

Come rendere più rapido il processo di integrazione delle applicazioni

La chiave sta in un alto livello di astrazione

INDICE

Applicazioni host: la natura del problema	pag. 1
Un esempio di programmazione	pag. 2
Un vantaggio aggiuntivo: la riduzione del traffico di rete	pag. 5
Integrazione rapida a livello host, assolutamente irrinunciabile	pag. 5
Una serie di soluzioni per l'integrazione	pag. 5
Attachmate	pag. 5

Come rendere più rapido il processo di integrazione delle applicazioni

La chiave sta in un alto livello di astrazione

Il tempo è denaro e specialmente in questo periodo le divisioni informatiche sono a corto di entrambi. Tuttavia, l'ampia diffusione di applicazioni customer-facing e le grandi aspettative di clienti tecnicamente preparati non lasciano loro altra scelta: devono garantire a clienti e partner l'accesso immediato a informazioni e servizi potendo contare esclusivamente sulle competenze di un personale sempre più ridotto.

Esiste anche un problema tecnico di notevole portata: la maggior parte dei dati e della logica di business che stanno dietro a queste nuove applicazioni risiede ancora su applicazioni host, dove generalmente non esiste una separazione netta tra dati e logica, per cui l'accesso a uno di questi due elementi a fini di integrazione si è sempre rivelata un'operazione lenta e difficile.

Verastream Host Integrator di Attachmate offre una modalità radicalmente nuova di integrazione delle applicazioni mainframe. In sintesi, Verastream Host Integrator trasforma dati e logica di business non strutturati in servizi accessibili in modo strutturato da parte degli sviluppatori di applicativi. In questo white paper viene data una descrizione tecnica generale dell'approccio adottato da Verastream Host Integrator, basato su un alto livello di astrazione, compreso l'uso di tabelle e procedure per accelerare il processo di integrazione.

Applicazioni host: la natura del problema

L'accesso programmatico ai dati e alla logica che risiedono sull'host attraverso l'interfaccia del terminale inizia generalmente con un processo di modellazione, durante il quale i soggetti che hanno familiarità con l'applicazione host selezionano schermate, definiscono input e output di dati e registrano le interazioni degli utenti. Il risultato del processo è un servizio che ingloba le funzionalità host e che viene collocato su un server middle-tier, da cui il servizio è accessibile tramite una API (application programming interface).

La manipolazione di applicazioni host attraverso una API può rivelarsi complessa. Come si vede nella fig. 1, una schermata standard contiene numerosi campi di input, set di record di dati che scorrono attraverso schermate multiple e una serie di opzioni di navigazione. Senza contare che, generalmente, l'esecuzione di una transazione host comporta l'interazione con un numero elevato di schermate dell'applicazione.

La maggior parte dei prodotti di integrazione host fornisce API di accesso all'applicazione a un livello di astrazione relativamente basso, per cui lo sviluppatore deve avere una conoscenza approfondita dell'applicazione host per riuscire a navigare attraverso le schermate, inserire ed estrarre dati.

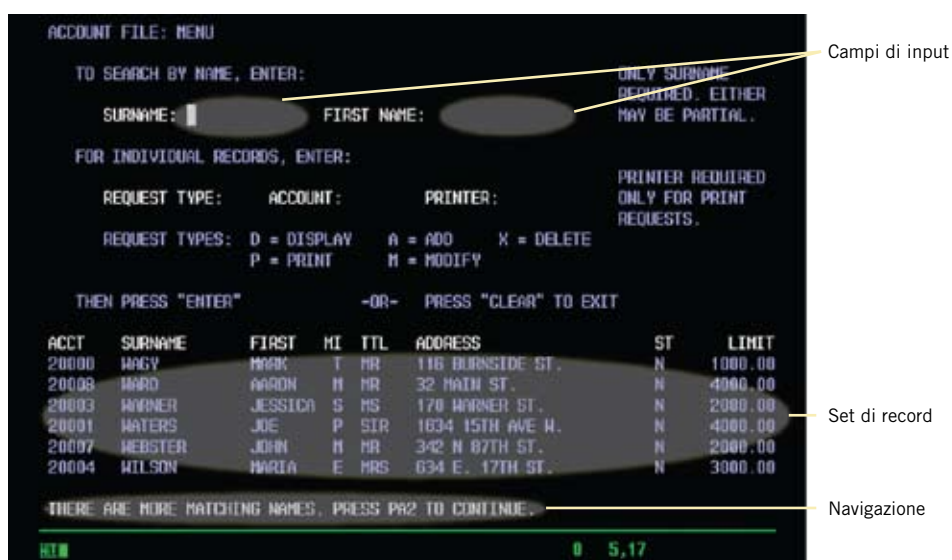


Fig. 1 – Schermata di un'applicazione host

Verastream Host Integrator abbina invece un'ampia gamma di API Java e COM a livelli supplementari di astrazione che permettono agli sviluppatori di applicativi di ignorare completamente la complessità delle applicazioni host sottostanti. Nella fig. 2 vengono illustrati i vari livelli di astrazione in un servizio Verastream Host Integrator, le cui API garantiscono l'accesso a ciascun livello di astrazione.

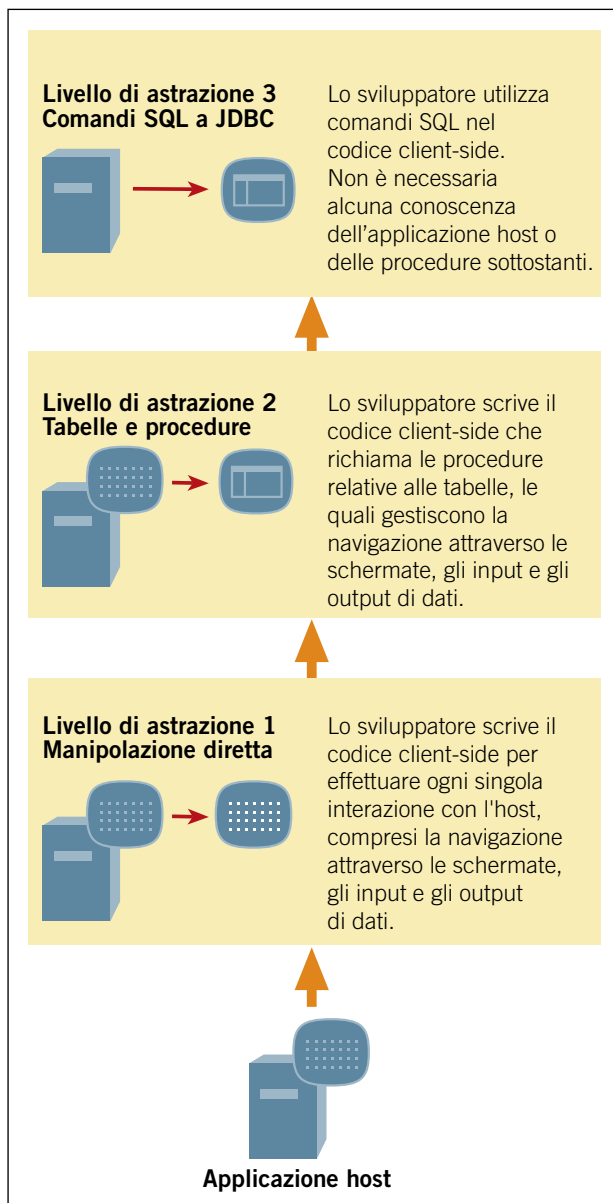


Fig. 2 – Livelli di astrazione di Verastream Host Integrator

Nel corso del processo di modellazione con Verastream, gli esperti delle applicazioni host definiscono le tabelle creando un elenco di colonne di database in cui a ciascun dato viene attribuito un nome. All'interno delle tabelle, le procedure mappano gli input e gli output di dati provenienti da una qualsiasi schermata dell'applicazione host e gestiscono automaticamente tutti i dettagli relativi alla navigazione e alle transazioni. Le procedure, comprese quelle di select, update, insert e delete, possono essere combinate per eseguire una serie complessa di transazioni.

Le tabelle e le procedure di Verastream Host Integrator consentono agli sviluppatori di lavorare con applicazioni host non strutturate come se si trattasse di un normale database. Nella fig. 3 viene visualizzata la finestra di dialogo Tabelle e procedure di Verastream Host Integrator.

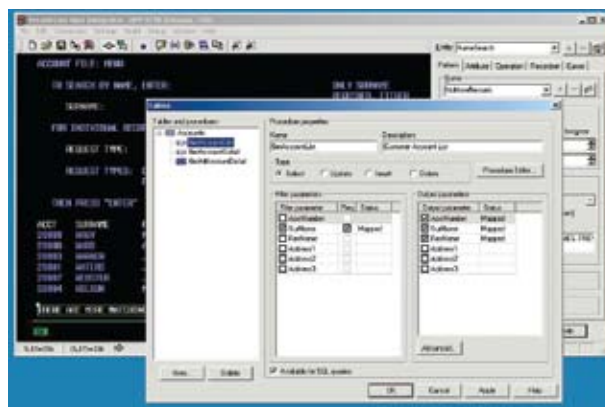


Fig. 3 – Tabelle e procedure di Verastream Host Integrator

Un esempio di programmazione

L'importanza dell'uso di un alto livello di astrazione per snellire l'integrazione può essere evidenziata da un semplice esempio di programmazione. Nel nostro esempio, l'utente di un'applicazione host vuole recuperare informazioni su tutti i clienti il cui cognome inizia con la lettera "w".

Nel caso di un'applicazione host non integrata, questa transazione inizia con l'inserimento della lettera "w" da parte dell'utente, dopo di che l'applicazione visualizza un elenco di numeri di conto relativi ai clienti il cui cognome inizia per "w". A questo punto l'utente, per recuperare le informazioni relative a

ciascun numero di conto, deve inserire manualmente il relativo numero. Questo processo comporta numerose operazioni ripetitive attraverso una serie di schermate. Non esiste infatti la possibilità di raccogliere le informazioni in un'unica schermata o di ordinare i risultati per numero di conto.

Utilizzando invece un'applicazione gestionale integrata, con il semplice inserimento della lettera "w" l'utente è in grado di ottenere risultati molto più soddisfacenti: quest'unica operazione genera infatti un elenco completo e ordinato di cui la tabella della fig. 4 porta un possibile esempio.

Numero di conto	Cognome	Nome	Indirizzo 1	Indirizzo 2
20000	Wagy	Mark	116 Burnside St	Portland, OR 97219
20001	Waters	Joe	1634 15th Ave W	San Diego, CA 92101
20003	Warner	Jessica	170 Warner St	Warner, MN 26582
20004	Wilson	Maria	634 E 17th St	Taos, NM 98765
20006	Winters	Shelly	1400 Snowy Lane	Portland, OR 38694
20007	Webster	John	342 N 87th St	Seattle, WA 98456
2008	Ward	Aaron	32 Main St	Plymouth, IN 74856

Fig. 4 – Risultato della transazione

La tabella della fig. 4 può essere generata in due modi diversi: utilizzando API di basso livello oppure un alto livello di astrazione. Nella fig. 5 viene messa a confronto la quantità di lavoro richiesta da ciascuno dei due metodi. Nella colonna centrale è riportata la notevole quantità di codice client necessaria per manipolare direttamente l'applicazione host utilizzando API di basso livello. Nella colonna di destra viene invece dimostrato come sia possibile ottenere lo stesso risultato tramite un unico comando SQL, una shortcut resa possibile dalle tabelle e dalle procedure di Verastream Host Integrator.

Fasi	Codice applicativo utilizzando una API di basso livello	Codice applicativo utilizzando tabelle e procedure Verastream
Inserire la lettera "w" nell'applicazione host	<pre>HashMap attributesMap = new HashMap(); Vector fieldNames = new Vector(); Vector detailRecords = new Vector(); attributesMap.put("Last Name", "w"); // write w to the LastName attribute mySession.setAttributes(attributesMap);</pre>	
Andare alla schermata dei risultati della ricerca	<pre>// navigate to entity NameSearch Results mySession.setCurrentEntity("NameSearchResults"); fieldNames.add("AcctNum"); // fetch records with the field name AcctNum Records = mySession.fetchRecords(0, fieldNames, null); // create an array of acct numbers</pre>	
Visualizzare i numeri di conto	<pre>String[] acctArray = new String[records.getCount()]; for (int i = 0; i < records.getCount(); i++) {</pre>	
Ordinare l'elenco dei numeri di conto	<pre>acctArray[i] = records.get(i).get("AcctNum"); } // sort the array of acct numbers Arrays.sort(acctArray); // additional attributes to get for each record in the recordset fieldNames.add("LastName"); fieldNames.add("FirstName"); fieldNames.add("Address1"); fieldNames.add("Address2"); // iterate thru the sorted acct number array from name search to get acct details for (int i = 0; i = acctArray.length; i++) { // navigate to the entity Main mySession.setCurrentEntity("Main");</pre>	
Ritornare alla schermata del menu principale Inserire uno dei numeri di conto	<pre>// get the acct number from the current record to write to the AcctNum attribute attributes.Map.put("AcctNum", acctArray[i]); // write the AcctNum attribute mySession.setAttributes(attributes.Map);</pre>	<pre>AppConnRecordSet records = mySession.executeSQLStatement("Select AcctNum, LastName, FirstName, Address1, Address2 from Accounts where LastName like 'w' order by AcctNum")</pre>
Scrivere il numero di conto nell'applicazione host	<pre>// navigate to the entity RecordDisplay mySession.setCurrentEntity("RecordDisplay");</pre>	
Andare alla schermata dove è visualizzato il record	<pre>// read the attributes for the acct number record = mySession.getAttributes(fieldNames);</pre>	
Recuperare i dettagli relativi al conto	<pre>// store account details in an array of records detailRecords.add(record);</pre>	
Salvare i dettagli del conto per un eventuale utilizzo futuro Tornare alla schermata del menu principale Ripetere l'operazione per ciascun numero di conto		

Fig. 5 – Confronto tra codifiche

L'esempio di transazione riportato in questa sede è abbastanza semplice. Con l'aumentare della complessità delle transazioni saranno notevolmente superiori anche i vantaggi offerti da Verastream in termini di rapidità di sviluppo.

Un vantaggio aggiuntivo: la riduzione del traffico di rete

La rapidità del processo di sviluppo di un'applicazione è solo una parte dell'accelerazione resa possibile dalle tabelle e dalle procedure Verastream.

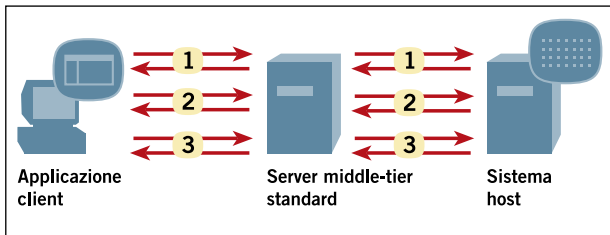


Fig. 6 Traffico di rete generato da API di basso livello

Se le applicazioni client vengono scritte in modo tale da interagire direttamente con le applicazioni host, ciascuna interazione comporta un percorso completo della rete (vedi fig. 6). Ad esempio, il codice API di basso livello della fig. 5 dà origine a oltre 20 percorsi completi della rete.

Se le applicazioni client vengono invece scritte in modo da poter usufruire delle tabelle e delle procedure Verastream, il traffico di rete viene ridotto e l'elaborazione delle transazioni è spostata dall'applicazione client al server middle-tier Verastream.

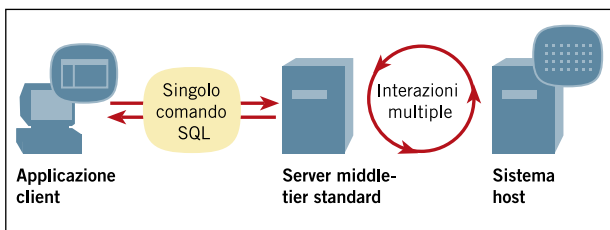


Fig. 7 Traffico di rete generato da un singolo comando SQL

Come appare evidente nella fig. 7, un comando SQL generato dal codice applicativo delle tabelle e delle procedure genera un unico percorso completo della rete tra l'applicazione client e il server Verastream Host

Integrator. Il resto dell'elaborazione delle transazioni viene eseguito solo tra il server e l'applicazione host, riducendo notevolmente il traffico di rete complessivo e incrementando la performance.

Integrazione rapida a livello host, assolutamente irrinunciabile

Il ritmo dell'innovazione a livello aziendale sta aumentando parallelamente alla necessità per le divisioni informatiche di ottenere maggiori risultati con minori risorse. L'esclusiva rappresentazione in forma di database di applicazioni host non strutturate da parte di Verastream Host Integrator consente un'integrazione più rapida riducendo la complessità del processo di sviluppo e ottimizzando le competenze dei singoli. Dopo il deployment, Verastream Host Integrator consente inoltre di continuare a rendere più veloce l'elaborazione di nuove applicazioni incrementando la performance della rete.

Una serie di soluzioni per l'integrazione

Verastream Host Integrator fa parte della suite di integrazione di applicazioni legacy Verastream Attachmate, una serie completa di tool di modernizzazione a livello mainframe, desktop e Web. Le nostre soluzioni offrono un'intera gamma di funzionalità di modernizzazione: semplice rejuvenation, presentazione personalizzata, integrazione avanzata ad alte prestazioni. I servizi generati da Verastream possono essere uniti tra di loro, abbinati e riutilizzati selettivamente per estendere la funzionalità legacy a nuove applicazioni o a nuovi utenti senza necessità di modificare il codice delle applicazioni legacy.

Attachmate

Attachmate fornisce soluzioni software evolute per l'emulazione di terminale, l'integrazione di applicazioni e le comunicazioni sicure. La business unit NetIQ offre soluzioni per la gestione delle applicazioni, i sistemi enterprise, la sicurezza e la compliance. Grazie alle nostre tecnologie, oltre 65.000 aziende in tutto il mondo sono ora in grado di far funzionare in modo nuovo e più proficuo le proprie risorse IT. Per ulteriori informazioni visitare il sito www.attachmate.com.



Sede Centrale
1500 Dexter Avenue North
Seattle, Washington 98109
TEL 206 217 7500
FAX 206 217 7515

Sede Centrale EMEA
Paesi Bassi
TEL +31 172 50 55 55
FAX +31 172 50 55 51

Sede Italiana di Milano
TEL +39 02 99060201
FAX +39 02 99044784

Sede Italiana di Roma
TEL +39 06 5423281
FAX +39 06 5408851

WEB <http://it.attachmate.com>
EMAIL informazioni.italia@attachmate.com

Per le informazioni sulle sedi locali, visitare <http://it.attachmate.com>.