

# *Jeppesen Solution Integrator Overview*

**DOCUMENT VERSION 1.0**

**OCTOBER 1, 2014**



# ***Jeppesen Solution Integrator Overview***

**DOCUMENT VERSION 1.0**

**Copyright © 2014, Jeppesen, All Rights Reserved**

# Contents

---

<b>Figures</b>	<b>v</b>
<b>Tables</b>	<b>vii</b>
<b>Introduction</b>	<b>1</b>
<b>Getting Started</b> .....	<b>3</b>
Jeppesen Solution Integrator Core Technologies .....	3
Jeppesen Solution Integrator Minimum Software Requirements .....	4
Purpose .....	5
Audience .....	5
References .....	5
<b>Why Jeppesen Solution Integrator?</b> .....	<b>6</b>
The Challenge .....	6
The Solution .....	7
<b>Inter-Application Communication Architecture</b> .....	<b>8</b>
Message Bus Architecture .....	8
Product .....	9
Messages .....	11
Adapter Functionality .....	13
Database .....	13
<b>Messaging and Standardization</b> .....	<b>14</b>
Message Standards .....	14
Common Interfaces .....	15
Sample Messages .....	15
Using the AOC Bookshelf .....	16
<b>Jeppesen Solution Integrator Feature Overview</b> .....	<b>18</b>
<b>System Configuration</b> .....	<b>20</b>
<b>Glossary</b>	<b>21</b>



# Figures

---

<b>Introduction</b>	<b>1</b>
Figure 1. Typical Point-to-Point Integration .....	6
Figure 2. Jeppesen Solution Integrator - Simplified .....	7
Figure 3. System Architectural Components .....	8
Figure 4. Sample Products in Relation to Jeppesen Solution Integrator .....	9
Figure 5. Communication Between Products Using Jeppesen Solution Integrator .	11
Figure 6. Message Sent to Jeppesen Solution Integrator Through Adapter .....	13
Figure 7. Sample View of AOC Bookshelf .....	16





# ***Tables***

---

<b>Introduction</b>	<b>1</b>
Table -1: JSI Feature Overview and Description .....	18



# *Introduction*

---

---

**IMPORTANT** Due to the nature of software development, Jeppesen Solution Integrator Overview may contain information that has become outdated as the ongoing development of the application has progressed. It is important to understand that the information in this draft is subject to change in future revisions of this document.

---



## Getting Started

Jeppesen Solution Integrator (JSI) provides an enhanced interface, which facilitates communication between applications (also referred to as “products” in this guide) within an Airline Operations Center (AOC) - and their partner applications requiring messaging to exchange information. Jeppesen Solution Integrator does this by routing discreet messages between the applications through a flexible, service-oriented architecture using XML message sets and approved transports (HTTP/HTTPS, and JMS). For a detailed description of Jeppesen Solution Integrator, and client configuration and setup, please consult the *Jeppesen Solution Integrator User’s Guide*, and/or the *Jeppesen Solution Integrator Client Developer’s Guide*.

## Jeppesen Solution Integrator Core Technologies

### Enterprise Service Bus (ESB)

- Open-source Apache Synapse™
  - Allowing for quick development of JSI instead of building a custom ESB or relying on high cost commercial ESB
  - Open source technology provides low cost deployment options to customers

### Application Server

- JBoss®

### Operating System

- Linux®, Windows Server®

### Database

- Oracle®, PostgreSQL

### Transport

- HTTP, HTTPS, JMS (support for WebSphere® MQ and Tibco® EMS)

# Jeppesen Solution Integrator Minimum Software Requirements

## Operating Systems

- Red Hat<sup>®</sup> Enterprise Linux<sup>®</sup> Server 5 (64 bit)
- Windows Server<sup>®</sup> 2012 SE

## Application Server

- JBoss<sup>®</sup> 6 (JBoss-6.0.0.Final)

## Runtime Environment

- Oracle<sup>®</sup> JRE 1.6 (64 bit JVM) for JBoss<sup>®</sup> Application Server

## Browser Software

- Microsoft<sup>®</sup> Internet Explorer<sup>®</sup> 7.0
- Mozilla<sup>®</sup> Firefox<sup>®</sup> 5.0

## Databases

- Oracle<sup>®</sup> 11g
- PostgreSQL 9.2

## Level 1 Hardware Configuration

- Based on approximately 5,000 messages per day, maintaining a 30 day history
  - One (1) Intel<sup>®</sup> Dual Core Processor at 3GHz
  - 4GB Memory
  - 100GB Application Server & Database Disk Storage

## Level 2 Hardware Configuration

- Based on approximately 30,000 messages per day, maintaining a 30 day history
  - Two (2) Intel<sup>®</sup> Xeon<sup>®</sup> Quad Core Processors at 3GHz
  - 8GB Memory
  - 100GB Application Server Disk Storage
  - 30GB Database Disk Storage

## Purpose

This document provides an overview of Jeppesen Solution Integrator. It is intended to provide software and system engineers with an overview of JSI architecture and messaging standards and direct them to other documents in the JSI documentation set.

---

**NOTE** It is not the intention of this document to provide the level of detail needed to install, upgrade, administer, or configure JSI (see [“References”](#) below).

---

The following topics are discussed:

- [“Why Jeppesen Solution Integrator?”](#) – The widespread industry challenge of integrating multiple applications within an AOC
- [“Inter-Application Communication Architecture”](#) – Key components of Jeppesen Solution Integrator architecture including systems, messages, and bus
- [“Jeppesen Solution Integrator Feature Overview”](#) – Jeppesen Solution Integrator message translation and message delivery features
- [“System Configuration”](#) – Jeppesen Solution Integrator defined

## Audience

This document is published for Jeppesen customers, partners, and integration experts who define messages to be routed through Jeppesen Solution Integrator.

## References

The following documents support JSI:

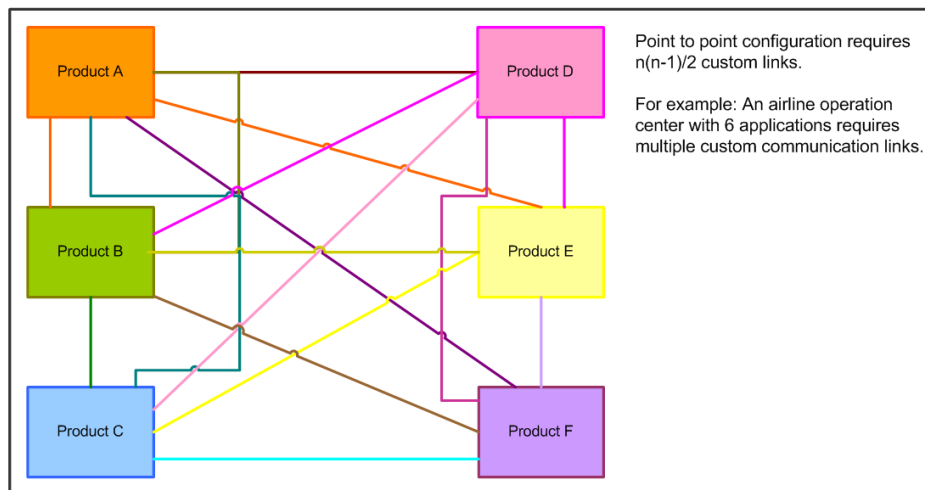
- *Jeppesen Solution Integrator Installation and Upgrade Guide*
- *Jeppesen Solution Integrator User’s Guide*
- *Jeppesen Solution Integrator Client Developer’s Guide*

## Why Jeppesen Solution Integrator?

Jeppesen Solution Integrator provides capabilities that greatly simplify communication between applications in an AOC.

### The Challenge

An AOC consists of a number of products/applications, and each application must be able to communicate with the other applications. This is a significant challenge because these applications often communicate via different standards and technologies. This results in an extremely complex integrated environment, consisting of numerous custom communication links between applications (see [Figure 1](#)). These custom integration efforts only work for these specific products and their versions. When upgrades are required, a new custom development effort is often the consequence, resulting in higher costs, long production cycles, and decreased operational efficiencies.



**Figure 1.** Typical Point-to-Point Integration

The typical non-JSI configuration introduces the following challenges:

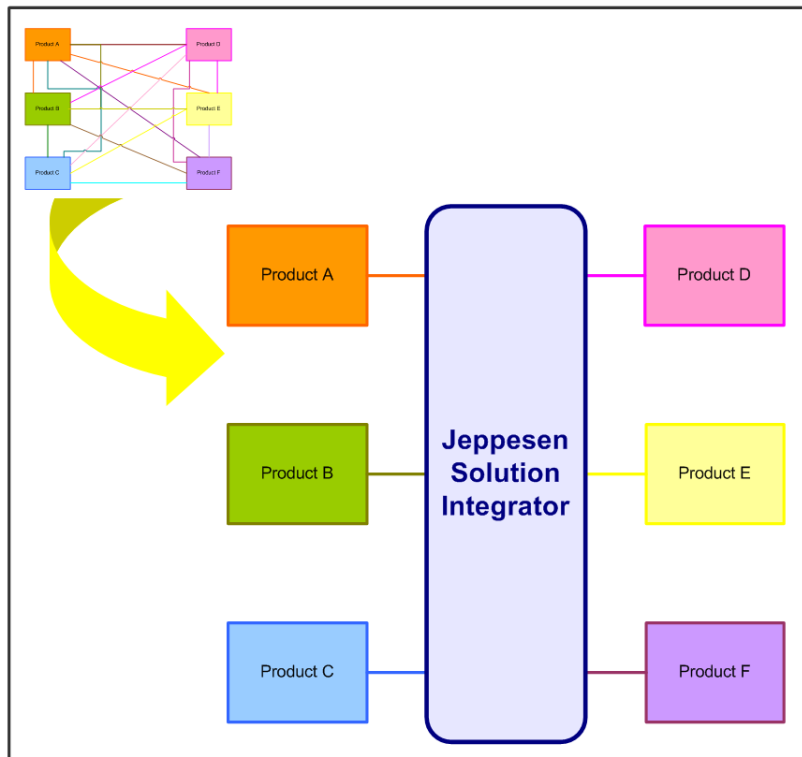
- Integrations are complex, consisting of multiple custom connections and applications.
- Implementations are expensive due to installation, setup, and maintenance of numerous connections.



- Each custom communication point introduces another single point of failure, which can result in costly outages and require more expensive backup solutions.
- Upgrades are complex, and changing any application can have costly ripple effects.

## The Solution

Jeppesen Solution Integrator introduces a way to more efficiently (and simply) integrate the applications in their environment by allowing Jeppesen Solution Integrator to receive, translate, and route messages between associated applications (see [Figure 2](#)).



**Figure 2.** Jeppesen Solution Integrator - Simplified

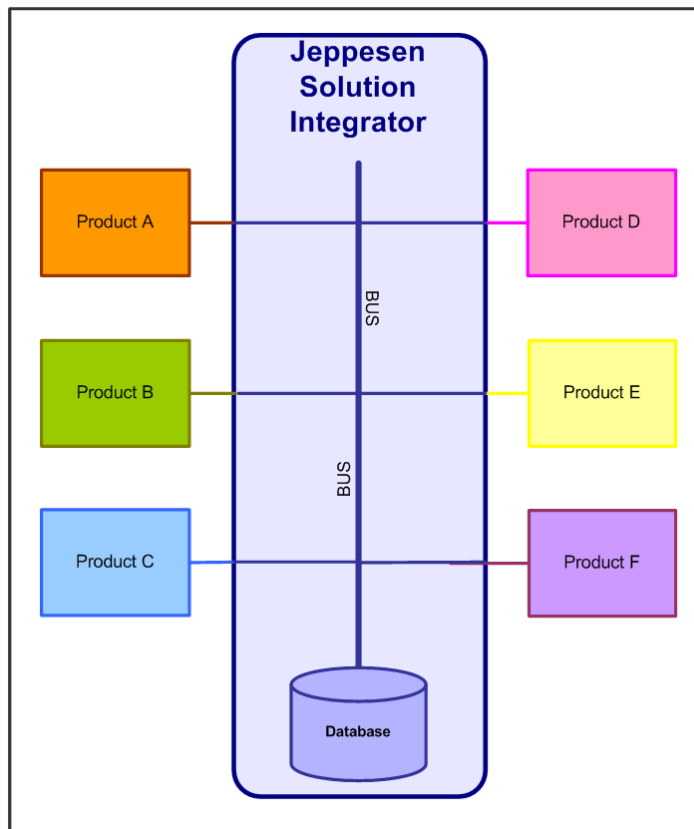
Jeppesen Solution Integrator simplifies the process by greatly reducing the number of custom links between different products. Rather than requiring direct (and custom) communication between each product, each product must only communicate with JSI, which then facilitates delivery of messages between products. This architectural model allows for more seamless integration with new or upgraded products into the operational environment.

# Inter-Application Communication Architecture

Jeppesen Solution Integrator routes messages to and from different products in an AOC. To do this, individual products rely on Jeppesen Solution Integrator to receive and standardize messages and in-turn process the messages by routing them as required.

## Message Bus Architecture

Individual products/applications communicate via Jeppesen Solution Integrator using bus type technology (see [Figure 3](#)), common transports, and data format (serialized objects).



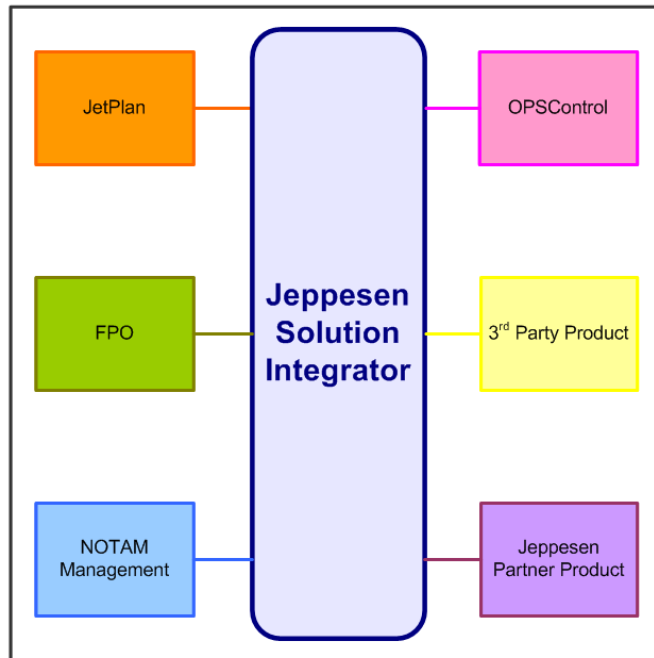
**Figure 3.** System Architectural Components

Unlike a point-to-point environment requiring numerous interfaces as described previously, centralized bus architecture allows the flexible reuse of business components using a service-oriented architecture.

This approach makes it easy to dynamically connect, mediate, and control services and interactions such as reconciliation of incompatible protocols, data formats, and interaction patterns of connected services. By bridging these differences with the built-in mediation functionality of a bus architecture, it is much easier to rapidly link communications among diverse services.

## Product

Jeppesen Solution Integrator can be used to integrate any product, regardless of vendor. Though the products are outside of the JSI architecture, it is helpful to understand their orientation in reference to other architectural features. [Figure 4](#) illustrates a few examples of products including Jeppesen’s JetPlan, NOTAM Management, and OPSControl.



**Figure 4.** Sample Products in Relation to Jeppesen Solution Integrator

Jeppesen partner products include applications designed by companies that Jeppesen has partner agreements with. Jeppesen ensures that all integration efforts include compatibility and testing with JSI. Third party products include applications other than Jeppesen or Jeppesen Partner applications.

Jeppesen Solution Integrator allows the user to set the following parameters when defining a product (see *Jeppesen Solution Integrator User's Guide* for details):

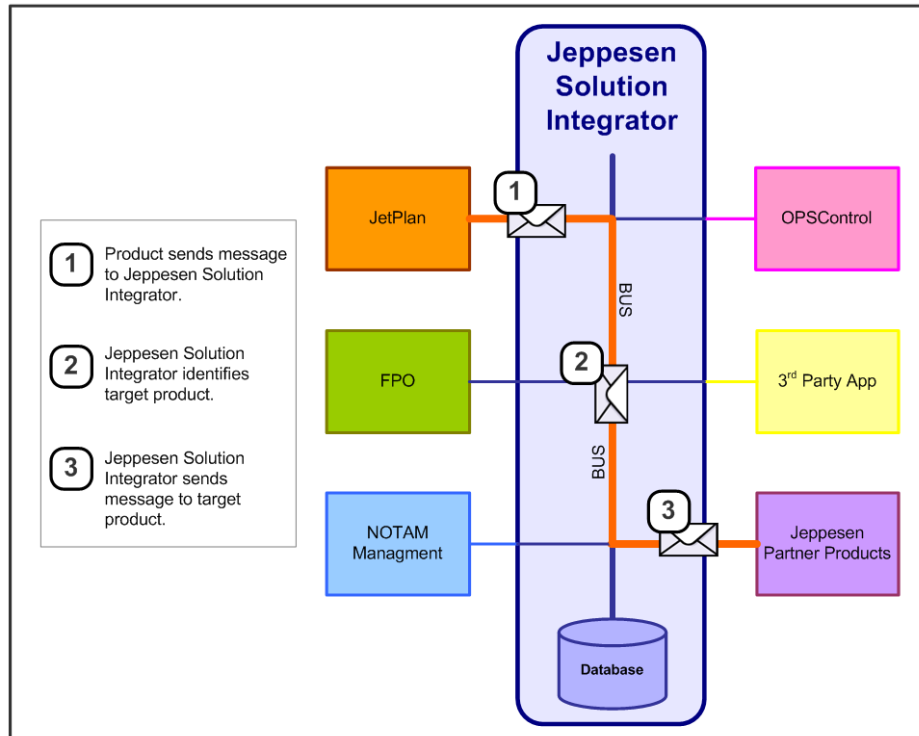
- **Message Format** - Plain XML, or SOAP
- **Throttling Time** - The minimum delay time in milliseconds between two publish/subscribe message deliveries to a product
- **Maximum Number of Retries** - allows the user to designate the maximum number of retries before placing a message into a Failed status
- **Number of Delivery Threads** - The number of delivery threads used to deliver non-FIFO messages
- **Delay Between Retries** - The designated period of time between the initial message processing attempt and the following re-attempt
- **Failed Status Duration** - The value that indicates how long the Publish/Subscribe messages in a specific queue are eligible for processing and delivery. When a specified Failed Status Duration has passed for a message, the message can no longer be processed and is assigned the Undeliverable status.

Additionally, Jeppesen Solution Integrator allows for Content Based Routing (CBR), Cross-Referencing, FIFO messaging, message delivery windows, and the inclusion of adapters.

Each message is stored in a database for audit and retransmission purposes, and Jeppesen Solution Integrator provides for querying and viewing of messages sent between applications by way of the Jeppesen Solution Integrator incorporated Message Store. The Message Store offers the user a way to view all messages being processed or processed through the system. Additionally, the Message Store allows for querying and viewing of messages sent between applications, and in some cases, the processing steps Jeppesen Solution Integrator has taken while processing the message - which can be invaluable in troubleshooting a failed message.

## Messages

Integrated systems must be able to send messages between applications using protocols understood by each party. In a standard point-to-point integration scheme, messages must pass between products using numerous custom links. Jeppesen Solution Integrator eliminates the need for these custom links by collecting, routing and distributing messages from each product.



**Figure 5.** Communication Between Products Using Jeppesen Solution Integrator

[Figure 5](#) depicts a message sent between the JetPlan product to the Jeppesen Partner Product traversing the JSI bus to be delivered.

JSI Message type can be categorized in one of two ways; Jeppesen defined messages (also referred to as “canned messages”), and custom user defined messages (also referred to as “customized messages”), in either case it is important that messages be well formed XML messages.

As each message enters Jeppesen Solution Integrator, the message is translated into a standard XML message format, which is routed to the target product. Because each product integrated with JSI understands the standardized messages, any product can send a message to any other product without requiring a custom link.

When selecting a transport for a particular application/product it is important that the correct transport be configured in JSI for the application - or the message will fail to process or be delivered. Currently, JSI can be configured to use the HTTP/HTTPS, or JMS (WebSphere® MQ, and Tibco® JNDI Service Provider) transport.

Jeppesen Solution Integrator Content Based Routing (CBR) offers users added routing functionality by allowing for distribution of a message to a recipient, or multiple recipients based on criteria contained within the message. CBR works by comparing the value in the message XPath against a content based routing rule XPath to determine if the message meets the criteria specified in the rule. If the message matches the criteria, the message processing is completed and the message is delivered. CBR can also be used to route a message to specific recipients by narrowing the criteria to meet the intended recipient while screening out the unintended recipients. For example, if a message only applies to a particular type of aircraft (777) or aircraft configuration, a CBR rule such as “If Aircraft Type matches 777 ...”, may be used. For details Content Based Routing use and associated rules see *Jeppesen Solution Integrator User’s Guide*.

During message processing, Jeppesen Solution Integrator makes optional message validation possible by accessing the message XSD, and validating the message XML against it. Each Jeppesen defined messages is loaded with an associated XSD and therefore validation is assured if selected, however, user defined custom messages need to have associated XSDs built and loaded in order to enable the validation functionality.

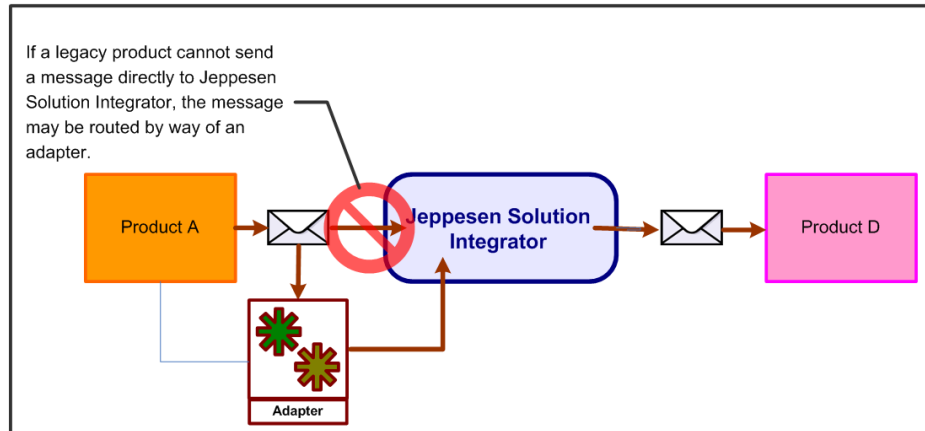
Alert functionality is incorporated within JSI to notify users of various error conditions. When a problem is detected the JSI Console displays an alert indicator to notify the user that such a condition exists, and logs the alert and associated data to assist with troubleshooting. Additionally Jeppesen Solution Integrator makes available an Email Alert Notification feature to provide the user the ability to designate an unlimited number of email addresses to be notified in the event of an Alert. The Email Alert Notification feature allows for rapid notification, and therefore expedient corrective action to resolve the issue causing the alert.

Messages may be uploaded using several methods in Jeppesen Solution Integrator (see *Jeppesen Solution Integrator User’s Guide* for methods) and depending on the method used there are certain files required. For instance, Message upload may be accomplished by the following methods and require the indicated files associated files:

- Defining an individual message; requires; one XSD file
- Uploading a bundled message set; requires; one MessageCatalog.xml file, and one or more XSD files (optional)
- Uploading an XSL set; requires; two or more XSL files

## Adapter Functionality

Legacy products may not support the XML interface defined by Jeppesen Solution Integrator. In this case, a custom translation may be needed to translate between the format used by the product and the XML required by JSI. [Figure 6](#) shows the role of the custom translation component, or adapter, within the architecture.



**Figure 6.** Message Sent to Jeppesen Solution Integrator Through Adapter

---

**NOTE** Adapter development is beyond of this document. See *Jeppesen Solution Integrator Developer's Guide* for more information pertaining to adapter development.

---

## Database

The JSI database contains configuration, message routing, and transaction logging data. Transaction logs contain messaging events providing an essential resource for event tracking, auditing and application troubleshooting. Jeppesen Solution Integrator can be deployed with a variety of relational databases.

## **Messaging and Standardization**

As previously stated; products integrated through Jeppesen Solution Integrator communicate using standardized Jeppesen defined messages, or custom user defined messages. This section provides additional details on standards and explains how and when these standards are published.

It is important to note that user defined custom messages must also conform to Jeppesen Solution Integrator messaging and standardization rules. For example, custom messages must be valid xml, and certain information must be present, such as an XSD, XPath to Name, and XPath to Version addresses. One significant difference between Jeppesen defined messages and user defined custom messages is in the way message names are formatted. Because Jeppesen defined messages follow the ALPHA, ALPHA, NUMBER, NUMBER, NUMBER naming convention (for example; AA001), user defined custom messages must use a different naming convention to preclude any inadvertent corruption, or overwrites of critical messages. Jeppesen recommended naming convention for user defined custom messages is ALPHA, ALPHA, ALPHA, NUMBER, NUMBER, NUMBER, NUMBER (for example; AAA0001), but any number of formats may be used as long as it differs from the standardized Jeppesen defined naming convention.

### **Message Standards**

The AOC Bookshelf defines the messaging requirements for each subsystem within an AOC.

- Message Overviews – Defines the common applications of messages within the associated system. Also provides links to the technical specifications and XML Schema Definition (XSD).
- XML Schema – Enforces messaging standardization within the messaging architecture. Each associated system has an XSD, which defines the messages associated with the system. For example, Crew Tracking messages are defined in a schema file named CrewTracking.xsd.
- Technical Specification (Spec) – Human readable documentation output of the XSD. Each associated system includes a spec that allows users to navigate through the system's complex types by clicking graphical or textual links.



## **Common Interfaces**

The messaging interface standards for Jeppesen Solution Integrator are documented in the AOC Bookshelf.

## **Sample Messages**

Jeppesen also provides sample XML files for almost every type of defined message. These sample files are linked from the AOC Bookshelf and from within each system's Message Overview document.

## Using the AOC Bookshelf

Jeppesen publishes the AOC Bookshelf (see [Figure 7](#)) as an easy, user friendly way to view all of the message related documents, XSDs, and sample XML associated with Jeppesen Solution Integrator online. AOC Bookshelf and associated documents can be viewed by navigating to: [http://www.jeppesen.com/download/CA\\_docs/bookshelf/START\\_HERE.html](http://www.jeppesen.com/download/CA_docs/bookshelf/START_HERE.html)  
 The XSDs, and XML samples (*Developers Toolkit*) can be viewed or downloaded (as a .zip file) from the same Website.

**AOC Bookshelf 14.2**  
23 September 2014

AIM Overview

Message Details

**Search the AOC Bookshelf**

Enter one or more key words and click the "Search" button. Documents containing the key words specified appear in a new window.  
\*\* Adobe® Reader 5.0 or greater is required.

**Overview**

Provides an overview of the Jeppesen Solution Integrator. Includes a solution review and architectural overview.

- [Jeppesen Solution Integrator \(JSI\) Overview](#)
- [AIM Glossary](#)

**Release Notes**

AOC Bookshelf Release Notes, including AIM installation instructions

- [AOC Bookshelf Release Notes](#)

**Developer Tool Kit**

XSDs and sample XMLs in a single .zip file.

- [Download](#)

Copyright © 2009-2014 Jeppesen. All rights reserved. Your use of the AOC Bookshelf and all supporting documentation is subject to a separate license agreement between you and Jeppesen, a copy of which is included in the zip file or can be obtained from Jeppesen. The AOC Bookshelf is delivered "AS IS" without warranty of any kind and is not guaranteed to be free from errors or defects. You rely on the AOC Bookshelf at your own risk. No support for the AOC Bookshelf is implied through its publication. The AOC Bookshelf is intended solely for use as a reference and examples of interfaces to Jeppesen systems. Jeppesen may revise, update or cease publication at any time, without notice. Building to the specifications set forth in the AOC Bookshelf does not mean that your intended integration needs will be met or that an interface will function as documented. We recommend contacting Jeppesen directly to discuss professional services options with respect to production application integration and validation efforts.

**Interface Control Documents**

Provides reference information for creating and implementing AIM messages. Includes all of Jeppesen's standard interfaces for AIM messaging.

Message Overviews	Spec	Source	Samples
<a href="#">Common Interfaces</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">AOC Message</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Airport Analysis (AA)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Airport Services (AS)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Alert Notification (AN)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">ATC (AT)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Binary Transport (BT)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Communications (CM)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Crew Tracking (CT)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Customer Data Services (CS)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">DatabaseManagement (DB)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Dispatch (DP)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Document Management (DM)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Electronic Flight Folder (EF)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Enterprise Resource Planning (EP)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Flight Following (FF)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Flight Planning (FP)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">GPS (GP)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Heartbeat (HB)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Maintenance (MX)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Navigation Data (ND)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Notam Management (NM)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Operations Control (OC)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Reservations (RS)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Schedule Planning (SP)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Training (TN)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">User Management (UM)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Weather (WX)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>
<a href="#">Weight and Balance (WB)</a>	<a href="#">html</a>	<a href="#">xsd</a>	<a href="#">xml</a>

Figure 7. Sample View of AOC Bookshelf

The AOC Bookshelf consists of the following tabs:

- **AIM Overview Tab**

Provides links to overview documents, XSDs, and technical specifications. The AIM Overview tab contains the following sections.

- **Overview:** Provides an overview of the Jeppesen messaging solution
- **Standards and Messaging:** Provides reference information related to AIM messaging, including Message version history, and message-flow system overviews
- **Release Notes:** Provides the most current information for the associated Bookshelf version, as well as instructions for installing the latest AIM to Jeppesen Solution Integrator.
- **Developer Tool Kit:** Provides a downloadable .zip file containing AIM/JSI XSDs, and XML sample messages.
- **Interface Control Documents:** Provides links to message overviews, message specifications, message source (XSD) files, and message samples (XMLs), which may all be viewed online.

- **Message Details Tab**

Provides a one-stop historical reference page for accessing versioning information on each message supported by the AOC Bookshelf.

---

**TIP** Links found within the Message Detail tab (Message Map Table) offer detailed information on each associated message. For example:

- \* Click links in the **Message Name** or **ID** column to view the message overview.
  - \* Click links in the **Overview** column to view the system overview.
  - \* Click links in the **Message Version** column to view the message specification (available for the most current version only).
-

# Jeppesen Solution Integrator Feature Overview

Jeppesen Solution Integrator includes a number of features and interfaces to support easy and efficient product maintenance and operation. These features are summarized in the following table.

**Table -1 JSI Feature Overview and Description**

<b>JSI Feature</b>	<b>Description</b>
JSI Console	The JSI Console (a web application) provides a single interface for users and support staff to manage Jeppesen Solution Integrator. This includes such activities as configuring services, as well as monitoring system status.
Message Store	Jeppesen Solution Integrator routes messages to and from products or applications in the system. Each message is database stored providing for querying and viewing of messages sent between applications.
Alerts	Jeppesen Solution Integrator generates alerts to inform the user if a problem is detected. If active alerts exist, a red, circular, numbered indicator badge appears. The number indicates the current number of active Jeppesen Solution Integrator alerts. Associated functionality includes the ability to designate Email Alert Notification recipients. These recipients receive email modifications if an alert occurs.
Data Translation – Cross Referencing	Cross Referencing ensures that all required and available information is sent in a JSI message. Cross referencing searches a data source for a value in an incoming message's data field(s) and uses the value to access or retrieve information located in another source. For example, if JSI receives the following XML message: <pre>&lt;airline&gt;&lt;icao&gt;UAL&lt;/icao&gt;&lt;/airline&gt;</pre> it can automatically look up and populate the iata code corresponding to the icao value. <pre>&lt;airline&gt;&lt;icao&gt;UAL&lt;/icao&gt;&lt;iata&gt;UA&lt;/iata&gt;&lt;/airline&gt;</pre> Cross Reference specifications are contained in a JSI database table. This table may be modified directly through database editing, use of the JSI upload capability, or through the use of the Manage Cross Reference Data tool in the JSI Console.

**Table -1 JSI Feature Overview and Description (continued)**

JSI Feature	Description
Message Delivery Window Support	<p>Message queuing allows future flight messages to be queued outside of an application's window of interest.</p> <p>For example, a customized application only cares about flights occurring within the next two days. The application is not interested and cannot process anything beyond that time frame. In this case, JSI queues messages sent to the application until they are within the two-day window.</p>
Content Based Routing (CBR)	<p>Content based routing provides the mechanism to route a message based on a comparison of the value of a given XPath element with the value specified in the rule.</p> <p>This value, when specified in the message, triggers the routing condition. That is, (using 1453 as an example) the rule triggers when the selected message for the product being modified contains the matching value (flight number 1453).</p> <p>To facilitate CBR specification a tool is provided within the JSI Console to extract a tree of valid XPaths for each canned message type. This tree is provided to the user during CBR specification.</p>
Message Cross Versioning	<p>Message Versioning provides compatibility between different JSI message versions.</p> <p>This is particularly helpful when one of the applications cannot handle the most current message version. For example, a third-party application may not support all of the required fields for the latest message.</p> <p>For example, a client Flight Planning application may be configured to use version one of a message, while the Operations Control application uses version two of the same message. When sending a message from Flight Planning to Operations Control, JSI converts version one to version two so that your Operations Control application can successfully process the message.</p> <p>Because each version transformation is unique (i.e. the code required to transform FP001 v1 to FP001 v2 is different than the code to transform OC001 v1 to OC001v2), Jeppesen provides each custom transformation from one message version to another on an as-needed basis.</p>
Custom Data Format Support – XSLT Message Transformation	<p>JSI delivers support for XML transformation to and from standard or proprietary formats.</p>
Configuration Import and Export Functionality	<p>Jeppesen Solution Integrator allows you to export existing system configurations, and import saved configurations.</p>

## ***System Configuration***

Each implementation of Jeppesen Solution Integrator can be unique - tailored to the specific environmental requirements of the implementing client. For each implementation, the following items are considered:

- Products (applications)
- What systems are required
- Any new products or upgrades to be implemented with JSI
- Hardware and software currently in use
- Communication protocols currently used between products

Jeppesen system engineers collect the customer environmental data to help create the JSI Configuration Specification.

This configuration document provides the Jeppesen deployment team with the basis for the implementation. See your Jeppesen deployment specialist for more information on JSI configuration.

# ***Glossary***

---





**AIM** — Aviation Integrated Messaging

**Adapter** — Legacy products may not support the XML interface defined by JSI Integrator. In this case, a custom translation may be needed to translate between the format used by the product and the XML required by JSI.

**Airline Operation Center (AOC)** — A center for airlines operations that unites various operational systems and departments in order to manage the daily operations of an airline. Also known as Systems Operations Control (SOC) and Operations Control Center (OCC). Jeppesen provides an application (JSI) to facilitate communication between the operational systems (i.e. products or applications) in an airline's operation center.

**Bus** — Individual products (applications) in the airline operation center communicate via a bus—a common transport (HTTP, HTTPS, JMS) and data format (serialized objects) standardized messaging engine.

**CBR** — Content Based Routing

**Crew Tracking** — A system that is used to manage the daily crew scheduling operations within an airline.

**JSI** — Jeppesen Solution Integrator

**Jeppesen Solution Integrator (JSI)** — Jeppesen Solution Integrator (JSI) provides a standard interface, which facilitates communication between different products (i.e. applications) within an airline operations center. JSI does this by routing discreet messages between the applications through a flexible, service-oriented architecture. Each message adheres to standards published by Jeppesen and is based on industry best practices.

**JMS** — The Java Message Service (JMS) API is a Java Message Oriented Middleware (MOM) API for sending messages between two or more clients. JMS is a specification developed under the Java Community Process as JSR 914.

**Product** — Applications integrated by JSI. The JSI can be used to integrate any product, regardless of vendor, in an airline operation center.

**SOAP** — SOAP is a protocol for exchanging XML-based messages over a computer network, normally using HTTP. SOAP forms the foundation layer of the Web services stack, providing a basic messaging framework that more abstract layers can build on.

**User Management** — A single system to manage users as well as application roles and permissions.

**XSD** — XML Schema Definition published as a W3C recommendation. XSDs define the messages that can be routed through JSI.