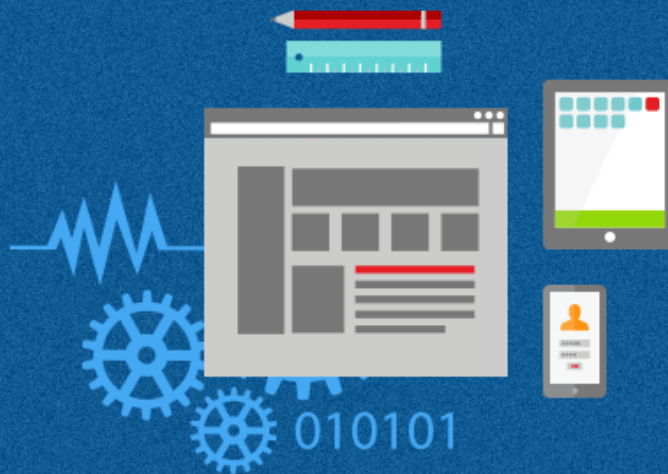




# SABRE APIs



## Sabre Travel Network TKT\_ElectronicDocumentServiceRQ User Guide Edition 2.0.0 January 20, 2022

**Document Edition 2.0.0** updated 1/20/2022 9:43 AM

This documentation is the confidential and proprietary intellectual property of the Sabre® business. Any unauthorized use, reproduction, preparation of derivative works, performance or display of this document or software represented by this document, without the express written permission of Sabre is strictly prohibited.

Sabre and the Sabre logo are trademarks and/or service marks. All other trademarks, service marks and trade names are the property of their respective owners.

© 2022 Sabre Inc. All rights reserved.

# Table of Contents

## 1 Introduction

|                                                         |   |
|---------------------------------------------------------|---|
| 1.1 Overview .....                                      | 5 |
| 1.2 Purpose of this Guide.....                          | 5 |
| 1.3 XML Examples in this Document .....                 | 6 |
| 1.4 Access Conditions.....                              | 6 |
| 1.5 Sabre Dev Studio.....                               | 6 |
| 1.6 TKT_ElectronicDocumentServicesRQ WSDL File .....    | 7 |
| 1.6.1 Changes from version 1.0.0 to version 2..00 ..... | 7 |

## 2 Understanding the TKT\_ElectronicDocumentServicesRQ Web Service

|                                                        |   |
|--------------------------------------------------------|---|
| 2.1 Introduction .....                                 | 8 |
| 2.2 Prerequisites .....                                | 8 |
| 2.3 Requirements.....                                  | 8 |
| 2.3.1 Eligible Documents.....                          | 8 |
| 2.4 The TKT_ElectronicDocumentServicesRQ Process ..... | 9 |
| 2.4.1 Display Request Scenarios.....                   | 9 |

## 3 TKT\_ElectronicDocumentServicesRQ Web Service Request

|                                                                                                             |    |
|-------------------------------------------------------------------------------------------------------------|----|
| 3.1 Introduction .....                                                                                      | 10 |
| 3.2 Request Attributes.....                                                                                 | 10 |
| 3.3 Request Elements.....                                                                                   | 10 |
| 3.3.1 STL:STL_Header.RQ Child Elements .....                                                                | 11 |
| 3.3.2 POS Child Elements .....                                                                              | 12 |
| 3.3.2.1 SearchParameters Child Elements .....                                                               | 12 |
| 3.3.2.2 Diagnostics Child Elements .....                                                                    | 13 |
| 3.4 TKT_ElectronicDocumentServicesRQ Request Examples .....                                                 | 13 |
| 3.4.1 Example Request to Display an Electronic Document with 13 Digit Document Number Only. ....            | 13 |
| 3.4.2 Example Request to Display an Electronic Document Specifying Document Type .....                      | 15 |
| 3.4.3 Example Request to Display an Electronic Document by Document Reference Number .....                  | 16 |
| 3.4.1 Example request to search and display an Electronic Document (EMD) Specifying the passenger name..... | 18 |

## 4 TKT\_ElectronicDocumentServicesRQ Web Service Response

|                                                                  |    |
|------------------------------------------------------------------|----|
| 4.1 Introduction .....                                           | 19 |
| 4.2 Response Attribute.....                                      | 19 |
| 4.3 Response Elements .....                                      | 19 |
| 4.3.1 Element Naming .....                                       | 19 |
| 4.3.2 STL:STL_Header.RS .....                                    | 21 |
| 4.3.3 Success/Error Results Elements .....                       | 22 |
| 4.3.4 DocumentDetailsDisplay Elements .....                      | 22 |
| 4.3.4.1 Considerations for WS Client / GUI Implementation .....  | 23 |
| 4.3.4.2 TKT_ElectronicDocumentServicesRQ Response Examples ..... | 24 |
| 4.3.4.2.1 Example ETR Green Screen Display .....                 | 24 |
| 4.3.4.2.2 Example ETR WS Response .....                          | 25 |
| 4.3.4.2.3 Example EMD-A Green Screen Display.....                | 33 |
| 4.3.4.2.4 Example EMD-A WS Response.....                         | 33 |
| 4.3.4.2.5 Example EMD-S Green Screen Display.....                | 38 |
| 4.3.4.2.6 Example EMD-S WS Response.....                         | 38 |
| 4.3.5 Diagnostics Element .....                                  | 42 |
| 4.3.6 Error Results Attributes and Elements .....                | 42 |
| 4.3.6.1 Error “type” Attributes .....                            | 42 |
| 4.3.6.2 SystemSpecificResults Element and Child Elements .....   | 43 |

## 5 Troubleshooting

|                                                                |    |
|----------------------------------------------------------------|----|
| 5.1 Introduction .....                                         | 44 |
| 5.2 List of Error Messages .....                               | 44 |
| 5.3 Example of Error Responses .....                           | 45 |
| 5.3.1 Non-Electronic Document Retrieval Error .....            | 45 |
| 5.3.1.1 Unexpected Subsystem Response.....                     | 46 |
| 5.3.2 Unable to Determine Validating Carrier .....             | 47 |
| 5.3.3 USG Authentication Errors.....                           | 49 |
| 5.3.3.1 Security Violations .....                              | 49 |
| 5.3.3.1.1 Invalid or Expired Security Token .....              | 49 |
| 5.3.3.1.2 No Branch Access.....                                | 50 |
| 5.3.4 Throttled Request/Response .....                         | 51 |
| 5.3.5 Invalid Service Version .....                            | 52 |
| 5.3.6 Invalid Request XML Format .....                         | 53 |
| 5.3.6.1 Invalid XML Namespace in RQ.....                       | 53 |
| 5.3.6.2 Invalid XML Element Name in RQ .....                   | 53 |
| 5.3.7 Database Response: Not Found.....                        | 54 |
| 5.3.8 Airline Issued EMD without Traveler Name in Search ..... | 55 |



# Introduction

## 1.1 Overview

---

TKT\_ElectronicDocumentServicesRQ, has been developed to deliver an XML version of the Sabre electronic ticket record (ETR) and electronic miscellaneous document (EMD) displays.

TKT\_ElectronicDocumentServicesRQ provides a quick and easy way for Sabre users to obtain from the *airline's version* of an ETR or EMD. The Web service will retrieve all the data from the airline's EDIFACT record of an ETR or EMD display request, translate the EDIFACT data elements to XML elements, then return the XML elements in a Web service response.

The service will return the data elements which are presented in the Sabre screen display using any of the following entries:

- WETR\*T[ticket number]
- WETR\*TR[ticket reference number from the \*T field in the PNR]
- WETR\*T[ticket number]/E
- WETR\*TR[ticket reference number from the \*T field in the PNR]/E
- WEMD\*T[ticket number]
- WEMD\*TR[emd reference number from the \*T field in the PNR]

**Note** Today, Sabre green screen users can request to see the standard ETR display (WETR\*), or an enhanced display (WETR\*/E) which shows some additional ticket data such as baggage allowance and validity dates. However, the TKT\_ElectronicDocumentServicesRQ Web service will return the data present in *both* the standard and enhanced display.

**Note** This Web service *does not* attempt to return the data elements present only in the Sabre Ticketing Database version of an ETR or EMD obtained by using a *WTDB entry*.

The Web service can be used by XML based web sites, online travel agencies, front end GUI Web services, and mobile applications.

Sabre does not provide a ready-to-use Web service client/GUI to use with this service. Sabre customers are responsible to create and maintain their own client/GUI to consume the XML RS generated by this service.

The details of the new service are described in following sections.

## 1.2 Purpose of this Guide

---

The purpose of this guide is to introduce the TKT\_ElectronicDocumentServicesRQ Web service in an easy to understand format. This guide is not a complete reference tool for XML design. Developers

should refer to the Sabre Dev Studio for additional information such as XML schemas, WSDL documents, metadata documents, etc.

It is assumed that a developer implementing this service is familiar with Sabre functionality \*or\* has access to a Sabre subject matter expert.

This guide provides:

- A description of the Web service features
- Descriptions of the attributes and elements used in the schema
- Examples of requests and responses for TKT\_ElectronicDocumentServicesRQ
- Common error messages that may be returned by the service along with the explanation as to why they occur, and suggestions for resolving the problem where applicable

### 1.3 XML Examples in this Document

---

The purpose of the XML examples provided in this document is to present possible XML content of the request and response.

Examples are for illustrative purposes only, and not all possible responses are captured in this user guide. In some sections, the XML examples are truncated in order to focus on the sections' topics.

**Note** TKT\_ElectronicDocumentServicesRQ is both the service name and action.  
<GetElectronicDocumentRQ> is the root element and will appear at the top of the XML.

### 1.4 Access Conditions

---

The Sabre *TKT\_ElectronicDocumentServicesRQ* Web service is a free product available for download from Sabre Dev Studios.

The following access requirements apply:

- Access is based on Employee Profile Record (EPR).
- All EPR groups should be auto provisioned with required OpenSystemsUser attribute. No additional security provisioning is required.
- Viewership of individual electronic documents is consistent with Sabre's existing security controls - the service will only return responses for electronic documents the user is entitled to see.

### 1.5 Sabre Dev Studio

---

The repository for all Sabre Web service information can be found at Sabre Dev Studio.

Here you can find documentation including schemas, WSDL files, code samples, workflows, and Sabre Web services related information.

Sabre Dev Studio is the most comprehensive reference tool available for XML developers.

Please visit the site and register for an account if you have not done so already.

## **1.6 TKT\_ElectronicDocumentServicesRQ WSDL File**

---

The WSDL file provides a description of the basic formats of Web service requests – how the service can be called, what parameters it expects, and what data structures it returns.

For the latest available links to WSDL files, check the TKT\_ElectronicDocumentServicesRQ Web service documentation on the Sabre Dev Studio website.

### **1.6.1 Changes from version 1.0.0 to version 2..00**

---

- Added StartDate and StartTime to the serviceCoupon of GetElectronicDocumentRS/DocumentDetailsDisplay/Ticket,
- Updating schema version from 1.0.0 to 2.0.0. As the changes in 1.0.0 are not backward compatible

# Understanding the TKT\_ElectronicDocumentServicesRQ Web Service

## 2.1 Introduction

---

This chapter provides a description of the TKT\_ElectronicDocumentServicesRQ Web service. It provides information about types of electronic documents that the service supports and some examples of how to integrate the new Web service into travel agency workflows.

This Web service adds new schema to the Sabre Dev Studios website. To utilize the TKT\_ElectronicDocuementServicesRQ Web service, you do not have to download the new schema, but it is recommended that you do so for your own development needs. It is also recommended that you download the current version of the WSDL.

**Note** TKT\_ElectronicDocumentServicesRQ is both the service name and action. <GetElectronicDocument> is the root element and will appear at the top of the XML requests and responses.

## 2.2 Prerequisites

---

Following are the prerequisites for using the Sabre TKT\_ElectronicDocumentServicesRQ Web service:

- Have met all requirements for accessing Sabre's Web services as per the requirements posted on the Sabre Dev Studio website.
- Have a client to consume the Web service, either Sabre Red Workspace or one that has been developed for your agency
- Have downloaded all the necessary files from Sabre Dev Studio

## 2.3 Requirements

---

The following information provides the rules and applicability of the TKT\_ElectronicDocuementServicesRQ Web Service.

### 2.3.1 Eligible Documents

---

The ElectronicDocumentServicesRQ Web service can be used display electronic tickets (ETRs) and electronic miscellaneous documents (EMDs) which meet the following criteria:

- The document must have been created in Sabre by the requestor's PCC or authorized branch as per the TJR \*\* or \*\* created in another GDS under the same IATA number present in the requestors TJR.
- The ticket must exist the airline carrier's ticket database (Sabre does not control the purge dates of electronic documents from the validating carrier's databases)

TKT\_ElectronicDocumentServices may not be used for:

- Paper Tickets (see section 5 for error responses if a paper ticket is included in the RQ)
- Electronic documents present in the Sabre Ticketing database but already purged from the validating carrier's database

## **2.4 The TKT\_ElectronicDocumentServicesRQ Process**

---

This section provides a brief description of the TKT\_ElectronicDocumentServicesRQ process.

### **2.4.1 Display Request Scenarios**

---

The following are some brief descriptions of TKT\_ElectronicDocumentServicesRQ Web service scenarios:

- The Web service client makes a request call which includes the minimum elements to fulfill the request, and any additional optional elements.
- The system will perform validations to determine if the requestor is entitled to view the document, and will send an EDIFACT display request to verify that the document is present in the validating carrier's electronic document database. The system will then translate the carrier's EDIFACT response to XML and return those data elements that are currently viewable in the Sabre green screen displays (WETR\*, WETR\*/E, WEMD\*).

Note: In some instances, an EDIFACT codes will be returned rather than Sabre's translation of that code in the green screen display. It will be up to the discretion of the Web service client consuming the XML response to decide how that code will be returned to the end user.

- The Web service response will include a response element to indicate that the transaction was successful.
- If any errors are detected at any point, the process will be interrupted, and an error message returned in the response.

# TKT\_ElectronicDocumentServicesRQ

## Web Service Request

### 3.1 Introduction

---

The purpose of this chapter is to briefly explain the content of the TKT\_ElectronicDocumentServicesRQ XML structure for requests.

This chapter should not be treated as complete information about the TKT\_ElectronicDocumentServicesRQ XML schema, but can be used as additional source of information for the TKT\_ElectronicDocumentServicesRQ definition available in the Sabre DEV Studio.

This section contains several tables identifying the XML elements and attributes important to understanding how to implement this new Web service. Additional elements may be present in the schema, which are not listed in the tables below. This is because the service utilizes a command template library which is share among multiple Sabre Web Services. The tables listened in this user guide are the ones utilized specifically by TKT\_ElectronicDocumentServicesRQ.

### 3.2 Request Attributes

---

Below is a table with the request attributes of TKT\_ElectronicDocumentServicesRQ:

| Attribute   | Description            | XPath                                 |
|-------------|------------------------|---------------------------------------|
| Version     | Version of the service | /GetElectronicDocumentRQ/@Version     |
| RequestType | W for current, H       | /GetElectronicDocumentRQ/@requestType |

### 3.3 Request Elements

---

Below is a table with the 4 top level request elements - STL\_Header, STL:POS, SearchParameters, and Diagnostic. Additional tables with child elements for the top level request elements are provided in their own subsections.

| Element    | Description                                                                                                                                              | XPath                                      |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------|
| STL_Header | Contains to and from PartyID's, CPAId, ConversationId, Service, Action, and MessageData elements                                                         | /GetElectronicDocumentRQ/STL:STL_Header.RQ |
| STL:POS    | Information about the point of sale (POS) and the agent requesting the electronic document display. Contains multiple child elements (see section 3.3.1) | /GetElectronicDocumentRQ/STL:POS           |

| Element          | Description                              | XPath                                     |
|------------------|------------------------------------------|-------------------------------------------|
| SearchParameters | Information about the document requested | /GetElectronicDocumentRQ/SearchParameters |
| Diagnostics      |                                          |                                           |

### 3.3.1 STL:STL\_Header.RQ Child Elements

Below is a table with the list of elements for the STL\_Header request element.

| Element         | Description                                                                                                     | XPath                                                          |
|-----------------|-----------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------|
| STL_Header      | Contains to and from PartyID's, CPAId, ConversationId, Service, Action, and MessageData elements                | /GetElectronicDocumentRQ/STL:STL_Header.RQ                     |
| ConversationID  | Identifies the set of related messages that make up a conversation between 2 parties – the requestor and the WS | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/ConversationID  |
| OrchestrationID | Used to track a set of service invocations and the sequence in which they are invoked.                          | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/OrchestrationID |
| Operation       | The action applied to the service                                                                               | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/Operation       |
| Service         | The name of the service being requested                                                                         | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/Service         |
| Messaging       | Provides an area for each system interacting with the message to record interactions.                           | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/Messaging       |
| Parameters      | Reserved for Sabre internal use                                                                                 | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/Parameters      |
| Security        | Provides security tokens for the system to use when authenticating the user/request.                            | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/Security        |
| DiagnosticData  | Reserved for Sabre internal use                                                                                 | /GetElectronicDocumentRQ/STL:STL_Header.RQ<br>/DiagnosticData  |

### 3.3.2 POS Child Elements

Below is a table for the elements of the Point of Sale (POS) request elements. They are defined as optional in the XML schema; however, their presence is verified by the service business validation layer. Therefore, the Company and Pseudo elements are required for the Web service to return an electronic document response.

| Element | Description                                                                                                                            | Example | XPath                                    |
|---------|----------------------------------------------------------------------------------------------------------------------------------------|---------|------------------------------------------|
| Company | The IATA assigned code of the GDS - it will be one of the three following codes:<br>1S for Sabre<br>1B for Sabre APAC<br>1F for INFINI | 1S      | /GetElectronicDocumentRQ/STL:POS/Company |

#### 3.3.2.1 SearchParameters Child Elements

Following is the list of child elements for SearchParameters.

**DocumentNumber** is 13 digit ticket number or by the ticket reference number in the PNR.

**DocumentType** is optional. If the DocumentType element is not in the request, the system will attempt to identify the type of document on a “best effort” basis. However if the system determines the wrong type, it could return a “DOCUMENT NOT FOUND” error incorrectly. It is therefore recommended that the DocumentType is always included in the request, when it is known.

**LineNumber** is the number corresponding to the number in the \*T field of the PNR

**CCVIEW** is valid only when the user’s Sabre EPR contains the keyword CCVIEW

**Passenger Name** is a required field to display an airline created EMD. User can search for the document by mentioning the passenger’s last name and the first name in the service request.

| Element        | Description                                                                            | Example            | XPath                                                         |
|----------------|----------------------------------------------------------------------------------------|--------------------|---------------------------------------------------------------|
| DocumentNumber | 13 digit document number<br>-OR-<br>Ticket reference number                            | 1251234567890      | /GetElectronicDocumentRQ/SearchParameters/DocumentNumber      |
| DocumentType   | 3 digit document type code                                                             | TKT<br>-OR-<br>EMD | /GetElectronicDocumentRQ/SearchParameters/DocumentType        |
| LineNumber     | The document reference from the *T field of a PNR                                      | 2, 3               | /GetElectronicDocumentRQ/SearchParameters/Document/LineNumber |
| CCView         | When present in the RQ, and the user credentials include the EPR keyword CCVIEW allows | CCVIEW             | /GetElectronicDocumentRQ/SearchParameters/Document/CCView     |

| Element  | Description                                                                                                                                                                                                                                                                                                                   | Example     | XPath                                              |
|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|----------------------------------------------------|
|          | the service to send unmasked credit card data unmasked.                                                                                                                                                                                                                                                                       |             |                                                    |
| Traveler | <p>Traveler search parameter should be used when DocumentType requested is EMD</p> <p>If document TYPE is other than EMD, Request Validation will ignore this Traveler search parameter.</p> <p>If document TYPE is EMD, but it is not Airline issued EMD, Request validation will ignore this Traveler search parameter.</p> | SAMPLE/TEST | /GetElectronicDocumentRQ/SearchParameters/Traveler |

When designing your client, it is recommended that you take steps to help the user avoid entering a “paper ticket” document number (or reference) because such a document will pass the schema/input validations, but error out at an internal subsystem call (see section 5). These “paper tickets” are often added to the PNR to cover agency fees.

### 3.3.2.2 Diagnostics Child Elements

---

The Diagnostics element is reserved for Sabre internal development use.

## 3.4 TKT\_ElectronicDocumentServicesRQ Request Examples

---

This section contains electronic ticket record (ETR) and electronic miscellaneous document (EMD) examples. The items noted with  red text are provided for explanation, they are not part of the Web service’s XML response.

### 3.4.1 Example Request to Display an Electronic Document with 13 Digit Document Number Only.

---

Below is the Sabre green screen request for an electronic ticket with the 13 digit ticket number.

To process the green screen request, the system examines the user’s credentials – established by the user’s EPR, PCC and LNIATA – determines if the users is entitled to receive a green screen response for this document.

### Example “green screen” request:

WETR\*T1397026404768«

Similarly, the TKT\_ElectronicDocumentServicesRQ request contains the same information required by the system to determine if the viewer is entitled to receive an XML response.

#### Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MessageHeader
      ns1:version=""
      xmlns:ns1="http://www.ebxml.org/namespaces/messageHeader">
      <ns1:From>
        <ns1:PartyId>999999</ns1:PartyId>
      </ns1:From>
      <ns1:To>
        <ns1:PartyId>123123</ns1:PartyId>
      </ns1:To>
      <ns1:CPAId>AAT</ns1:CPAId>
      <ns1:ConversationId>ABC123</ns1:ConversationId>
      <ns1:Service>TKT_ElectronicDocumentServicesRQ</ns1:Service>
      <ns1:Action>TKT_ElectronicDocumentServicesRQ</ns1:Action>
      <ns1:MessageData>
        <ns1:MessageId/>
        <ns1:Timestamp/>
      </ns1:MessageData>
    </ns1:MessageHeader>
    <ns1:Security
      xmlns:ns1="http://schemas.xmlsoap.org/ws/2002/12/secext">
      <ns1:BinarySecurityToken>Shared/IDL:IceSess\SessMgr:1\0.IDL/Common/!ICESMS\STSA
!ICESMSLB\STS.LB!-3180139424635212657!74232!0</ns1:BinarySecurityToken>
```

```

    </ns1:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetElectronicDocumentRQ
    Version="2.0.0"
    xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
    <ns1:STL_Header.RQ
      xmlns:ns1="http://www.sabre.com/ns/Ticketing/EDocStL"/>
    <ns2:POS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDocStL">
      <ns2:Company>1F</ns2:Company> ← Sabre "primehost", either 1S for Sabre, 1B for Abacus,
                                   or 1F for INFINI
      <ns2:Pseudo>WD40</ns2:Pseudo> ← Pseudo City Code (PCC) of agency
    </ns2:POS>
    <SearchParameters>
      <DocumentNumber>1397026404768</DocumentNumber> ← 13 digit document number specified
                                                         by the user
    </SearchParameters>
  </GetElectronicDocumentRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### 3.4.2 Example Request to Display an Electronic Document Specifying Document Type

---

As previously stated, the system will attempt to determine the type of document if document type is not specified on a "best effort" basis. However, Sabre strongly recommends sending the document type along with the document number whenever the document type is known.

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MessageHeader
      ns1:version=""
      xmlns:ns1="http://www.ebxml.org/namespaces/messageHeader">
      <ns1:From>
        <ns1:PartyId>999999</ns1:PartyId>
      </ns1:From>
      <ns1:To>

```

```

    <ns1:PartyId>123123</ns1:PartyId>
  </ns1:To>
  <ns1:CPAId>AAT</ns1:CPAId>
  <ns1:ConversationId>ABC123</ns1:ConversationId>
  <ns1:Service>TKT_ElectronicDocumentServicesRQ</ns1:Service>
  <ns1:Action>TKT_ElectronicDocumentServicesRQ</ns1:Action>
  <ns1:MessageData>
    <ns1:MessageId/>
    <ns1:Timestamp/>
  </ns1:MessageData>
</ns1:MessageHeader>
<ns1:Security
  xmlns:ns1="http://schemas.xmlsoap.org/ws/2002/12/secext">

<ns1:BinarySecurityToken>Shared/IDL:IceSess\SessMgr:1\0.IDL/Common/!ICESMS\STSB!IC
ESMSLB\STS.LB!-3187583529165862005!1264209!0</ns1:BinarySecurityToken>
</ns1:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetElectronicDocumentRQ
    Version="2.0.0"
    xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
    <ns1:STL_Header.RQ
      xmlns:ns1="http://www.sabre.com/ns/Ticketing/EDocStl"/>
    <ns2:POS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDocStl">
      <ns2:Company>1S</ns2:Company>
      <ns2:Pseudo>61N1</ns2:Pseudo>
    </ns2:POS>
    <SearchParameters>
      <DocumentNumber>7954573753197</DocumentNumber>
      <DocumentType>EMD</DocumentType>
    </SearchParameters>
  </GetElectronicDocumentRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### 3.4.3 Example Request to Display an Electronic Document by Document Reference Number

---

The user may also request the electronic document by referencing the appropriate line number from the \*T field of a PNR present in the user's AAA.

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MessageHeader
      ns1:version=""
      xmlns:ns1="http://www.ebxml.org/namespaces/messageHeader">
    <ns1:From>

```

```

    <ns1:PartyId>999999</ns1:PartyId>
  </ns1:From>
  <ns1:To>
    <ns1:PartyId>123123</ns1:PartyId>
  </ns1:To>
  <ns1:CPAId>AAT</ns1:CPAId>
  <ns1:ConversationId>ABC123</ns1:ConversationId>
  <ns1:Service>TKT_ElectronicDocumentServicesRQ</ns1:Service>
  <ns1:Action>TKT_ElectronicDocumentServicesRQ</ns1:Action>
  <ns1:MessageData>
    <ns1:MessageId/>
    <ns1:Timestamp/>
  </ns1:MessageData>
</ns1:MessageHeader>
<ns1:Security
  xmlns:ns1="http://schemas.xmlsoap.org/ws/2002/12/secext">
  <ns1:BinarySecurityToken>Shared/IDL:IceSess\SessMgr:1\0.IDL/Common/!ICESMS\STSA!IC
  ESMsLB\STS.LB!-3186513034027242097!1821896!0</ns1:BinarySecurityToken>
</ns1:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetElectronicDocumentRQ
    Version="2.0.0"
    xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
    <ns1:STL_Header.RQ
      xmlns:ns1="http://www.sabre.com/ns/Ticketing/EDocStl"/>
    <ns2:POS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDocStl">
      <ns2:Company>1S</ns2:Company>
      <ns2:Pseudo>WD40</ns2:Pseudo>
    </ns2:POS>
    <SearchParameters>
      <DocumentType/>
      <LineNumber>3</LineNumber>
    </SearchParameters>
  </GetElectronicDocumentRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### 3.4.1 Example request to search and display an Electronic Document (EMD) Specifying the passenger name

---

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MessageHeader
      ns1:version=""
      xmlns:ns1="http://www.ebxml.org/namespaces/messageHeader">
      <ns1:From>
        <ns1:PartyId>999999</ns1:PartyId>
      </ns1:From>
      <ns1:To>
        <ns1:PartyId>123123</ns1:PartyId>
      </ns1:To>
      <ns1:CPAId>AAT</ns1:CPAId>
      <ns1:ConversationId>ABC123</ns1:ConversationId>
      <ns1:Service>TKT_ElectronicDocumentServicesRQ</ns1:Service>
      <ns1:Action>TKT_ElectronicDocumentServicesRQ</ns1:Action>
      <ns1:MessageData>
        <ns1:MessageId/>
        <ns1:Timestamp/>
      </ns1:MessageData>
    </ns1:MessageHeader>
    <ns1:Security
      xmlns:ns1="http://schemas.xmlsoap.org/ws/2002/12/secext">
      <ns1:BinarySecurityToken>Shared/IDL:IceSess\SessMgr:1\0.IDL/Common/!ICESMS\STSB!ICESMSLB\S
      TS.LB!-3187583529165862005!1264209!0</ns1:BinarySecurityToken>
    </ns1:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRQ
      Version="2.0.0"
      xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
      <ns1:STL_Header.RQ
        xmlns:ns1="http://www.sabre.com/ns/Ticketing/EDocStl"/>
      <ns2:POS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDocStl">
        <ns2:Company>1S</ns2:Company>
        <ns2:Pseudo>61N1</ns2:Pseudo>
      </ns2:POS>
      <SearchParameters>
        <DocumentNumber>7951573753197</DocumentNumber>
        <DocumentType>EMD</DocumentType>
        <Traveler>
          <FirstName>TEST</FirstName>
          <LastName>SWS</LastName>
        </Traveler>
      </SearchParameters>
    </GetElectronicDocumentRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

# TKT\_ElectronicDocumentServicesRQ

## Web Service Response

### 4.1 Introduction

---

This section describes the TKT\_ElectronicDocumentServicesRQ XML structure for responses. It is not intended to supply information of all the elements and attributes. For complete information, please refer to schema documentation in Sabre Dev Studio. Examples of responses are included in the scenarios in section 5 of this document.

This section contains several tables identifying the XML elements and attributes important to understanding how to implement this new Web service. Additional elements may be present in the schema, which are not listed in the tables below. This is because the service utilizes a command template library which is share among multiple Sabre Web Services. The tables listened in this user guide are the ones utilized specifically by TKT\_ElectronicDocumentServicesRQ.

### 4.2 Response Attribute

---

The TKT\_ElectronicDocumentServicesRQ response consists of the following attributes and high level elements.

| Attribute | Description                                                                            | XPath                             |
|-----------|----------------------------------------------------------------------------------------|-----------------------------------|
| Version   | Initial release is version 1.0<br>Please refer to Sabre Dev Studios for latest version | /GetElectronicDocumentRS/@Version |

### 4.3 Response Elements

---

The response elements are the XML tag pairs which contain the data about the document display response and about the electronic document itself.

The DocumentDetailsDisplay element contains the information about the document which a Sabre green screen user normally sees in the WETR\*, WETR\*/E and WEMD\* displays.

#### 4.3.1 Element Naming

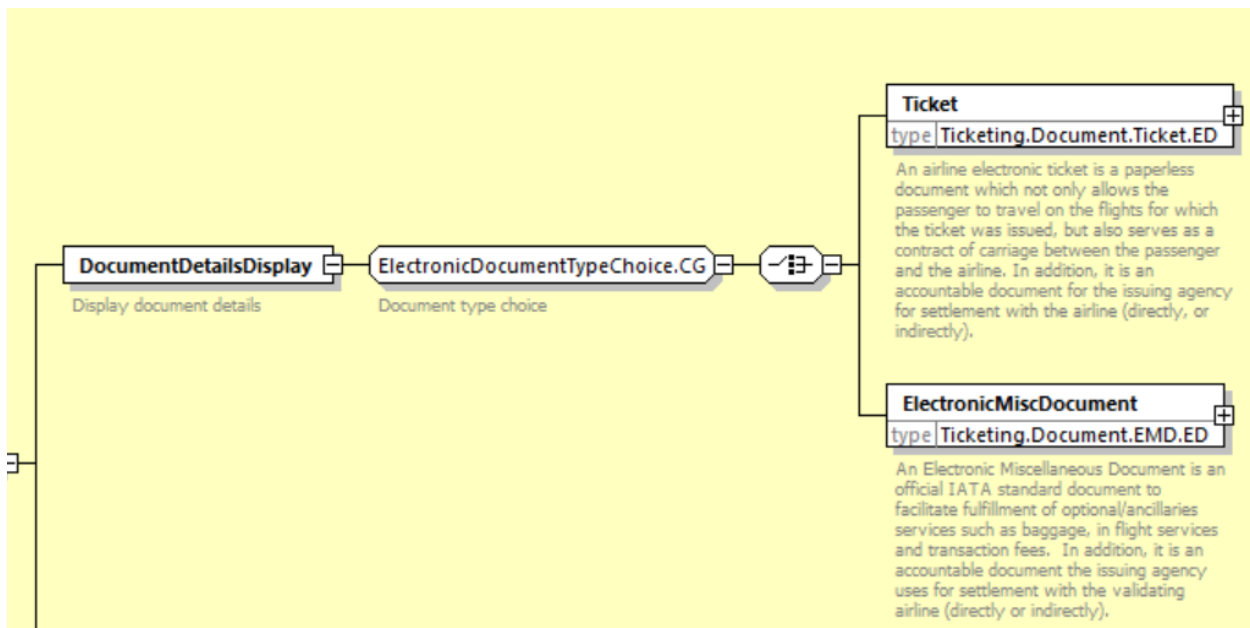
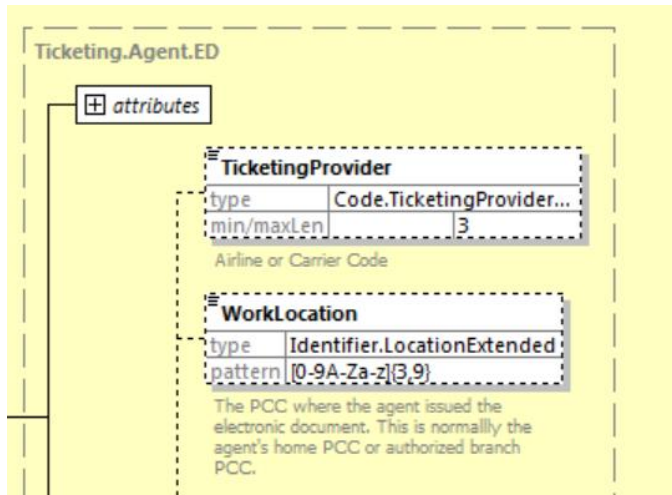
---

In some cases, the child element names are closely aligned with travel industry terminology (e.g. IataNumber), but in other cases they depart from industry terminology due to internal development requirements (e.g. WorkLocation is the element name for the Sabre PCC).

This User Guide does not attempt to fully document the XML response elements. It is therefore highly recommended that you use the annotation in the XML schema itself if you have questions about the function of a particular XML element.

## Examples:

Below are examples of the schema annotation for the elements `<TicketingProvider>`, `<WorkLocation>`, `<Ticket>` and `<ElectronicMiscDocument>` as seen through the Altova XMLSpy tool.



| Element                | Description                                                                                                                                                                      | XPath                                                      |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|
| STL:STL_Header.RS      | Contains OrchestrationID and Results                                                                                                                                             | /GetElectronicDocumentRS/STL:STL_Header.RS                 |
| OrchestrationID        |                                                                                                                                                                                  | /GetElectronicDocumentRS/STL:STL_Header.RS/OrchestrationID |
| Results                |                                                                                                                                                                                  | /GetElectronicDocumentRS/STL:STL_Header.RS/Results         |
| Agent                  | Agent information – ticketing provider                                                                                                                                           | /GetElectronicDocumentRS/Agent                             |
| TransactionInfo        |                                                                                                                                                                                  | /GetElectronicDocumentRS/TransactionInfo                   |
| DocumentDetailsDisplay | Contains the data elements present in the green screen document display. Examples are provided in section 5. Please refer to the schema for full documentation of data elements. | /GetElectronicDocumentRS/DocumentDetailsDisplay            |
| DocumentHistoryDisplay | Reserved for future use                                                                                                                                                          | /GetElectronicDocumentRS/DocumentHistoryDisplay            |
| Diagnostic             | Reserved for Sabre internal development use                                                                                                                                      | n/a                                                        |

#### 4.3.2 STL:STL\_Header.RS

The Header contains the OrchestrationID element and the Results element which contains business messages warnings and errors.

| Element          | Description                                                                                                             | XPath                                                      |
|------------------|-------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|
| Orchestration ID | Information indicating the result of the service call, orchestration and transaction identifiers, errors, and warnings. | /GetElectronicDocumentRS/STL:STL_Header.RS/OrchestrationID |
| Results          | Success, Warnings and Errors                                                                                            | /GetElectronicDocumentRS/STL:STL_Header.RS/Results         |

### 4.3.3 Success/Error Results Elements

---

The Results element has two child elements to either communicate the success of the display request, or send an error advising the failure of the display request. The success and Error elements are mutually exclusive - you can only have one of them in the Web service response, not both.

| Element | Description                                                                                                                                                                                                                                                                  | XPath                                                       |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|
| Success | Contains elements identifying the source (ExchangeServices) and the system (T2).<br>The success element will be present if the exchange request has been successful                                                                                                          | /GetElectronicDocumentRS/STL:STL_Header.RS /Results/Success |
| Error   | Contains elements identifying the source (ExchangeServices) and the system (T2) as well as an element containing the error message texts. The error element will be present if an error condition has been encountered and the exchange transaction has not been successful. | /GetElectronicDocumentRS/STL:STL_Header.RS /Results/Error   |

### 4.3.4 DocumentDetailsDisplay Elements

---

The DocumentDetailsDisplay element contain the data that is normally seen in the WETR\* and WEMD\* green screen displays.

The DocumentDetailsDisplay element has 2 primary child elements – one for ETRs and one for EMD

| Element                | Description                                                               | XPath                                                                  |
|------------------------|---------------------------------------------------------------------------|------------------------------------------------------------------------|
| Ticket                 | Contains the data elements for an Electronic Ticket Record (ETR)          | /GetElectronicDocumentRS/DocumentDetailsDisplay/Ticket                 |
| ElectronicMiscDocument | Contains the data elements for an Electronic Miscellaneous Document (EMD) | /GetElectronicDocumentRS/DocumentDetailsDisplay/ElectronicMiscDocument |

It is important to understand these elements in the context of the XML structure. Some examples are provided in this section, but it is highly recommended that you review the element's annotations in the schema itself when making design decisions for your Web service client/GUI.

#### 4.3.4.1 Considerations for WS Client / GUI Implementation

---

When designing/updating your Web service client/GUI which will consume the TKT\_ElectronicDocumentServicesRQ, please consider the following:

- The data **labels** (e.g. BRDOFF, FARE CALCULATION, FF, FCI, etc.) shown in the Sabre green screen response **are not returned** as values in XML elements in the Web service response because they are not present in the carrier's EDIFACT response to the display request. They are coded into the Sabre green screen displays to provide context to the data returned by the carrier in an EDIFACT message.

The XML tags of the DocumentDetailsDisplay child elements are for the most part self-explanatory, and are closely aligned with the data labels in the Sabre green screen display. However, if you are unsure in which DocumentDetailsDisplay child element certain data is returned, please refer to the annotation in the TKT\_ElectronicDocumentServicesRQ schema on Sabre Dev Studios.

- Certain data items are not returned exactly as they appear in the Sabre green screen response. This is due to differences in data structure for EDIFACT and XML.
  - Example: The Sabre green screen shows ARUNK whereas the Web service RS shows an XML attribute of "A". This is because in EDIFACT, an ARUNK segment is identified by the **absence** of a CPN segment.
- Additional data elements which do not appear in the Sabre green screen do appear in the Web service response. This is because the additional data might be useful to Sabre customers, or that TKT\_ElectronicDocumentServicesRQ might more easily integrate with other existing and future Sabre Web services.
  - Example: The Sabre green screen display shows the date/month for validity dates, while the Web service response returns date/month/year.
  - Example: The Sabre green screen display shows only the marketing carrier, while the Web service response returns both marketing carrier and operating carrier.
  - Example: The Sabre green screen WETR\* display does not show commission but the Web service response returns the commission amount and the commission percentage, when present in the EDIFACT version of the ticket.

#### 4.3.4.2 TKT\_ElectronicDocumentServicesRQ Response Examples

This section contains examples of the Sabre Green Screen electronic document displays, and the corresponding Web service Responses for the same examples.

##### 4.3.4.2.1 Example ETR Green Screen Display

When implementing the TKT\_ElectronicDocumentServicesRQ Web service, you may wish to compare some of your own electronic documents' green screen displays to the Web service response to make design decisions about your client/GUI.

Below is an example of an Electronic Ticket as seen in the enhanced Sabre green screen display (WETR\*/E).

```
ELECTRONIC TICKET RECORD                                CRS CODE:0011
INV:3232643          CUST:1234567890          PNR:XBWYQ
TKT:6077026404769/70  ISSUED:09FEB18    PCC:WD40    IATA:52812874
NAME:SALAMA/SAMHAR          FF:EY100097448780
NAME REF:123456789012345678901234    TOUR ID:ITIABCD
FORM OF PAYMENT          FCI: 0
1    CASH
CPN  A/L  FLT  CLS  DATE  BRDOFF  TIME  ST F/B          STAT
1    EY   160   Y  23FEB  DFWAUH  1905   OK YRTUS/YF          OPEN
                        NVB:23FEB    NVA:23FEB    BAG:2PC
2    EY   224   Y  28FEB  AUHDEL  1435   OK YRTUS/YF          OPEN
                        NVB:28FEB    NVA:28FEB    BAG:2PC
3                                ARUNK
4    EY   378   Y  07MAR  BAHAUH  1020   OK YRTUS/YF          OPEN
                        NVB:07MAR    NVA:07MAR    BAG:2PC
5    EY   161   Y  11MAR  AUHDFW  0920   OK YRTUS/YF          OPEN
                        NVB:11MAR    NVA:11MAR    BAG:2PC

FARE      USD6021.00
TOTAL     USD6172.21

FARE CALCULATION
DFW EY AUH Q75.00EY DEL Q DFWDEL2.50Q DFWDEL228.00 2738.50/-BAH+
MD<<
  EY AUH EY DFW Q BAHDFW2.50Q BAHDFW218.00 2756.50NUC6021.00END +
ROE1.00 XFDFW4.5

TAX BREAKDOWN
TAX      1.10YR TAX      36.60US TAX      5.65YC TAX      7.00XY
TAX      3.96XA TAX      5.60AY TAX      40.80AE TAX      2.80TP
TAX      5.60ZR TAX      19.00F6 TAX      18.60BH TAX      4.50XF

ENDORSEMENT
NON ENDO/ REF
```

#### 4.3.4.2.2 Example ETR WS Response

The Web service (XML) version of the same electronic ticket from section 4.3.3.1 is shown below. As you can see, the XML tags of the DocumentDetailsDisplay child elements are similar to the data labels in the Sabre green screen display.

In the XML example below, notes are provided in **red bold** text for response elements which are different than the green screen data labels. The red bold text is not a part of the TKT\_ElectronicDocumentServices response, itself - it is provided for reference only.

Response elements that are the same as the green screen response data labels – or very similar – are not noted.

#### Response

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRS
      Version="2.0.0"
      xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
      <STL:STL_Header.RS
        timeStamp="2018-02-09T15:08:41"
        xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl">
        <STL:OrchestrationID
          seq="0">TKTHLI700-20559-1471999380-1518210521239-115-edoc:0</STL:OrchestrationID>
        <STL:Results>
          <STL:Success>
            <STL:System>TKT-DS</STL:System>
            <STL:Source>EDOC</STL:Source>
          </STL:Success>
        </STL:Results>
      </STL:STL_Header.RS>
    <Agent
      sine="AQ"
      type="GDS">
```

```

<TicketingProvider>EY</TicketingProvider>    ← this is the validating carrier
<WorkLocation>WD40</WorkLocation>           ← Sabre PCC that issued the ETR
<HomeLocation>MKE</HomeLocation>            ← location in issuing agency's TJR
<IsoCountryCode>US</IsoCountryCode>
<IataNumber>52812874</IataNumber>
<CRSCode>0011</CRSCode>
</Agent>
<TransactionInfo
  sequence="1">
<LocalDateTime>2018-02-09T15:08:41.784-06:00</LocalDateTime>
<SystemDateTime>2018-02-09T15:08:41.784-06:00</SystemDateTime>
<SystemProvider>1S</SystemProvider>          ← code for the Sabre GDS
</TransactionInfo>
<DocumentDetailsDisplay>
<Ticket
  number="6077026404769"
  type="TKT">
<Details>
  <TourNumber>ITIABCD</TourNumber>
  <Reservation>
    <Sabre>XBUWYQ</Sabre>                    ← this is the Sabre PNR
  </Reservation>
  <LocalIssueDateTime>2018-02-09T00:00:00.000-00:00</LocalIssueDateTime>
  <AgencyBillingNumber>1234567890</AgencyBillingNumber> ← this is the DK Number
  <InvoiceNumber>3232643</InvoiceNumber>
  <FareCalculationPricing>0</FareCalculationPricing>    ← this is the FCI code
</Details>
<Customer>
  <Traveler>
    <Name>SAMHAR SALAMA</Name>
    <FirstName>SAMHAR</FirstName>
    <LastName>SALAMA</LastName>
    <ExternalNumber>123456789012345678901234</ExternalNumber> ← name reference nbr
  </Traveler>

```

```

</Customer>
<Affinity>
  <FrequentFlyer>
    <Provider>EY</Provider>
    <Number>100097448780</Number>    ← frequent flyer number
  </FrequentFlyer>
</Affinity>
<ServiceCoupon
  coupon="1">    ← flight coupon 1
  <MarketingProvider>EY</MarketingProvider>    ← marketing carrier code
  <MarketingFlightNumber>160</MarketingFlightNumber>
  <OperatingProvider>EY</OperatingProvider>    ← operating carrier code (will be different than
marketing provider on code share flights)
  <ClassOfService
    name="Y">Y</ClassOfService>
  <FareBasis>YRTUS/YF</FareBasis>    ← Fare basis / ticket designator
  <StartLocation>DFW</StartLocation>    ← board point (BRD)
  <StartDateTime>2018-02-23T19:05:00</StartDateTime>    ← departure date/time
  <EndLocation>AUH</EndLocation>    ← off point (OFF)
  <NotValidBeforeDate>2018-02-23</NotValidBeforeDate>
  <NotValidAfterDate>2018-02-23</NotValidAfterDate>
  <BookingStatus>OK</BookingStatus>    ← PNR segment status, equivalent to HK
  <CurrentStatus>OPEN</CurrentStatus>    ← ticket coupon status
  <BagAllowance
    amount="2" code="PC"/>    ← Baggage allowance, 2 pieces
</ServiceCoupon>
<ServiceCoupon
  coupon="2">    ← flight coupon 2
  <MarketingProvider>EY</MarketingProvider>
  <MarketingFlightNumber>224</MarketingFlightNumber>
  <OperatingProvider>EY</OperatingProvider>
  <ClassOfService
    name="Y">Y</ClassOfService>
  <FareBasis>YRTUS/YF</FareBasis>
  <StartLocation>AUH</StartLocation>

```

```

<StartDateTime>2018-02-28T14:35:00</StartDateTime>
<EndLocation>DEL</EndLocation>
<NotValidBeforeDate>2018-02-28</NotValidBeforeDate>
<NotValidAfterDate>2018-02-28</NotValidAfterDate>
<BookingStatus>OK</BookingStatus>
<CurrentStatus>OPEN</CurrentStatus>
<BagAllowance
  amount="2" code="PC"/>
</ServiceCoupon>
<ServiceCoupon
  coupon="3" type="A"/> ← flight coupon 3 - ARNK
<ServiceCoupon
  coupon="4"> ← flight coupon 4
  <MarketingProvider>EY</MarketingProvider>
  <MarketingFlightNumber>378</MarketingFlightNumber>
  <OperatingProvider>EY</OperatingProvider>
  <ClassOfService
    name="Y">Y</ClassOfService>
  <FareBasis>YRTUS/YF</FareBasis>
  <StartLocation>BAH</StartLocation>
  <StartDateTime>2018-03-07T10:20:00</StartDateTime>
  <EndLocation>AUH</EndLocation>
  <NotValidBeforeDate>2018-03-07</NotValidBeforeDate>
  <NotValidAfterDate>2018-03-07</NotValidAfterDate>
  <BookingStatus>OK</BookingStatus>
  <CurrentStatus>OPEN</CurrentStatus>
  <BagAllowance
    amount="2" code="PC"/>
  </ServiceCoupon>
  <ServiceCoupon
    coupon="5"> ← flight coupon 5
    <MarketingProvider>EY</MarketingProvider>
    <MarketingFlightNumber>161</MarketingFlightNumber>
    <OperatingProvider>EY</OperatingProvider>

```

```

<ClassOfService
  name="Y">Y</ClassOfService>
<FareBasis>YRTUS/YF</FareBasis>
<StartLocation>AUH</StartLocation>
<StartDateTime>2018-03-11T09:20:00</StartDateTime>
<EndLocation>DFW</EndLocation>
<NotValidBeforeDate>2018-03-11</NotValidBeforeDate>
<NotValidAfterDate>2018-03-11</NotValidAfterDate>
<BookingStatus>OK</BookingStatus>
<CurrentStatus>OPEN</CurrentStatus>
<BagAllowance
  amount="2"
  code="PC"/>
</ServiceCoupon>
<RelatedDocument>
  <Conjunctive>
    <Number>6077026404770</Number> ← next ticket number in conjunctive series. In this case, for
flight coupon 5.
  </Conjunctive>
</RelatedDocument>
<Amounts>
  <New>
    <Base>
      <Amount
        currencyCode="USD"
        decimalPlace="2">6021.00</Amount> ← base amount
      <Text>6021.00</Text> ← base amount or text – varies based on settlement plan
    </Base>
    <Total>
      <Amount
        currencyCode="USD"
        decimalPlace="2">6172.21</Amount> ← total amount
      <Text>6172.21</Text>
    </Total>
  </New>

```

```

<Other>
  <Commission>
    <Amount
      currencyCode="USD"
      decimalPlace="2">722.52</Amount> ← commission amount
    <PercentageRate>12.00</PercentageRate> ←commission percentage
    <Text>01200</Text> ←commission percentage or text, varies
  </Commission>
</Other>
</Amounts>
<Taxes>
  <New>
    <Tax
      code="YR">
        <Amount currencyCode="USD"
          decimalPlace="2">1.10</Amount>
        </Tax>
      <Tax
        code="US">
          <Amount currencyCode="USD"
            decimalPlace="2">36.60</Amount>
          </Tax>
        <Tax
          code="YC">
            <Amount currencyCode="USD"
              decimalPlace="2">5.65</Amount>
            </Tax>
          <Tax
            code="XY">
              <Amount currencyCode="USD"
                decimalPlace="2">7.00</Amount>
              </Tax>
            <Tax
              code="XA">

```

```

    <Amount currencyCode="USD"
    decimalPlace="2">3.96</Amount>
  </Tax>
  <Tax
    code="AY">
    <Amount currencyCode="USD"
    decimalPlace="2">5.60</Amount>
  </Tax>
  <Tax
    code="AE">
    <Amount currencyCode="USD"
    decimalPlace="2">40.80</Amount>
  </Tax>
  <Tax
    code="TP">
    <Amount currencyCode="USD"
    decimalPlace="2">2.80</Amount>
  </Tax>
  <Tax
    code="ZR">
    <Amount currencyCode="USD"
    decimalPlace="2">5.60</Amount>
  </Tax>
  <Tax
    code="F6">
    <Amount currencyCode="USD"
    decimalPlace="2">19.00</Amount>
  </Tax>
  <Tax
    code="BH">
    <Amount currencyCode="USD"
    decimalPlace="2">18.60</Amount>
  </Tax>
  <Tax
    code="XF">

```

```

        <Amount currencyCode="USD"
decimalPlace="2">4.50</Amount>
    </Tax>
</New>
</Taxes>
<Payment
sequence="1" type="CA">
    <Total>
        <Amount
            currencyCode="USD"
            decimalPlace="0">0</Amount>
    </Total>
    <Other>
        <Details>Intentionally left empty</Details>
    </Other>
</Payment>
<Remark>
    <Endorsements sequence="0">NON ENDO/ REF//VALID ONLY IN EY</Endorsements>
</Remark>
<FareCalculation>
    <New>DFW EY AUH Q75.00EY DEL Q DFWDEL2.50Q DFWDEL228.00 2738.50/-BAH EY AUH EY
DFW Q BAHDFW2.50Q BAHDFW218.00 2756.50NUC6021.00END ROE1.00 XFDFW4.5</New>
</FareCalculation>
</Ticket>
</DocumentDetailsDisplay>
</GetElectronicDocumentRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

#### 4.3.4.2.3 Example EMD-A Green Screen Display

```
WEMD*T7954573753501«
ELECTRONIC MISCELLANEOUS DOCUMENT RECORD
INV:3232412          CUST:                      PNR:BIDGWL
EMD:7954573753501    ISSUED:26JAN18      PCC:WD40    IATA:52812874
NAME:DOUGLAS/KATHRYN          FF:VA7380011585
NAME REF:1234567890          TOUR ID:
FOP: CASH
RFIC:C/BAGGAGE
CPN  A/L  BRDOFF  DATE  QTY          AIR EXTRA          STAT
1    VA    LAXBNE  N/A    2  BAGGAGE          OPEN
2    VA    BNELAX  N/A    2  BAGGAGE          OPEN

CPN SERVICE DETAIL                      BASE AMT
1  BAGGAGE UP TO 35KG                      USD17.20
2  BAGGAGE UP TO 35KG                      USD17.20

BASE AMT:AUD42.68                      EMD TTL:USD34.40
TAX AMT:                      EQUIV AMT:USD34.40

ENDORSEMENTS/RESTRICTIONS:REFUNDABLE
SETTLEMENT AUTHORIZATION:
EMD TYPE:ASSOCIATED                      ‡
MD«
ASSOCIATED ELECTRONIC TICKET NUM:7957016147778                      ‡
CPN ASSOCIATION
ET CPN - EMD CPN
1        1
2        2
```

#### 4.3.4.2.4 Example EMD-A WS Response

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRS
      Version="2.0.0"
      xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
      <STL:STL_Header.RS
```

```

timeStamp="2018-02-09T15:53:28"
  xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStL">
<STL:OrchestrationID
seq="0">TKTHLI700-20559-1471999380-1518213208736-189-edoc:0</STL:OrchestrationID>
<STL:Results>
  <STL:Success>
    <STL:System>TKT-DS</STL:System>
    <STL:Source>EDOC</STL:Source>
  </STL:Success>
</STL:Results>
</STL:STL_Header.RS>
<Agent
  sine="ATW"
  type="GDS">
  <TicketingProvider>VA</TicketingProvider>
  <WorkLocation>WD40</WorkLocation>
  <HomeLocation>MKE</HomeLocation>
  <IataNumber>52812874</IataNumber>
</Agent>
<TransactionInfo
  sequence="1"
  type="SALE">
  <LocalDateTime>2018-02-09T15:53:29.599-06:00</LocalDateTime>
  <SystemDateTime>2018-02-09T15:53:29.599-06:00</SystemDateTime>
  <SystemProvider>1S</SystemProvider>
</TransactionInfo>
<DocumentDetailsDisplay>
  <ElectronicMiscDocument
    number="7954573753501"
    type="EMD">
  <Details>
    <Reservation>
      <Sabre>BIDGWL</Sabre>
    </Reservation>

```

```

<LocalIssueDateTime>2018-01-26T00:00:00.000-00:00</LocalIssueDateTime>
<ItineraryType>I</ItineraryType>
<ElectronicMiscType>A</ElectronicMiscType>
<ReasonForIssue>C</ReasonForIssue>
<InvoiceNumber>3232412</InvoiceNumber>
</Details>
<Customer>
  <Traveler>
    <Name>DOUGLAS/KATHRYN</Name>
    <FirstName>KATHRYN</FirstName>
    <LastName>DOUGLAS</LastName>
    <ExternalNumber>1234567890</ExternalNumber>
  </Traveler>
</Customer>
<Miscellaneous>
  <Fee
    coupon="1" description="BAGGAGE UP TO 35KG"
    quantity="2">
  <Total>
    <Amount currencyCode="USD"
    decimalPlace="2">17.20</Amount>
    <Text>17.20</Text>
  </Total>
  </Fee>
  <OptionalService
    reason="0C7">
    <FeeProvider>VA</FeeProvider>
  </OptionalService>
  <CouponDetails
    coupon="1" description="BAGGAGE UP TO 35KG"
    quantity="2">
    <MarketingProvider>VA</MarketingProvider>
    <StartLocation>LAX</StartLocation>
    <EndLocation>BNE</EndLocation>
    <CouponStatus>OPEN</CouponStatus>

```

```

</CouponDetails>
<AssociatedTicketNumber
  couponNumber="1">7957016147778</AssociatedTicketNumber>
<Affinity>
  <FrequentFlyer>
    <Provider>VA</Provider>
    <Number>7380011585</Number>
  </FrequentFlyer>
</Affinity>
</Miscellaneous>
<Miscellaneous>
  <Fee
    coupon="2" description="BAGGAGE UP TO 35KG"
    quantity="2">
  <Total>
    <Amount currencyCode="USD"
    decimalPlace="2">17.20</Amount>
    <Text>17.20</Text>
  </Total>
</Fee>
<OptionalService
  reason="0C7">
  <FeeProvider>VA</FeeProvider>
</OptionalService>
<CouponDetails
  coupon="2" description="BAGGAGE UP TO 35KG"
  quantity="2">
  <MarketingProvider>VA</MarketingProvider>
  <StartLocation>BNE</StartLocation>
  <EndLocation>LAX</EndLocation>
  <CouponStatus>OPEN</CouponStatus>
</CouponDetails>
<AssociatedTicketNumber
  couponNumber="2">7957016147778</AssociatedTicketNumber>
<Affinity>

```

```

<FrequentFlyer>
  <Provider>VA</Provider>
  <Number>7380011585</Number>
</FrequentFlyer>
</Affinity>
</Miscellaneous>
<Amounts>
  <New>
    <Base>
      <Amount
        currencyCode="AUD"
        decimalPlace="2">42.68</Amount>
      <Text>42.68</Text>
    </Base>
    <Equivalent>
      <Amount
        currencyCode="USD"
        decimalPlace="2">34.40</Amount>
      <Text>34.40</Text>
    </Equivalent>
    <Total>
      <Amount
        currencyCode="USD"
        decimalPlace="2">34.40</Amount>
      <Text>34.40</Text>
    </Total>
  </New>
</Amounts>
<Payment
  sequence="1" type="CA">
  <Total>
    <Amount
      currencyCode="USD"
      decimalPlace="2">34.40</Amount>
    </Total>
  </Total>

```

```

    <Other>
      <Details>Intentionally left empty</Details>
    </Other>
  </Payment>
  <Remark>
    <Endorsements sequence="1">REFUNDABLE</Endorsements>
  </Remark>
</ElectronicMiscDocument>
</DocumentDetailsDisplay>
</GetElectronicDocumentRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

#### 4.3.4.2.5 Example EMD-S Green Screen Display

```

WEMD*T6073010532202«
ELECTRONIC MISCELLANEOUS DOCUMENT RECORD
INV:                CUST:                PNR:BDMVAF
EMD:6073010532202   ISSUED:15MAR18      PCC:61N1   IATA:99999992
NAME:SALAMA/SAMHAR          FF:EY100097448780
NAME REF:1234567890          TOUR ID:
FOP: CASH

DOM/INT: I           SETTLEMENT AUTHORIZATION:
RFIC:D/FINANCIAL IMPACT

AIR EXTRA:BOOKING FEE
CPN:1   SVC CITIES:LAX          SVC DATE:      QTY:1      STAT:OPEN

PRESENT TO:EY AIRLINE
PRESENT AT:EY AIRPORT

BASE AMT:USD50.00          EMD TTL:USD148.00
TAX AMT:USD98.00          EQUIV AMT:

ENDORSEMENTS/RESTRICTIONS:TESTING ENDORSEMENT LINE IN EMDS
EMD TYPE:STAND-ALONE
REFERENCED ELECTRONIC TICKET NUM:

```

#### 4.3.4.2.6 Example EMD-S WS Response

```

<?xml version="1.0" encoding="UTF-8"?>

```

```

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRS
      Version="2.0.0"
      xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
      <STL:STL_Header.RS
        timeStamp="2018-03-15T15:47:06"
        xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl">
        <STL:OrchestrationID
          seq="0">TKTHLI700-3229-1974375660-1521146826120-32-edoc:0</STL:OrchestrationID>
        <STL:Results>
          <STL:Success>
            <STL:System>TKT-DS</STL:System>
            <STL:Source>EDOC</STL:Source>
          </STL:Success>
        </STL:Results>
      </STL:STL_Header.RS>
      <Agent
        sine="ATW"
        type="GDS">
        <TicketingProvider>EY</TicketingProvider>
        <WorkLocation>61N1</WorkLocation>
        <HomeLocation>LHR</HomeLocation>
        <IataNumber>99999992</IataNumber>
      </Agent>
      <TransactionInfo
        sequence="1"
        type="SALE">
        <LocalDateTime>2018-03-15T15:47:06.43-05:00</LocalDateTime>
        <SystemDateTime>2018-03-15T15:47:06.43-05:00</SystemDateTime>
        <SystemProvider>1S</SystemProvider>
      </TransactionInfo>
      <DocumentDetailsDisplay>
        <ElectronicMiscDocument
          number="6073010532202"
          type="EMD">
          <Details>
            <Reservation>
              <Sabre>BDMVAF</Sabre>
            </Reservation>
            <LocalIssueDateTime>2018-03-15T00:00:00.000-00:00</LocalIssueDateTime>
            <ItineraryType>I</ItineraryType>
            <ElectronicMiscType>S</ElectronicMiscType>
            <ReasonForIssue>D</ReasonForIssue>
          </Details>
          <Customer>
            <Traveler>
              <Name>SALAMA/SAMHAR</Name>
              <FirstName>SAMHAR</FirstName>
              <LastName>SALAMA</LastName>
              <ExternalNumber>1234567890</ExternalNumber>
            </Traveler>
          </Customer>

```

```

<Miscellaneous>
  <Fee
    coupon="1" description="BOOKING FEE"
    quantity="1">
    <Total>
      <Amount currencyCode="USD"
        decimalPlace="2">50.00</Amount>
      <Text>50.00</Text>
    </Total>
    </Fee>
  <OptionalService
    reason="997">
    <FeeProvider>EY</FeeProvider>
    <PresentTo>EY AIRLINE</PresentTo>
    <PresentAt>EY AIRPORT</PresentAt>
  </OptionalService>
  <CouponDetails
    coupon="1" description="BOOKING FEE"
    quantity="1">
    <MarketingProvider>EY</MarketingProvider>
    <StartLocation>LAX</StartLocation>
    <CouponStatus>OPEN</CouponStatus>
  </CouponDetails>
  <Affinity>
    <FrequentFlyer>
      <Provider>EY</Provider>
      <Number>100097448780</Number>
    </FrequentFlyer>
  </Affinity>
</Miscellaneous>
<Amounts>
  <New>
    <Base>
      <Amount
        currencyCode="USD"
        decimalPlace="2">50.00</Amount>
      <Text>50.00</Text>
    </Base>
    <TotalTax>
      <Amount
        currencyCode="USD"
        decimalPlace="2">98.00</Amount>
      <Text>98.00</Text>
    </TotalTax>
    <Total>
      <Amount
        currencyCode="USD"
        decimalPlace="2">148.00</Amount>
      <Text>148.00</Text>
    </Total>
  </New>
</Amounts>
<Taxes>
  <New>
    <Tax

```

```

        code="AA" sequence="1">
        <Amount currencyCode="USD"
        decimalPlace="2">10.00</Amount>
        <Text>10.00</Text>
    </Tax>
    <Tax
        code="BB" sequence="2">
        <Amount currencyCode="USD"
        decimalPlace="2">28.00</Amount>
        <Text>28.00</Text>
    </Tax>
    <Tax
        code="CC" sequence="3">
        <Amount currencyCode="USD"
        decimalPlace="2">48.00</Amount>
        <Text>48.00</Text>
    </Tax>
    <Tax
        code="DD" sequence="4">
        <Amount currencyCode="USD"
        decimalPlace="2">12.00</Amount>
        <Text>12.00</Text>
    </Tax>
</New>
</Taxes>
<Payment
    sequence="1" type="CA">
    <Total>
    <Amount
        currencyCode="USD"
        decimalPlace="2">148.00</Amount>
    </Total>
    <Other>
        <Details>Intentionally left empty</Details>
    </Other>
</Payment>
<Remark>
    <Endorsements sequence="1">TESTING ENDORSEMENT LINE IN EMDS</Endorsements>
</Remark>
</ElectronicMiscDocument>
</DocumentDetailsDisplay>
</GetElectronicDocumentRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### 4.3.5 Diagnostics Element

---

The Diagnostics element is reserved for internal use

### 4.3.6 Error Results Attributes and Elements

---

If the system is unable to return the document details, instead of sending the **Success** element in the Header, it will return the **Error** element.

The **Error** element with either contain,

1. just the **type** and **status** *attributes*

*\*or\**

2. the **SystemSpecificResults** element which has child elements - **ErrorMessage** and **ShortText**

#### 4.3.6.1 Error “type” Attributes

---

The Error element contains the type and status attributes.

| Attribute | Description                         | Schedule Change Path                                             |
|-----------|-------------------------------------|------------------------------------------------------------------|
| Type      | Error from the internal application | /GetElectronicDocumentRS/STL:STL_Header.RS/Results/Error/@type   |
| Status    | Impact of the error on processing   | /GetElectronicDocumentRS/STL:STL_Header.RS/Results/Error/@status |

#### 4.3.6.2 SystemSpecificResults Element and Child Elements

The SystemSpecificResults element contains the ErrorMessage and ShortText child elements.

| Element      | Description                                                   | Example                                                                                                                    | XPath                                                                                       |
|--------------|---------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| ErrorMessage | Contains the text of the message and a message code attribute | Unable to determine airline for document beginning with 060. Please verify 13-digit document number and try request again. | /GetElectronicDocumentRS/STL:STL_Header.RS/Results/Error/SystemSpecificResults/ErrorMessage |
| ShortText    | Contains text which identifies the category of the error.     | Internal error                                                                                                             | /GetElectronicDocumentRS/STL:STL_Header.RS//Results/Error/SystemSpecificResults/ShortText   |

#### ErrorMessage Attribute

The ErrorMessage element has one attribute to uniquely identify the warning or error with a specific code.

| Attribute | Description                                  | XPath                                                                                             |
|-----------|----------------------------------------------|---------------------------------------------------------------------------------------------------|
| Code      | Unique code of the error or warning message. | /GetElectronicDocumentRS/STL:STL_Header.RS/Results/Error/SystemSpecificResults/ErrorMessage/@code |

# Troubleshooting

## 5.1 Introduction

This section provides a list of error messages, their causes, and XML examples of some of them.

There are a few steps that the TKT\_ElectronicDocumentServicesRQ Web service makes during transaction processing and at each of these steps, an error message will be returned if the system detects the problem.

These steps are:

- Checking TJR for IATA number and branch access to determine if the user is entitled to view the electronic document.
- If requesting by document reference number, determine that a PNR is present in the user's Sabre session.
- Verifying the document in the RQ exists in the airline carrier's database.
- EPR check for CCVIEW to determine if the user is allowed to view a credit card number unmasked.

## 5.2 List of Error Messages

If the system is unable to process the document display request for any reason, the background processes will stop, and an error message will be returned in the Web service response.

The following table contains error messages that can be returned during an electronic document display from the TKT\_ElectronicDocumentServicesRQ Web service.

| Error Code | Error Name                                 | Error Element returned in RS                                                                                                                                  | Short Error Message |
|------------|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|
| 97008      | EDOC_WS_PNR_CALL_FAILURE                   | No PNR in AAA                                                                                                                                                 | Error in PNR Client |
| 97009      | EDOC_WS_PNR_CALL_FAILURE                   | Call to PNR subsystem failed                                                                                                                                  | Error in PNR Client |
| 97010      | EDOC_WS_DOCUMENT_TYPE_RESOLVER_FAILED      | Unable to retrieve document type from an internal subsystem.                                                                                                  | n/a                 |
| 97011      | EDOC_WS_VALIDATING_CARRIER_RESOLVER_FAILED | Internal error: Unable to determine airline for document beginning with [3 digit airline code]. Please verify 13-digit document number and try request again. | Internal error      |

| Error Code | Error Name                            | Error Element returned in RS                                                                                                                                                                                                    | Short Error Message                |
|------------|---------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| 97012      | EDOC_WS_HUB_REQUEST_PROCESSING_ERROR  | T2 HUB Request cannot be null                                                                                                                                                                                                   | n/a                                |
| 97013      | EDOC_WS_HUB_RESPONSE_PROCESSING_ERROR | Unable to find the document for the document number in your request. Please verify that your request includes a valid 13-digit document number or a valid line number from the *T field of the PNR and try your request again.  | 401                                |
| 97014      | EDOC_WS_NOT_SUBSCRIBER_SESSION        | Request security validation failed: Unable to process request due to user credentials. Service is available to Sabre travel agency subscribers only.                                                                            | Request security validation failed |
| 97015      | EDOC_WS_EDOC_REQUEST_PROCESSING_ERROR | Internal error: Unable to process request. Use either the 13-digit document number or document reference number from *T in the PNR, and try your request again.                                                                 | Internal error                     |
| 97015      | EDOC_WS_EDOC_REQUEST_PROCESSING_ERROR | Internal error: Unable to find a document for the document reference number in your request. Please check document reference number (*T in the PNR), or make request with 13-digit document number.                             | Internal error                     |
| 97019      | EDOC_WS_VIEWERSHIP_VALIDATION_ERROR   | Request security validation failed: Unable to process request due to security rules. Service is available to the location (Pseudo City Code or PCC) which issued the document, or an authorized branch of the issuing location. | Request security validation failed |

## 5.3 Example of Error Responses

This section provides several example requests and responses when an error condition is present.

### 5.3.1 Non-Electronic Document Retrieval Error

TKT\_ElectronicDocumentServicesRQ will only retrieve electronic documents. If the user attempts a request with a non-electronic document – e.g. a “paper ticket”, an error will be returned in the WS response.

### 5.3.1.1 Unexpected Subsystem Response

---

#### Request

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MessageHeader
      ns1:version=""
      xmlns:ns1="http://www.ebxml.org/namespaces/messageHeader">
      <ns1:From>
        <ns1:PartyId>999999</ns1:PartyId>
      </ns1:From>
      <ns1:To>
        <ns1:PartyId>123123</ns1:PartyId>
      </ns1:To>
      <ns1:CPAId>AAT</ns1:CPAId>
      <ns1:ConversationId>ABC123</ns1:ConversationId>
      <ns1:Service>TKT_ElectronicDocumentServicesRQ</ns1:Service>
      <ns1:Action>TKT_ElectronicDocumentServicesRQ</ns1:Action>
      <ns1:MessageData>
        <ns1:MessageId/>
        <ns1:Timestamp/>
      </ns1:MessageData>
    </ns1:MessageHeader>
    <ns1:Security
      xmlns:ns1="http://schemas.xmlsoap.org/ws/2002/12/secext">
      <ns1:BinarySecurityToken>Shared/IDL:IceSess\SessMgr:1\0.IDL/Common/!ICESMS\STSA!ICE
      SMSLB\STS.LB!-3168880030685001073!1530160!0</ns1:BinarySecurityToken>
    </ns1:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRQ
      Version="2.0.0"
      xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
      <ns1:STL_Header.RQ
        xmlns:ns1="http://www.sabre.com/ns/Ticketing/EDocStl"/>
      <ns2:POS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDocStl">
        <ns2:Company>1S</ns2:Company>
        <ns2:Pseudo>AM20</ns2:Pseudo>
      </ns2:POS>
      <SearchParameters>
        <DocumentNumber>8900727271250</DocumentNumber>
      </SearchParameters>
    </GetElectronicDocumentRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

#### Response

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
```

```

<SOAP-ENV:Body>
  <GetElectronicDocumentRS
    Version="2.0.0"
    xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
    <STL:STL_Header.RS
      messageID="TKTHLI701-18250-1772263587-1520958747360-281-edoc"
      timeStamp="2018-03-13T11:32:27"
      xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl">
      <STL:OrchestrationID
        seq="0">TKTHLI701-18250-1772263587-1520958747360-281-edoc</STL:OrchestrationID>
      <STL:Results>
        <STL:Error
          status="Incomplete" timeStamp="2018-03-13T11:32:27"
          type="Application">
          <STL:Source>EDOC WS</STL:Source>
          <STL:System>TKT-WS</STL:System>
          <STL:SystemSpecificResults>
            <STL:ErrorMessage
              code="21213">Unexpected subsystem response: T2 HUB Response
Validation</STL:ErrorMessage>
            <STL:ShortText>Unexpected subsystem response</STL:ShortText>
          </STL:SystemSpecificResults>
          <STL:SystemSpecificResults>
            <STL:ErrorMessage code="97013">No agreement between two carriers : 1S and
XD</STL:ErrorMessage>
            <STL:ShortText>9001</STL:ShortText>
          </STL:SystemSpecificResults>
        </STL:Error>
      </STL:Results>
    </STL:STL_Header.RS>
  </GetElectronicDocumentRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### Reason

The user is requesting a non-electronic document. The service is able to perform some validations but fails a validation in a subsystem.

### Solution

Resubmit request with a valid 13 digit electronic document number, or a valid ticket reference number for an electronic document.

## 5.3.2 Unable to Determine Validating Carrier

---

If the user manually enters a 13 digit electronic document number, but the first 3 digits are not associated with an airline, the system will return an error.

### Request

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"

```

```

    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header>
  <ns1:MessageHeader
    ns1:version=""
    xmlns:ns1="http://www.ebxml.org/namespaces/messageHeader">
    <ns1:From>
      <ns1:PartyId>999999</ns1:PartyId>
    </ns1:From>
    <ns1:To>
      <ns1:PartyId>123123</ns1:PartyId>
    </ns1:To>
    <ns1:CPAId>AAT</ns1:CPAId>
    <ns1:ConversationId>ABC123</ns1:ConversationId>
    <ns1:Service>TKT_ElectronicDocumentServicesRQ</ns1:Service>
    <ns1:Action>TKT_ElectronicDocumentServicesRQ</ns1:Action>
    <ns1:MessageData>
      <ns1:MessageId/>
      <ns1:Timestamp/>
    </ns1:MessageData>
    </ns1:MessageHeader>
  <ns1:Security
    xmlns:ns1="http://schemas.xmlsoap.org/ws/2002/12/secext">

<ns1:BinarySecurityToken>Shared/IDL:IceSess\\SessMgr:1\\.0.IDL/Common/!ICESMS\\STSB!ICE
SMSLB\\STS.LB!-3186557900998506619!1244673!0</ns1:BinarySecurityToken>
  </ns1:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetElectronicDocumentRQ
    Version="2.0.0"
    xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
    <ns1:STL_Header.RQ
      xmlns:ns1="http://www.sabre.com/ns/Ticketing/EDocStl"/>
    <ns2:POS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDocStl">
      <ns2:Company>1S</ns2:Company>
      <ns2:Pseudo>61N1</ns2:Pseudo>
    </ns2:POS>
    <SearchParameters>
      <DocumentNumber>0601995160777</DocumentNumber>
    </SearchParameters>
  </GetElectronicDocumentRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

## Response

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <GetElectronicDocumentRS
    Version="2.0.0"
    xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
    <STL:STL_Header.RS
      messageID="TKTHLI702-15752-578892685-1516643524387-32-edoc"
      timeStamp="2018-01-22T11:52:04"

```

```

    xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl">
<STL:OrchestrationID
  seq="0">TKTHLI702-15752-578892685-1516643524387-32-edoc</STL:OrchestrationID>
<STL:Results>
  <STL:Error
    status="Incomplete" timeStamp="2018-01-22T11:52:04"
    type="Application">
    <STL:Source>EDOC WS</STL:Source>
    <STL:System>TKT-WS</STL:System>
    <STL:SystemSpecificResults>
      <STL:ErrorMessage code="97011">Internal error: Unable to determine airline for
document beginning with 060. Please verify 13-digit document number and try request
again.</STL:ErrorMessage>
      <STL:ShortText>Internal error</STL:ShortText>
    </STL:SystemSpecificResults>
    </STL:Error>
  </STL:Results>
</STL:STL_Header.RS>
</GetElectronicDocumentRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

**Reason:**

The user is requesting a document with an incorrect validating carrier number

**Solution:**

Resubmit request with a valid 3 digit validating airline number.

### 5.3.3 USG Authentication Errors

---

USG is Sabre's Universal Service Gateway. This type of error is returned when authorization to access Web services fails for user.

#### 5.3.3.1 Security Violations

---

##### 5.3.3.1.1 Invalid or Expired Security Token

---

```

Error
?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://schemas.xmlsoap.org/soap/envelope/">
  <soap-env:Header>
    <eb:MessageHeader xmlns:eb="http://www.ebxml.org/namespaces/messageHeader"
eb:version="1.0" soap-env:mustUnderstand="1">
      <eb:From>

```

```

    <eb:PartyId eb:type="URI">123123</eb:PartyId>
  </eb:From>
  <eb:To>
    <eb:PartyId eb:type="URI">999999</eb:PartyId>
  </eb:To>
  <eb:CPAId>HDQ</eb:CPAId>
  <eb:ConversationId>SYSTEM-T2DAILY-SYSTEM</eb:ConversationId>
  <eb:Service>TKT_ElectronicDocumentServicesRQ</eb:Service>
  <eb:Action>ErrorRS</eb:Action>
  <eb:MessageData>
    <eb:MessageId>180743552951180150</eb:MessageId>
    <eb:Timestamp>2018-03-02T15:21:35</eb:Timestamp>
    <eb:RefToMessageId>mid:20001209-133003-
2333@clientofsabre.com</eb:RefToMessageId>
  </eb:MessageData>
</eb:MessageHeader>
  <wsse:Security xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/12/secext">
    <wsse:BinarySecurityToken valueType="String"
EncodingType="wsse:Base64Binary">${ICE_SECURITY_TOKENM}</wsse:BinarySecurityToken>
  </wsse:Security>
</soap-env:Header>
<soap-env:Body>
  <soap-env:Fault>
    <faultcode>soap-env:Client.InvalidSecurityToken</faultcode>
    <faultstring>Invalid or Expired binary security token:
${ICE_SECURITY_TOKENM}</faultstring>
    <detail>
      <StackTrace>com.sabre.universalservices.base.session.SessionException:
errors.session.USG_INVALID_SECURITY_TOKEN</StackTrace>
    </detail>
  </soap-env:Fault>
</soap-env:Body>
</soap-env:Envelope>

```

## Reason

The system is unable to authenticate the security token

## Solution

Contact your Sabre Account Director. Please remember to provide your user ID, domain, organization and the environment you use.

### 5.3.3.1.2 No Branch Access

---

## Error

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRS
      Version="2.0.0"
      xmlns="http://www.sabre.com/ns/Ticketing/EDoc">
      <STL:STL_Header.RS

```

```

messageID="TKTHLI702-8445-1810146969-1520009516659-97-edoc"
timestamp="2018-03-02T10:51:56"
xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl">
<STL:OrchestrationID
  seq="0">TKTHLI702-8445-1810146969-1520009516659-97-edoc</STL:OrchestrationID>
<STL:Results>
  <STL:Error
    status="NotProcessed" timestamp="2018-03-02T10:51:56"
    type="Validation">
    <STL:Source>EDOC WS</STL:Source>
    <STL:System>TKT-WS</STL:System>
    <STL:SystemSpecificResults>
      <STL:ErrorMessage code="97019">Request security validation failed: Unable to
process request due to security rules. Service is available to the location (Pseudo
City Code or PCC) which issued the document, or an authorized branch of the issuing
location.</STL:ErrorMessage>
      <STL:ShortText>Request security validation failed</STL:ShortText>
    </STL:SystemSpecificResults>
  </STL:Error>
</STL:Results>
</STL:STL_Header.RS>
</GetElectronicDocumentRS>
</SOAP-ENV:Body>

```

#### Reason

User PCC does not have permission to retrieve the document specified in the RQ – security authorization failure.

#### Solution

Verify if the values of search parameters fields are correct and if needed, amend them and try again. If you need additional assistance, use your provided Sabre Web services contacts. If you are unsure, contact your Sabre Account Director. Please remember to provide your user ID, domain, organization and the environment you use.

### 5.3.4 Throttled Request/Response

---

In the event that there are too many requests for the system to handle, a user may experience a “throttled request/response”.

#### Error

```

<GetElectronicDocumentRS xmlns:ns2="http://www.sabre.com/ns/Ticketing/EDoc/1.0"
  xmlns="http://www.sabre.com/ns/Ticketing/EDocStl"
  Version="2.0.0">
  <Header>
    <OrchestrationID seq="0">TKTHLI702-16241-13246280-
1496255121939-8-edoc</OrchestrationID>
    <Results>
      <Error status="NotProcessed" timestamp="2018-01-
15T13:25:28.247" type="Application">
        <Source>EDOC</Source>
        <System>TKT-DS</System>
      </Error>
    </Results>
  </Header>
</GetElectronicDocumentRS>

```

```

        <Header>
        <TransactionInfo>
        <TransactionInfo sequence="1">
        <LocalDateTime>2017-12-20T15:35:40.289-06:00</LocalDateTime>
        <SystemDateTime>2017-12-20T15:35:40.289-06:00</SystemDateTime>
        <SystemProvider>1S</SystemProvider>
        </TransactionInfo>

```

## Reason

The service is at peak usage.

## Solution

Wait for a few minutes and try again. If the problem persists, contact your Sabre Web services support. If you are unsure, contact your Sabre Account Director.

## 5.3.5 Invalid Service Version

---

### Error

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRS xmlns="http://www.sabre.com/ns/Ticketing/EDoc"
Version="2.0.0">
      <STL:STL_Header.RS
xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl" messageID="TKTHLI702-8445-
1810146969-1520004172154-42-edoc" timeStamp="2018-03-02T09:22:52">
        <STL:OrchestrationID seq="0">TKTHLI702-8445-1810146969-
1520004172154-42-edoc</STL:OrchestrationID>
        <STL:Results>
          <STL:Error status="NotProcessed" timeStamp="2018-03-
02T09:22:52" type="Validation">
            <STL:Source>EDOC WS</STL:Source>
            <STL:System>TKT-WS</STL:System>
            <STL:SystemSpecificResults>
              <STL:ErrorMessage code="21212">Invalid
service version specified in XML request: XML schema version 2.0.0 is not
supported</STL:ErrorMessage>
              <STL:ShortText>Invalid service version
specified in XML request</STL:ShortText>
            </STL:SystemSpecificResults>
          </STL:Error>
        </STL:Results>
      </STL:STL_Header.RS>
    </GetElectronicDocumentRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

### Reason

The version specified in the Web service request is not supported.

## Solution

Change the version to the latest version.

### 5.3.6 Invalid Request XML Format

---

#### 5.3.6.1 Invalid XML Namespace in RQ

---

##### Error

```
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://schemas.xmlsoap.org/soap/envelope/">
  <soap-env:Header/>
  <soap-env:Body>
    <soap-env:Fault>
      <faultcode>soap-env:Client.InvalidEbXmlMessage</faultcode>
      <faultstring>Unable to create envelope from given source: Error on
line 34 of document : The prefix "STM" for element "STM:Compay" is not bound. Nested
exception: The prefix "STM" for element "STM:Compay" is not bound.</faultstring>
      <detail>
        <StackTrace>javax.xml.soap.SOAPException: Unable to create
envelope from given source: Error on line 34 of document : The prefix "STM" for
element "STM:Compay" is not bound. Nested exception: The prefix "STM" for element
"STM:Compay" is not bound.</StackTrace>
      </detail>
    </soap-env:Fault>
  </soap-env:Body>
</soap-env:Envelope>
```

##### Reason

The XML namespace prefix is not bound.

##### Solution

Fix namespace binding or use proper prefix

#### 5.3.6.2 Invalid XML Element Name in RQ

---

##### Error

```
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://schemas.xmlsoap.org/soap/envelope/">
  <soap-env:Header/>
  <soap-env:Body>
    <soap-env:Fault>
      <faultcode>soap-env:Client.InvalidEbXmlMessage</faultcode>
      <faultstring>Unable to create envelope from given source: Error on
line 34 of document : The end-tag for element type "STL:Compan" must end with a
'&gt;' delimiter. Nested exception: The end-tag for element type "STL:Compan" must end
with a '&gt;' delimiter.</faultstring>
      <detail>
```

```

        <StackTrace>javax.xml.soap.SOAPException: Unable to create
envelope from given source: Error on line 34 of document : The end-tag for element
type "STL:Compan" must end with a '&gt;' delimiter. Nested exception: The end-tag for
element type "STL:Compan" must end with a '&gt;' delimiter.</StackTrace>
    </detail>
  </soap-env:Fault>
</soap-env:Body>
</soap-env:Envelope>

```

### Reason

The XML request sent doesn't match the <<xxx>> XSD schema. The parameters provided in the search are invalid.

### Solution

Verify XML request with the XSD schema version you used and correct it.

If you need additional assistance, use your provided Sabre Web services contacts. If you are unsure, contact your Sabre Account Director.

## 5.3.7 Database Response: Not Found

---

### Error

```

SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <GetElectronicDocumentRS
xmlns="http://www.sabre.com/ns/Ticketing/EDoc" Version="2.0.0">
      <STL:STL_Header.RS
xmlns:STL="http://www.sabre.com/ns/Ticketing/EDocStl" messageID="TKTHLI702-8445-
1810146969-1520007872151-90-edoc" timeStamp="2018-03-02T10:24:32">
        <STL:OrchestrationID
seq="0">TKTHLI702-8445-1810146969-1520007872151-90-edoc</STL:OrchestrationID>
        <STL:Results>

<STL:Error status="Incomplete" timeStamp="2018-03-02T10:24:32" type="Application">

<STL:Source>EDOC WS</STL:Source>

<STL:System>TKT-WS</STL:System>

<STL:SystemSpecificResults>

<STL:ErrorMessage code="21213">Unexpected subsystem response: T2 HUB Response
Validation</STL:ErrorMessage>

<STL:ShortText>Unexpected subsystem response</STL:ShortText>

</STL:SystemSpecificResults>

<STL:SystemSpecificResults>

<STL:ErrorMessage code="97013">Unable to find the document for the document number in
your request. Please verify that your request includes a valid 13-digit document

```

```

number or a valid line number from the *T field of the PNR and try your request
again.</STL:ErrorMessage>

<STL:ShortText>401</STL:ShortText>

</STL:SystemSpecificResults>

</STL>Error>

                                                                    </STL:Results>
                                                                    </STL:STL_Header.RS>
</GetElectronicDocumentRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

## Reason

There is actually not an error, but a database response. The document requested was not found. The user may have entered an invalid PCC and document number combination in the search.

## Solution

Verify if the values of search parameters fields are correct and if needed, amend them and try again.

If you need additional assistance, use your provided Sabre Web services contacts. If you are unsure, contact your Sabre Account Director.

## 5.3.8 Airline Issued EMD without Traveler Name in Search

---

### Error

```

<GetElectronicDocumentRS
  Version="2.0.0"
  xmlns=http://www.sabre.com/ns/Ticketing/EDoc>
  <STL:STL_Header.RS
    messageID="LTXLD_TKT_05-39169-792444566-1628623823224-50103-edoc"
    timeStamp="2021-08-10T14:30:23"
    xmlns:STL=http://www.sabre.com/ns/Ticketing/EDocStl>
    <STL:OrchestrationID
      seq="0">LTXLD_TKT_05-39169-792444566-1628623823224-50103-
edoc</STL:OrchestrationID>
    <STL:Results>
      <STL>Error
        status="NotProcessed" timeStamp="2021-08-10T14:30:23"
        type="Validation">
          <STL:Source>EDOC WS</STL:Source>
          <STL:System>TKT-WS</STL:System>
          <STL:SystemSpecificResults>
            <STL:ErrorMessage code="97019">Request security validation failed: Unable to
process request due to security rules. Service is available to the location (Pseudo
City Code or PCC) which issued the document, or an authorized branch of the issuing
location.</STL:ErrorMessage>
            <STL:ShortText>Request security validation failed</STL:ShortText>

```

```
</STL:SystemSpecificResults>  
</STL:Error>  
</STL:Results>  
</STL:STL_Header.RS>  
</GetElectronicDocumentRS>
```

**Reason**

A traveler name was not entered for an Airline issued EMD.

**Solution**

Resubmit request with the passenger name for the requested EMD.