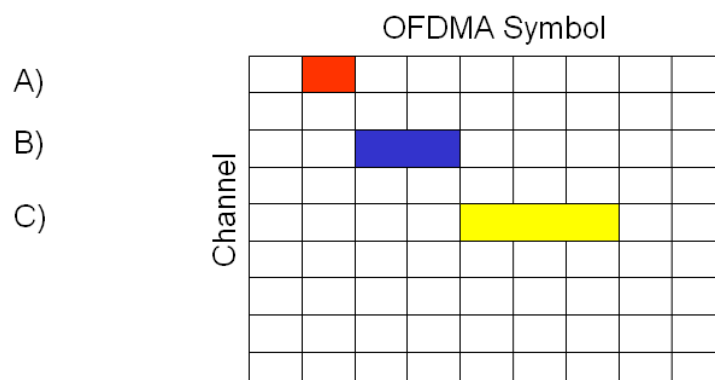
Uno **Slot** inteso nel OFDMA PHY richiede per essere definito sia una dimensione temporale che il sottocanale (frequenza) su cui è allocato ed è la minima unità possibile di allocazione dei dati. La definizione di uno slot-OFDMA dipende dalla struttura del simbolo OFDMA, che varia da uplink a downlink, da FUSC a PUSC, e dalla permutazione delle sottoportanti distribuite o adiacenti.

A) Per downlink_FUSC uno slot è un sottocanale per un simbolo OFDMA.

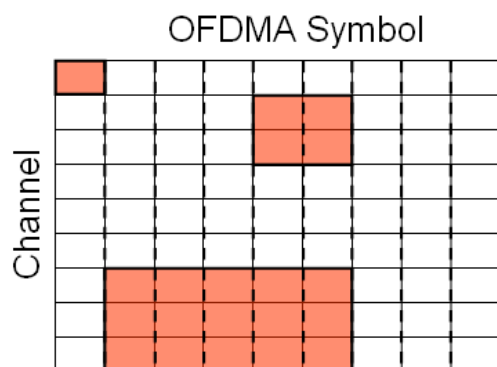
B) Per downlink_PUSC uno slot è un sottocanale per due simboli OFDMA.

C) Per uplink PUSC uno slot è un sottocanale per tre simboli OFDMA.

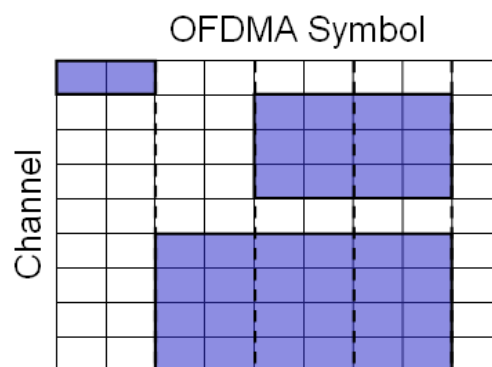
D) Per la permutazione delle portanti adiacenti uno slot è un sottocanale per uno, due o tre simboli OFDMA.



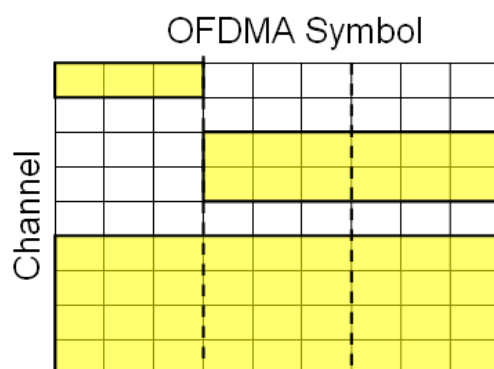
Nell'OFDMA, una **Data Region** è una allocazione bidimensionale di un gruppo di sottocanali adiacenti in un gruppo di simboli OFDMA continui. Questa allocazione può essere visualizzata come un rettangolo.



A) fattore_X=1



B) fattore_X=2



C) fattore_X=3

Le successive tabelle illustrano che indipendentemente dalla modalità di funzionamento, ogni slot ha una disposizione 48 sottoportanti, allocate tutte in un simbolo (caso A), su due simboli (caso B) oppure su tre (caso C).

La ripartizione è illustrata in opportuni schemi riportati dallo standard.

A) Downlink FUSC

	128	512	1024	2048
# used	107	427	851	1703
# channel	2	8	16	32

SYMBOL #0

Data subcarriers: 1536

Pilot subcarriers: 166

Data subcarriers per channel: **48**

B) Downlink PUSC

	128	512	1024	2048
# used	85	421	841	1681
# channel	3	15	30	60

SYMBOL #0

Data subcarriers: 1440

Pilot subcarriers: 240

Data subc. per channel: **24**

SYMBOL #1

Data subcarriers: 1440

Pilot subcarriers: 240

Data subc. per channel: **24**

C) Uplink PUSC

	128	512	1024	2048
# used	97	409	841	1681
# channel	4	17	35	70

SYMBOL #0

Data subcarriers: 840

Pilot subcarriers: 840

Data subc. per channel: **12**

SYMBOL #1

Data subcarriers: 1680

Pilot subcarriers: 0

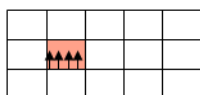
Data subc. per channel: **24**

SYMBOL #2

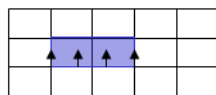
Data subcarriers: 840

Pilot subcarriers: 840

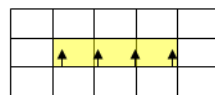
Data subc. per channel: **12**



A) 48 carriers



B) 24+24 carriers



C) 12+24+12 carriers

= 48!

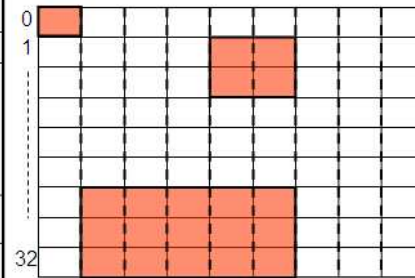
Un'interpretazione delle modalità di allocazione delle sottoportanti può essere:

PUSC: suddivide le portanti tra i sottocanali e le distribuisce in frequenza in modo diverso per ogni settore minimizzando e mediando l'interferenza. La distribuzione in frequenza serve anche a garantire una certa diversità di frequenza. In uplink può introdurre una rotazione con effetti simili al frequency hopping del CDMA.

FUSC: Simile al PUSC, massimizza l'effetto di diversità e aggiunge flessibilità all'allocazione

Parameter	Value	Comments
Number of DC Subcarriers	1	Index 1024
Number of Guard Subcarriers, Left	173	
Number of Guard Subcarriers, Right	172	
Number of Used Subcarriers (N_{used})	1703	Number of all subcarriers used within a symbol, including all possible allocated pilots and the DC carrier.
Pilots		
VariableSet #0	71	0,72,144,216,288,360,432,504,576,648,720,792,864,936,1008,1080,1152,1224,1296,1368,1440,1512,1584,1656,48,120,192,264,336,408,480,552,624,696,768,840,912,984,1056,1128,1200,1272,1344,1416,1488,1560,1632,24,96,168,240,312,384,456,528,600,672,744,816,888,960,1032,1104,1176,1248,1320,1392,1464,1536,1608,1680
ConstantSet #0	12	39,645,1017,1407,330,726,1155,1461,351,855,1185,1545
VariableSet #1	71	36,108,180,252,324,396,468,540,612,684,756,828,900,972,1044,1116,1188,1260,1332,1404,1476,1548,1620,1692,12,84,156,228,300,372,444,516,588,660,732,804,876,948,1020,1092,1164,1236,1308,1380,1452,1524,1596,1668,60,132,204,276,348,420,492,564,636,708,780,852,924,996,1068,1140,1212,1284,1356,1428,1500,1572,1644
ConstantSet #1	12	261,651,1143,1419,342,849,1158,1530,522,918,1206,1701
Number of data subcarriers	1536	
Number of data subcarriers per subchannel	48	
Number of Subchannels	32	
PermutationBase		3, 18, 2, 8, 16, 10, 11, 15, 26, 22, 6, 9, 27, 20, 25, 1, 29, 7, 21, 5, 28, 31, 23, 17, 4, 24, 0, 13, 12, 19, 14, 30

OFDMA Symbol

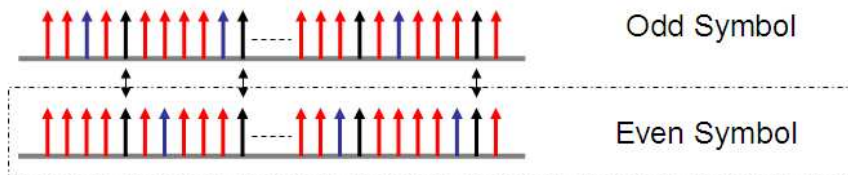
A) $N_{GS}=1$

SYMBOL#0

Data subcarriers: 1536

Pilot subcarriers: 166

Data subcarriers per channel: 48

Passo 1): Fisso Sottoportanti **Pilota Fisse** (Costant#1=12, Costant#2=12)Passo 2): Fisso Sottoportanti **Pilota Variabili** (Variable#1=71, Variable#2=71)

$$PilotsLocation = VariableSet\#x + 6 \cdot (FUSC\_SymbolNumber \bmod 2)$$

Passo 3): Divido le restanti $1702 - (12 + 12) - (71 + 71) = 1536$ sottoportanti, in sottocanali:

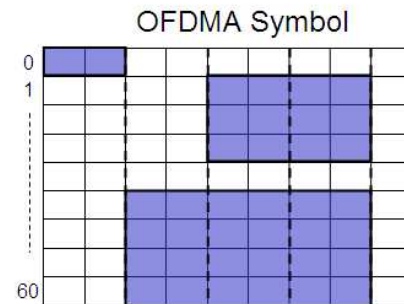
$$subcarrier(k, s) = N_{subchannels} \cdot n_k + \{p_1[n_k \bmod N_{subchannels}] + IDcell\} \bmod N_{subchannels}$$

where

$subcarrier(n, s)$ is the subcarrier index of subcarrier n in subchannel s ,
 s is the index number of a subchannel, from the set $\{0 \dots N_{subchannels}-1\}$,
 $n_k = (k + 13 \cdot s) \bmod N_{subcarriers}$,
 where k is the subcarrier-in-subchannel index from the set $\{0 \dots N_{subcarriers}-1\}$,
 $N_{subchannels}$ is the number of subchannels,
 $p_1[j]$ is the series obtained by rotating $\{PermutationBase\}$ cyclically to the left s times,
 $ceil[j]$ is the function that rounds its argument up to the next integer,
 $IDcell$ is an integer ranging from 0 to 31, which identifies the particular BS segment and is specified by MAC layer,
 $X_{mod(k)}$ is the remainder of the quotient X/k (which is at most $k-1$).

$$32Ch \times 48SubC = 1536$$

Parameter	Value	Comments
Number of DC subcarriers	1	Index 1024
Number of Guard subcarriers, Left	184	
Number of Guard subcarriers, Right	183	
Number of used subcarriers (N_{used})	1681	Number of all subcarriers used within a symbol, including all possible allocated pilots and the DC carrier.
Number of subcarriers per cluster	14	
Number of clusters	120	
Renumbering sequence	1	Used to renumber clusters before allocation to subchannels: 6, 108, 37, 81, 31, 100, 42, 116, 32, 107, 30, 93, 54, 78, 10, 75, 50, 111, 58, 106, 23, 105, 16, 117, 39, 95, 7, 115, 25, 119, 33, 71, 22, 98, 28, 79, 17, 63, 27, 72, 29, 86, 5, 101, 49, 104, 9, 68, 1, 73, 36, 74, 43, 62, 20, 84, 52, 64, 34, 60, 66, 48, 97, 21, 91, 40, 102, 56, 92, 47, 90, 33, 114, 18, 70, 15, 110, 51, 118, 46, 83, 45, 76, 57, 99, 35, 67, 53, 85, 59, 113, 11, 82, 38, 88, 19, 77, 3, 87, 12, 89, 26, 65, 41, 109, 44, 69, 8, 61, 13, 96, 14, 103, 2, 80, 24, 112, 4, 94, 0
Number of data subcarriers in each symbol per subchannel	24	
Number of subchannels	60	
PermutationBase12 (for 12 subchannels)	6,9,4,8,10,11,5,2,7,3,1,0	
PermutationBase8 (for 8 subchannels)	4	7,4,0,2,1,5,3,6



B) $N_{GS}=2$

SYMBOL #1

Data subcarriers: 1440

Pilot subcarriers: 240

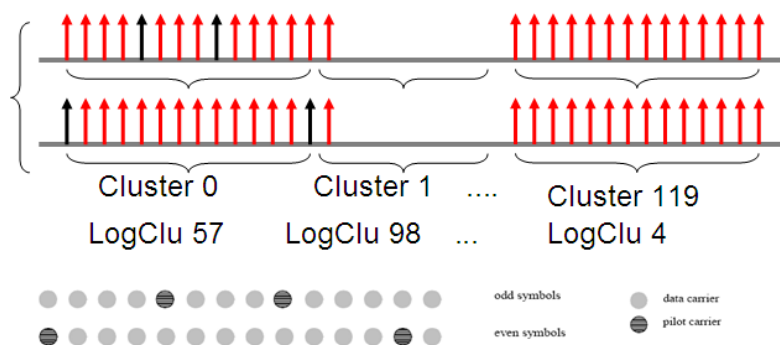
Data subc. per channel: **24**

SYMBOL #0

Data subcarriers: 1440

Pilot subcarriers: 240

Data subc. per channel: **24**



Odd symbol

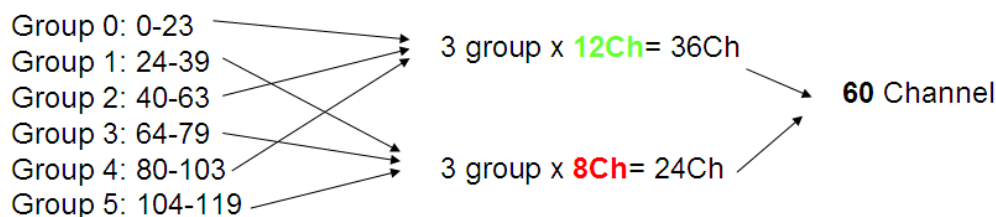
Even symbol

NOTA:

120 Cluster x

14Subcarriers =

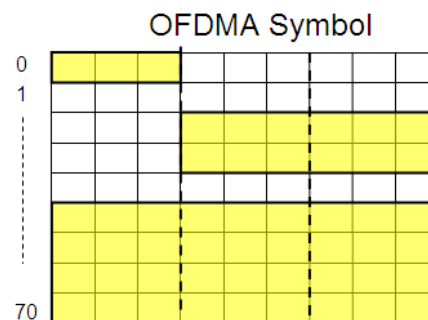
1680 SubCarriers



I gruppi (0,2,4) hanno 24 cluster da 12 SubC. Ogni 2 simboli ho: $2 \times 24 \times 12 = 48 \times 12\text{Ch}$

I gruppi (1,3,5) hanno 16 cluster da 12 SubC. Ogni 2 simboli ho: $2 \times 16 \times 12 = 48 \times 8\text{Ch}$

Parameter	Value
Number of dc subcarriers	1
N_{used}	1681
Guard subcarriers: Left, Right	184, 183
TilePermutation	6, 48, 58, 57, 50, 1, 13, 26, 46, 44, 30, 3, 27, 53, 22, 18, 61, 7, 55, 36, 45, 37, 52, 15, 40, 2, 20, 4, 34, 31, 10, 5, 41, 9, 69, 63, 21, 11, 12, 19, 68, 56, 43, 23, 25, 39, 66, 42, 16, 47, 51, 8, 62, 14, 33, 24, 32, 17, 54, 29, 67, 49, 65, 35, 38, 59, 64, 28, 60, 0
$N_{subchannels}$	70
$N_{subcarriers}$	48
N_{tiles}	420
Tiles per subchannel	6



C) $N_{GS}=3$

SYMBOL#0

Data subcarriers: 840

Pilot subcarriers: 840

Data subc. per channel: **12**

SYMBOL#1

Data subcarriers: 1680

Pilot subcarriers: 0

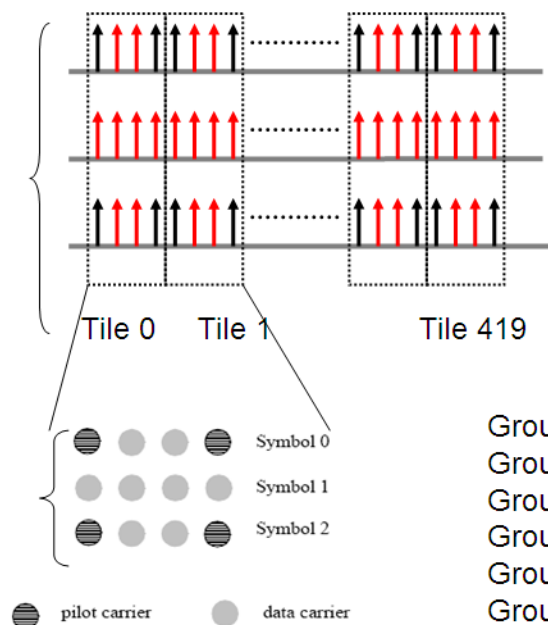
Data subc. per channel: **24**

SYMBOL#2

Data subcarriers: 840

Pilot subcarriers: 840

Data subc. per channel: **12**



Symbol 0

NOTA
420 Tile x 4SubCar
=1680

Symbol 1

Symbol 2

NOTA
1 Tile = 8SubCar

Group 0: Tile 0-69
Group 1: Tile 70-139
Group 2: Tile 140-209
Group 3: Tile 210-279
Group 4: Tile 280-349
Group 5: Tile 350-419

1 Subch=1Tile x $\forall$ gruppo= 6 Tile =6x8SubC=48 subcarrier utili

70 Subchannel

Caratterizzazione del sistema ricevitore:

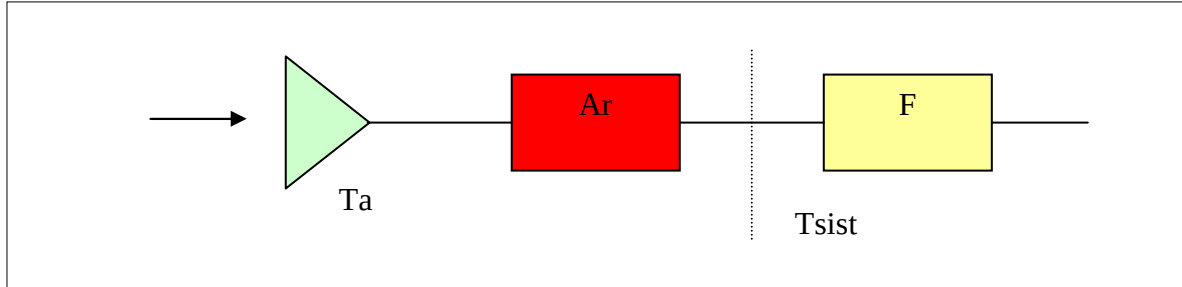


Figura 1: Schema del ricevitore

Calcolo della temperatura di sistema:

$$T_{sist} = \frac{T_a}{A_r} + T_0 \frac{(A_r - 1)}{A_r} + T_0 (F - 1)$$

Schema della catena di trasmissione/ricezione:

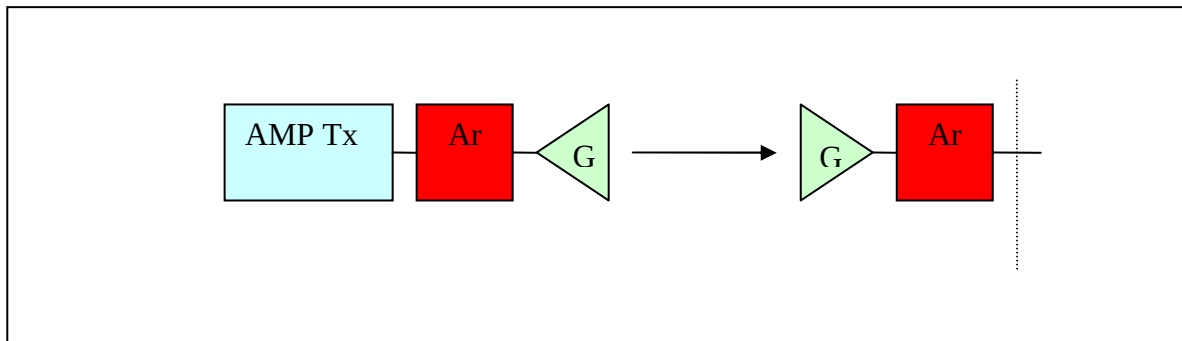


Figura 2: Schema del collegamento

Calcolo della Potenza Utile Ricevuta:

$$P_{U_{ricevuta}} = \frac{\left(\left[P_{TX} + G_{TX} + G_{RX} - Ar_{TX} - Ar_{RX} - P_{Loss} - P_{Shadowing} \right]_{dB} \right)_{lineare}}{(1 + G) \cdot N_{Subcarriers}}$$

con $G=1/4, 1/8, 1/16, 1/32$

Calcolo della Potenza di Rumore:

$$P_{Noise} = k \cdot T_{sist} \cdot B_r = k \cdot T_{sist} \cdot \left(\frac{\log_2 M}{T_s} \right) \text{ con } M=64, 16, 4$$

Definizione del Rapporto segnale rumore a radiofrequenza per singola sottoportante:

$$W = \frac{P_{U_{ricevuta}}}{P_{Noise}}$$

Considerando gli interferenti:

$$W_{eff} = W [dB] - \Delta W [dB]$$

dove

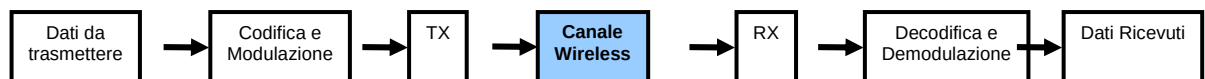
$$\Delta W [dB] = \sum P_{u\ interferenti}$$

$$P_{eb} = \frac{1}{\log_2 L} \cdot \frac{L-1}{L} \cdot \operatorname{erfc} \sqrt{\frac{3(L-1)}{L+1} \cdot \frac{\log_2 L}{(L-1)^2} \cdot W_{eff} \cdot |H_f|^2}$$

Modello di Path-Loss e Shadowing

Un modello di *canale wireless* deve prevedere:

- modello di *propagazione* per la caratterizzazione del **Path Loss**
- modello di *multipath* per la caratterizzazione del **Fading**
- modello spaziale per la correlazione dello **Shadowing**
- valutazione degli effetti dell'**Interferenza**



Per quello che riguarda la caratterizzazione del terreno in cui si propaga il segnale elettromagnetico, il modello di Erceg implementato prevede 3 categorie di terreno:

- A: i terreni fortemente collinari, con una densità di alberi elevata, caratterizzati da una forte attenuazione;
- B: i terreni con caratteristiche intermedie, colline dolci e mediamente alberate, con caratteristiche intermedie rispetto ai casi precedenti;
- C: i terreni più pianeggianti con pochi alberi, caratterizzata da una attenuazione minore.

A ciascuna tipologia di terreno è possibile associare un modello di multipath per la caratterizzazione del fading, secondo il modello SUI, secondo la tabella riportata di seguito:

Canali SUI (STANFORD UNIVERSITY INTERIM CHANNEL MODEL)	Categoria del terreno
5-6	A
3-4	B
1-2	C

Per il calcolo dell'attenuazione dovuta al Path Loss viene utilizzato il modello (Erceg), che propone la seguente formula:

$$PL = A + 10 \cdot \gamma \cdot \log_{10}(d / d_0)$$

dove i parametri valgono:

$$\begin{cases} A = 20 \cdot \log_{10}\left(\frac{4\pi d_0}{\lambda}\right) & d_0 = 100\text{m} \\ \gamma = (a - b \cdot h_b + c / h_b) + x\sigma_\gamma & x \in N[0,1] \end{cases}$$

dove le costanti vengono scelte opportunamente secondo le indicazioni della tabella:

PARAMETRI DEL MODELLO	CATEGORIA DI TERRENO		
	A (Hilly/moderate-to heavy tree density)	B (Hilly/light tree density or flat/moderate-to-heavy tree density)	C (Flat/light tree density)
a	4.6	4.0	3.6
b [m]	0.0075	0.0065	0.0050
c [m-1]	12.6	17.1	20.0
σ_γ	0.57	0.75	0.59

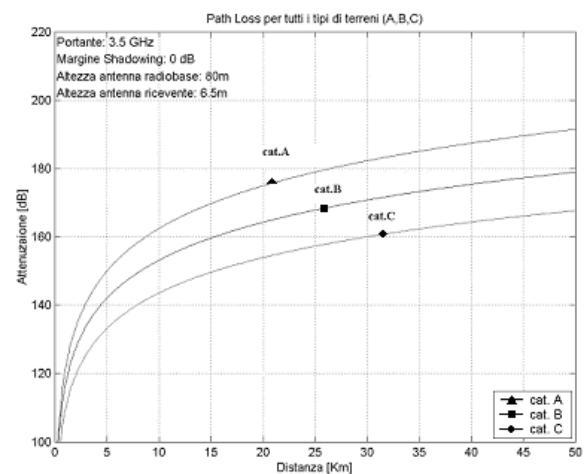
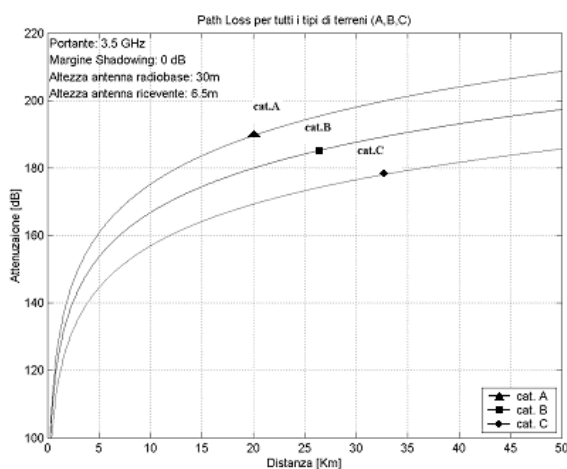
Esiste inoltre un modello modificato, anch'esso adottato nel simulatore WiMAX per considerare opportunamente le frequenze e le altezze delle antenne utilizzate:

$$PL_{\text{mod}} = PL + \Delta PL_h + \Delta PL_f$$

I parametri utilizzati, secondo il modello sono quelli riportati in tabella:

	Per terreno A, B	Per Terreno C
Fattore correttivo altezza antenne:	$\Delta PL_h = -20 \log_{10} \left(\frac{h}{2} \right)$	$\Delta PL_h = -10.8 \log_{10} \left(\frac{h}{2} \right)$
Fattore correttivo frequenza portante:	$\Delta PL_f = 6 \log_{10} \left(\frac{f[\text{MHz}]}{2000} \right)$	

Di seguito vengono riportati due esempi di PathLoss calcolato per diverse altezze delle antenne e per diverse categorie di terreni:



Per quello che riguarda lo shadowing, quest'ultimo viene modellizzato con statistica lognormale, così come proposto dal modello di Erceg, utilizzando la formula:

$$s = y\sigma = y\mu_{\sigma} + yz\sigma_{\sigma}$$

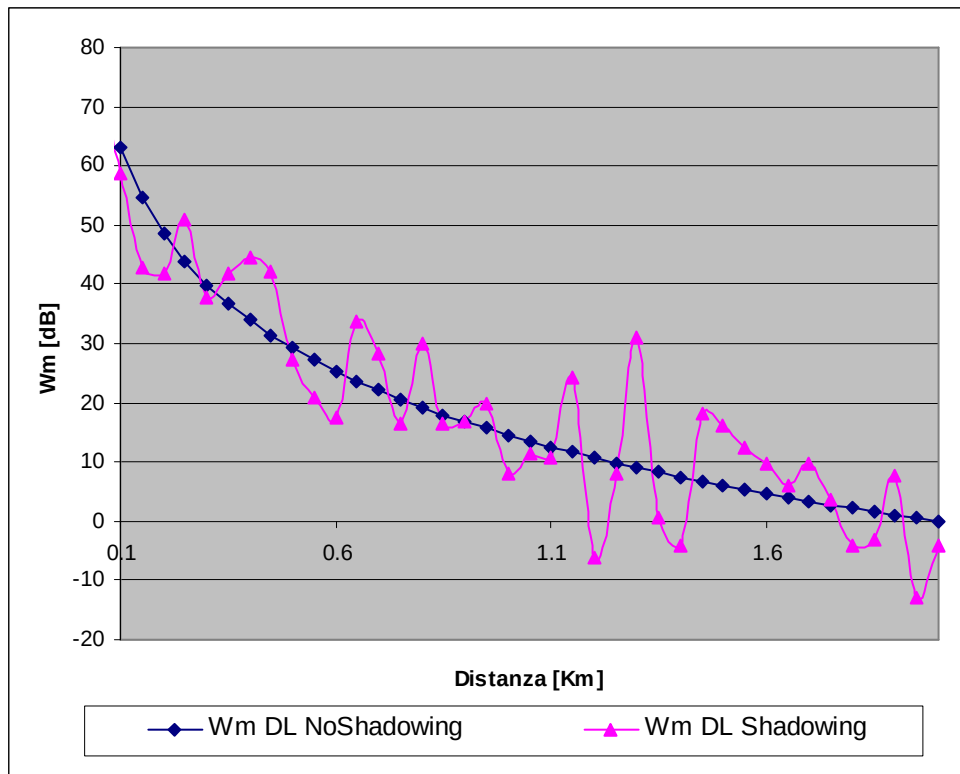
dove valgono:

- $y \in N[0,1]$
- $\sigma \in N[\mu_{\sigma}, \sigma_{\sigma}]$
- $z \in N[0,1]$

e i restanti parametri sono ricavati dalla tabella:

PARAMETRI DEL MODELL O	CATEGORIA DI TERRENO		
	A (Hilly/moderate-to heavy tree density)	B (Hilly/light tree density or flat/moderate-to-)	C (Flat/light tree density)
μ_{σ}	10.6	9.6	8.2
σ_{σ}	2.3	3.0	1.6

Di seguito è riportato un esempio degli effetti della presenza di Shadowing sul rapporto segnale rumore (Terreno di tipo C)



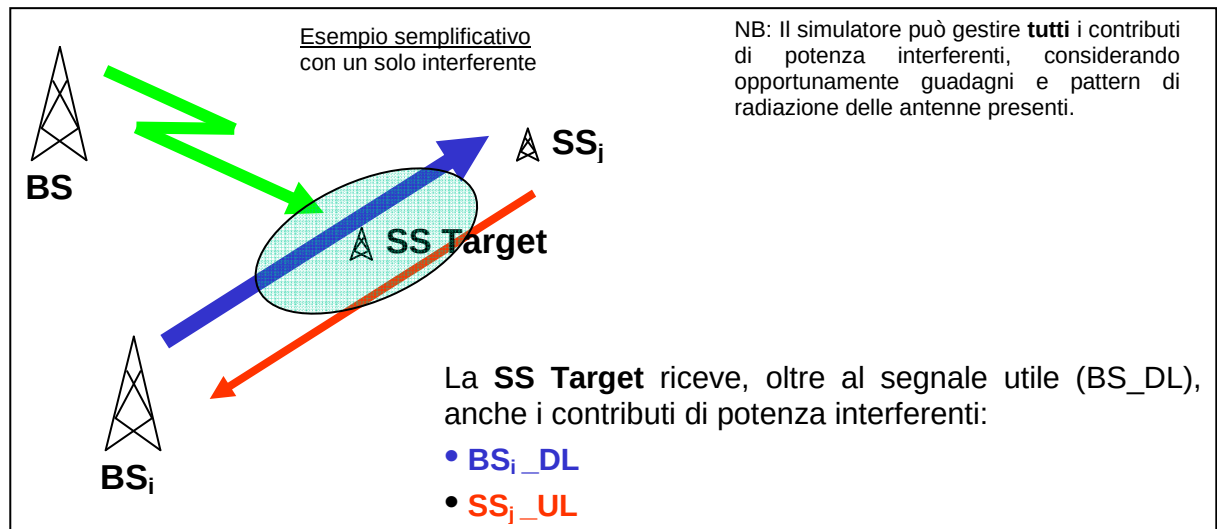
Gestione dell'Interferenza

L'interferenza tra le stazioni radiobase WiMax e i diversi terminali è tenuta opportunamente in conto all'interno del simulatore, facendo valere le seguenti ipotesi:

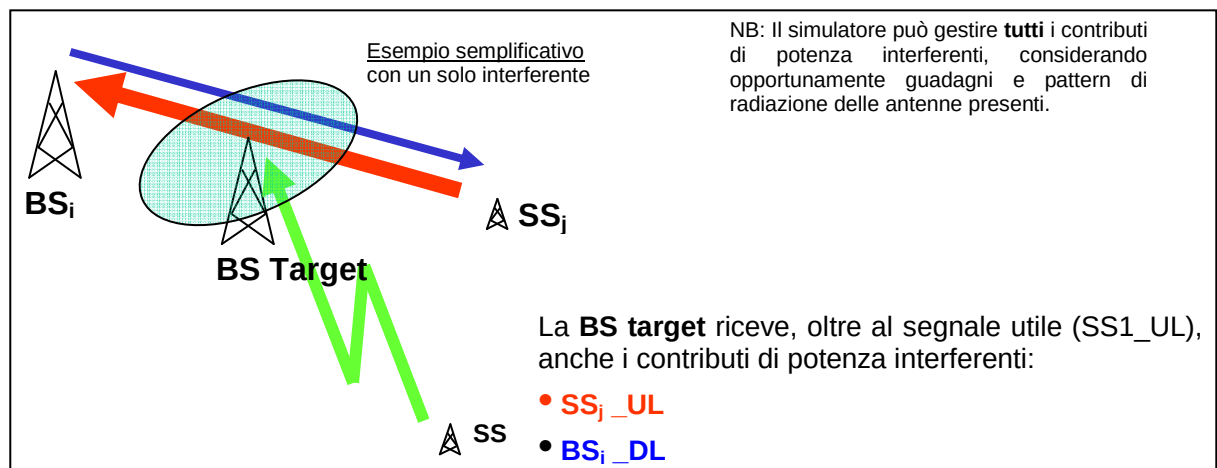
- La potenza interferente viene supposta gaussiana e perciò assimilata alla potenza di rumore nel calcolo del SNR
- Attualmente non si considera l'interferenza tra DL/UL (*TDD mode*: la rete è sincrona; *FDD mode*: DL e UL lavorano a bande diverse)

Nelle figure sottostanti sono riportati i due casi di Downlinke e Uplink.

CASO DOWNLINK



CASO UPLINK



Codifica di canale

Lo standard IEEE802.16e, grazie all'utilizzo della modulazione OFDMA riesce a garantire un collegamento anche ad utenti che sono in condizioni di NLOS rispetto alla stazione radiobase e si muovono rispetto ad essa. Lo standard IEEE802.16e prevede l'utilizzo della modulazione adattativa: si utilizzano cioè diverse modulazioni (ciascuna caratterizzata da un diverso bit-rate e tolleranza al rumore) per riuscire a raggiungere tutti gli utenti e fornire un servizio efficiente. Lo standard implementa, inoltre, alcune tecniche di codifica ed interleaving per migliorare le prestazioni e rendere la trasmissione più affidabile e meno sensibile al rumore. La codifica di canale è dunque composta da tre fasi, nell'ordine:

1. Randomizer
2. Forward Error Correction (FEC)
3. Interleaving

La fase di Randomizer aiuta a rendere incorrelate le sequenze di bit della sorgente, mentre la fase di FEC consiste in un semplice codificatore convoluzionale con rate variabile. Eventuali codificatori (Reed-Solomon, Turbocodici) possono essere concatenati, in maniera opzionale. Queste tecniche aiutano a recuperare frame contenenti errori dovuti al fading selettivo in frequenza o ad errori in burst, migliorando notevolmente le prestazioni. I dati in uscita dalla fase di interleaving vengono successivamente modulati. In fase di decodifica devono essere effettuate le operazioni complementari, ovviamente applicate in ordine inverso.



Tabella 1: Codifica di canale prevista per IEEE802.16

Le diverse combinazioni di codifica e modulazione sono riportate nella Tabella 2. Tabella 2: Modulazioni e codifica :

Modulazione	Bit/sottopor	Bit Totali x slot [Byte]	Byte Info x slot
QPSK (1/2)	2	96 [12]	6
QPSK (3/4)	2	96 [12]	9
16QAM (1/2)	4	192 [24]	12
16QAM (3/4)	4	192 [24]	18
64QAM (1/2)	6	288 [36]	18
64QAM (2/3)	6	288 [36]	24
64QAM (3/4)	6	288 [36]	27

Tabella 2: Modulazioni e codifica

Nello standard viene indicato un algoritmo che stabilisce la lunghezza della parola di codice in funzione di alcuni parametri:

- j : parameter dependent on the modulation and FEC rate
- n : number of allocated subchannels
- k : floor (n/j)
- m : n modulo j

Number of subchannels	Subchannels concatenated
$n \leq j$	1 block of n subchannels
$n > j$	$(k-1)$ blocks of j subchannels 1 block of $\text{ceil}((m+j)/2)$ subchannels 1 block of $\text{floor}((m+j)/2)$ subchannels

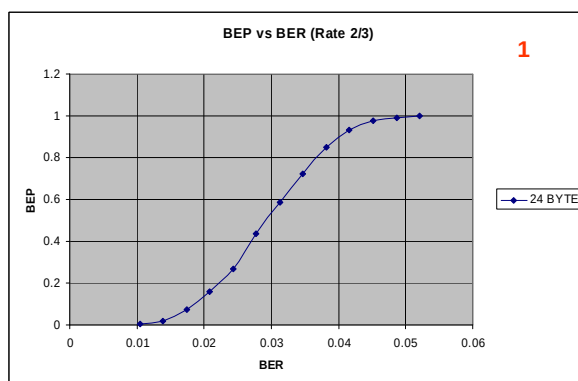
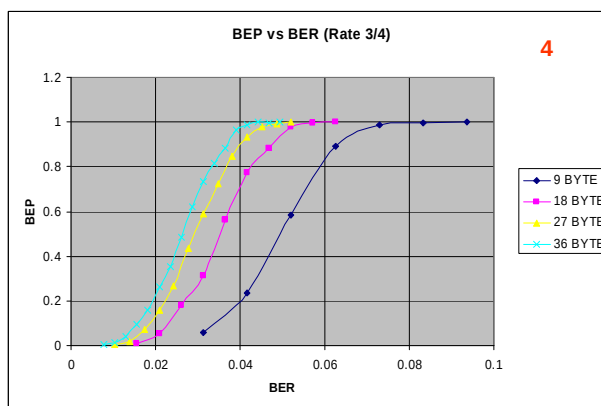
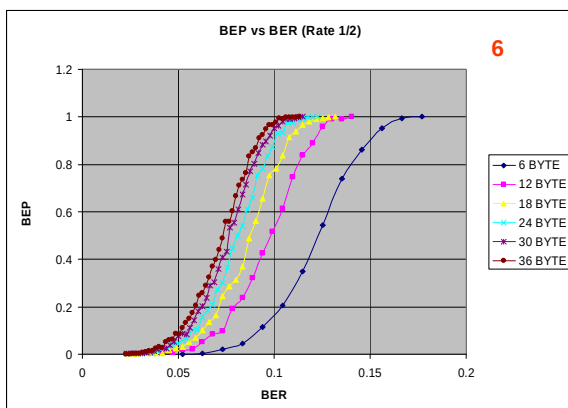
Modulation and rate	j
QPSK 1/2	$j = 6$
QPSK 3/4	$j = 4$
16-QAM 1/2	$j = 3$
16-QAM 3/4	$j = 2$
64-QAM 1/2	$j = 2$
64-QAM 2/3	$j = 1$
64-QAM 3/4	$j = 1$

Le lunghezze delle parole di codice ammissibili sono raggruppate nella tabella sottostante, divise per modulazione e ritmo di codifica. Per la trasmissione dei dati, esistono 7 combinazioni Codifica / Modulazione (freccie azzurre), e 11 diverse combinazioni Codifica / Lunghezza del Payload (freccie rosse), così come riportato in tabella. Per ulteriori informazioni si faccia riferimento allo standard IEEE802.16e-2005.

	QPSK		16 QAM		64 QAM		
Encoding rate	R=1/2	R=3/4	R=1/2	R=3/4	R=1/2	R=2/3	R=3/4
Data payload (bytes)	6 ←						
		9 ←					
	12 ←		12				
	18 ←	18 ←	18	18			
	24 ←		24			24 ←	
		27 ←					27
	30 ←						
	36 ←	36 ←	36	36	36		
	↑	↑	↑	↑	↑	↑	↑

Per ciascuna delle 11 combinazioni sono state ottenute delle curve che legano BER sul canale alla PER a valle della decodifica. Le 11 diverse combinazioni di codice / lunghezza sono così riassumibili:

- 6 curve R=1/2 [Dim: 6, 12, 18, 24, 30, 36 byte]
- 4 curve R=3/4 [Dim: 9, 18, 27, 36 byte]
- 1 curve R=2/3 [Dim: 24 byte]



Le curve sono state ottenute con un tool esterno al software e sono fornite in ingresso al simulatore mediante il file di input FileNBE_PEP.txt.

Procedure di Aggancio

A seconda della posizione della SS (MS) scelgo:

- quale BS fornirà il servizio (**Criterio:** stazione + vicina, considerando opportunamente le direzioni delle antenne)
- la modulazione da utilizzare (**Criterio:** retroazione su W fino a superamento soglia)

fisso SNR a cui effettuare la simulazione (Interferenza calcolata a Runtime)

Modulazione	Soglia W [dB]
QPSK (1/2)	qualunque
QPSK (3/4)	2
16QAM (1/2)	4
16QAM (3/4)	6
64QAM (1/2)	8
64QAM (2/3)	10
64QAM (3/4)	12

Tab 1. Tabella delle soglie di W utilizzate

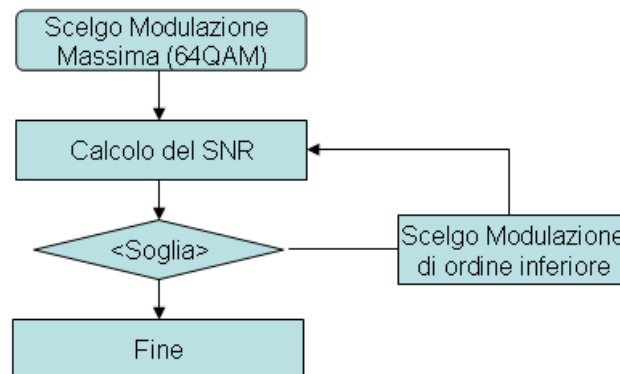


Fig 5. Algoritmo di scelta della modulazione da utilizzare

Simulazione con modello semi analitico

La simulazione della trasmissione dei bit trasmessi su ogni singola sottoportante è effettuata con un metodo semi-analitico. Grazie ad un generatore di fading per il canale si ottengono delle curve della funzione di trasferimento correlate in tempo e frequenza (Modello di Jakes). Considerando il corretto ordine dei simboli OFDMA, per ciascun simbolo si utilizza la “fotografia” nel dominio delle frequenze all’istante considerato per il simbolo OFDMA. Per ciascuna sottoportante del simbolo OFDMA considerato si utilizza il valore della funzione di trasferimento corrispondente alla coppia tempo, frequenza.

Bisogna ricordare che la matrice tempo-frequenza che viene riempita durante la fase dei mappatura utilizza nel dominio delle frequenze i canali logici. Per poter recuperare le sottoportanti associate a ciascun canale logico è necessario recuperare i dati dal file di configurazione delle permutazioni IDperm0.txt. Attenzione: è da ricordare che alcune permutazioni allocano le 48 sottoportanti dati in 1, 2 o 3 simboli OFDMA contigui. E’ cioè necessario cioè memorizzare simbolo per simbolo il numero di bit errati, prima di aver ricevuto tutti i 48 e prima di prendere una decisione sulla correttezza della codeword.

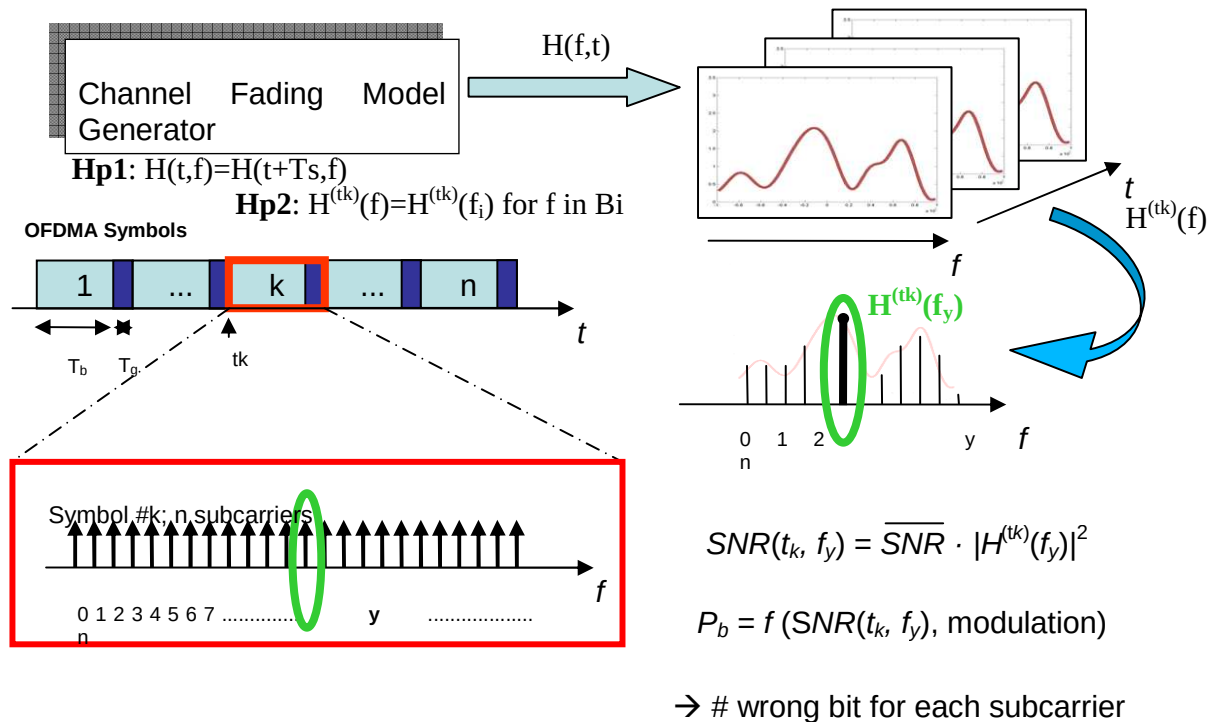
Lo schema è rappresentato in figura.

Per calcolare la probabilità di errore per ogni singolo bit che viene trasmesso su ogni singola sottoportante all’interno dello stesso tempo di simbolo OFDMA si utilizza la formula riporta:

$$P_{eb} = \frac{1}{\log_2 L} \cdot \frac{L-1}{L} \cdot \operatorname{erfc} \sqrt{\frac{3(L-1)}{L+1} \cdot \frac{\log_2 L}{(L-1)^2} \cdot W_{eff} \cdot |H_f|^2}$$

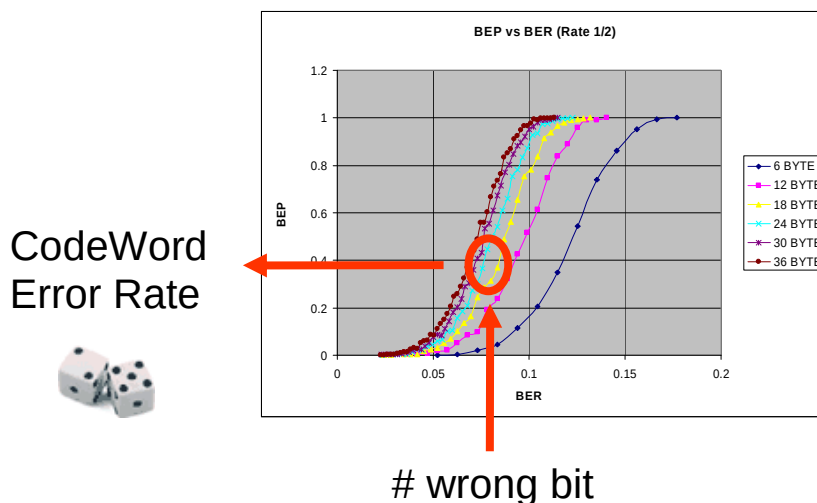
Per ulteriori dettagli far riferimento al paragrafo 5.5.

Il valore W_{eff} è già ugualmente descritto in 5.5.



A questo punto è possibile, a partire dal numero di bit errati in ciascuna codeword ricavare la probabilità di perdere l'intera codeword, leggendo opportune lookup table che devono essere fornite come input al simulatore.

Le curve per le 11 combinazioni di codifica/rate/lunghezza vengono appunto utilizzate come lookup table: a partire dalla BER sulla codeword posso ricavare la probabilità che la decodifica non vada a buon fine.



Livello Data Link

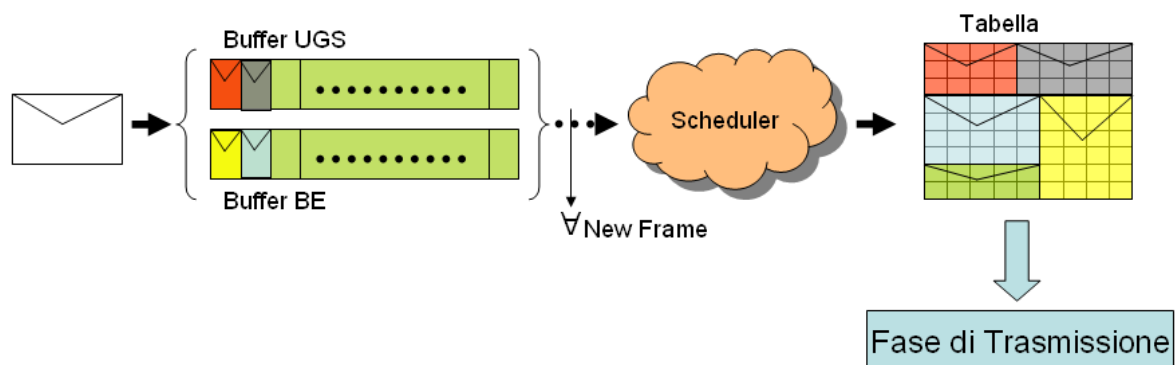
Lo scopo principale del Livello Data Link sono:

- Amministrare la capacità del canale radio condiviso distribuendola fra gli utenti e le loro connessioni
- Specificare il formato delle PDU e la loro modalità di trasmissione
- Implementare meccanismi di ARQ e recupero delle informazioni perdute
- Differenziazione e gestione di diverse classi di traffico

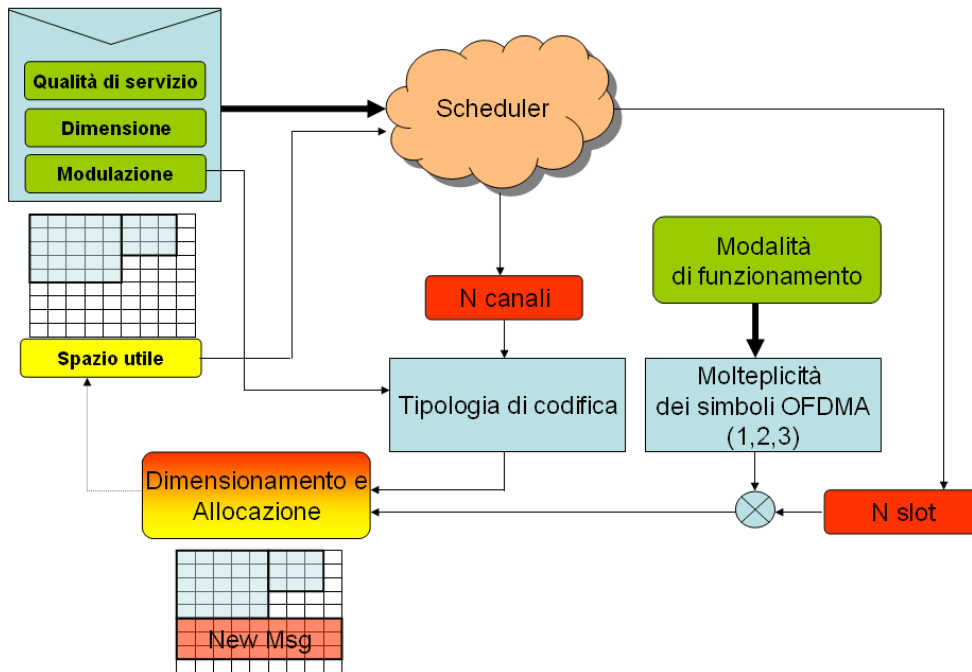
La QoS permette di differenziare il servizio tra le connessioni in base a:

- contratto
- esigenze del tipo di servizio (dati, voce, video,...)

I pacchetti inviati dal Simulatore di livello Trasporto vengono accodati nei Buffer, rispettando le specifiche di qualità, dove restano in attesa fino all'inizio di un nuovo frame. Successivamente lo Scheduler processerà in ordine e secondo le specifiche di servizio i pacchetti in coda.



Lo schema a blocchi che illustra il funzionamento dello scheduler è illustrato nel diagramma sottostante:

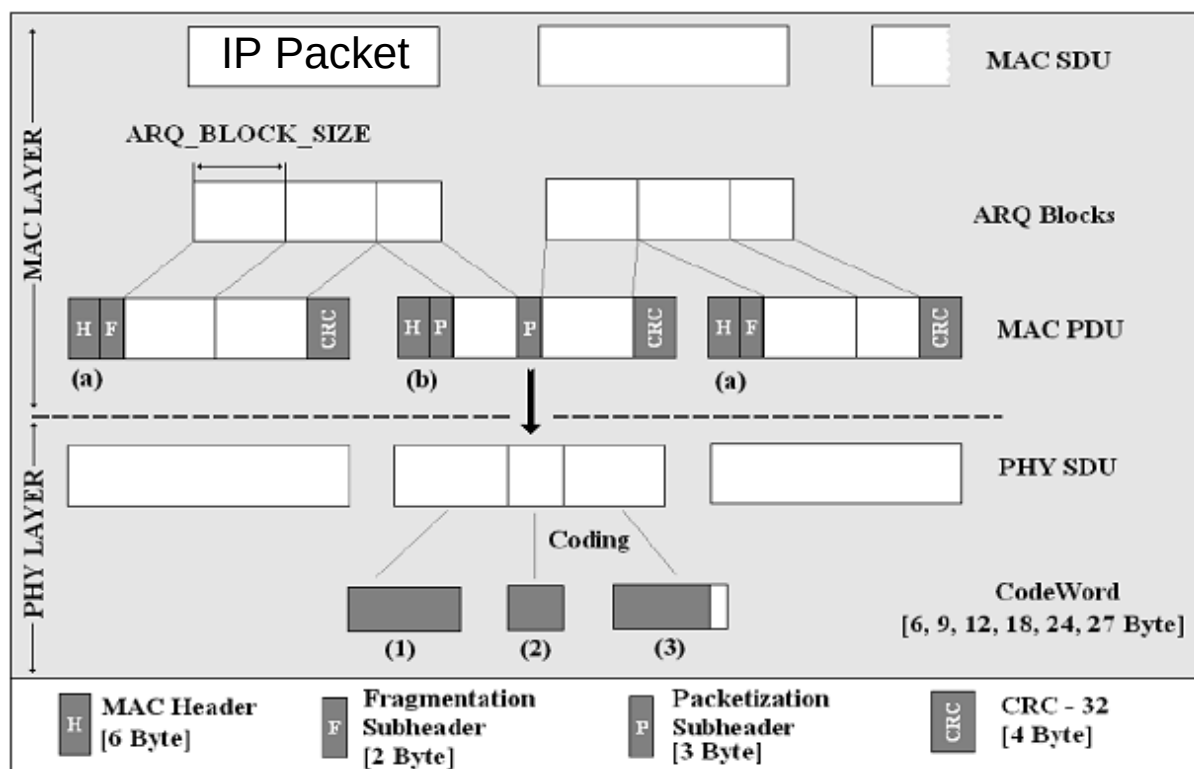


FASE 1: conversione dei pacchetti IP(SDU) in ARQBLOCK

La dimensione del singolo AB è fissata dal parametro opportuno letto dal file di configurazione.

Ogni singolo AB viene marcato mediante:

- ➔ gli ID della BS, della SS e la direzione della trasmissione
- ➔ il numero di sequenza del pacchetto IP passato dall'ULS, di cui l'AB fa parte
- ➔ il numero totale di AB che compongono il pacchetto IP e il numero progressivo
- ➔ il contatore di tutti gli AB trasmessi
- ➔ viene inoltre memorizzato un puntatore al AB_BUFFER in cui verrà archiviato l'AB

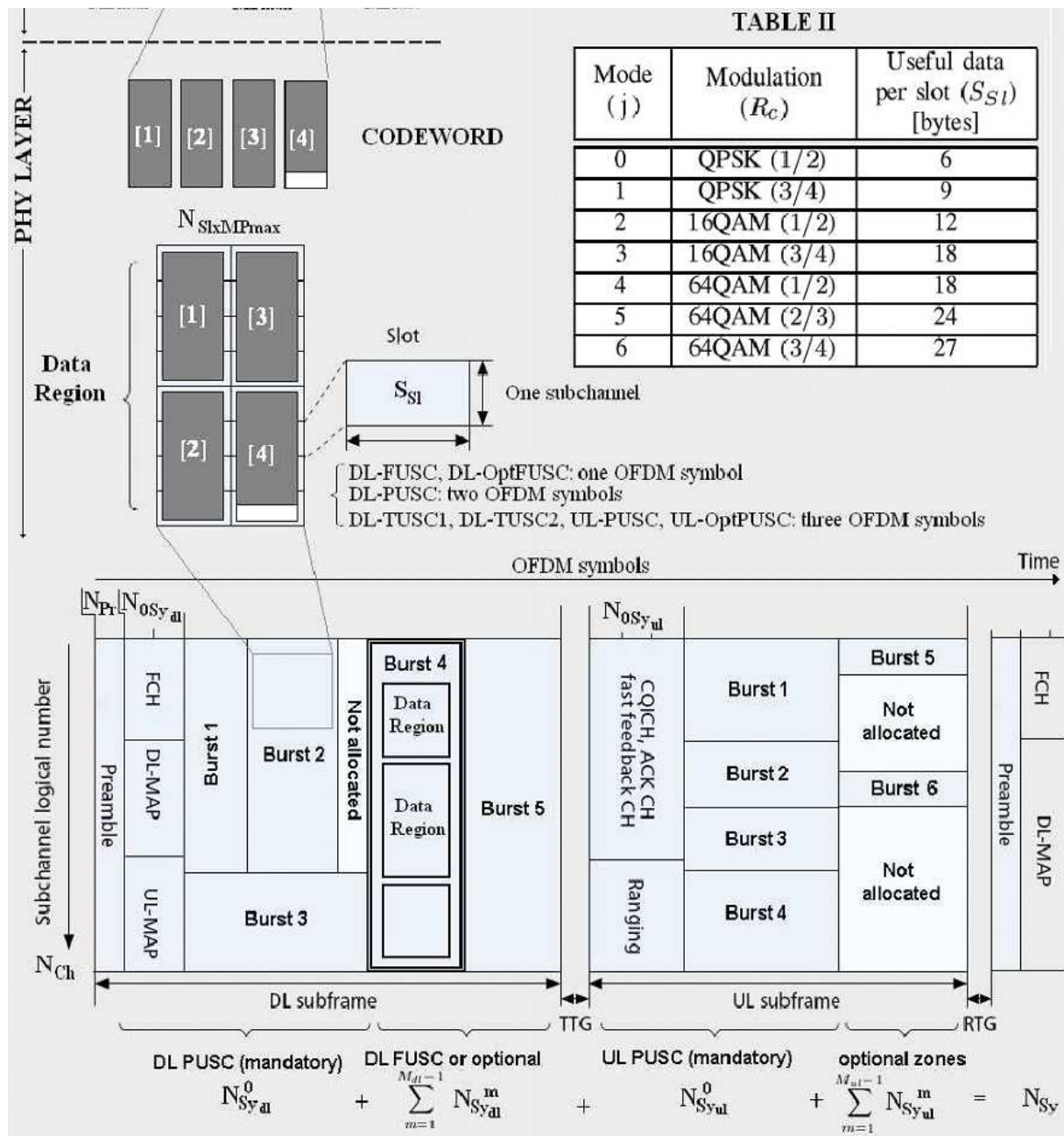


FASE 2: formazione di una PDU a partire dai singoli AB

Al momento opportuno dai Buffer in cui sono immagazzinati gli AB vengono create le PDU, secondo le regole e la QoS prevista.

La creazione delle PDU si sviluppa in tre passi:

AB_ProgettaPDU → CreaPDU_DATA → CreaPDU

FASE 3: trasmissione PHY delle PDU mediante codeword

QPSK (1/2) occupo 60 slots

QPSK (3/4) occupo 40 slots

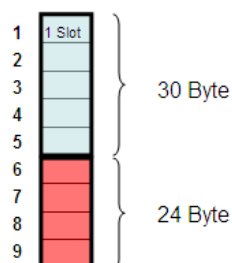
QPSK (3/4) occupo 30 slots

Modulazione	Slot Size [Byte]
QPSK (1/2)	6
QPSK (3/4)	9
16QAM (1/2)	12
16QAM (3/4)	18
64QAM (1/2)	18
64QAM (2/3)	24
64QAM (3/4)	27

Ipotesi: Numero slot da concatenare [n]= 9

Modulazione	j	$k=\text{floor}(n/j)$	$m=n\%j$	
QPSK (1/2)	6	1	3	1 blocco da 5 sottocanali e 1 da 4
QPSK (3/4)	4	2	1	1 blocco da 4 sottocanali, 1 da 3 e 1 da 2
16QAM (1/2)	3	3	0	2 blocchi da 3 sottocanali, 1 da 2 e 1 da 1
16QAM (3/4)	2	4	1	3 blocchi da 2 sottocanali, 1 da 2 e 1 da 1
64QAM (1/2)	2	4	1	3 blocchi da 2 sottocanali, 1 da 2 e 1 da 1
64QAM (2/3)	1	9	0	9 blocchi da 1 sottocanale
64QAM (3/4)	1	9	0	9 blocchi da 1 sottocanale

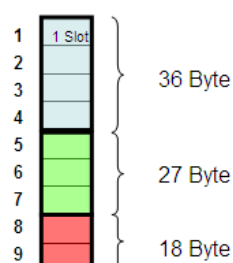
1 slot = 6 byte



QPSK (1/2)

[54 Byte]

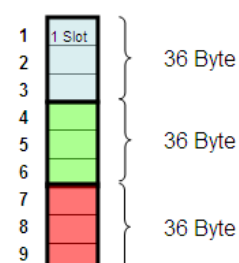
1 slot = 9 byte



QPSK (3/4)

[81 Byte]

1 slot = 12 byte

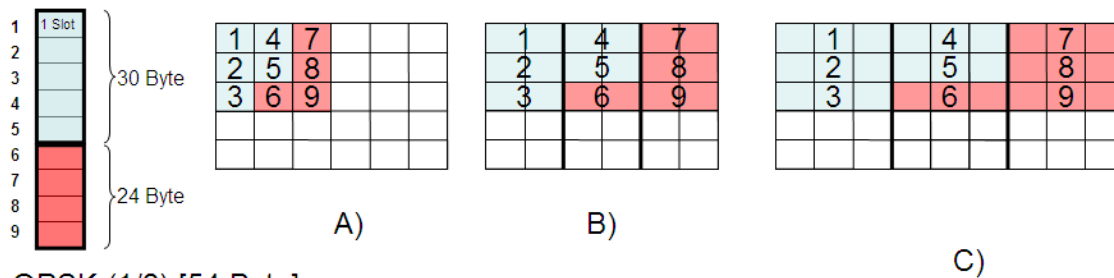


16QAM(1/2)

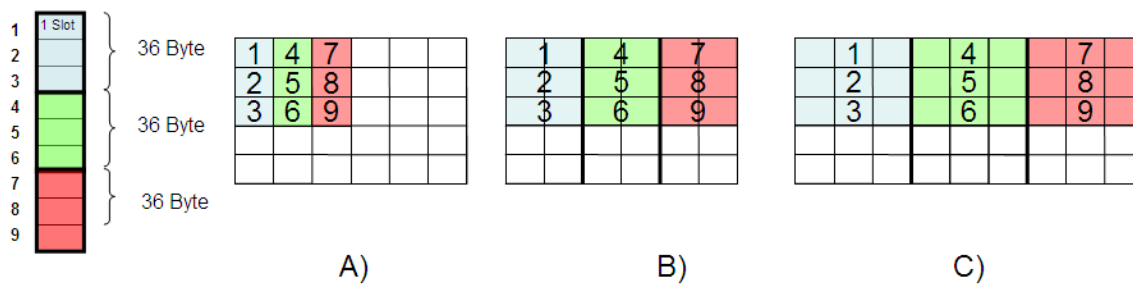
[108 Byte]

Modalità: DL-FUSC (A), DL-PUSC (B), UL-PUSC (C),

HP: Numero slot allocati [n]= 9, 162 bytes da trasmettere (54Bx3)



HP: Numero slot allocati [n]= 9, 324 bytes da trasmettere (108Bx3)



FASE 4: ricezione delle codeword e ricostruzione delle PDU

FASE 5: controllo MAC

Nel simulatore WiMax è implementato il meccanismo di Automatic Repeat Request, così come previsto dallo standard.

L'ARQ è una tecnologia per il controllo degli errori:

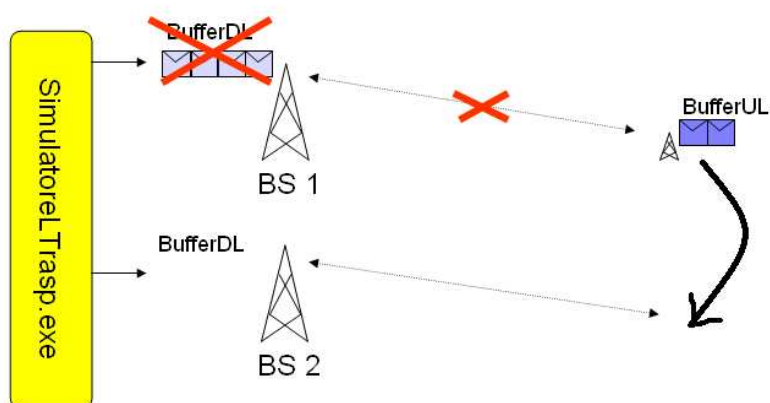
- complementare al FEC
- opera a livello MAC
- ritrasmette le PDU MAC ricevute errate
- permette di dimensionare il funzionamento normale del FEC con una BER più elevata di quella tollerata dal servizio
- è molto onerosa perché richiede i feedback (ACK)
- è più efficiente delle ritrasmissioni TCP perché è più rapida, non è end-to-end, ritrasmette solo i blocchi errati e non interi pacchetti IP

Radio Resource Management

Per quello che riguarda la procedura di Handover, valgono le seguenti:

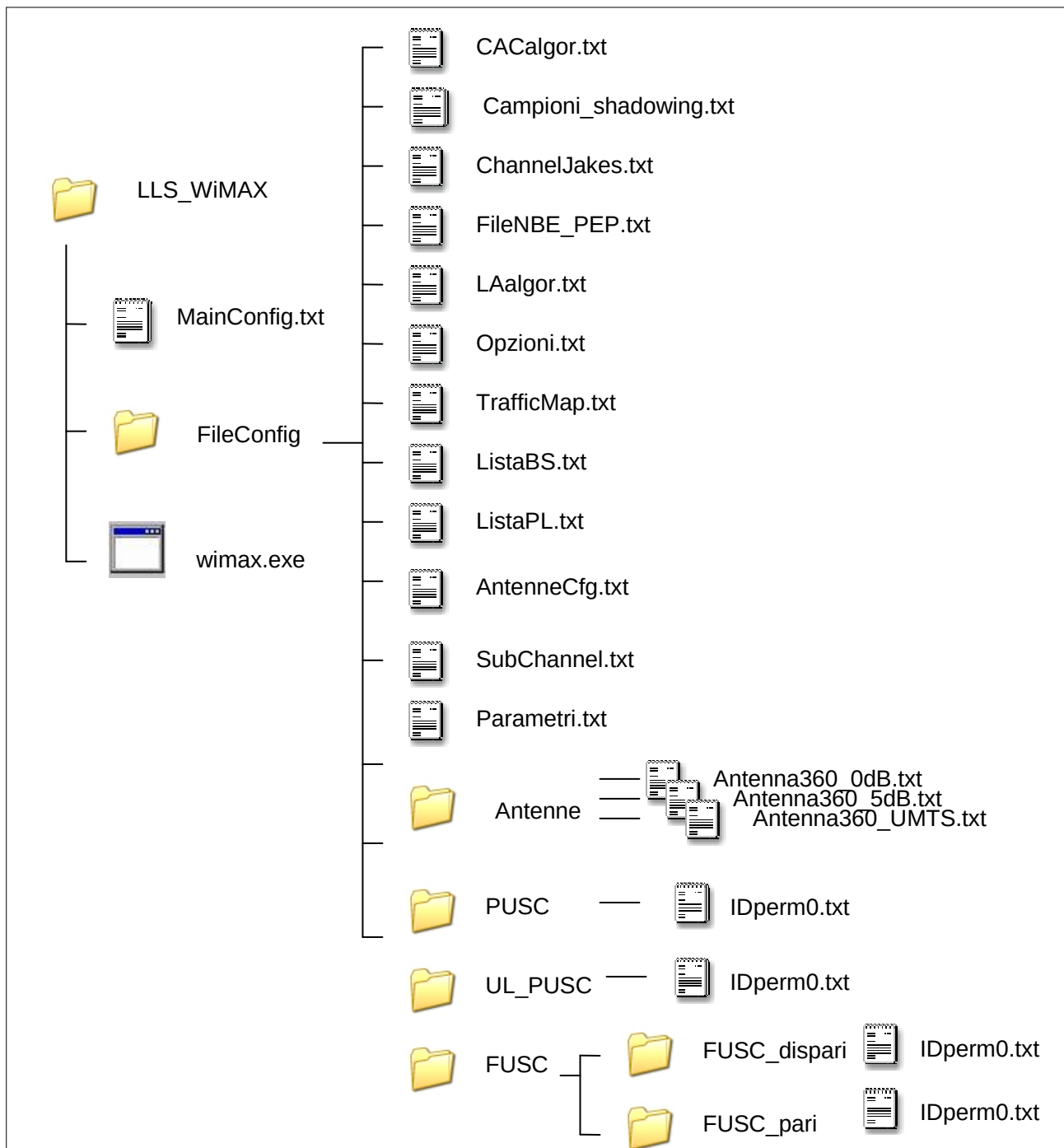
- quando un utente viene spostato, viene ricalcolata qual è la BS a cui collegarsi
- se diversa dalla precedente, effettuo Handover
- chiudo vecchia connessione, attivo quella nuova
- i msg in UL vengono mantenuti in memoria dalla SS

- i msg in DL vengono persi e il SLT deve provvedere a ritrasmetterli alla nuova BS



File di Input e Configurazione

L'interfacciamento con gli strati superiori utilizza le primitive documentate nel manuale di SHINE. La configurazione del simulatore WiMAX è realizzata attraverso semplici file di testo. In particolare si distingue un file di configurazione principale (MainConfig.txt), e altri file ausiliari che devono essere necessariamente essere forniti:



In generale:

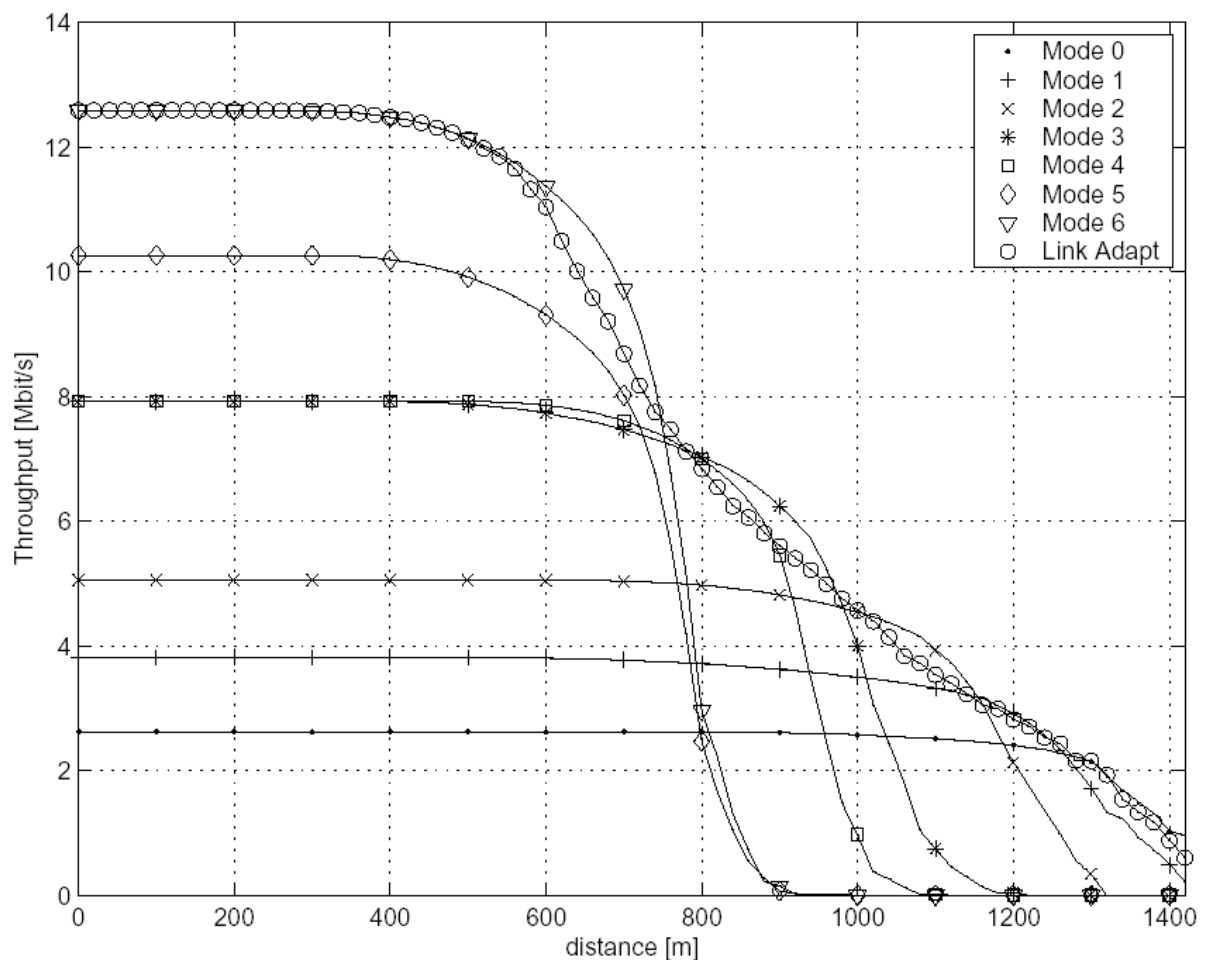
1. MainConfig.txt: file principale di configurazione
2. AntennaCfg.txt: file di configurazione dei diversi tipi antenne da utilizzare
3. ID_perm0.txt: file sdi allocazione dei sottocanali
4. CACalgor.txt: file di configurazione degli algoritmi di Call Admission Control
5. Campioni_shadowing.txt: file che contiene i campioni dello shadowing
6. ChannelJakes.txt: file di configurazione del simulatore PHY nel caso si vogliano considerare realizzazioni del canale valutate runtime.
7. FileNBE_BEP.txt: file con le lookup table per il livello fisico
8. LAalgor.txt: file di configurazione degli algoritmi di Link Adaptation

9. Opzioni.txt: infine gran parte delle funzionalità del simulatore possono essere attivate attraverso un file di definizione delle opzioni .
10. Parametri.txt: file di configurazione dei parametri del sistema
11. ListaBS.txt: file delle Basi attive
12. ListaPL.txt: file dei valori dei modelli di PatLoss per ogni Base attiva
13. TrafficMap.txt: file di configurazione degli algoritmi di scheduling
14. SubChannel.txt: file di configurazione dei sottocanali

Si consiglia di inserire i file di configurazione entro una cartella apposita (p.es. Configurazione).

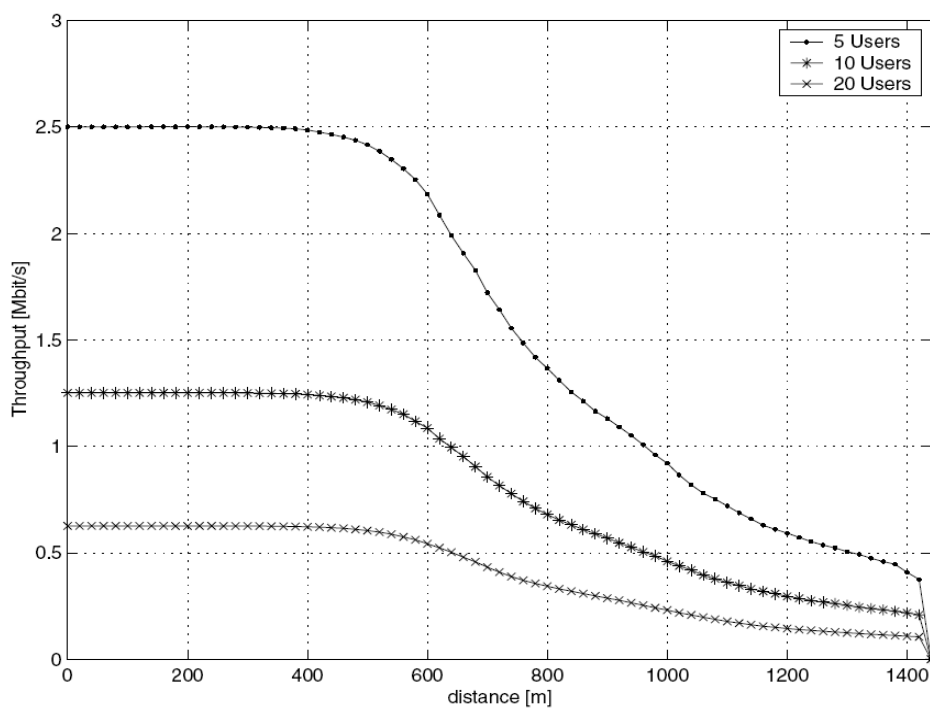
Esempi di simulazione

THR di 1 utente (FTP saturazione, BE) per diverse combinazioni di modulazione e codifica al variare della distanza. Valutazione delle prestazioni di un semplice algoritmo di LA

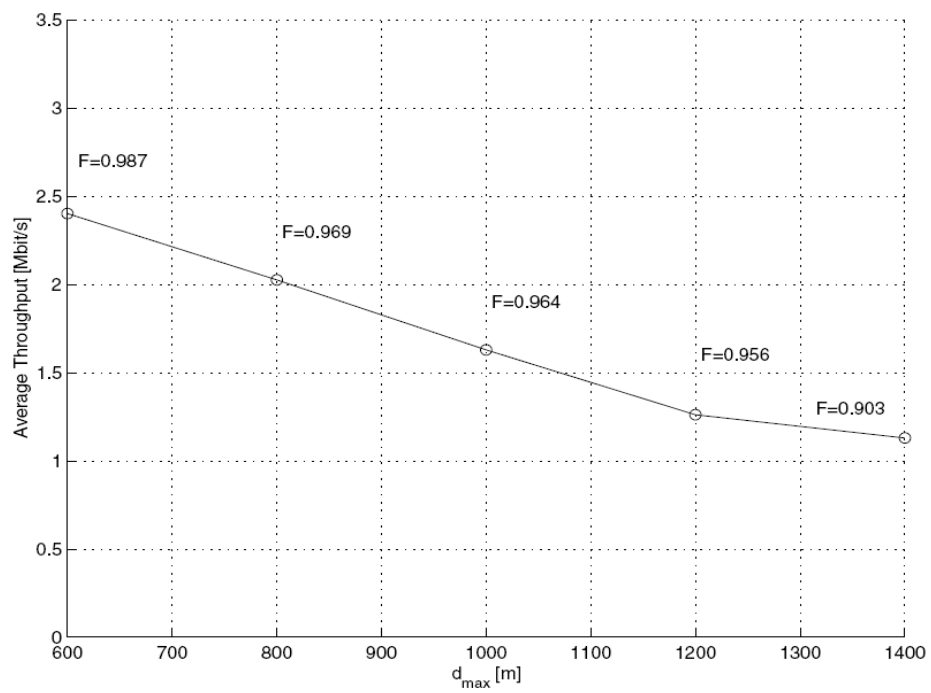


Modo
[0] QPSK (1/2)
[1] QPSK (3/4)
[2] 16QAM (1/2)
[3] 16QAM (3/4)
[4] 64QAM (1/2)
[5] 64QAM (2/3)
[6] 64QAM (3/4)

Average THR di 5,10,20 utenti (FTP saturazione, BE) tutti alla stessa distanza

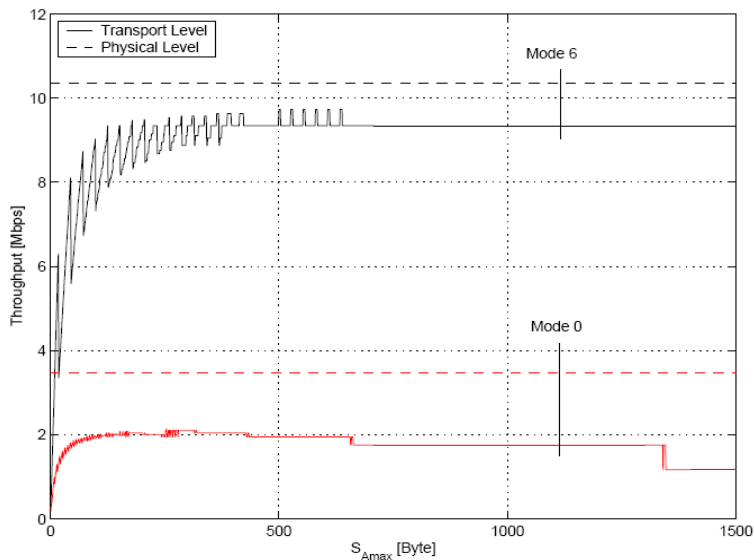


Average THR di 5 utenti (FTP saturazione, BE) ugualmente distanziati tra [0 e Dmax].
Valutazione della Fairness (F).



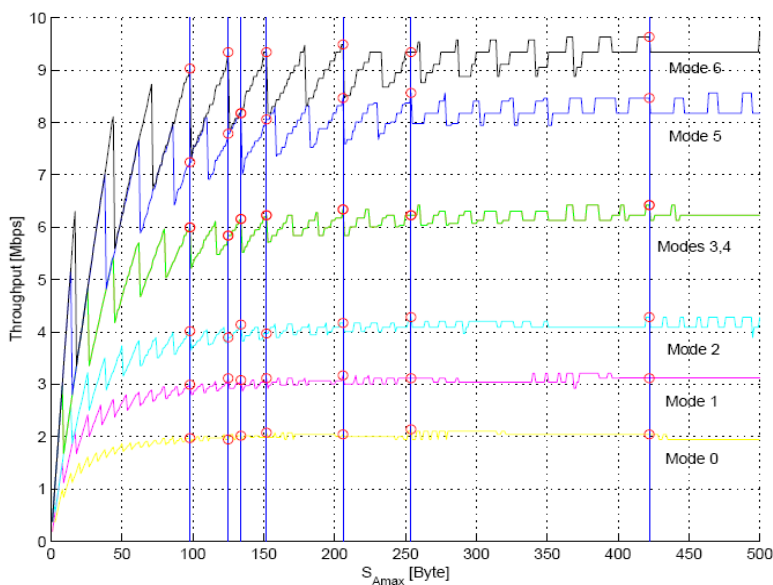
THR di 1 utente (FTP saturazione, BE) per diverse dimensioni dell'ArqBlock_Size per diversi modi di funzionamento

Confronto tra Throughput di Livello Fisico (Teorico) e quello a Livello Trasporto

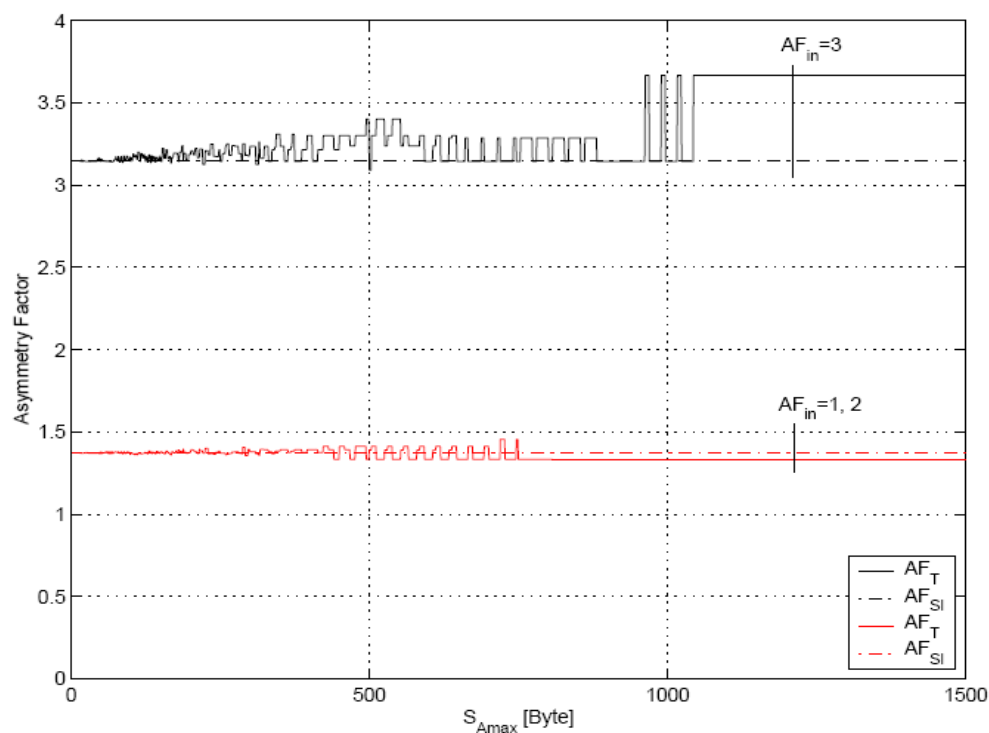


THR di 1 utente (FTP saturazione, BE) per diverse dimensioni dell'ArqBlock_Size per diversi modi di funzionamento

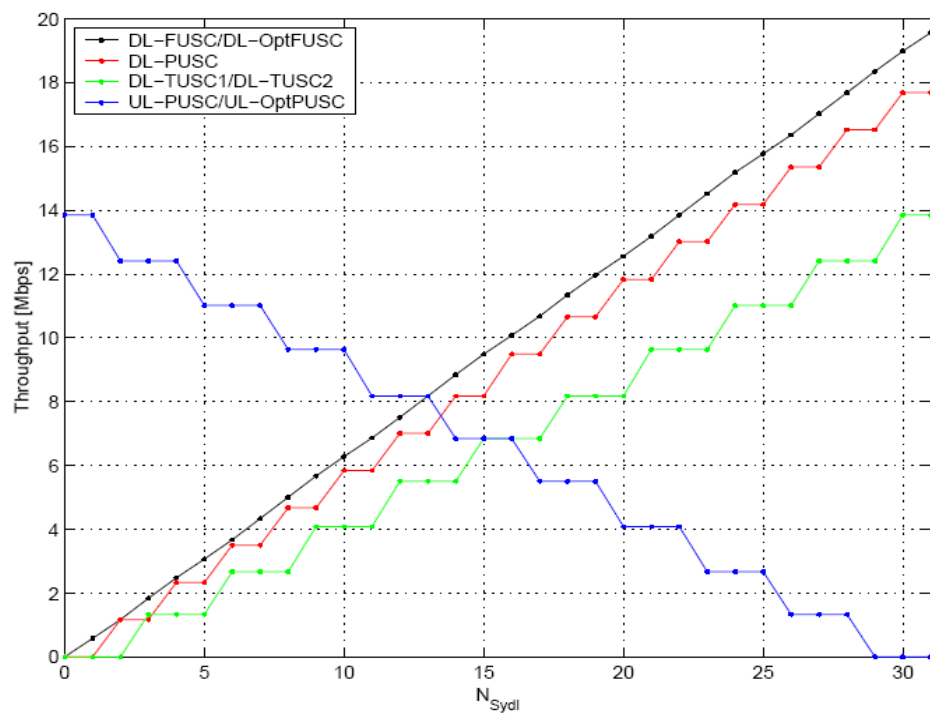
Ricerca dei valori di ArqBlock_Size che massimizzano il Throughput per tutti i modi di funzionamento



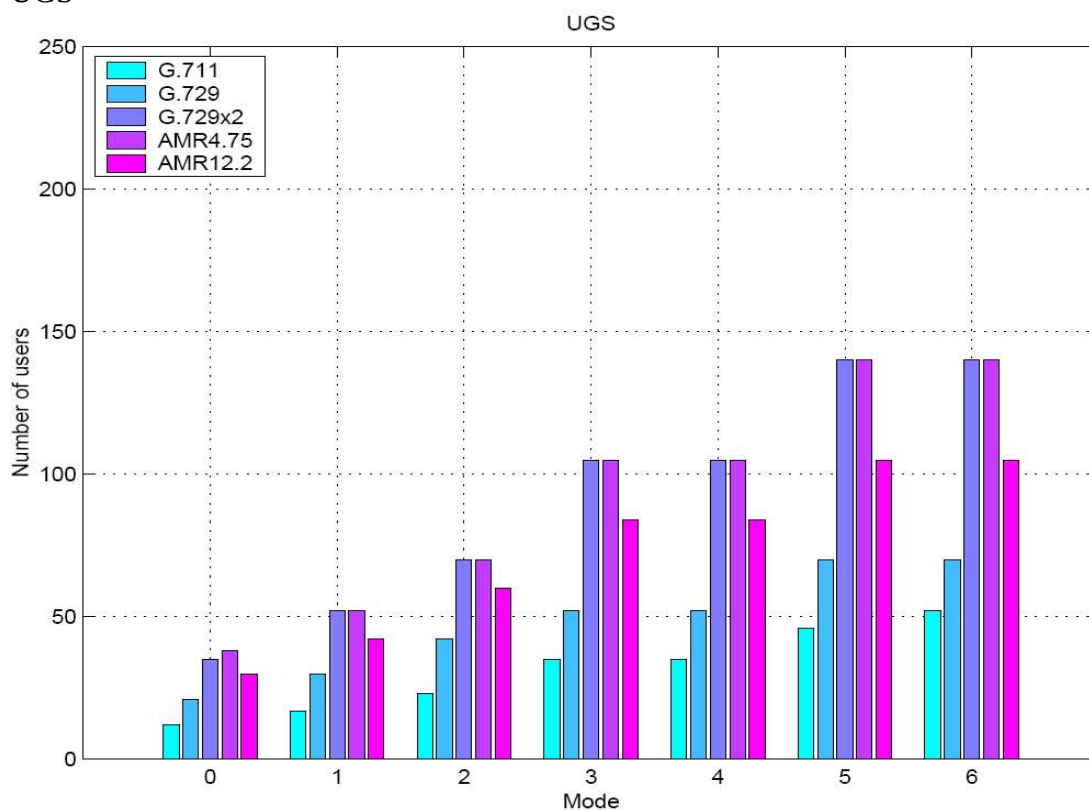
Valutazione del Fattore di Asimmetria (AF) tra DL e UL, per sistemi TDD Information Slot vs Throughput



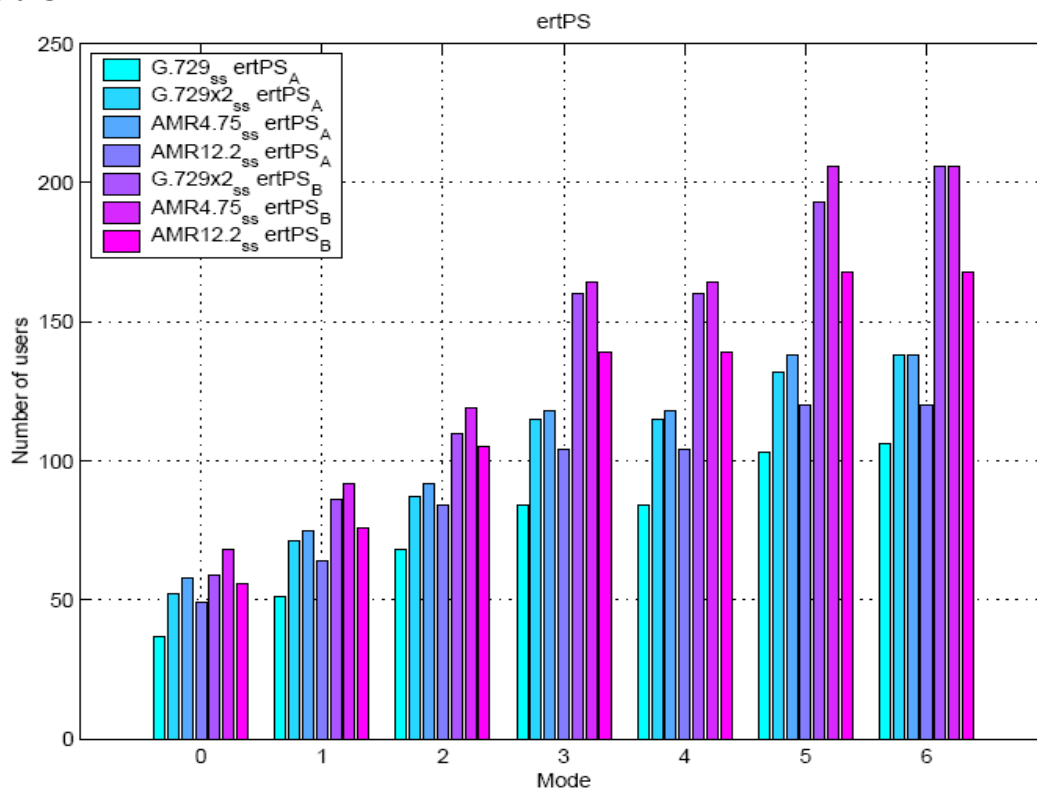
Valutazione del Throughput a Livello Trasporto al variare del numero di simboli riservati alla fase di DL (in TDD) Modo 6, ArqBlock_Size=204B



Valutazione del numero di utenti servibili per i diversi modi, per diversi VoIP Codec, in UGS



Valutazione del numero di utenti servibili per i diversi modi, per diversi VoIP Codec, in ertPS



ALLEGATI
