

Oggetto :	<b>2G, Aspetti connessi con l'interoperabilità applicativa</b>	<b>Revisione</b>
	<b>Descrizione dell'architettura generale di 2G con riferimento alle funzionalità strutturali per l'accesso di applicazioni esterne alle informazioni del sistema.</b>	<b>1.0</b>
		<b>11/10/2011</b>

## Sommario

1. Premessa .....	1
2. Cos'è 2G.....	1
3. Architettura attuale e limiti all'interoperabilità .....	1
4. L'architettura 2G.....	3
5. Impiego di 2G per l'interoperabilità e futura architettura RIA .....	4

### 1. Premessa

La presente informativa descrive in termini generali le principali caratteristiche della piattaforma 2G con espresso riferimento ai vantaggi offerti in termini di interoperabilità applicativa, ovvero di accesso condiviso alle informazioni di sistema da parte di più applicazioni, anche prodotte da aziende diverse. Trattandosi di una nota divulgativa i concetti tecnici sono espressi in forma semplificata.

### 2. Cos'è 2G

**2G è il nome assegnato alla nuova tecnologia Gesinf** per lo sviluppo e l'impiego di applicazioni complesse, quali ad esempio Team Gov. Oltre ad essere un ambiente di produzione, sviluppato e utilizzato presso Gesinf per la realizzazione dei programmi, **2G comporta l'adozione di una specifica architettura presso i clienti che la utilizzano**, architettura definita in termini tecnici come SOA (*basata sui servizi*) e RIA (*basata su interfacce web evolute*) della quale più avanti verrà data breve spiegazione. Nei piani Gesinf la diffusione della nuova tecnologia porterà alla migrazione graduale di tutte le principali applicazioni attualmente in uso presso i Clienti entro alcuni anni, compatibilmente con le esigenze e le tempistiche specifiche.

Caratteristica importante delle applicazioni basate su 2G è che le stesse potranno convivere con gli attuali software nell'ambito dello stesso cliente, ovvero che anche solo una parte delle funzionalità dei programmi attuali (ad esempio specifiche gestioni o moduli appositamente realizzati) potranno essere utilizzate sfruttando le nuove tecnologie, senza necessità di sostituire integralmente il software attualmente a regime.

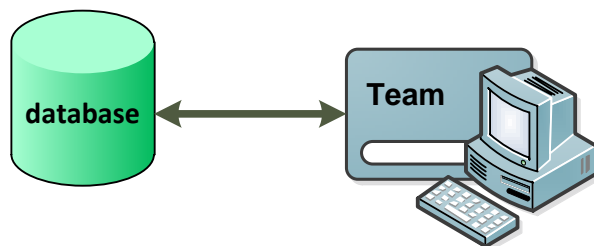
Altra caratteristica importante è la possibilità di distinguere una applicazione 2G in due componenti indipendenti: a) lo strato dei servizi (o *servizi Server*), oggetto della presente nota informativa in quanto impiegabile anche autonomamente per l'interoperabilità applicativa; b) l'interfaccia RIA, che costituisce l'insieme delle maschere web per l'uso del sistema, non oggetto del presente documento e non ancora resa disponibile da Gesinf.

### 3. Architettura attuale e limiti all'interoperabilità

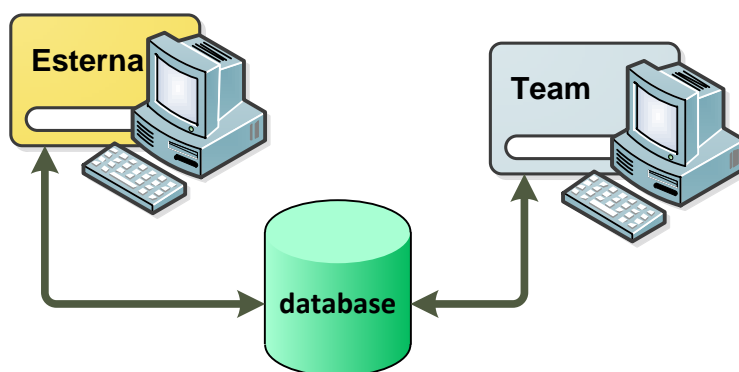
L'attuale architettura delle applicazioni Gesinf, ad esempio Team Gov, è basata sulla presenza di un *Database*<sup>1</sup> (insieme di archivi che contengono i dati del cliente) e di un programma *Client*, scritto da Gesinf, che gira sulle singole postazioni di lavoro degli utenti. I singoli *Client* leggono e scrivono dati

<sup>1</sup> In realtà i *database* impiegati sono più di uno per ciascuna applicazione, ma non è rilevante ai fini della presente descrizione.

tramite un collegamento *diretto* al database, al quale accedono utilizzando ciascuno le proprie credenziali (utente e password).



Nell'attuale software il programma Gesinf è pertanto contenuto all'interno delle singole stazioni di lavoro e il database funziona da contenitore di dati<sup>1</sup>. Questa architettura, fin qui utilizzata nella grande maggioranza delle applicazioni aziendali, presenta una serie di limiti sia di natura tecnica che per quanto riguarda l'accesso ai dati da parte di applicazioni esterne, ad esempio quelle di un altro produttore di software. Relativamente a quest'ultimo aspetto (l'accesso dall'esterno ai programmi Gesinf), l'architettura perseguibile attualmente è sempre basata sull'accesso diretto ai dati:



Questa modalità comporta una serie di limiti:

1. **Uso di internet.** Le applicazioni esterne devono essere utilizzate nell'ambito della rete locale dove risiedono i dati, ad esempio non è possibile che siano in collegamento internet remoto (in questi casi sarebbe possibile uno scambio dati tramite l'invio e la ricezione di archivi, ma non è equiparabile ad un accesso in tempo reale).
2. **Restrizioni di Accesso.** Per quanto sul database possano essere definite delle restrizioni di accesso, queste non sono in genere sufficienti per garantire una reale protezione dei dati. Ad esempio: è possibile impedire l'accesso in lettura ad una intera tabella ma, se questa è visibile, non è possibile limitare l'accesso solo a determinati campi o determinate registrazioni; è possibile impedire la scrittura ma, se concessa, la stessa non può essere vincolata facilmente a regole di processo.
3. **Tracciamento delle operazioni.** Salvo il ricorso a tecniche particolarmente onerose, inapplicabili nella maggior parte dei casi reali, non è possibile conservare una traccia esatta di tutte le operazioni compiute dalle applicazioni terze e risulta pertanto difficile, se non impossibile, stabilire quale applicazione ha prodotto un determinato dato o ha modificato dati già esistenti nel database.
4. **Logica di Elaborazione.** Salvo che per una parte delle regole presenti nel sistema<sup>2</sup>, non è possibile controllare le registrazioni effettuate dall'applicazione esterna, poiché il programma Gesinf è al di fuori

<sup>1</sup> In realtà il programma Gesinf è contenuto in parte anche nel *database* (logica server) ma il concetto qui è fortemente semplificato.

<sup>2</sup> Il riferimento è alla logica contenuta nel database, comune a tutte le applicazioni, che tuttavia riguarda solo parte delle regole del programma.

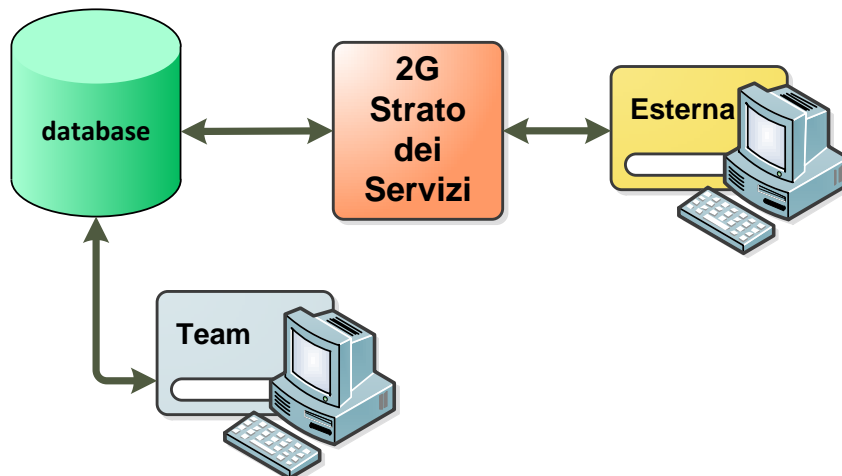
del collegamento tra questa e il database, e non può quindi intervenire per impedire scritture anomale o parziali.

5. **Standard di Comunicazione.** Seppure quasi tutti i produttori di software siano in grado di connettersi direttamente ad un database come quello in oggetto, le informazioni relative alle tabelle ed ai campi delle stesse e le politiche di inserimento/modifica sono di natura proprietaria, ovvero il fornitore non ha linee guida da seguire per l'inserimento/modifica e deve pertanto riferirsi alla documentazione di sistema e del database, nei limiti per la quale è disponibile. Tale aspetto è particolarmente importante per quanto riguarda la stabilità del sistema in caso di aggiornamento dei programmi Gesinf: in questi casi, la variazione ad una struttura dati può comportare l'errato funzionamento dell'applicazione esterna, che dovrà pertanto essere nuovamente adeguata.
6. **Configurazione degli accessi.** La configurazione delle restrizioni e delle possibilità concesse alle applicazioni esterne è possibile, ma necessita di una discreta serie di attività tecniche che possono rendere complessa l'operazione o rendere difficilmente manutenibile il sistema.

#### 4. L'architettura 2G

La piattaforma 2G, coerentemente con lo sviluppo tecnologico intervenuto negli ultimi anni nel settore informatico, consente di affrontare efficacemente le problematiche di interoperabilità applicativa attraverso i concetti del *multi-tier* (letteralmente *multi-strato*) e di *SOA - Service Oriented Architecture* (letteralmente *Architettura orientata ai servizi*). Tale tecnica, oltre ad offrire una serie di vantaggi tecnici ed operativi non oggetto del presente documento, rappresenta un punto di arrivo per quanto riguarda l'apertura dei sistemi verso l'esterno, nascendo proprio in virtù di tali problematiche.

In 2G il database è separato dalle applicazioni esterne attraverso l'introduzione di uno *Strato dei Servizi*<sup>1</sup> (*Service Layer*), il quale si occupa di fornire alle stesse le funzionalità del sistema (i *Servizi* appunto) gestendo le operazioni sui dati:



In pratica il *service layer* di 2G rappresenta una sorta di *ufficio relazioni esterne* rispetto al sistema: espone una serie di servizi a disposizione dei terzi per leggere/scrivere dati, controllandone prima i contenuti, ed eventualmente adattandoli alle esigenze del sistema, e tiene traccia di tutto ciò che viene effettuato dall'esterno. L'applicazione esterna non ha bisogno di conoscere l'ubicazione o la struttura del database e, soprattutto, può chiedere informazioni allo strato dei servizi su quali dati sono disponibili, ed in quale formato vanno richiesti, utilizzando un protocollo standard, universalmente diffuso e indipendente dall'hardware e dal software utilizzato: l'*XML* (una sorta di *esperanto* che, ad oggi, è lo standard consolidato nell'interscambio dati).

I vantaggi di tale soluzione sono molteplici:

<sup>1</sup> In termini tecnici 2G espone *Web Services* basati su *SOAP*, generalmente su protocollo *https*.

1. **Uso di internet.** Le applicazioni esterne possono interagire con il sistema anche tramite internet che, anzi, rappresenta uno dei mezzi principali di comunicazione in queste architetture.
2. **Restrizioni di Accesso.** Lo strato dei servizi 2G consente di effettuare restrizioni a qualunque livello dei dati, in lettura/scrittura, senza le limitazioni imposte dal database.
3. **Tracciamento delle operazioni.** 2G traccia automaticamente qualunque chiamata dall'esterno che, in qualche modo, possa condurre ad una modifica di dati, ed è in grado - se richiesto - di tracciare anche qualunque informazione letta all'esterno, a diversi livelli di dettaglio facilmente configurabili. In pratica 2G può registrare qualunque attività condotta da un'applicazione esterna in forma completa.
4. **Logica di Elaborazione.** 2G integra le regole di aggiornamento dei dati, lato applicativo, ed è in grado di verificare che le chiamate dall'esterno siano coerenti con esse prima di operare nel database. 2G può anche trasformare i dati ricevuti prima di effettuare scritture, sulla base di regole specifiche.
5. **Standard di Comunicazione.** Come già accennato, 2G comunica attraverso un protocollo standard indipendente dal database e dal software esterno, e fornisce automaticamente documentazione *online* circa la formattazione dei messaggi e le informazioni reperibili, informazioni che il software esterno generalmente utilizza in modo automatico e guidato<sup>1</sup>. I servizi sono indipendenti da eventuali variazioni strutturali alla base dati, pertanto le applicazioni terze possono basare l'interscambio dati su formati auto-documentati, che non ne inficiano il funzionamento a parità di informazioni necessarie al servizio (questo significa, ad esempio, che lo scambio dei dati è relativamente immune da aggiornamenti tecnici al database di Team).
6. **Configurazione degli accessi.** 2G prevede una serie di strumenti per la configurazione rapida delle applicazioni esterne, che consentono di tenere sotto controllo diritti e restrizioni in modo agevole e auto-documentato.

Relativamente alle applicazioni Gesinf su tecnologia tradizionale, è importante sottolineare che le stesse continuano ad operare tramite collegamento diretto al database - come si evince dallo schema sopra indicato - senza alcun cambiamento tecnico, ovvero coesistono, nell'ambito del sistema, insieme allo strato di servizi che si occupa delle applicazioni esterne. Eventuali nuovi moduli espressamente concepiti potranno invece sfruttare lo strato dei servizi 2G per poter operare su web, tramite un nuovo tipo di interfaccia (ad esempio moduli per la rappresentazione di cruscotti aziendali o la visualizzazione di dati direzionali in forma aggregata, sfruttando i vantaggi del web).

## 5. Impiego di 2G per l'interoperabilità e futura architettura RIA

2G è attualmente proposto solo per la componente relativa allo strato dei servizi (*service layer*) e pertanto il suo impiego è principalmente rivolto alle esigenze di interoperabilità tra applicazioni, locali o remote. Tuttavia il progetto di conversione tecnologica riguarda l'introduzione graduale di interfacce web in sostituzione delle attuali procedure Team, fino al completo abbandono della piattaforma attualmente in uso presso i clienti a favore di un sistema interamente SOA/RIA (basato su servizi e interfaccia web evoluta). In questo senso, il *service layer* di 2G può essere visto anche come il primo passo verso l'evoluzione del sistema sulle nuove tecnologie, oltre a costituire, da subito, la base sulla quale realizzare specifiche interfacce web 2.0 che operano sui dati dell'applicazione.

---

<sup>1</sup> Tecnicamente, il riferimento è alla presenza del WSDL (Web Services Description Language) ed agli schemi XSD (XML schema).