



SOVERAIN

233 S. Wacker Drive, Suite 9425, Chicago, IL 60606
www.Soverain.com

SecureLink Web Services

Web Services for the SecureLink API

June 2008



Table of Contents

Introduction to SecureLink Web Services	3
Overview	3
Introduction to Web Services	3
Web Services Functionality in Transact 6.5	3
Buyer Profile Web Service.....	3
Order Entry Web Service.....	3
SecureLink Web Service	4
Multi-Item Web Service	4
Functionality Available via Web Services.....	5
SecureLink API Limitations	5
Interoperability	5
Deployability	5
SecureLink API Enabled with Web Services Advantages.....	6
Interoperability	6
Usability	6
Reusability	6
Deployability	6
Ubiquity	7
Accessing SecureLink Web Services	8
Requirements.....	8
Software Requirements	8
System Requirements	8
Download and Installation	8
Download	8
Installation	8

Introduction to SecureLink Web Services

Overview

The Transact 6.5 release provides Web Services for the SecureLink API. This feature enables merchants to integrate the SecureLink API into their client applications written in other popular languages such as Java. Using SecureLink Web Services, merchants can develop their own customized stores and related services on heterogeneous platforms including Windows, Linux, HP-UX, Sun-Solaris and other UNIX operating systems.

Introduction to Web Services

Web services have developed with the evolution of the Internet. The intent of a *web service* is to use the Internet as a transactional tool rather than simply a visual tool. These application-to-application interactions are driven by, and built on, existing standards such as:

- Extensible Markup Language (XML)
- Hypertext Transfer Protocol (HTTP)
- Simple Object Access Protocol (SOAP)
- Universal Description, Discovery, and Integration (UDDI)
- Web Services Description Language (WSDL)

Clients of web services do not need to know how the services themselves are implemented – this adds significantly to the flexibility and integration capabilities of web services.

Web Services Functionality in Transact 6.5

In Transact 6.5, the SecureLink functionality is offered as a set of Web Services including: Buyer Profile Management, Order Entry Management, SecureLink Utilities, Multi-Item, and commerce utilities among others. These functions can be called from any application external to Transact and SecureLink and can be integrated with external applications and products through the Internet. More detail on the SecureLink Web Services follows.

Buyer Profile Web Service

The Buyer Profile Web Service enables merchants to manage buyer profiles from their remote custom applications. Using the Buyer Profile Web Service, merchants can create a new registered buyer entry; retrieve a registered buyer's information; and modify a registered buyer's information such as address, phone number, email address and any custom information.

Order Entry Web Service

The Order Entry Web Service allows a buyer to make purchases without going through the Transact order form GUI. This is useful for Merchants interested in having a separate (and generally custom) order entry interface.

The Order Entry Web Service processes an order by gathering information about a purchase, including that information into a special order object, and sending that order object to Transact for validation and processing.

SecureLink Web Service

The SecureLink Web Service allows merchants to integrate the Transact store generation features in their remote custom applications - for example, to generate digital offers from product catalogs in a Content Server. Using the SecureLink Web Service, merchant can:

- Generate Digital Offers for tangible/digital goods for the store
- Generate Digital Offers for digital subscriptions and discount coupons for the store
- Generate Digital Offers for micro-transaction credit and micro-transaction goods for the store

Multi-Item Web Service

The Multi-item web service allows merchants to implement remote custom shopping carts (using any third-party applications) and submit the shopping cart items to Transact for processing. Using the Multi-item web service, merchants are able to submit items in a remote client's custom shopping cart to Transact's shopping cart using a single web service call to Transact.

Functionality Available via Web Services

This section discusses the limitations of the legacy SecureLink API and the advantage of using SecureLink Web Services.

SecureLink API Limitations

Limitation of the SecureLink API fall into two categories as highlighted below: interoperability and deployability.

Interoperability

The SecureLink API is written in C/C++; as such, applications which would like to use the API must be written in C/C++ as well. Developers using CORBA or RMI or JNI can implement varying levels of interoperability, but none of these languages enable full application interoperability. Because the SecureLink API is written in C/C++, applications developed in different languages such as Visual Basic, C#, and Java cannot access the API directly.

Deployability

The current architecture typically works inside of a private network(s) as shown in Figure 1. As the SecureLink libraries reside on the machine where SecureLink is installed, the applications making use of the SecureLink API must be deployed on the same machine. Additionally, the C/C++ client applications using the SecureLink API must run on the Sun Solaris platform.

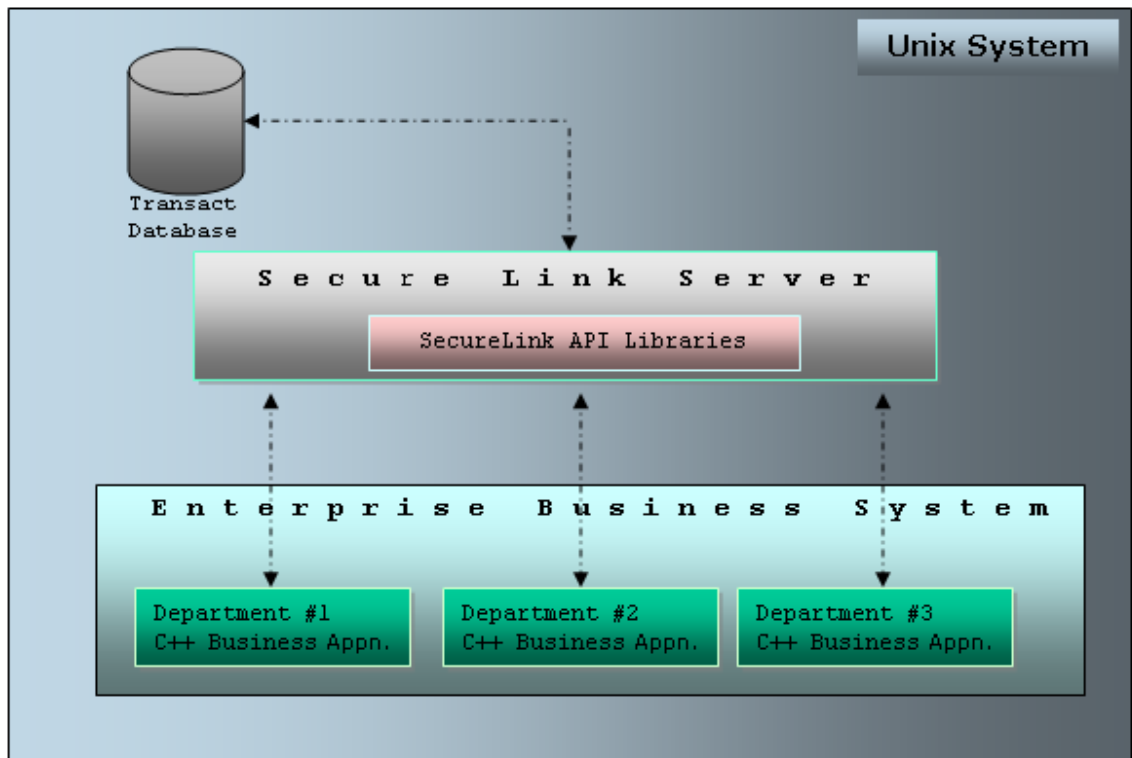


Figure 1. Existing Transact Framework

SecureLink API Enabled with Web Services Advantages

In Transact 6.5, the supported client languages are C++ and Java. However, it is quite possible to support additional languages that adhere to the Web Services protocols with minimal effort because Web Services are not based on languages or on a programming data model. As a result, custom applications written in C++ or Java programming languages can talk to the SecureLink Order Entry Web Services. And, the client applications accessing the Web Service can run on any platform or operating system. Applications running on Windows, Unix (various flavors) and Linux can make use of the SecureLink API Web Services. Key advantages of providing Web Services for the SecureLink/OrderEntry C/C++ API are highlighted in the following sections. Figure 2 illustrates the integration of Web Services into the Transact framework.

Interoperability

Interoperability is the most significant benefit of Web Services. Web Services typically work outside of private networks, offering developers a non-proprietary route to their solutions. Services developed are therefore likely to have a longer life-span, offering a better return on investment of the developed service.

Web Services also let developers use their preferred programming languages. Java, C++, VBScript, JavaScript, and Perl all have one or more Web Service APIs maintained by an array of vendors. In addition, due to the use of standards-based communications methods, Web Services are virtually platform-independent.

Usability

Web Services allow the business logic of SecureLink API to be exposed over the web. This gives third-party custom applications the freedom to choose the Web Services that they need with which to integrate, rather than customizing the third-party applications. Based on the requirements, programmers just need to include additional application-specific business logic on the client-side. This allows developers to use the Web Services in the languages and tools with which they are familiar.

For example, a programmer building a store (or a department within a store) can use the SecureLink Web Services in their remote applications to create digital offers. Similarly, the Finance/MIS department may be interested in the number of registered buyers and their corresponding segment of the state. As such, they can integrate the Buyer Profile Web Services in their custom finance applications to manage or use Transact's buyer profiles database.

Reusability

Web Services do not provide a component based model of application development, but rather a near zero-coding deployment of such services. This makes it easy to reuse Web Service components in other services as well as to deploy legacy code as a Web Service.

Deployability

Client applications using the Web Services are deployed over standard Internet technologies making it possible to deploy the applications behind the firewall while still connecting to servers running anywhere on the Internet. And, due to the use of coding standards, underlying security (such as SSL) is already built-in.

Ubiquity

The SOAP protocol and its industry-wide support promises to make services available to users anywhere, e.g. in cellphones, pocket PCs, PDAs, embedded systems, and desktop applications.

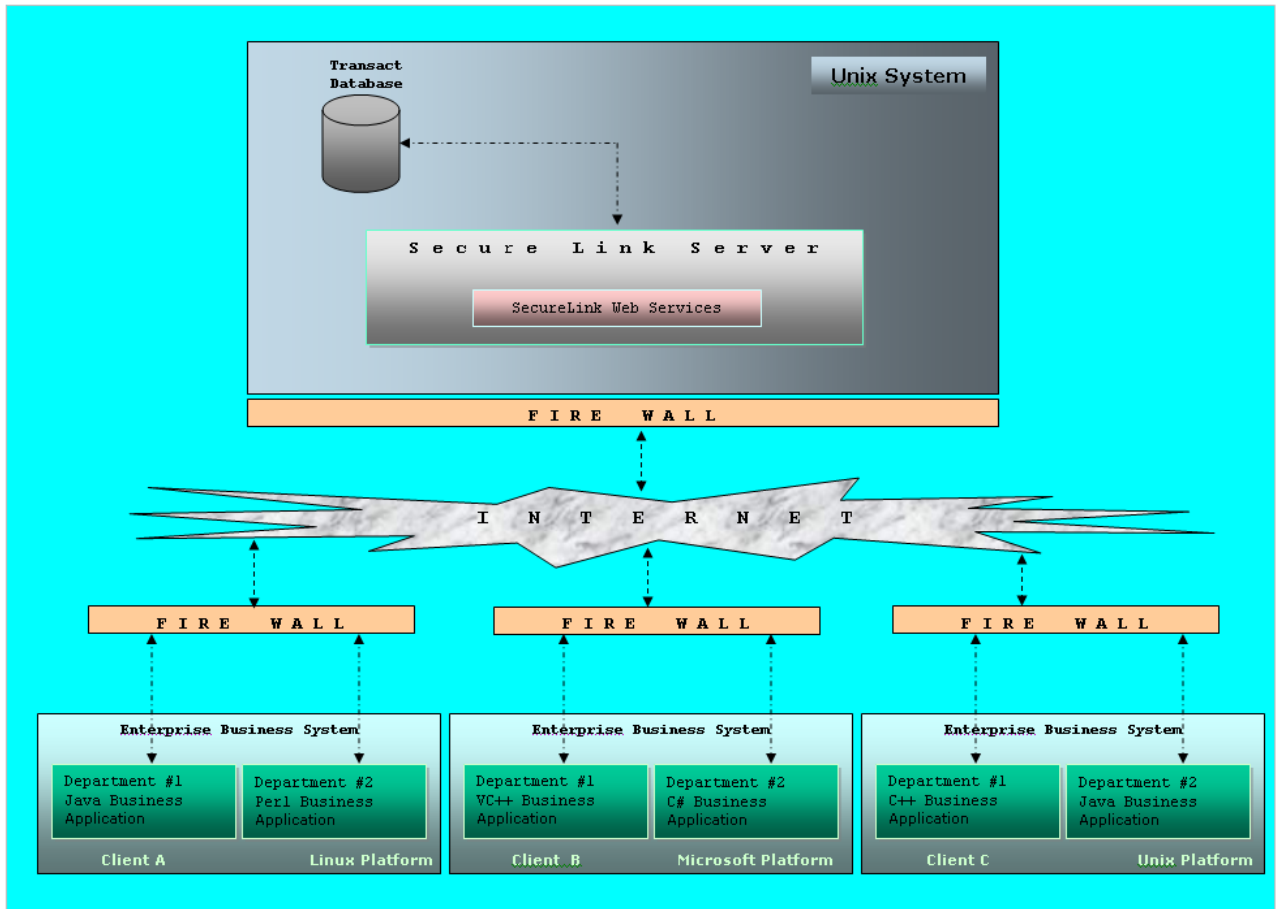


Figure 2. Transact Framework with Web Services

Accessing SecureLink Web Services

This section includes functionality requirements and instructions for accessing the installation package.

Requirements

The Order Entry Web Services Kit (OEWSK) Ver 6.5 has the following software and system requirements.

Software Requirements

The following software is required:

1. OEWSK 6.5 (if already using SecureLink 6.0)
2. SecureLink 6.5 (which includes the OEWSK 6.5)
3. SecureLink Order Entry SDK
4. gSOAP and Apache Axis SOAP Toolkits
5. C++ and Java compilers

System Requirements

The OEWSK 6.5 requires the following system environment:

1. Solaris 9 Operating System
2. 10 MB disk space

Download and Installation

Download

The Order Entry Web Services Kit 6.5 and the SecureLink 6.5 is available for download from the Soverain Technical Support website "Downloads" section. The password-protected Technical Support site is accessible via <http://www.soverain.com>.

Installation

Installation instructions are included in the *OEWSK 6.5 Installation and Configuration Guide* available at the Technical Support site "Documentation" section.