Galileo Web Services

Product Overview

30 January 2009

Company Confidential

Table of Contents

Overview

This overview is designed to provide a high-level introduction to Galileo Web Services (GWS), including a brief introduction to Web Services technologies, as well as a description of how Web Services can be used to interface with the Apollo® and Galileo® Computer Reservations Systems (CRS).

GWS is designed to support data transfer between client travel applications and the Apollo and Galileo CRSs. GWS is a collection of Web Services that provide client applications with access to key functionality on the CRS.

What Benefits Do Galileo Web Services Offer?

By standardizing communications between the Apollo and Galileo CRSs, Galileo Web Services offer a number of potential benefits for designing, implementing, and marketing travel-related applications. By combining current Galileo products with new technology standards, GWS offers a more streamlined and robust method for designing travel applications.

The benefits of GWS include:

- Platform independence.
- Encapsulated business logic that streamlines transactions with the CRS.
- More connectivity options.
- CRS communications components hosted by Galileo.
- Increased security for transactions using SSL protocols.
- Use of gzip compression to reduce file sizes and increase performance.

Platform independence. Because GWS adheres to cross-platform Web Services standards, GWS does not restrict the environment for developing or deploying client applications. GWS client applications can be deployed in any environment that supports HTTP (Hypertext Transfer Protocol), including Windows® and UNIX, without requiring bridges or specialized interfaces. In addition, a wide variety of languages and development developer toolkits can be used to design and deploy client applications for GWS.

Encapsulated business logic that streamlines transactions with the CRS. As GWS evolves, more and more Web Services will encapsulate the business logic needed to communicate with the CRS into a more streamlined, intuitive framework. As a result, developers focus on writing their applications rather than communicating with the CRS. The streamlined usability of GWS could potentially reduce development time by 30 to 70%, depending on the type of application and development environment.

CRS communication components hosted by Galileo. Previously, developers needed to include complicated code in their applications for identifying and routing transactions with the CRS. Galileo's XML Select API product streamlined communications with the CRS for customer-hosted applications. GWS hosts any required components at Galileo, which internally supports identifying, routing, and transforming transactions with the CRS.

Increased security for transactions using SSL protocols. GWS now supports SSL (Secure Sockets Layer) protocol to manage sessions. SSL is used to create an encrypted connection, which adds another layer of security on to transactions of confidential data.

Use of gzip compression to reduce file sizes and increase performance. The gzip (GNU zip) compression utility reduces the file sizes of responses sent from GWS, which speeds response times and creates a more efficient experience for the traveler.

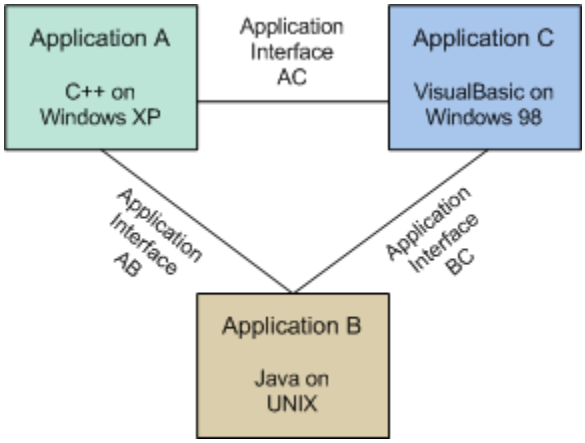
What Are Web Services?

Web Services are essentially a series of standards that allow disparate systems to communicate with each other using a common framework. Even though these systems communicate via a common Web Service standard, their internal structure is not dictated by Web Services. Services and client applications do not require a specific kind of infrastructure; their language, platform, and other structures need only be compatible with Web Services.

Web Services represent an *evolution* in communication between applications, rather than a *revolution*. Most of the technologies behind Web Services are not new. However, these existing technologies has been combined and refined to create a common system of communication. Most importantly, a set of common standards have been assigned to these technologies so that systems using Web Services have a single, explicitly defined method for communicating with each other.

Application Interfaces Before Web Services

Web Services create greater flexibility because they remove the need for applications to programmatically understand each other. The following diagram shows how three applications, written in different languages on different platforms, would need specialized interfaces, or bridges, to communicate with each other:



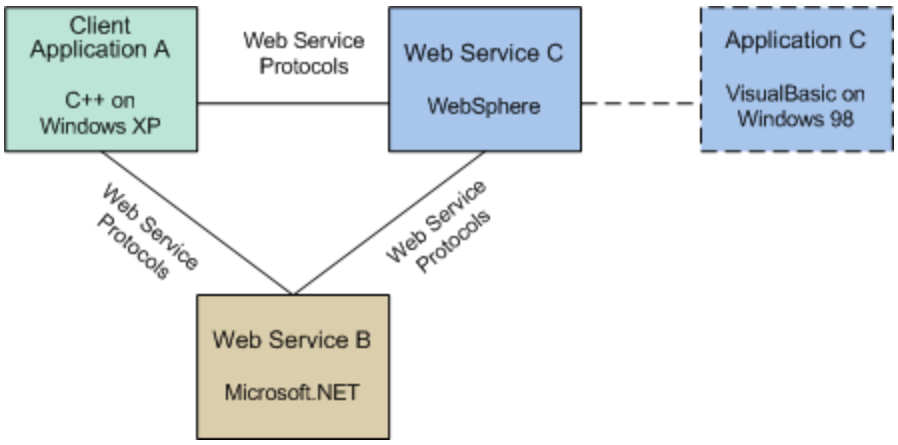
Without a common system of communication, each application requires its own specialized interface with another application. For example, the interface that Application A requires to communicate with Application B is completely different from the interface that Application A must use for Application C.

In some cases, pre-existing interfaces or bridges can be purchased to translate between disparate languages and platforms. In other cases, however, a proprietary applications interface must be specifically designed and deployed to communicate with the other application. Such specialized design can often consume huge amounts of financial and employee resources, as well as add another layer of complication to product design, implementation, and maintenance.

Application Interfaces with Web Services

The introduction of Web Services allows all of these applications to communicate via common Web Services standards (such as XML, HTTP, and SOAP). The specific protocols for data transfer are less important than the fact that all of the services and applications adhere to the same standard.

See *Appendix A: Know Your Web Services Jargon* (page 20) for more details about the specific standards, protocols, and formats used in GWS and other Web Services.



In the new scenario, Application B and C have now become Web Services that both use standard Web Service protocols to transfer information. The fact that Application B uses Microsoft.NET on a Microsoft platform as the infrastructure for its Web Service, while Application C uses WebSphere on a UNIX platform for its Web Service, is irrelevant. Both Web Services still communicate outside of their infrastructure using the same protocols.

In this scenario, Application A has *not* become a Web Service; it remains a standard application. However, the code in Application A has been modified to make the application a Web Services client that uses the same SOAP, XML, and HTTP standards to communicate with the Web Services servers.

Web Service B and Web Service C respectively represent two different methods for implementing Web Service functionality:

- In Web Service B, all of the data that the client application or service requires is stored within the Web Service itself.
- Web Service C communicates with Application C, which continues to supply the required functionality for the application. The dotted line in the chart shows the relationship of Web Service C as an intermediary between Application C and the other services and applications.

Galileo Web Services use both of these scenarios when communicating with client travel applications. Some of the GWS services communicate with the CRS via Galileo's XML Select product (similar to Web Service C). Other GWS services encapsulate all of the data that a client applications needs within the Web Service itself (like Web Service B).

What Are Galileo Web Services?

Galileo Web Services (GWS) use Web Services technologies to support data transfer between client travel applications and the Apollo® and Galileo® Computer Reservations Systems (CRS). GWS is a collection of Web Services that provide access to key functionality on the CRS.

The Web Services available through GWS provide access to a wide variety of travel-related functionality. Currently, GWS is divided into several different types of services that transact with the CRS using different models for organizing data:

- **XML Select Web Service**
  The XML Select Web Service mirrors the host functionality of the CRSs by containing individual transactions with the CRS that can be strung together to create a business process. These transactions are the same as those transactions found on Galileo's XML Select API product. For example, the XML Select Web Service contains individual transactions for air availability, air faring, selling a segment, reserving an itinerary, selecting a seat, and end transacting a file.

  The XML Select Web Service the functionality of the existing XML Select product as a Web Service. Business functions include transactions for:

  - Air, car, hotel, and cruise shopping, availability, and booking.
  - Faring and fare manipulation.
  - Using sessions.
  - Manipulating PNRs/Booking Files, itineraries, and segments.
  - PRO-File/Client File manipulation.
- **Encapsulated Web Services**
  Several Web Services in GWS provide a higher level of abstraction by encapsulating a specific business process into a single transaction with GWS. For example, Itinerary eBL can combine over six individual transactions into a single call to GWS.
- **Cruise Web Services**
  The Cruise Web Services provide the ability to shop, book, modify, and cancel cruises via a single API that normalizes data across multiple vendors. Additionally, the service creates a PNR/BF on the CRS to allow all existing reporting functionality.

XML Select Web Service

Based on Galileo by Travelport's XML Select product, this Web Service provides the functionality of the existing XML Select product as a Web Service. The XML Select Web Service also supports terminal transactions, which can be used for functions not currently support through structured data (XML Select transactions).

The XML Select Web Service is useful for client applications that require specialized business processes or detailed control of specific transactions within a business process. The XML Select Web Service includes transactions for a variety of travel-related business functions. Archived transactions that are fully functional also remain available for backwards compatibility.

Please note that the availability or detailed functionality may vary by CRS.

Booking Records (PNR/BF) and Itineraries

GWS provides the following functionality for manipulating a Passenger Name Record (PNR) or Booking File (BF).

- Creating and maintaining Primary PNR/BF data, such as names, phones, addresses, and profiles.
- Creating and maintaining Secondary PNR/BF data, such as such as form of payment, customer and corporate identifiers, and various types of remarks including general remarks, SSRs, OSI, and ticketing remarks.
- Modifying an existing PNR/BF.
- Canceling individual travel segments in a PNR/BF or canceling an entire PNR/BF.
- Retrieving a PNR/BF by record locator, carrier code and flight, or traveler name.
- Retrieving a PNR/BF that is currently active.
- Displaying a PNR/BF.
- Dividing a single booking into two separate bookings.
- Ignoring (ignore transacting) changes to a PNR/BF.
- Ignoring (Ignore transacting) and re-retrieving active PNR/BF.
- Finishing (end transacting) an active PNR/BF, which allows PNR/BFs to be created or modified, and then committed to the CRS in the same call.

Segments

Functionality for individual travel segments in a PNR/BF includes:

- Inserting segments.
- Canceling segments.
- Performing passive segment sells.
- Reordering itinerary requests.
- Modifying or deleting hotel segment data, including reservation dates, room types, and optional data.
- Modifying or deleting car segment data, including reservation dates, car types, and optional data.
- Selling air, car, and hotel segments.
- Performing direct air segment sells.
- Modifying air segments through date, class, flight number, timings, and stopover indicators.
- Modifying air segments through rebooking.
- Selling seats for air segments.

Sessions/Security

GWS provides seamless internal support for transactions that require a sessioned environment. The following functionality is handled by either the identifiers in the Host Access Profile or XML Select Web Services methods that support sessions:

- Signing on to and signing off of a session.
- Changing the password required to sign on to a session.
- Allowing a Service Bureau or other approved entity to emulate a client travel agency or organization.

Sessions/Security (continued)

In addition the XML Select Web Service contains transactions to support the following session functionality:

- Identifying work area information and changing work areas.
- Changing duty codes.

Air

Air functionality in the XML Select Web Service includes:

- General Air Availability, which supplies scheduled flights between a given city pair on a given day, and indicates whether seats are available on those flights. Various qualifiers are available to narrow the search criteria.
- Tariff and Rules Display, which supplies general fares based on specific dates or range of dates. Rules associated with a specific tariff fare can then be displayed.
- Flight Information (FLIFO), which provides real-time flight information for participating air vendors, including actual, as well as scheduled departure and arrival times.
- In-Flight Service Information, which supplies scheduled information, including: scheduled departure and arrival times, type of aircraft, meal service, and elapsed flying time.
- Seat Maps, which supply available seats for specified flight segments.

Fares

Fare functionality in the XML Select Web Service includes:

- Standard Fare Quote based on the schedule and class of service booked.
- Manual Fare Quotes on the Galileo CRS.
- Flight-Specific Fare Quote for a specified single flight or specified multiple flights.
- Class-Specific Fare Quote, which obtains a fare quote for a specified flight and class of service, regardless of availability.
- Best Buy Fare Quote (also known as Dollar Saver Fare Quote), which finds the best price for the booked itinerary, taking into account all classes that are currently available on the flights that make up the itinerary and includes the rebooking details in the response.
- Best Buy Quote (also known as Fare Quote Finder), which prices a booked itinerary at the lowest available fare. Options return display as a specific itinerary, with flight segments that most closely match the original flight times.
- (Best Buy Quote) This transaction uses the origin/destination and date information from a booked itinerary to find the best price options for a proposed journey. An option will not display in the response unless it is cheaper than the original booked fare.
- Super Best Buy (also known as Shopper Fare Quote or Low Fare Shopping), which supplies availability and fare options for an unbooked itinerary.
- Fare Quote with Rules Display, which supplies the rules and policies associated with an airfare quote.
- Calendar Shopping, which combines several functions into one feature. Allows travelers to select a tariff fare, and then provides travel dates and valid flight availabilities for the selected tariff.
- Validated tariff fares, which returns all applicable fares for a specified date or dates. Currently available on Apollo for US and Canadian markets. Note that tariff fares do not indicate availability for the requested itinerary.

Fares (continued)

- Non-validated tariff fares, which provide all published fares between two cities without validating applicability.
- Fare quote rules, which supply the requirements or penalties associated with a specific fare quote.
- Ticketing Point Mileage, which returns the distance between two or more locations for a customer's travel plans. This function is used for mileage based fares to determine if the customer traveling is within the maximum point mileage allowed for those city pairs.

Queues

Queue functionality in the XML Select Web Service includes:

- Signing in to and out of a Queue.
- Managing queuing functions, such as removing a PNR/BF from a Queue.
- Supplying Queue statistics, such as Queue counts and Queue lists.

Stored Fares (ATFQ/FF)

Stored fare functionality in the XML Select Web Service includes:

- Storing an airfare quote for later ticketing.
- Verifying the price of a booked airfare quote (Apollo only).
- Re-pricing a stored and booked airfare quote, which ensures that the stored price reflects any changes made to an itinerary.
- Re-pricing a fare quote with updated ticketing modifiers.
- Canceling a stored airfare quote.
- Retaining a previously stored fare.

Car

Car transactions include the following functionality in the XML Select Web Service:

- Car Description, which provides information about a specific rental car and associated vendor information, such as policies and hours of operation.
- Car Index, which supplies a list of all car vendors and locations within the specified city. Various qualifiers are available to narrow the search criteria.
- Standard Car Availability, which supplies a list of available rental cars and their rates. Various qualifiers are available to narrow the search criteria.
- Car Matrix, which provides a basic availability call that focuses on returning a maximum amount of car rates in the fastest time possible. This transaction was created for web users attempting to fill out a grid of car rates where the axes represent car vendors by car types. This function provides the data necessary to complete this grid in one call.
- Car Rules, which displays car vendor rules and requirements that must be met in order to qualify for certain rates. General rules information can be returned, or more specific rules for a sold car segment.
- Reference Points, which provide a list of locations in a city or metro area, such as an area attraction, business center, hospital, or government office. Reference points can be used to facilitate the location of a rental car vendor.

Rail

Rail transactions include the following functionality in the XML Select Web Service.

- Making reservations and printing tickets on the SNCF platform.

Hotel

Hotel transactions include the following functionality in the XML Select Web Service.

- Standard Hotel Availability Supplies a list of hotel properties with available rooms, as well as a partial list of available rates for each property. Optional qualifiers, such as reference point, rate code, and room type, can be added to modify the response.
- Complete Hotel Availability Supplies a complete list of available rates for a specific property.
- Hotel Description Supplies additional information about a selected hotel property, such as room types and descriptions, directions, amenities, and hotel policies.
- Hotel Index Returns a list of properties for a specified availability, regardless of availability.
- Hotel Rules Displays hotel vendor rules and requirements that must be met in order to qualify for certain rates.
- Reference Points, which provide a list of locations in a city or metro area, such as an area attraction, business center, hospital, or government office. Reference points can be used to facilitate the location of a hotel property.

Cruise

Cruise transactions include the following functionality in the XML Select Web Service.

- Vendor search, which locates cruise vendors for specified regions and dates.
- Available for sailings, rate codes, categories, and cabins.
- Holding (reserving) and releasing cabins from the cruise vendor's inventory.
- Package options to finalize a selected cabin reservation, and request features and options for the selected cruise packages.
- Modified transportation options to accommodate selected package options, such as pre-cruise tours.
- Preliminary pricing estimates, per person and total, which do not require specific passenger and cabin details.
- Pricing for a selected cruise package and options.
- Payment rules permitted by a specific cruise vendor.
- Payment terms for a reserved package, and information about incremental payments.
- Selling cruise segments.
- Canceling a booked cruise segment before payment has been finalized.

Traveler Profiles and Corporate Data

- Manipulating Client Files™/PRO-files™, which store traveler and travel provider information that can be added as a shortcut to a PNR/BF.
- Invoking Custom Check rules, which check for the presence or absence of information in the PNR/BF to verify the booking against a corporate policy.
- Invoking TravelScreen™, which stores personal preferences that can be applied to availability and fare displays for air, cars, and hotels. TravelScreen also stores personal travel information that can be transferred into the PNR/BF.

Ticketing/Document Production

- Retrieving the itinerary and the fare construction from an original electronic ticket.
- Validating that booked itineraries are eligible for electronic ticketing.
- Obtaining information on valid tickets for the passenger for paper and/or electronic tickets.
- Enabling printer devices to be linked and displayed, and enabling maintenance of the TINS (Ticket Number Stock) table.
- Allowing electronic tickets to be revalidated against an updated itinerary.
- Voiding or unvoiding tickets, or marking ticket stock as spoiled.
- Retrieving and confirming the refund or exchange price for a ticketed booking.

Additional Functionality

- Displaying the agency code, contact information, and additional information for a requested agency.
- Verifying credit card numbers. Verification of addresses is also available by contract.
- Displaying the local date and time for a requested city or airport.
- Provides the currency conversion rates used by Apollo and Galileo for rates and fares.
- Managing Miscellaneous Charge Orders (MCO), which includes building MCO information, obtaining information for MCO displays, and issuing MCO documents.
- Issuing Ticket Agent Service Fees (TASF).

Encapsulated Web Services

The Encapsulated Web Services in GWS combine multiple functions for a specific business process into a single transaction with GWS. For example, Itinerary eBL can combine five or more individual transactions into a single call to GWS, depending on the number of travel segments in an itinerary. The applicability of Encapsulated Web Services may vary depending on the type of client application and associated business processes.

Encapsulated Web Service	Description
Flight Information eBL	Provides the real-time status of a flight, such as actual, as well as scheduled, departure and arrival times, departure and arrival gates, and cancellation or delay information.
Itinerary eBL	Retrieves detailed itinerary information in one display.
Travel Codes Translator eBL	Provides translation for Galileo's *local data*, which provides mappings between travel-industry names and codes recognized by the CRSs.
Image Viewer eBL	Provides images, rich media, and descriptions to over 65,000 hotel properties, with additional properties being added regularly.
PNRQueueAndRetrieve eBL	Retrieves and displays a booking in one call by encapsulating queue and re-display functionality into one transaction.
Reservation Builder eBL	Combines booking functions into one call to the CRS. **Note**: This Web Service is retired from new implementations, and is no longer functional in the EMEA (Europe, Middle East, and Africa) markets. Customers that currently use this Web Service will continue to be supported.
Trip Planner eBL	Encapsulates shopping, evaluating, and ranking air, car, and hotel selections for a specific trip in one call. Returns complete trips and alternatives, with prices and rankings against preferences and profiles. **Note**: This Web Service is retired from new implementations, and is no longer functional in the EMEA (Europe, Middle East, and Africa) markets. Customers that currently use this Web Service will continue to be supported.

Flight Information eBL

Flight Information eBL provides the current status of a flight such as actual and scheduled departure and arrival times; departure and arrival gates; and cancellation or delay information.

Flight Information (often referred to as FLIFO) is typically displayed for flights that are either already in progress or scheduled for the current day. Because FLIFO is updated throughout the course of the flight, it is an accurate source of data. Although this information can be obtained prior to a flight's departure, it is only truly relevant on the day of travel. Until the departure date, the data includes scheduled information (in-flight services) only, which is subject to change.

Not all carriers support FLIFO. In addition, FLIFO is delivered by the carriers themselves. Therefore, there may be differences between the amount or type of data provided by each carrier, as well as the frequency of updates.

Itinerary eBL

Itinerary eBL provides allows users to view updated trip information in one display. Itinerary eBL provides a comprehensive travel itinerary that contains the latest air, car, and hotel booking information, as well as other non-host reservation data, such as tours and cruises. Itinerary eBL can also provide fare information about the itinerary.

The functionality offered by this service is based on Galileo's ViewTrip™ product, which is accessed at viewtrip.com. Enhanced functionality for this product now provides access to the Apollo, Galileo, and Worldspan systems.

After entering record data, such as a reservation number and last name, users can request basic information a full itinerary containing the basic information plus any non-host (W3) segments, which may be decoded.

Travel Codes Translator eBL

Travel Codes Translator eBL provides translation for Galileo's *reference data* (also known as *local data*), which is a collection of data that provides mappings between travel-industry names and codes recognized by the Apollo and Galileo CRSs. These mappings include information for data such as air, car, and hotel vendors, cities and airports, classes of service, and meal types.

Because the CRSs process only encoded forms of many types of data, local data codes are necessary to communicate with the CRS. Conversely, GUI end-user applications require more decoded data to display travel information in human-readable text. For example, the *BGO* city code decodes to *Bergen* and the *Kahului Airport* encodes to *OGG*. In addition, local data also serves to maintain relationships between types of data, such as cities, states, and countries.

While some types of reference data, such as city and airport codes, are standardized, other types of local data are obtained directly from travel vendors. Because these codes change on a regular basis, the Encode/Decode Service allows for more efficient translation of reference data.

Image Viewer eBL

Image Viewer eBL provides images, descriptions, and rich media, such as videos and virtual tours, to over 65,000 hotel properties, with additional properties being added regularly from multiple suppliers. With Image Viewer eBL, hotel description and image requests can be made either separately or in a single call, and up to 20 properties can be included in a single request. Responses also combine image and description data, based on the request. All request and response data is in XML.

PNRQueueAndRetrieve eBL

PNRQueueAndRetrieve eBL queues, retrieves, and displays a booking in one transaction. The PNRQueueAndRetrieve Web Service encapsulates two functions: queue and re-display. When the user submits a request, GWS performs an end transact (ET) which saves the PNR/BF data, then routes for the PNR/BF to the designated queue.

After the PNR/BF is successfully queued, GWS re-retrieves and displays the PNR. The data in the PNR/BF can then be reviewed or modified.

Reservation Builder eBL

Reservation Builder eBL reserves a completed itinerary (PNR/Booking File) on the CRS. This service comprises a number of transactions that are available individually through the XML Select Web Service. However, Reservation Builder eBL completes the booking in a single web service call to the CRS.

In its most basic form, the client application supplies the required PNR/Booking file data for the trip, and a record locator is returned.

Note: Please note that Reservation Builder eBL is retired from new implementations, and is no longer functional in the EMEA (Europe, Middle East, and Africa) markets. Customers that currently use this Web Service will continue to be supported.

The PNRBFManagement transaction the XML Select Web Service can be used to obtain the same functionality.

Trip Planner eBL

The Trip Planner eBL encapsulates the business logic for shopping, evaluating, and ranking air, car, and hotel selections for a specific trip. Individual transactions for shopping, availability, and faring are contained in the XML Select Web Service. However, this service allows users, particularly business travelers, to save time by providing complete trip plans that more closely meet the traveler's requirements.

With Trip Planner eBL, users can enter travel preferences, such as vendors, departure times, rates, car types, and location. For each preference, they can optionally weight that preference to make it more or less favorable for a trip.

One call to the Trip Planner returns complete trips and alternatives with prices. The information that Trip Planner eBL returns can then be converted into a Reservation Builder eBL request to book the trip.

Note: Please note that Trip Planner eBL is retired from new implementations, and is no longer functional in the EMEA (Europe, Middle East, and Africa) markets. Customers that currently use this Web Service will continue to be supported.

The PNRBFManagement transaction the XML Select Web Service can be used to obtain the same functionality.

Cruise Web Services

The Cruise Web Services provide the ability to shop, book, modify, and cancel cruises via a single API that normalizes data across multiple vendors. Additionally, the service creates a booking file (PNR/BF) on the Apollo® or Galileo® CRS, depending on the travel provider, to allow all existing reporting functionality. The Cruise Web Services allow users to implement one product for their entire cruise booking needs.

While cruise functionality can also be obtained through individual transactions within the XML Select Web Service, the Cruise Web Services:

- Provide a common XML user interface that supports all available APIs and creates common codes and messages.
- Significantly increase content and functionality including regional pricing, promotions, group functionality, ship content, and detailed explanations of all features and options, including shore excursions, pre- and post-night hotels, and shipboard amenities.
- Support modification and cancellation requests for previous versions and the new common version of XML requests.
- Enhance performance with updated connectivity and more robust technology.
- Provide the most up-to-date information for flights, itinerary changes, and payment updates because booking details are retrieved directly from the vendor's database.
- Provide Web Services to Galileo Web Service (GWS) and XML Select customers that cover all cruise-related functions and all connection types.
- Support international rates.

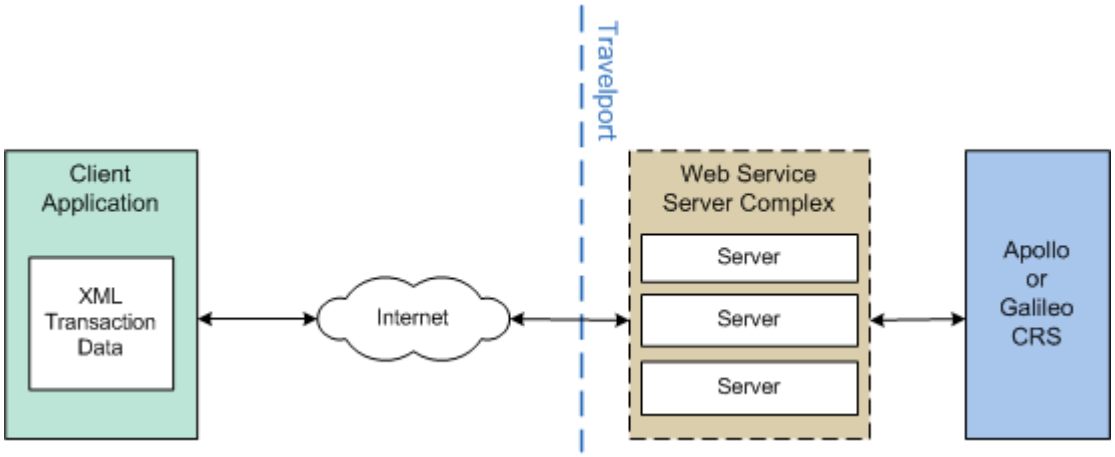
How Do Galileo Web Services Work?

Galileo Web Services combine Galileo's XML Select API with Web Services technologies. The XML Select product performs two key functions:

- Managing communications between client applications and the Apollo or Galileo CRS.
- Converting between XML data from client applications and Galileo's proprietary structured data formats, which are used by the CRS.

However, with GWS, the XML Select components are hosted by Galileo on secured servers. By hosting XML Select at Galileo, customers do not need to install, configure, and maintain XML Select components on-site.

The following graphic shows the basic flow of information in a transaction that uses GWS to communicate with the CRS. For more information about the details of data transfer in a Web Services environment, please refer to *Appendix A: Know Your Web Services Jargon* (page 20).



Sending Requests

The following description explains how a message is sent from a client application to the CRS.

Client Application

The client application creates a message with the request data in an XML (Extensible Markup Language) format.

Client applications can be written in any language or platform that supports Web Services technologies or protocols. See *Requirements and Recommendations* (page 16) for further details about supported environments.

Regardless of the programming language or platform, the application code must use XML for the actual request data that is sent to the CRS. This XML transaction is wrapped in a *SOAP envelope*. SOAP (Simple Object Access Protocol) is XML data that directs the message to be delivered to the correct Web Service, and allows the message to be appropriately processed by the Web Service.

Internet Connection

The message is sent from the client application to the GWS servers, which are hosted by Galileo by Travelport.

Messages in Web Services are sent via HTTPS (Hypertext Transfer Protocol Secure), which is a common method of transferring HTML and XML data. The session is managed by a SSL (Secure Sockets Layer) protocol.

The amount of network bandwidth is the most important factor in planning for adequate capacity. Regardless of the type of network connection, the size of the connection has the most direct affect on the speed and capacity of the client application.

Web Services Server Complex

The Web Services servers receive and process the request message.

The SOAP message that was attached to the request message channels the request to appropriate Web Service, and essentially instructs the Web Service in how to process the message. A Host Access Profile (HAP) is included with the XML application data to identify the sender of the transaction to the Web Service.

Some Web Services messages end at the Web Services servers, while others are forwarded to the CRS to be converted and processed. How messages are handled depends on the type of service. For example, Travel Codes Translator eBL, which maps between travel-industry codes to names, has all of its functionality located within the Web Service itself. The decode request and response is handled completely within the service. However, an air availability request requires additional processing by the XML Select API and the CRS. In this case, the Web Services act as a sort of intermediary for processing the air availability request and response data.

The XML Select API takes the XML data that has been received and processed through the Web Service and essentially translates the data from XML into Galileo's proprietary structured data format. The structured data is then processed by the Galileo or Apollo CRS.

The Web Services servers reside on *server cluster*. A server cluster is a series of servers that are networked to each other to form a unified interface for data. Clusters provide excellent fault tolerance; if one server in a cluster fails, other servers accommodate the failed server's load. Additional servers can also be easily added to increase capacity of a cluster. This power and redundancy allows Galileo to manage large volumes of data and expand the fault tolerance of a system.

Retrieving Responses

The response message follows the same path as the request message, in reverse. The proprietary structured data response is sent from the CRS to the Web Services Server Complex, which converts the response into an XML format.

This response is then wrapped in a SOAP envelope and forwards the message back to the client application. Again, the response is sent via HTTP through either an Internet connection. After the client application receives the message, it takes XML response data and displays that data within the client application.

How Are Galileo Web Services Different from XML Select?

Galileo Web Services (GWS) use XML Select components to translate data between client applications that use XML and the proprietary host data used by the CRS. With GWS, the XML Select components are hosted by Galileo by Travelport on secured servers. With the XML Select API product, customers host the XML Select components and maintain connections to the CRS.

XML Select, as a separate product, is available for customers who choose to manage the connections to the CRS themselves. For customers who choose to migrate to GWS, the XML Select Web Service contains the current XML Select transactions in a Web Services format. Changing an existing XML Select client application to Web Services simply involves changing the code to send data to the CRS via GWS, rather than directly to the CRS via XML Select components. Because the interface with Web Services is simpler than the interface with the XML Select components, the modified code is typically more streamlined than the original code for XML Select.

GWS also eliminates the need to manage GTIDs (Global Terminal Identifiers) that are associated with individual sessions or transactions. Because GTIDs are maintained on the Galileo servers, customers no longer need to obtain or manage specific numbers or ranges of GTIDs.

Requirements and Recommendations

Planning for GWS includes familiarity with not only the required technologies and skill sets, but also preliminary planning to design a client application that can appropriately support Web Services for the expected amount of traffic within the scope of available resources.

GWS Standards and Protocols

GWS conforms to the following industry standards and protocols.

Message Envelope	SOAP 1.1 and SOAP 1.2	http://www.w3.org/TR/SOAP/
Message Transfer	HTTP 1.0 and HTTP 1.1	http://www.w3.org/Protocols/
Encryption	SSL 3.0	http://wp.netscape.com/eng/ssl3/
Web Service Description Language	WSDL 1.1 and WSDL 2.0	http://www.w3.org/TR/wsdl
Data Transfer	XML 1.0	http://www.w3.org/XML

Tools and Environments

Any development tools or environments that are compatible with the above standards and protocols can be used. These tools and environments include, but are not limited to:

- Microsoft.NET
- IBM WebSphere® Studio Application Developer
- Apache Axis (replacing Apache SOAP)
- The Mind Electric GLUE™
- SOAP::Lite for PERL

Developer Skills

Developers should have familiarity with XML, as well as the standards and protocols listed in the in this section. Particularly when using the XML Select Web Service, knowledge of the CRS business model is also helpful.

Connectivity

GWS is supported through an Internet connection, which must be secured via an SSL (Secure Sockets Layer) protocol, as well as the security functions provided for GWS. Because Web Services formats and standards are not limited to use on the Internet, the same protocols apply to the message, regardless of the type of network connection.

Security

The GWS servers are secured, and access is granted only to the specific Web Services for which the client is licensed. Internet connections to GWS must also be encrypted via an SSL (Secure Sockets Layer) protocol.

Capacity Planning

All capacity planning decisions should initiate from the projected scope and requirements of the client application. Capacity relates to the functionality of two basic components:

- The design and required functionality for the client application.
- The size and type of network connection.

The expected volume of Web Service calls is the key factor in capacity planning. Issues for determining expected volume include:

- The expected look-to-book ratio between requests to the CRS and actual bookings.
- The expected number of sessioned vs. sessionless transactions.
- The types of Web Service calls planned, and their expected message sizes.

Support for Galileo Web Services

Technical support is provided during the development of new applications and modification to existing applications. Galileo by Travelport supports GWS only; the programming technique, language, or environment is not supported.

Online Help

GWS will include a web site that provides theoretical and practical information for using Galileo Web Services, including:

- A general overview of Web Services.
- Descriptions of each Galileo Web Service.
- Recommendations and requirements for application design, capacity planning, and security.
- Getting started guides.
- Sample transactions for each XML Select Web Service request and response.
- API Developer Notes, which provide practical assistance for developing against GWS, and designing client applications for specific business models.
- Steps for migrating existing XML Select applications to GWS.
- Detailed information about how to use each Web Service, including the individual transactions within the XML Select Service.
- A troubleshooting guide.

Training

Formal classroom training is not provided for Galileo Web Services. However, training for general Web Services tools and environments is provided by a number of training vendors.

Technical Support

Global API support is available for Galileo Web Services. For critical emergencies, 24 x 7support is also available.

GWS Sample Site

Galileo Web Services offers a Sample Web Site at:
http://testws.galileo.com/GWSSample/Pages/MainPage/MainFrame.aspx.

The Sample Site allows you to:

- Test Web Service requests for air, car, and hotel.
- Read an Overview of GWS architecture.
- Access the GWS Help System for details, instructions, and reference information.
- View and copy sample code for several languages and platforms to provide guidelines for development of client applications.

The features available on the Sample Site depend on your level of access:

- Everyone can use the sample requests and read the Overview.
- Prospective customers, who receive a preliminary user name and password from Galileo, can also review the Documentation section, including the GWS Help System.
- Licensed customers, who receive a user name and password, can also access the sample code in the View Code section.

Learn More About Galileo Web Services

There are several ways to learn more about Galileo Web Services.

Web Site

Additional information about Galileo Web Services is available at **ais.galileo.com**.

Contact Us

Travel providers can contact one of our Galileo by Travelport's regional offices. Travel suppliers can contact our Supplier Business Solutions group.

Asia/Pacific Regional Office

10/F, Golden Centre
188 Des Voeux Road Central
Hong Kong

Tel: 852 2821 2288
Fax: 852 2821 2299

Europe, Middle East, and Africa Regional Office

Galileo House
Axis Park
10 Hurricane Way
Langley
Berkshire SL3 8AG
United Kingdom

Tel: 44 (0) 1753 288000
Fax: 44 (0) 1753 288001

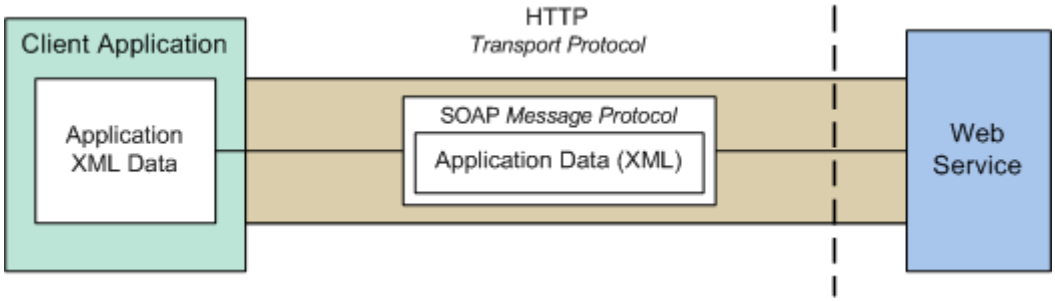
The Americas Regional Office

Travelport GDS - U.S. Headquarters
300 Galleria Parkway
Atlanta, GA 30339
United States

Email: **New.sales@galileo.com (New Sales Inquiries)**

Appendix A: Know Your Web Services Jargon

Understanding the underlying concepts behind Web Services helps to better illustrate why Web Services streamline and transform communication with the CRS. For all of the jargon that surrounds Web Services, the basic premise of Web Services is not very complex. The following diagram shows the basic flow of data between a client application and a Web Service.



XML (Extensible Markup Language)

XML is the format for message data used with Web Services.

What is XML?

Like its "cousin," HTML, XML is not a programming language, but rather a system for tagging chunks of data so that they can be sent and received in a defined, self-descriptive fashion. HTML tells you how the data should *display*. XML not only controls the display of data, but also tells you *what* the data is and *how* to use the data.

XML is commonly used across the Internet, but can also be used to transport information in a number of other environments. XML has already been used for several years to transfer data between applications. In the case of GWS, XML is already used to communicate with the CRSs in the XML Select product.

XML has several particularly useful qualities that can be applied to Web Services:

- *Customizable tags* that can be used to precisely define content for a specific application. The tags, together with the data described by the tags, form an XML *element*.
- Developers typically use an XML schema to describe specific XML data and define how to use that data. For example, the schema for a cruise application would define the types of data, such as cabin types, insurance waivers, and ports of call. The schema would also define the relationship between data elements, for example, by specifying that a port of call has to have a country element associated to it.
- *Data storage* within the application itself, which can increase product usability, reduce transaction time, and decrease server usage.
- *Separation of display and content*, in which changing the display of content does not affect the content itself.

How do Web Services use XML?

XML is used in two main ways for Web Services:

- The actual application data that is sent between systems is formatted in XML. For GWS, the data that is sent between a client travel application and the Galileo Web Services is in XML.
- Message protocols and supporting data that are used to send the application data are also written in XML.

SOAP (Simple Object Access Protocol)

SOAP is a specific kind of XML message that is "wrapped around" the XML application data. SOAP directs the application data to the Web Services, and defines how the data is processed by the Web Service.

Written in XML, the SOAP message is then wrapped in a *SOAP envelope* around the application data, also written in XML. SOAP indicates:

- How a message is being sent.
- Where the message is being sent.
- What kind of message is being sent.

For example, in GWS, a SOAP message for an air availability request would indicate that the XML data for the transaction should be sent:

- From the client application via HTTP (the standard for Web Services data transaction).
- Specifically to the GWS *XML Select Web Service* (which processes air availability requests).
- As a request to submit an XML transaction (messages to the CRS can be sent as either terminal or XML transactions). SOAP does not indicate that this particular XML transaction is an air availability request; that information lies in the attached XML data for the client application. Rather, SOAP tells the Web Service to perform whatever functions are required for the transaction.
- For example, the SOAP request for air availability includes a SubmitXml function. This function indicates that the XML Select Web Service should process the request by submitting it to XML Select as an XML transaction. The air availability request itself is processed by XML Select and the CRS, not by the Web Service.

WSDL (Web Services Definition Language)

The WSDL provides a compilation of the functions that can be performed by a specific Web Service.

The WSDL is another XML document that contains the XML schema about how to access and use the functions for a specific Web Service. For example, the WSDL for the XML Select Web Service contains information about functions such as submitting XML transactions, receiving terminal transactions, and beginning sessions. When a SOAP message is sent to the Web Service from the client application, the data about how to use that Service has been obtained from the WSDL.

Typically, the WSDL programmatically interfaces with the developer toolkit. Before designing a client application, developers create a proxy (copy) of the WSDL, which then provides them with the Web Service parameters need to design their application and data structures.

HTTPS (Hypertext Transfer Protocol Secure)

HTTPS is the method by which XML application data and SOAP data are sent between client applications and Web Services.

HTTPS is a communications protocol for transferring data that is already commonly used on the Internet. HTTPS is used extensively on the World Wide Web to transfer HTML from Web servers to internet browsers. The **https://** prefix in front of a web site address indicates that HTTPS is the protocol being used by that web site.

Because XML and HTML are related, XML can also use HTTPS as a transfer protocol. Web Services do not strictly require the use of HTTPS; however, it is currently the most commonly used transfer protocol for secured Web Services. GWS uses SSL (Secure Sockets Layer) protocol to manage HTTPS sessions.

Appendix B: Glossary

BF (Booking File)
A unique identifier that serves as the record for a specific travel plan. Contains a record locator, passenger information, travel provider information, as well as selected or reserved itinerary information. Referred to as a Passenger Name Record (PNR) on Apollo and a Booking File (BF) on Galileo.

PNR (Passenger Name Record)
A unique identifier that serves as the record for a specific travel plan. Contains a record locator, passenger information, travel provider information, as well as selected or reserved itinerary information. Referred to as a Passenger Name Record (PNR) on Apollo and a Booking File (BF) on Galileo.

Sessioned Transaction
A transaction designed to work with other transactions to perform a specific process. Sessioned transactions can be "strung together" to form an entire process, such as an Air Availability or Hotel Sell; each transaction interacts separately with the CRS. Users must sign on to the CRS to begin a session and sign off to end the session. Sessioned transactions are typically used for desktop products or terminal emulation clients, where users sign on and remain signed on for longer periods of time. Most agency-specific products rely on a sessioned environment.

Sessionless Transactions
A transaction designed to occur individually. A series of transactions is created by the client application and then sent to the CRS at one time. Sessionless transactions are frequently used for web-based applications where users do not sign on and do not remain signed on for longer periods of time.

SOAP (Simple Object Access Protocol)
XML data that directs the XML message data from a client application be delivered to the correct Web Service, and allows the message to be appropriately processed and returned by the Web Service.

Structured Data
Formalized data formats, as well as associated procedures and data records that are collectively referred to as *structured data*. Groups of structured data define the data format for each type of transaction.

Terminal Transaction
A transaction of raw, unstructured data. Terminal data was the original form for all data on the CRS.

XML (Extensible Markup Language)
A metalanguage that was developed specifically to provide standardized data transport across the Internet. Basically, a metalanguage is a language that describes another language. In this case, XML is a metalanguage that describes and standardizes how internet "language" is communicated from one system to another. XML is not a programming language, but rather a system for tagging chunks of data so that they can be sent and received in a self-defined fashion.

WSDL (Web Services Definition Language)
An XML document that contains the XML schema about how to access and use the functions for a specific Web Service. For example, the WSDL for the XML Select Web Service contains information about functions such as submitting XML transactions, receiving terminal transactions, and beginning sessions.