xConnect IDMS Architecture

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev** | **Date** | **Author** | **Description** |
| 1.3 | 8/28/2012 | Ted Crane | Version 1.3 Updates |
|  | 10/11/2012 | Ted Crane |  |

**Document Approvers & Sign-Off**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Approver** | **Role** | **Document Accept/Reject** |
| 09/05/2012 | Ric Merrifield | Release Manager | In Review |
| 09/05/2012 | John Stiehl | Release Manager | In Review |
| 09/05/2012 | Adam Parrish | Product Tech Manager | In Review |

**Table of Contents**

[1 Introduction 4](#_Toc338142570)

[1.1 Description 4](#_Toc338142571)

[1.2 Purpose 4](#_Toc338142572)

[1.3 Scope 4](#_Toc338142573)

[1.4 Assumptions and Constraints 4](#_Toc338142574)

[1.5 Definitions 5](#_Toc338142575)

[2 The xConnect IDMS Environment 6](#_Toc338142576)

[3 Prerequisites 6](#_Toc338142577)

[4 Processing Sequences 7](#_Toc338142578)

[4.1 Book xBand Request Sequence 7](#_Toc338142579)

[4.2 Book xBand Sequence 8](#_Toc338142580)

[4.3 Tap To Assign Sequence 9](#_Toc338142581)

[4.4 Reassign xBand Sequence 10](#_Toc338142582)

[4.5 Public Experience Change (PxC) Sequence 11](#_Toc338142583)

[5 Messages 12](#_Toc338142584)

[5.1 Book xBand Request Message 12](#_Toc338142585)

[5.2 Book xBand Message 15](#_Toc338142586)

[6 IDMS Database Schema 17](#_Toc338142587)

[7 IDMS Listener Configuration 18](#_Toc338142588)

[8 High Availability 21](#_Toc338142589)

[9 Cache 21](#_Toc338142590)

[10 Troubleshooting 22](#_Toc338142591)

# Introduction

## Description

IDMS is a cache of guest and xBand data to support high performance lookups by other systems. In addition IDMS has been used as temporary stand in for SF/OneView during demos and tests.

IDMS is comprised of two components.

* IDMS RESTFul web services, which allow other application to retrieve guest, xBand, and celebration data. The available web services are documented in the IDMS Interface Control document.
* IDMS Listener, which consumes messages from several JMS topics and writes data to the IDMS data store using the IDMS RESTFul web services. The IDMS Listener is one service provided by the xConnect JMS Listener. Services that listen to other JMS messages are documented in the TBD document.

## Purpose

Description of the system architecture for the xConnect IDMS Listener software that reads and messages transmitted from xBMS and caches information about the relationship between guest and the media used by the guest to interact with the xConnect system.

IDMS will be the source for other xConnect systems in the LDU to retrieve xBMS Link ID. This ID will be used by xConnect in messaging guest information to systems outside xConnect.

## Scope

This is an early draft of the xConnect IDMS Listener software, and does not cover all functionality to be provided in the production system.

## Assumptions and Constraints

A Progress Software Sonic MQ Enterprise Service Bus has been installed and configured in a location that can be connected to by the IDMS Listener software.

A SQL Server 2008 R2 Server has been installed and configured with the appropriate xBRMS database schema in a location that can be connected to by the IDMS Listener software.

xBMS is operating and publishing the expected messages on the SonicMQ Enterprise Service Bus.

The IDMS web services are available to retrieve guest information.

## Definitions

For this draft, the following acronyms are used:

|  |  |
| --- | --- |
| **Term** | **Definition** |
| xConnect | Code, scripts, APIs, and database schemas which comprise the unifying messaging, management, and reporting software which ties the hardware together into a coherent solution |
| xBand | RF-enabled wristband |
| xTP | Experience TouchPoint, a Disney-themed short range HF reader or "touch" device |
| DAP | Disney Access Portal, a free-standing xTP configured with a biometric reader at park entry or as a sole touch point stanchion for access to attractions |
| xBR | Long range RFID reader with uni- or omni-directional antennae |
| xBRC | xBand Reader Controller which manages xBRs, xTPs, and DAP devices |
| xBMS | xBand Management System code and database, which is the system of record for media, i.e. cards and bands, and the association between media and Guests via link ID architecture. |
| Media | xBands, cards or other devices that store a public or secure ID that are read by the xConnect system. |
| BOG | Be Our Guest restaurant in Fantasyland. |
| DME | Disney’s Magical Express. |

# The xConnect IDMS Environment



Figure 1 xConnect IDMS Environment

# Prerequisites

This section addresses network, hardware, and software prerequisites to support the xConnect IDMS Listener software. The xConnect system has been installed following the instructions outlined in the *NGE Alpha Lab Installation Guide*.

# Processing Sequences

## Book xBand Request Sequence



Figure 2 Book xBand Request Sequence

The Book xBand Request sequence is initiated by a travel plan event, typically a file import of reservation data. The reservation data is provided and imported in batches.

From this data xBMS generates a Book xBand Request message on the Enterprise Service Bus.

The IDMS Listener uses the reference ID in this message to call to the xBMS Provide xBand Request Details web service to retrieve additional data about the Book xBand Request.

For additional information on both the Book xBand Request message and the response to the Provide xBand Request Details, see the [Book xBand Request Message](#_Book_xBand_Request_1) section in this document.

In production the IDMS Listener will listen to both the XBMS.XBANDREQUEST and XBMS.IDMS.XBANDREQUEST topics (shown as JMS Topic in the sequence diagram). The same message structure will be transmitted on both topics. The XBMS.XBANDREQUEST topic will contain messages about guests imported into xBMS from DME and Dreams reservations systems. The XBMS.IDMS.XBANDREQUEST topic will be used for guests specifically imported for the BOG restaurant.

The Provide xBand Request Details response can contain multiple guests, so the Create Guest message between the IDMS Listener and IDMS may occur multiple times for each message received on the JMS Topic.

If the call to the Provide xBand Request Details web service fails, processing is discontinued for the current message. Should the message be queued and retried?

## Book xBand Sequence



Figure 3 Book xBand Sequence

The Book xBand sequence is initiated by a Fulfilled Band event, typically a file import of vendor fulfillment data.

From the data received from the fulfillment vendor, xBMS then generates a Book xBand event on the Enterprise Service Bus. This message indicates that an xBand needs to be associated to a guest.

The IDMS Listener will first check if the guest exists in IDMS by the provided guest type and guest id. If the guest doesn’t exist, the message is ignored, since the guest should already be in xBMS, and have been created in IDMS through the Book xBand Request Sequence. For situation where the guest doesn’t exist, see the [Tap To Assign Sequence](#_Tap_To_Assign).

If the guest exists, then the xBand information included in the message is used to create the xBand data in IDMS. Then the association between the xBand and the guest is created in IDMS.

If the guest has a gxp-link-id in IDMS, then the IDMS Listener will notify GXP about the new band associated to the guest. The IDMS Listener will send a notification that the band is being associated.

## Tap To Assign Sequence



Figure 4 Tap To Assign Sequence

The Tap To Assign Sequence represents the provisioning of xBands for use in the BOG restaurant to assume the role of a Puck or Cast Member. The Puck represents a guest that doesn’t have a personal band, and the xBand is enclosed in a container given to the guest when they check in. xBands used for the Cast Member role attached to a device used by a cast member such as busser or server.

The sequence begins with an application calling the xBMS Tap To Assign web service endpoint. This call will generate a Book xBand Message. The processing of this message is similar to the Book xBand Sequence, but in the case of the Tap To Assign Sequence, the guest does not already exist in IDMS. An anonymous guest is created in IDMS, with the specified guest type and guest id. This sequence also assumes the xBand needs to be created. xBands that have been previously associated with a guest will be handled in the [Reassign xBand Sequence](#_Reassign_xBand_Sequence). Once the xBand has been created in IDMS, an association to the anonymous guest will be created. Unlike the Book xBand Sequence, GxP will not be notified of this association, as the guest will not have a gxp-link-id.

## Reassign xBand Sequence



Figure 5 Reassign xBand Sequence

Prior to xBands being shipped to guests, guests that pre order a meal in the BOG restaurant will need to be associated to one of the existing xBands being used in the Puck role. To accomplish this, an application will call the xBMS Modify xBand Details web service to transfer the xBand to the guest. xBMS will generate a Book xBand Message on the Enterprise Service Bus. Unlike the Book xBand Sequence or Tap To Assign Sequence, both the guest and the xBand will already exist in IDMS. So data for both the guest and the xBand will be retrieved from IDMS. This data will be used to remove the existing association between the anonymous guest and xBand in IDMS and create a new association between the actual guest and xBand in IDMS. Then GxP is informed of the ownership change on the IDMS.NOTIFIER.GXP queue.

Once the meal has been served to the guest, the Puck will be removed from their table, and reassigned back to the anonymous guest. Then the Puck can be reused in the restaurant for guests that haven’t pre ordered.

## Public Experience Change (PxC) Sequence



Figure 6 Public Experience Change (PxC) Sequence

The Publish Experience Change Sequence allows the IDMS cache of identifiers to stay in sync with the SF/OneView, which is the system of record for the guest identifiers IDMS is caching. The IDMS Listener will listen for a Change Event message on the SF.OV.EXPERIENCEASSOCIATION topic and call the Provide Guest Identifiers IDMS web service to retrieve the existing guest identifiers stored in IDMS. These identifiers will be compared to the identifiers received in the Change Event message, and any guest identifiers not found in IDMS will be added. Only identifiers associated with a guest, and not those associated with a guest’s xBand, will be added to IDMS. IDMS will store all xBand data from xBMS instead.

Part of 1.3.2 Release.

# Messages

This section documents the messages the read and processed by the xConnect IDMS Listener system processes. Sample events provided from xBMS are included to help document the flow of data through the xConnect IDMS Listener system.

## Book xBand Request Message

The Book xBand Request message is documented on the xBMS wiki page:

<https://wiki.nge.wdig.com/display/NGE/xBMS+Event+-+Book+xband+Request>

A sample Book xBand Request Message:

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

<businessEvent>

<guestIdentifier></guestIdentifier>

<eventType>BOOK</eventType>

<subType></subType>

<location>XBMS.XBANDREQUEST</location>

<timestamp>March 28, 2011 9:39:01 PM UTC</timestamp>

<referenceId>EE692F11-8982-47EB-9717-24EFB9C15D35</referenceId>

<payload><![CDATA[]]></payload>

<correlationId>21ffefd0-c279-46d8-b978-fe268868b509</correlationId>

</businessEvent>

From this message, the referenceId value is used to retrieve the details of the request, as documented on the xBMS wiki.

<https://wiki.nge.wdig.com/display/NGE/xBMS+Provide+xBand+Request>

This endpoint provides a response in JSON, as shown in the following example:

{

"customizationSelections" : [

{

"xbandOwnerId" : "0E164AC1-6230-4DC5-A5D5-C5F30F60C43F",

"self" : "/customization-selections/0E164AC1-6230-4DC5-A5D5-C5F30F60C43F",

"xband" : "/xband/0E164AC1-6230-4DC5-A5D5-C5F30F60C43F",

"birthDate" : "1975-02-01T05:00:00Z",

"firstName" : "Jane",

"guestId" : 10000,

"guestIdType" : "transactional-guest-id",

"primaryGuest" : true,

"lastName" : "Doe",

"bandProductCode" : "B11111",

"bandAccessories" : [

{

"bandAccessoryCode": "100000",

"bandAccessoryType": "100000"

}

],

"printedName" : "Jane's Band",

"customizationSelectionId" : "0E164AC1-6230-4DC5-A5D5-C5F30F60C43F",

"createDate" : "2010-07-26T05:00:00Z",

"updateDate" : "2010-07-26T05:00:00Z",

"entitlements" : [ "STANDARD", "LEVEL\_2\_5" ],

"qualifyingIds" : [

{

"qualifyingId": 1234579,

"qualifyingIdType": "travel-component-id"

},

{

"qualifyingId": 1234578,

"qualifyingIdType": "travel-component-id"

},

{

"qualifyingId": 4387621045,

"qualifyingIdType": "bundle-id"

},

{

"qualifyingId": 4387621048,

"qualifyingIdType": "bundle-id"

},

],

"confirmedCustomization" : false,

},

{

"xbandOwnerId" : "D7C0EF05-5078-4ECD-A3DF-D53A65E0A75C",

"self" : "/customization-selections/8D60B701-22EA-487D-8950-D185B46D4EDC",

"xband" : "/xband/8D60B701-22EA-487D-8950-D185B46D4EDC",

"birthDate" : "1980-01-01T05:00:00Z",

"firstName" : "Joey",

"guestId" : 10001,

"guestIdType" : "transactional-guest-id",

"primaryGuest" : false,

"lastName" : "Tsai",

"bandProductCode" : "B11111",

"bandAccessories" : [

{

"bandAccessoryCode": "100000",

"bandAccessoryType": "100000"

}

],

"printedName" : "Joey's band",

"customizationSelectionId" : "8D60B701-22EA-487D-8950-D185B46D4EDC",

"createDate" : "2010-07-26T05:00:00Z",

"updateDate" : "2010-07-26T05:00:00Z"

"entitlements" : [ "NONE" ],

"qualifyingIds" : [

{

"qualifyingId": 1234578,

"qualifyingIdType": "travel-component-id"

}

],

"confirmedCustomization" : false,

},

],

"requestAddress" :

{

"confirmedAddress" : false,

"phoneNumber" : "867-5309",

"address" : {

"address1" : "#4 Privet Drive",

"city" : "Atlanta",

"country" : "US",

"postalCode" : "30328",

"state" : "GA"

}

},

"resortReservations": [

{

"arrivalDate" : "2010-09-26T04:00:00Z",

"departureDate" : "2010-10-09T04:00:00Z",

"facilityId" : 280010388,

"travelSegmentId": 1234578,

"travelComponentId": 87654321

},

{

"arrivalDate" : "2010-09-26T04:00:00Z",

"departureDate" : "2010-10-09T04:00:00Z",

"facilityId" : 280010400,

"travelSegmentId": 1234579,

"travelComponentId": 87654320

}

],

"shipment" : {

"method": "PRIMARY\_GUEST\_ADDRESS\_BEST",

"carrier" : "UPS",

"carrierLink" : "http://www.ups.com/WebTracking/track",

"shippingDate" : "2010-08-27T04:00:00Z",

"trackingNumber" : 66,

"address" : {

"address1" : "#4 Privet Drive",

"city" : "Atlanta",

"country" : "US",

"postalCode" : 30328,

"state" : "GA"

}

},

"state" : "COMPLETED",

"createDate" : "2010-07-26T05:00:00Z",

"updateDate" : "2010-07-26T05:00:00Z",

"customizationEndDate" : "2010-08-26T05:00:00Z",

"xbandRequestId" : "B2512223-7314-46EE-9687-8B6159ECAD08",

"self" : "/xband-requests/B2512223-7314-46EE-9687-8B6159ECAD08",

"reorder" : "/xband-requests/B2512223-7314-46EE-9687-8B6159ECAD08/reorder",

"options": "/reorder-options/458414CE-602C-4ECC-B01D-81D4BE7EC29C",

"order": "/orders/85E5B54C-3727-4A57-8B13-5570D4B3657D",

"acquisitionId" : "100077",

"acquisitionIdType" : "travel-plan-id",

"acquisitionStartDate" : "2010-08-26T05:00:00Z",

"acquisitionUpdateDate" : "2012-01-26T05:00:00.568Z",

}

For each guest item in the customization selections list, the transactional guest ID (or what every guest id type is specified) is first used to check if the guest already exists. If the guest doesn’t exist, the highlighted fields are used to create a new guest in IDMS.

## Book xBand Message

The Book xBand message is documented on the the following xBMS wiki page:

<https://wiki.nge.wdig.com/display/NGE/xBMS+Event+-+Book+xband+Request>

A sample Book xBand Message:

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

<businessEvent>

<guestIdentifier></guestIdentifier>

<eventType>BOOK</eventType>

<subType></subType>

<location>XBMS.XBAND</location>

<timestamp>March 28, 2011 9:39:01 PM UTC</timestamp>

<referenceId>EE692F11-8982-47EB-9717-24EFB9C15D35</referenceId>

<payload><![CDATA[]]></payload>

<correlationId>21ffefd0-c279-46d8-b978-fe268868b509</correlationId>

</businessEvent>

From this message, the referenceId value is used to retrieve the details of the request from the Provide xBand Details endpoint, as documented on the following xBMS wiki page:

<https://wiki.nge.wdig.com/display/NGE/xBMS+Provide+xBand+Details>

JSON, like the following example, is returned:

{

"assignmentDateTime" : "2011-02-22T04:09:47Z",

"externalNumber" : 4320000000000000,

"printedName" : "Carol Xband3",

"productId" : "B11012",

"publicId" : 1234,

"secondaryState" : "ORIGINAL",

"secureId" : 66,

"self" : "/xband/0192B866-85BD-4369-9E85-84237315A110",

"shortRangeTag" : 166,

"state" : "ACTIVE",

"guestId" : "0915EFB4-3B7D-45B4-93FD-D7145941D484",

"guestIdType" : "swid",

"xbandId" : "0192B866-85BD-4369-9E85-84237315A110",

"xbandRequest" : "/xband-requests/6315A9F3-C19C-4530-8337-A73AA21C33D3",

"xbandOwnerId" : "6422FA29-6BDB-4831-9F5F-6A2A165E3FA1",

"options" : "/xband-options/0192B866-85BD-4369-9E85-84237315A110"

"history" : "/xband-history/0192B866-85BD-4369-9E85-84237315A110"

}

The highlighted values are used to create a band in IDMS. The xbandOwnerID is used to associate the guest and the band in IDMS.

# IDMS Database Schema



Figure 4 IDMS Database Schema

The IDMS database is the persistence mechanism for the caching guest and xBand data retrieved from xBMS messages and web service endpoints.

# IDMS Listener Configuration

The IDMS Listener depends on both the environment.properties file located in /etc/nge/config, as well as the config table in the xBRMS database.

The environment.properties file contains the information the IDMS Listener needs to connect to the database to read additional configuration information. All passwords are also stored in the environment.properties file, so they can be encrypted and stored in the file. The standard NGE decryption process is used to decrypt the passwords so they can be used by the IDMS Listener.

The following settings must appear in the environment.properties:

|  |  |
| --- | --- |
| **Property Name** | **Description** |
| nge.xconnect.jms.databasedriver | Database driver to be used when connecting to the configuration database. Currently only net.sourceforge.jtds.jdbc.Driver is supported. |
| nge.xconnect.jms.dbserver.url | URL used by JDBC to connect to the configuration database. |
| nge.xconnect.jms.dbserver.uid | Database user to connect to the configuration database. |
| nge.xconnect.jms.dbserver.pwd | Password of the database user used to connect to the configuration database. |
| nge.xconnect.jms.c3p0.maxPoolSize | Number of connections in the connection pool used by the IDMS Listener. |
| nge.xconnect.jms.c3p0.maxStatementsPerConnection | Maximum number of statements, per connection allowed by the connection pool. |
| nge.xconnect.jms.c3p0.maxStatements | Maximum number of statements, total allowed by the connection pool. |
| nge.xconnect.jms.xbms.broker.pwd | Password for the user used to connect to the JMS topics and queues used by the IDMS Listener. |
| nge.xconnect.jms.xbms.dbserver.pwd | Password used by the IDMS Listener when writing to the database. (Deprecated, will be removed in a future release) |

The remaining configuration for the IDMS Listener is stored in the config table in the database specified in the nge.xconnect.jms.dbserver.url setting in the environment.properties. The config table is deserialized into memory, and used by the IDMS Listener at run time.

The config database has three columns:

* class – which represents the Java class the configuration is deseralized into.
* property – the name of the property, which maps to the field name in the Java class.
* value – value of the property used at run time.

All IDMS Listener configuration data has the class name of “XbmsListenerConfig”, and the following properties are required:

|  |  |
| --- | --- |
| **Property** | **Description** |
| databasedriver | Database driver to be used when connecting to the destination database. (Deprecated, will be removed in a future release) |
| databaseurl | URL used by JDBC to connect to the destination database. (Deprecated, will be removed in a future release) |
| databaseuser | Database user to connect to the configuration database. (Deprecated, will be removed in a future release) |
| idmsurl | Root URL of IDMS that will receive updates from the IDMS Listener. |
| jmsbroker | Name of the JMS Broker used for listening for messages. |
| jmsretryperiod | Number of milliseconds to wait to retry connecting when a connection is lost or cannot be established. |
| jmsuser | Username used to connect to the JMS Broker. |
| publishqueue | Name of queue to publish band association and ownership changes. |
| queuebroker | Name of the JMS broker to connect to for publishing messages. |
| xbandrequesttopic | Name of the topic that Book xBand Request messages will be received on. |
| xbandtopic | Name of the topic that Book xBand messages will be received on. |
| xbmsurl | Root URL used for web service calls to xBMS. |

# High Availability

To be supported in version 1.4.

# Cache

The cache is an optional component that may be used to reduce network load between the IDMS server and database. It stores, in memory, results from the database up to some configured maximum number of objects. Objects retrieved that would exceed the maximum number of allowed objects are not stored in the cache. The cache may configured to be cleared once a day or never at all.

The status message will present cache details when the cache is enabled:

<statusMessage>Guest objects 1023/1000000

Band objects 2098/3000000

Cache is enabled</statusMessage>

The following is an example configuration from /etc/nge/config/environment.properties that allows for caching a maximum of one million guests, three million bands, and is cleared each day at 2:00am:

nge.xconnect.idms.cache.enable=true

nge.xconnect.idms.cache.maxGuests=1000000

nge.xconnect.idms.cache.maxBands=3000000

nge.xconnect.idms.cache.resetTime=02:00

|  |  |
| --- | --- |
| **Property** | **Description** |
| nge.xconnect.idm.cache.enable | Set to ‘true’ to turn on caching. Default: false |
| nge.xconnect.idm.cache.maxGuest | The maximum number of guest objects to hold in memory. Default: 1000000 |
| nge.xconnect.idm.cache.maxBands | The maximum number of band objects to hold in memory. Default: 3000000 |
| nge.xconnect.idm.cache.resetTime | Time of day (HH:mm, 24-hr clock) to clear cache. Default: “” |

# Troubleshooting

Please refer to the *Troubleshooting.docx* for troubleshooting instructions.