

TELEMACO - Sistema Telematico per l'invio delle Comunicazioni DbcXml DbcPlus - GaiaTpl

Il presente documento illustra le nuove funzionalità di invio telematico delle Comunicazioni degli Orari Pianificati e delle relative rappresentazioni cartografiche [TELEMACO].

Dato il sostanziale parallelismo di implementazione, quanto segue vale in linea di massima sia per l'applicazione **DbcPlus** che per **GaiaTpl**.

Il documento è sostanzialmente destinato agli utilizzatori finali degli applicativi **DbcPlus** e **GaiaTpl** [trasportisti, addetti aziendali], quindi la parte specificamente informatica è ridotta al minimo indispensabile, mentre vengono analizzati in dettaglio gli aspetti relativi al reale utilizzo pratico del sistema.

Concetti:

Per consentire una migliore efficienza complessiva del sistema informativo dell'Osservatorio Regionale Trasporti, viene introdotto un meccanismo completamente nuovo per la generazione, l'invio e la successiva acquisizione ed elaborazione delle Comunicazioni Orari che le Aziende TPL sono tenute a comunicare periodicamente all'Osservatorio.

Tuttavia il nuovo sistema (per quanto decisamente innovativo), non si configura affatto come una brusca discontinuità rispetto a quanto già utilizzato negli anni precedenti; al contrario, rappresenta la naturale evoluzione e maturazione del sistema nel segno di una sostanziale continuità.

Vecchio sistema:

- le singole Aziende dovevano generare una Comunicazione nei formati standardizzati **DbcTxt** e **DbcGis** [sia utilizzando i propri applicativi Aziendali, sia utilizzando i SW gratuiti OTRT **DbcPlus** / **GaiaTpl**, sia utilizzando un mix di entrambi]
- la Comunicazione veniva poi inviata in Osservatorio [a volte su supporto CD-ROM, più frequentemente come allegato e-mail]
- gli addetti all'Osservatorio procedevano alla verifica di plausibilità del contenuto delle Comunicazioni pervenute [in maniera sostanzialmente manuale]
- infine tutte le Comunicazioni pervenute finivano per alimentare il Database Corse [DBC] relativo alla copertura integrata dei servizi TPL sull'intero territorio regionale [sempre in maniera sostanzialmente manuale]

Nuovo sistema:

- le singole Aziende generano una Comunicazione nel nuovo formato standardizzato **DbcXml** [che è semplicemente una *versione formalmente ristrutturata e razionalizzata* dei precedenti **DbcTxt** e **DbcGis**]
- la Comunicazione viene poi inviata direttamente inviata ad un server [invio telematico], utilizzando un canale di trasmissione sicuro e verificato.
- Il SW di gestione presente sul server provvede automaticamente alla verifica preliminare di plausibilità della Comunicazione pervenuta.
- Dopo che un addetto all'Osservatorio ha dato un esplicito consenso [dietro verifica estesa], la Comunicazione finisce per aggiornare automaticamente Database Corse [DBC] relativo alla copertura integrata dei servizi TPL sull'intero territorio regionale

Quindi sostanzialmente il nuovo sistema introduce positivi elementi di maggiore automazione nei processi, che rendono praticamente possibile l'aggiornamento anche molto frequente del DBC; ovviamente l'obiettivo da perseguire è quello di avere costantemente disponibile la mappa

aggiornata che descrive l'offerta reale dei servizi TPL [da utilizzare p.es. per la pubblicazione *on-line* sul **Portale Regionale della Mobilità** <http://www.muoversiintoscana.it>]

Glossario:

La lettura del glossario non è affatto indispensabile [può anche essere saltata a piè pari], tuttavia è decisamente consigliabile per non trovarsi in difficoltà più avanti ...

XML: si tratta di uno standard internazionale che permette la rappresentazione rigorosamente formalizzata di qualsiasi tipo di dato strutturato. Rappresenta lo *standard de facto* per le comunicazioni dati tramite Web.

DbcXml: è il nuovo formato basato su XML adottato dall'OTRT per la rappresentazione delle Comunicazioni degli orari e delle cartografie delle reti TPL.

DbcXml level 1: è la versione di *DbcXml* che permette la rappresentazione degli orari e delle reti TPL senza richiedere la stretta conformità dei percorsi rispetto agli elementi del Grafo Stradale RT; il supporto al formato *level 1* è implementato dall'applicazione **DbcPlus**.

DbcXml level 2: è la versione di *DbcXml* che permette la rappresentazione degli orari e delle reti TPL in rigorosa conformità rispetto agli elementi del Grafo Stradale RT; il supporto al formato *level 2* è implementato dall'applicazione **GaiaTpl**.

LZMA: è un algoritmo di compressione dei dati particolarmente efficiente e pubblicamente disponibile in forma assolutamente gratuita [*open source*]; è molto più efficiente p.es. dei comuni ZIP o RAR. L'applicazione di riferimento per LZMA è **7-Zip** [gratuito, *open source*]

Indirizzo IP: è un numero che identifica in maniera assolutamente univoca qualsiasi apparecchiatura connessa ad Internet. P.es. **159.213.32.254** è l'indirizzo IP associato al server principale della Regione Toscana.

Porta IP: a ciascun indirizzo IP sono associate 32.535 possibili porte di comunicazione separate. Di regola le porte con numeri bassi sono associate ai servizi universalmente diffusi: p.es. la porta 21 è associata ai servizi *FTP*, la porta 25 è associata alla ricezione della posta elettronica, la porta 80 è associata ai servizi *Web* etc etc

Nome di dominio: è un identificativo letterale che corrisponde ad un indirizzo IP; p.es. www.rete.toscana.it corrisponde a 159.213.32.254. Un nome di dominio è molto più facilmente memorizzabile di una arida sequenza numerica.


URL: è l'identificativo che caratterizza qualsiasi risorsa disponibile sul Web. P.es. <http://www.rete.toscana.it/sett/pta/trasporti/sommario.htm> identifica la pagina iniziale del sito RT dedicato al TPL. Una URL comunemente è composta da tre elementi distinti:

- L'identificativo del protocollo di comunicazione [nel ns. esempio *http*]
- Il nome di dominio del server [nel ns. esempio www.rete.toscana.it]
- L'identificativo della risorsa [*target*] all'interno del server [nel ns. esempio */sett/pta/trasporti/sommario.htm*]

TCP: è il protocollo di comunicazione di rete finalizzato alle connessioni punto-punto tra un *trasmittente* ed un *ricevente*. In pratica tutti i normali servizi Internet [Web, eMail, FTP etc] sono basati su TCP.

HTTP: è il protocollo di comunicazione basato su *TCP* utilizzato universalmente dai servizi Web; una sessione *HTTP* avviene di norma tra un **WebServer** [Apache, IIS] ed un **browser** [Internet Explorer, Firefox, Netscape etc]. Per regola i servizi *HTTP* sono disponibili sulla porta **80** del server. Le comunicazioni *HTTP* avvengono in chiaro, quindi almeno teoricamente possono essere intercettate e falsificate da terze parti fraudolente o malevole. *HTTP* dovrebbe essere assolutamente ben familiare a tutti, visto che viene correntemente utilizzato quando si apre una connessione internet su qualsiasi sito Web [p.es. <http://www.rete.toscana.it>]

SSL/TSL: sono due ulteriori protocolli di comunicazione, sempre basati su *TCP*, che applicano una *cifratura* sui dati trasmessi / ricevuti. Dato che le *chiavi crittografiche* utilizzate sono note solo ed esclusivamente al *server* ed al *client*, una trasmissione *SSL/TSL* è affidabile e sicura, nel senso che non può essere intercettata o manomessa da terze parti ostili.

HTTPS: è il protocollo di comunicazione basato su *TCP* e su *SSL/TSL* strettamente analogo ad *HTTP*. Per regola i servizi *HTTPS* sono disponibili sulla porta **443** del server. Le comunicazioni *HTTPS* avvengono in maniera criptata, quindi non possono essere intercettate e falsificate da terze parti. Di regola i comuni browser [Internet Explorer, Firefox] visualizzano un piccolo lucchetto  quando si sta utilizzando una connessione protetta e sicura *HTTPS*. Le connessioni *HTTPS* sono comunemente utilizzate per i servizi bancari via Internet, per i pagamenti *on-line* tramite carta di credito etc.

Crittografia asimmetrica [a chiave pubblica / chiave privata]: si tratta di un sistema di crittografia moderno e sofisticato ritenuto molto difficilmente violabile. Ciascun utente del sistema dispone di una chiave privata [che deve custodire gelosamente, altrimenti la sicurezza è violata] e di una chiave pubblica [che al contrario può essere comunicata a chiunque senza rischi per la sicurezza]. Sono possibili i seguenti utilizzi:

- Il mittente cifra utilizzando la propria chiave privata; il ricevente decifra utilizzando la chiave pubblica del mittente. In questo modo il ricevente è assolutamente certo che il messaggio proviene realmente dal mittente; dato che la chiave pubblica del mittente può essere nota anche a terzi, il mittente non è affatto sicuro che solo il ricevente possa leggere il messaggio.
- Altrimenti il mittente può cifrare utilizzando la chiave pubblica del ricevente; il ricevente decifra utilizzando la propria chiave privata. In questo modo il mittente è assolutamente certo che solo il ricevente potrà leggere il messaggio. Però il ricevente non può essere certo che il messaggio provenga realmente dal mittente, visto che potrebbe provenire da chiunque conosca la chiave pubblica.
- Infine si possono combinare entrambi i criteri; in questo caso sia il mittente che il ricevente possono essere sicuri che non si sono verificate in alcun modo intromissioni di terzi nella comunicazione. Quindi la comunicazione può essere ritenuta realmente sicura, affidabile e certificata.

Certificati digitali: si tratta di piccoli file di modeste dimensioni che contengono all'interno tutte le *chiavi crittografiche* necessarie per identificare in modo sicuro sia il mittente che il ricevente. Un certificato digitale deve essere rilasciato da una *CA* [*Certification Authority*]. Di regola è opportuno proteggere i certificati con una *password* o con una *passphrase*, in modo tale da evitare usi abusivi in caso di trafugamento, smarrimento etc.

Password: si tratta di una sequenza arbitraria di caratteri posta a protezione di un *elemento sensibile* destinato a rimanere segreto; solo se la password viene digitata in modo assolutamente conforme [e quindi si dimostra praticamente di conoscere il segreto] è ammesso l'accesso all'elemento sensibile che si intende proteggere. In genere le password dovrebbero essere lunghe almeno una decina di caratteri [compresi cifre, segni di interpunzione etc] e non dovrebbero avere un significato umanamente comprensibile [per scoraggiare gli attacchi basati sull'uso di un dizionario].

- I seguenti sono esempi di *passwords assolutamente deprecabili*, in quanto decisamente troppo deboli per reggere ad un eventuale attacco malevolo:
 - mario, arezzo, cavallo, GiorgioRossi
- Invece i seguenti sono esempi di *passwords forti ed affidabili*:
 - 12Jkw#2rt;-#12asm, aWIPj78;jkPo8ed, ad90io-ab09-56hN

Passphrase: come mostra l'esempio precedente, è quasi impossibile pretendere che un operatore umano possa memorizzare una password sufficientemente robusta; quindi è preferibile utilizzare una *passphrase*, cioè una frase convenzionale facilmente memorizzabile [purché sufficientemente lunga e non banalmente scontata], come p.es. le seguenti:

- Mio cugino Enrico è nato nel 1962
- Adoro i bucatini all'amatriciana
- La capitale della Francia non è Versailles

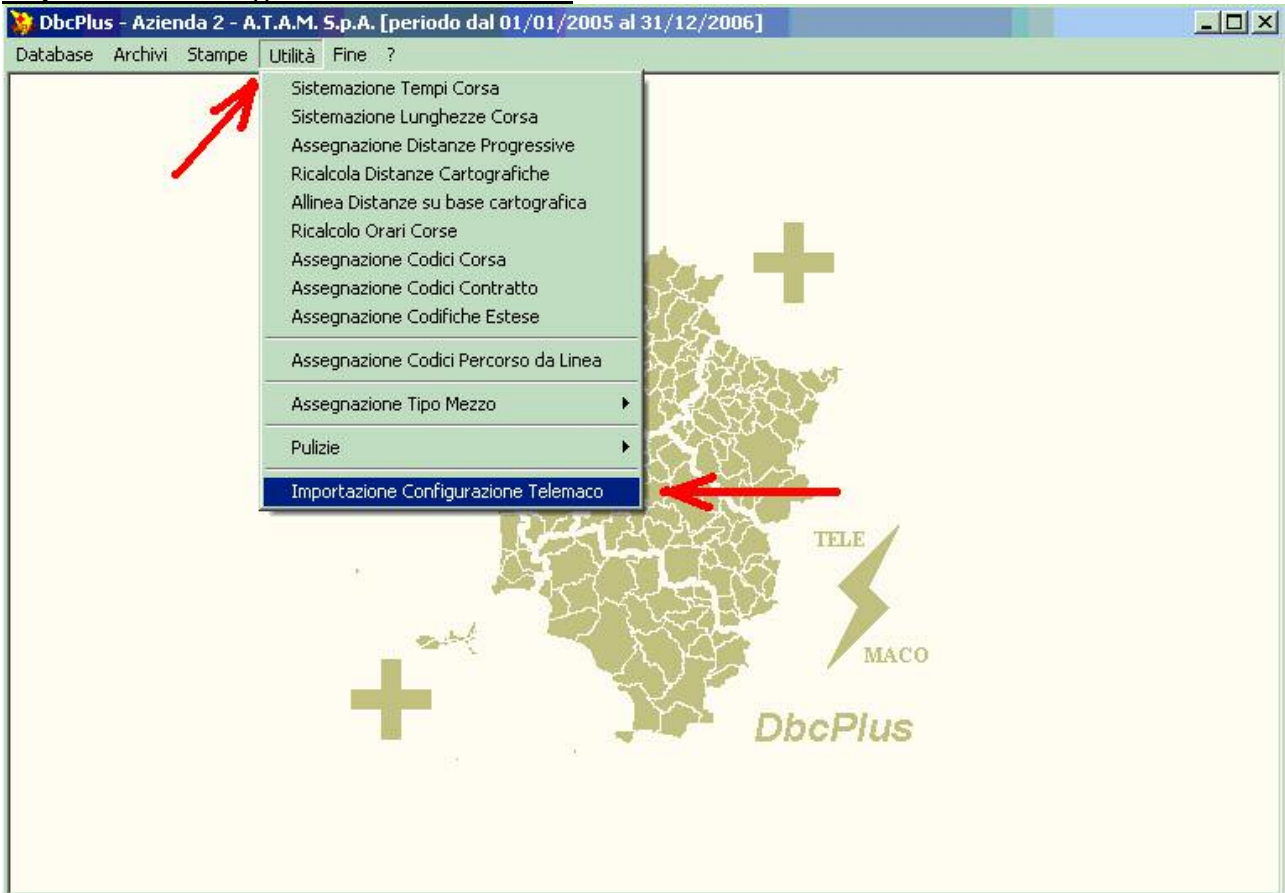
CA – Certification Authority: è una qualsiasi organizzazione in grado di emettere dei *certificati digitali* e di rendersi garante della loro autenticità e legittimità. Di norma le comuni CA di tipo commerciale o istituzionale [banche, enti etc] agiscono in quanto delegate in modo gerarchico da altre CA di livello superiore.

CA di primo livello: al vertice del sistema piramidale globale delle CA istituzionali sono collocate le 3 o 4 monopoliste mondiali del settore [*VeriSign, Twate etc*]. Per definizione le catene di certificazione che derivano gerarchicamente da una CA di primo livello si considerano automaticamente affidabili e sicure. Ma affidarsi ad un servizio commerciale di certificazione è economicamente oneroso [*l'emissione di ciascun singolo certificato ha un costo ...*].

CA autogestite: in alternativa è perfettamente possibile realizzare delle CA autonome che autogarantiscono la propria affidabilità, evitando in questo modo di passare per i sistemi commerciali di certificazione. Le prestazioni effettive del sistema sono assolutamente identiche. L'unico handicap è che una CA autogestita è assolutamente sconosciuta ai normali SW [*che quindi la ritengono inaffidabile per definizione*] a meno che l'utente non ritenga di concedere esplicitamente la propria fiducia. Le CA autogestite sono uno strumento assolutamente adeguato per gestire in piena sicurezza una comunità ristretta al cui interno tutti si conoscono e comunicano personalmente.

HowTo – come fare per ...

Importare la configurazione di Telemaco:



Per prima cosa occorre importare il *pod* contenete la configurazione da utilizzare per la connessione Telematica ed il proprio certificato digitale ... si tratta del file *NomeCognome.tlmc* ricevuto a suo tempo dall'OTRT ...

N.B. la configurazione del Server ed il certificato sono condivisi tra DbcPlus e GaiaTpl; quindi se si utilizzano entrambi gli applicativi sul solito PC basta importare la configurazione una sola volta per abilitarli entrambi.

Installare DbcSecureSend:

Quindi bisogna verificare che sul PC che si utilizza sia installato il componente ausiliario *DbcSecureSend* [fa parte delle risorse SW standard fornite dall'OTRT].

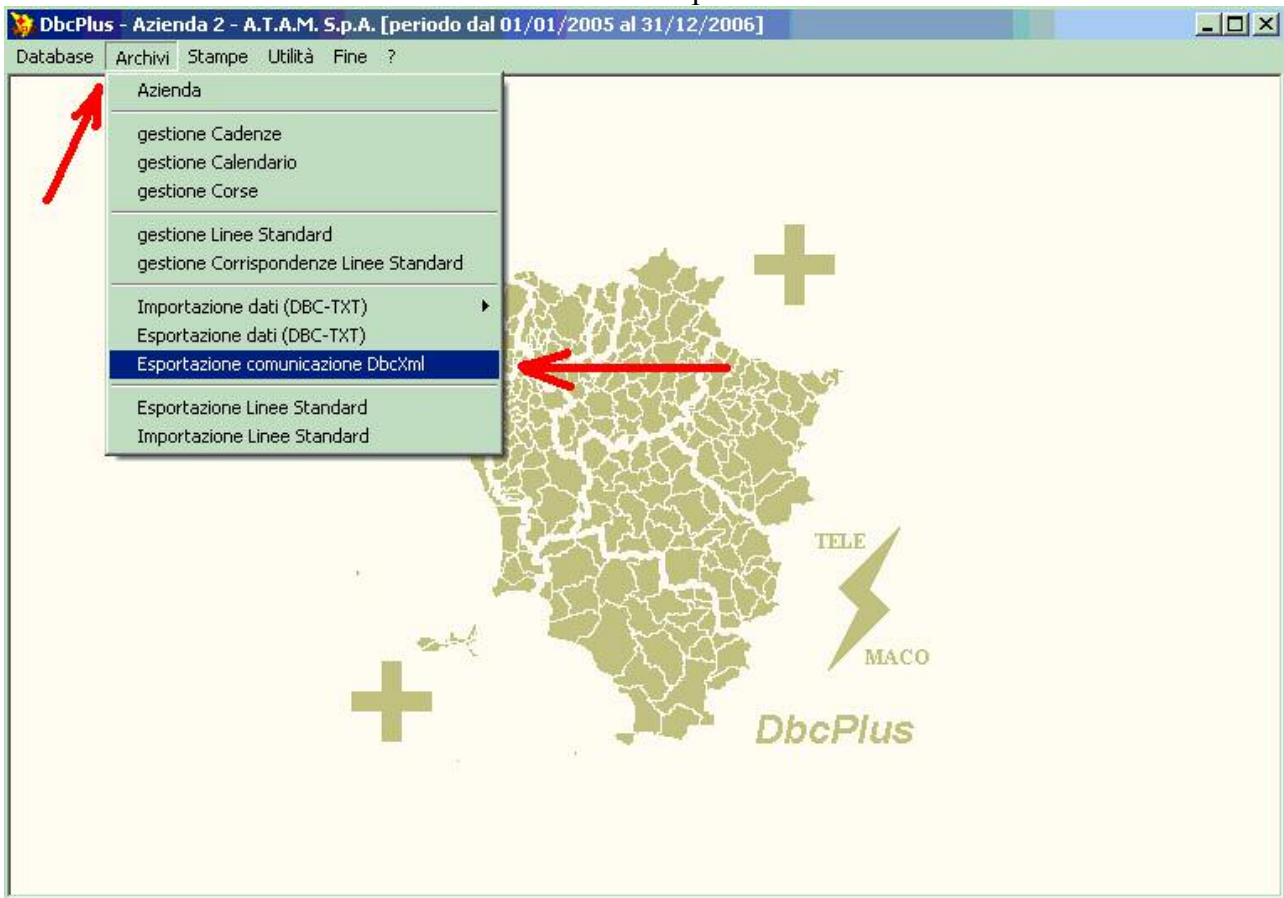
Questo componente opera congiuntamente a *DbcPlus* e/o *GaiaTpl* ed implementa le funzionalità *SSL/TSL* necessarie per gestire la comunicazione criptata e sicura *HTTPS*.

N.B. non è necessario installare due volte *DbcSecureSend* se si utilizza sia *DbcPlus* che *GaiaTpl*, visto che entrambe le applicazioni utilizzano un'unica identica copia di *DbcSecureSend*.

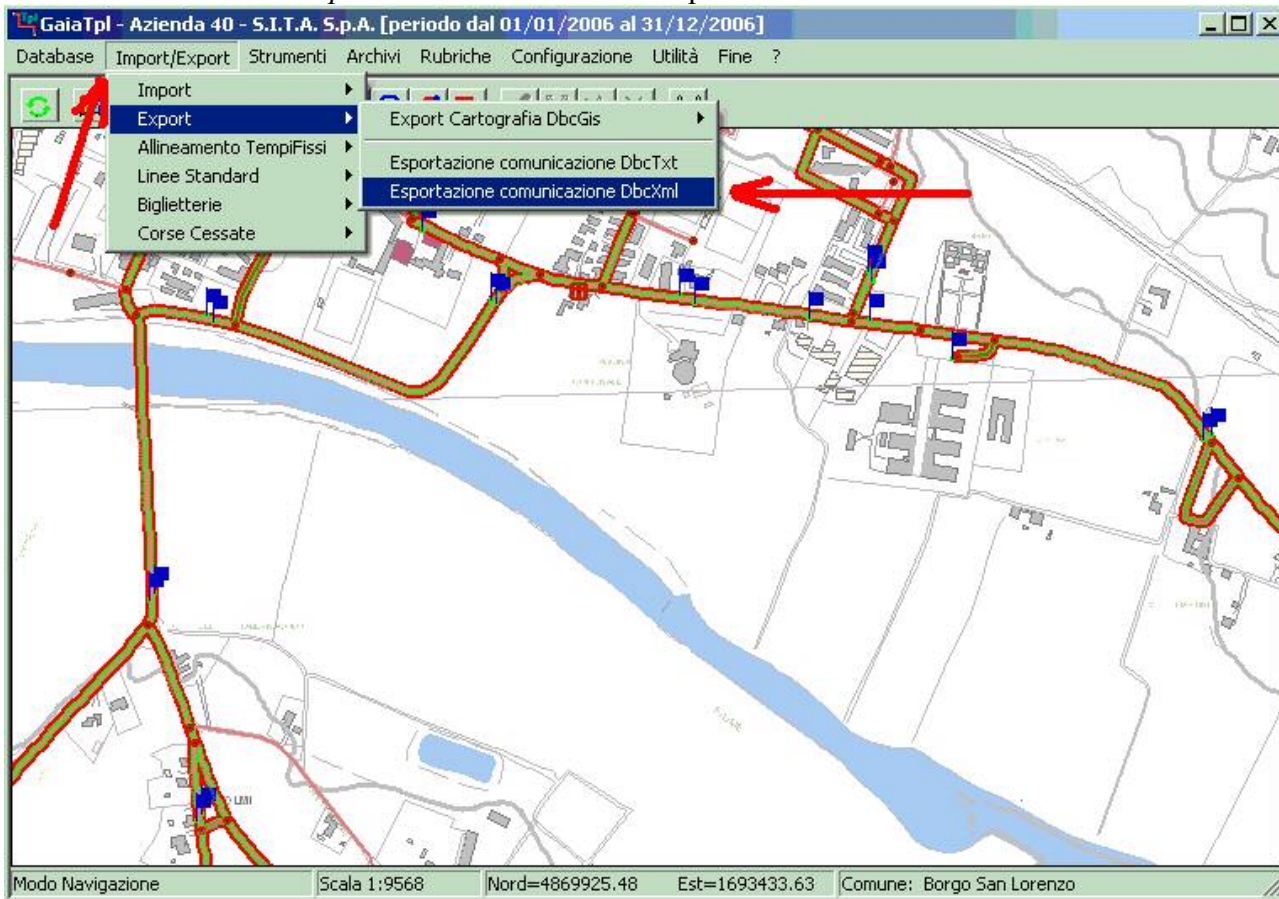
N.B. (2): non è affatto indispensabile installare *DbcSecureSend* se non si prevede di utilizzare la funzione di invio telematico; sia *DbcPlus* che *GaiaTpl* richiamano *DbcSecureSend* solo ed esclusivamente quando devono procedere ad un invio telematico su connessione *HTTPS*.

Generare ed inviare una Comunicazione DbcXml:

Nel caso si utilizzi *DbcPlus* la funzione si trova nella posizione indicata ...



Nel caso si utilizzi *GaiaTpl* la funzione si trova nella posizione indicata ...



N.B. per potere procedere alla generazione della Comunicazione *DbcXml* occorre assolutamente che il DB di partenza contenga dati corretti ed affidabili. In caso contrario apparirà il seguente messaggio:



In questa eventualità non sarà possibile procedere alla generazione delle Comunicazione; occorrerà salvare il report degli errori e quindi procedere alla correzione degli errori segnalati.

Oltre ai soliti ben noti criteri di convalida dei dati già implementati negli anni precedenti, ora sia *DbcPlus* che *GaiaTpl* richiedono tassativamente i seguenti elementi:

- Per ciascuna singola corsa devono essere presenti le informazioni di codifica estesa relative all'attribuzione del Lotto di Gara, dell'Azienda Associata esercente etc
- Deve necessariamente essere definita la struttura normalizzata del nomenclatore ufficiale relativo alla codifica delle Linee.

Fase I – impostazione dei criteri di filtro DbcXml:

The screenshot shows a dialog box titled "Parametri Estrazione Comunicazione DbcXml". It contains the following sections:

- Filtro Periodo Validità:** "Dal" field with date 31/05/2006, "Al" field with date 31/12/2006.
- Tipo Comunicazione:** Radio buttons for "Budget Contrattuale", "Piano di Servizio" (selected), "Progetto", and "Test".
- Filtro Lotto di Gara:** A dropdown menu showing "0002 - LOTTO ARETINO".
- Filtro Ente Contrattuale:** A dropdown menu with "Tutto" selected.
- Filtro Azienda Gestione:** A dropdown menu with "Tutto" selected.
- Filtro Azienda SubAppalto:** A dropdown menu with "Tutto" selected.
- Modalità estrazione:** Radio buttons for "invio telematico" (selected) and "generazione locale".
- Modalità compressione:** Radio buttons for "compressione LZMA" (selected) and "non-compresso".
- Buttons:** "OK" and "Annulla".

Per *default* i criteri di generazione della comunicazione DbcXml sono quelli illustrati:

- **Filtro periodo di validità:** vengono estratte le corse ed i calendari di servizio validi *a partire dalla data odierna* fino alla data finale del periodo contenuto all'interno del DB. *Si suppone che per buona norma l'Azienda comunichi le variazioni al piano di servizio prima di renderle esecutive ...*
- **Tipo di comunicazione:** per regola l'Azienda dovrebbe inviare delle Comunicazioni che identificano un nuovo *Piano di Servizio* variato / modificato / ristrutturato rispetto alla Comunicazione precedente ...
- **Filtro Lotto di Gara:** si suppone che una Comunicazione DbcXml venga di regola inviata dall'Azienda Scarl e quindi si riferisca ad un intero *Lotto di Gara* integrato ...
- **Modalità Estrazione:** di norma la Comunicazione deve essere immediatamente inviata all'Osservatorio Regionale [*invio telematico*]; in alternativa è possibile salvare localmente il file XML [*generazione locale*]. Solo nel caso della *generazione locale* è possibile attivare tutti i vari filtri di selezione disponibili.
- **Modalità Compressione:** nel caso di invio telematico della Comunicazione, il file XML verrà automaticamente compresso tramite LZMA. Nel caso di *generazione locale* è possibile scegliere se generare un file XML *non compresso* oppure un file *compressato LZMA*. Di norma il file compressato LZMA occupa solo il 10% dello spazio richiesto dal corrispondente XML non compresso.

The image shows a software dialog box titled "Parametri Estrazione Comunicazione DbcXml". It contains several sections for configuring data extraction:

- Filtro Periodo Validità:** "Dal" 31/05/2006, "Al" 31/12/2006.
- Filtro Lotto di Gara:** "0002 - LOTTO ARETINO".
- Filtro Ente Contrattuale:** "Tutto".
- Filtro Azienda Gestione:** "Tutto".
- Filtro Azienda SubAppalto:** "Tutto".
- Modalità estrazione:** "invio telematico" (selected), "generazione locale".
- Modalità compressione:** "compressione LZMA" (selected), "non compresso".
- Tipo Comunicazione:** "Budget Contrattuale", "Piano di Servizio" (selected), "Progetto", "Test". This section is circled in red.

Buttons for "OK" and "Annulla" are at the bottom.

Si ponga attenzione al fatto che è comunque possibile inviare all'Osservatorio Regionale i seguenti Tipi di Comunicazione:

- **Budget Contrattuale** il Budget Contrattuale descrive un Contratto di Servizio [si suppone con validità quantomeno annuale] così come definito inizialmente tra l'Azienda [Scarl] e l'Ente [Provincia] che gestiscono un Lotto di Gara. Per sua natura il Budget Contrattuale può essere comunicato una sola volta [ad inizio anno / periodo] e non può subire modifiche successive, visto che rappresenta la consistenza teorica di riferimento per il Contratto di Servizio in oggetto. Il Budget Contrattuale ha implicitamente anche valore di Piano di Servizio iniziale.
- **Piano di Servizio** un Piano di Servizio descrive la struttura reale e complessiva dei servizi TPL, così come risulta a seguito di tutte le variazioni / ristrutturazioni apportate. Quindi un *Piano di Servizio* modifica implicitamente il *Budget Contrattuale* di riferimento. Non ci sono limiti al numero ed alla frequenza di invio dei *Piani di Servizio*; in effetti è necessario inviare un nuovo *Piano di Servizio* ogni qual volta che si apporta una qualsiasi modifica ai servizi [teoricamente, anche nel caso di variazione di un singolo orario di transito su una singola corsa]
- **Progetto** una *Comunicazione di Progetto* non ha un reale valore esecutivo, quindi non andrà mai ad aggiornare il DataBase Corse, non varierà le consistenze contrattuali etc.
- **Test** infine un *Test* rappresenta un invio di prova; la comunicazione verrà sottoposta all'intero ciclo di convalida formale, ma non verrà memorizzata in alcun modo permanente.

Criteri di filtro applicati ai dati:

La generazione della Comunicazione *DbcXml* avviene sempre in stretta conformità con i criteri di filtro impostati: quindi non vengono mai generati riferimenti ad elementi ridondanti oppure inutilizzati. Se p.es. si esporta selezionando un solo Ente Contrattuale per un periodo temporale circoscritto, nella Comunicazione *DbcXml* appariranno solo ed esclusivamente:

- Le corse relative all'Ente purché effettuate in almeno un giorno del periodo richiesto
- Le fermate ed i percorsi necessari per rappresentare le corse di cui sopra.

Qualsiasi altro dato verrà omesso dalla Comunicazione (corse di natura diversa da quella indicata, fermate e percorsi non utilizzati dalle corse selezionate etc)

Fase II – impostazione dei parametri di connessione al Server TELEMACO:

Parametri Connessione TELEMACO

Autenticazione

Autenticazione richiesta dal WebServer

Autenticazione richiesta dalla CGI

UserName:

Password:

Indirizzo posta elettronica

Tutte le successive notifiche relative allo stato della comunicazione verranno comunicate a questo indirizzo di posta elettronica.

Accetta anche indirizzi multipli del tipo:
a.aaaa@xxxx.it; b.bbbb@xxxx.it

Url server TELEMACO

Server: Target:

Protocollo di comunicazione

HTTP [connessione non protetta]

HTTPS [connessione sicura - criptata]

Sicurezza e Crittografia

non occorre alcun certificato

richiede un certificato digitale

OK Annulla

Nessuna preoccupazione; se la fase di importazione dei parametri di connessione [descritta in precedenza] è già stata effettuata con successo si tratta solo di confermare le importazioni predefinite così come fornite dall'Osservatorio Regionale Trasporti.

Fase III – conferma generazione Comunicazione:

A questo punto inizia la vera e propria generazione della Comunicazione *DbcXml*; dopo una breve attesa [dovuta alla lettura del DB ed alla generazione del file XML] apparirà il seguente messaggio:



Se si tratta di una *generazione locale* l'elaborazione termina a questo punto; se invece si tratta di un *invio telematico* si prosegue ...

Fase IV – inserimento della passphrase:

Solo nel caso in cui i parametri di connessione richiedano l'utilizzo di un certificato digitale apparirà il seguente dialogo:



E' necessario introdurre manualmente la *password* [o meglio, la *passphrase*], proprio per garantire in modo assolutamente certo che sia effettivamente il legittimo titolare del certificato ad effettuare l'invio della Comunicazione.

Infatti un ipotetico utilizzo fraudolento o non autorizzato di un Certificato smarrito o trafugato fallirebbe, in quanto ben difficilmente l'eventuale malintenzionato potrebbe anche essere a conoscenza della *passphrase* associata al Certificato.

Fase V – compressione ed invio telematico della Comunicazione:

A questo punto inizia la vera e propria trasmissione telematica della Comunicazione:

- Per prima cosa il file XML [generalmente di dimensioni rilevanti; diverse decine di MB, tipicamente] viene compresso secondo LZMA; il file compresso è di dimensioni molto inferiori, tipicamente pochi MB, quindi la fase successiva [trasmissione vera e propria] procederà molto più speditamente.
- Quindi viene contattato il Server TELEMACO e, se possibile, viene inviato il file in *upload*.



a