Oracle InterConnect -

A case study integrating Keystone and E-Business 11i

Darren Wenham

Senior Consultant

Theta systems Ltd

Background

Clifford Chance, the worlds largest Legal Practice Firm, had evaluated a number of proposals for their new Global Practice Management System (GPMS). After an extensive evaluation process Clifford Chance indicated that no single application had the full functionality to satisfy the requirements of their proposed GPMS. With this understanding two applications were combined to constitute the GPMS, these applications were Keystone Professional 2002 (KP) and Oracle E-Business 11i (11i).

Business Issue

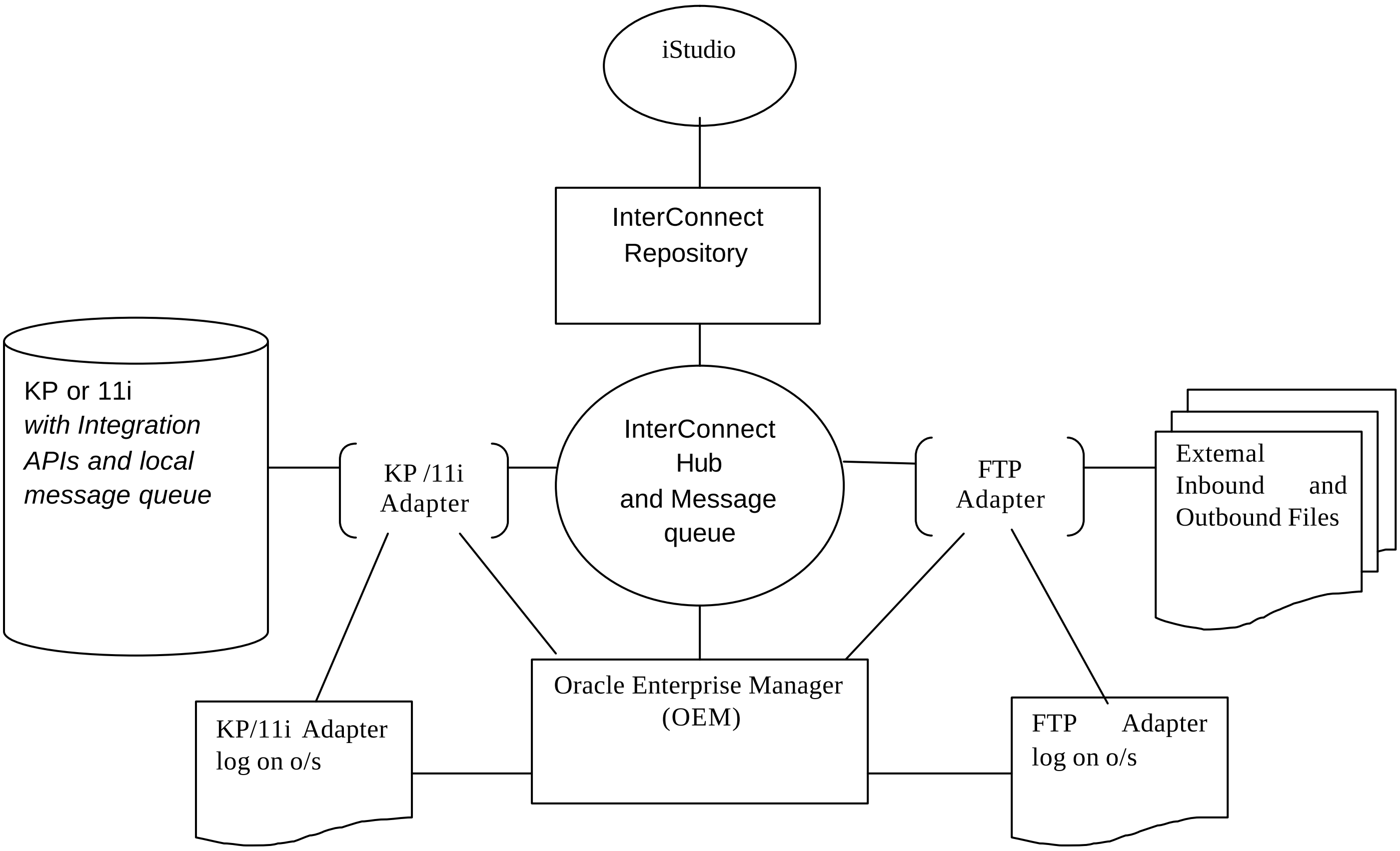
Clifford Chance requires both KP and 11i to function as a single application in the case where transactional data is common to both. Transactions requiring integration must be automatically recognised transmitted seamlessly and occur regardless of destination applications availability.

Proposed Solution

The Proposal included the use of a relatively new Oracle Messaging product known as Oracle InterConnect. This product along with the development of custom software to extract and apply transactional data between applications and a custom application for monitoring and alerting of exceptions would deliver a product to suit client requirements.

KP and 11i Business analysts and developers would work together to define a repository of integration definitions within InterConnect, using a GUI tool known as iStudio. InterConnect components were installed into both KP and 11i and configured to communicate directly with this repository and the InterConnect Hub. As InterConnect is a messaging product data would be packaged into messages then transacted and managed by the InterConnect HUB and adapters. The InterConnect hub functions as the message manager ensuring delivery of messages and application independence. Management of the Interconnect components, message logs and exceptions would be available through Oracle's Enterprise Management product.

Diagram of proposed solution


iStudio
InterConnect Repository
KP or 11i
with Integration APIs and local message queue
KP /11i Adapter
InterConnect Hub and Message queue
FTP Adapter
Extemal Inbound and Outbound Files
Oracle Enterprise Manager (OEM)
KP/11i Adapter log on o/s
FTP Adapter log on o/s

Description of Components

- **IStudio** – a GUI tool used by Business analysts and Developers to define the InterConnect repository metadata for KP and external applications.
- **InterConnect Hub** – Schema residing in an Oracle database controls the communication of messages between KP and External applications. KP/11i and External applications will only communicate with this hub and never directly to each other.
- **InterConnect Repository**– Contains definitions for KP and External applications Business Objects, Events, mappings, transformations and procedural business logic. This metadata is stored within Oracle supplied tables and used during the creation, publishing, transmission and application of messages between KP and External applications.
- **KP/11i Adapter** – An InterConnect supplied communication pipe. Delivers outgoing messages to the InterConnect Hub, monitors Hub for incoming messages and applies incoming messages to the application.
- **FTP Adapter –** An InterConnect supplied communication pipe. This adapter is capable of uploading and downloading InterConnect messages in XML format from a file system and FTP Server to/from the Interconnect HUB.
- **Oracle Enterprise Manager** – With the installation of InterConnect plug-ins this Oracle product allows management of InterConnect components via the standard OEM console.
- **KP/11i Integration APIs –** Includes InterConnect Stub APIs exported from the repository and installed into the KP/11i applications, and non-InterConnect objects such as database triggers, KPAPI views/triggers and Integration APIs defined by developers to trigger/apply integration events.

InterConnect Adapters

Communication between an application and the Hub is established using InterConnect Adapters. Oracle supplies a set of standard adapters for InterConnect to Application connectivity, such as AQ, XML, SAP and Oracle RDBMS. An adapter contains monitors for incoming and outgoing messages, quantities are configurable to suit transactional volumes.

This proposal will utilise the following InterConnect supplied adapters;

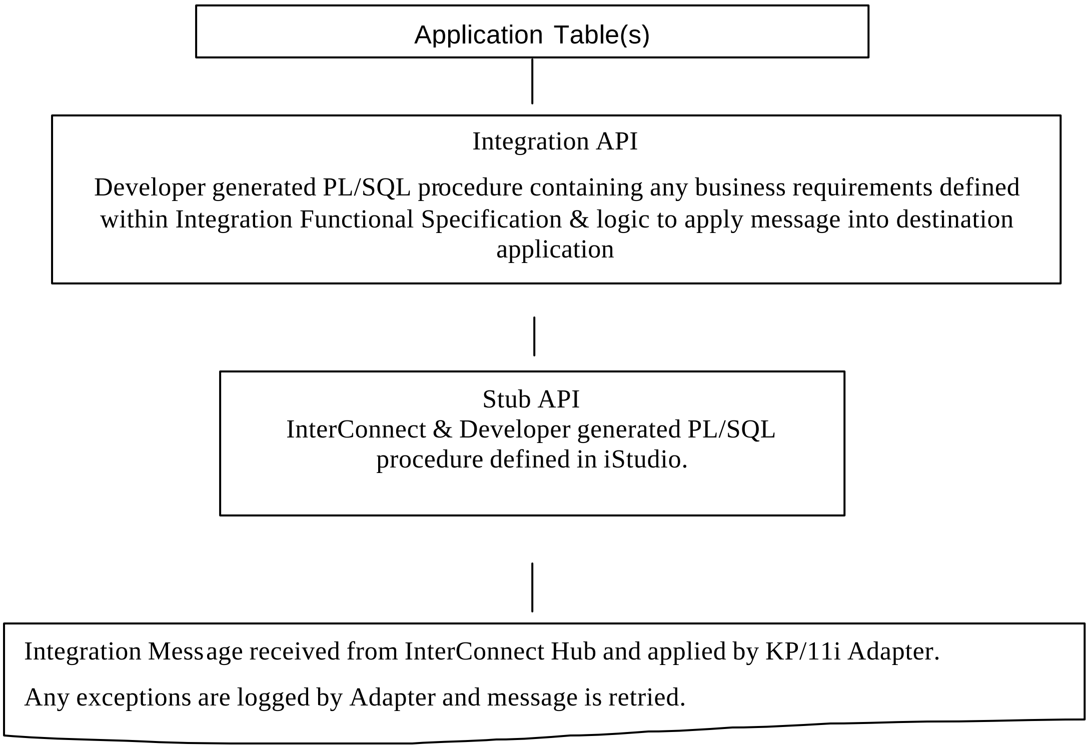
- Database adapter for KP/11i communication since both are configured into Oracle databases
- AQ adapter for communication with an Oracle Advanced Queue object
- FTP adapter for loading of XML files from the file system
- PeopleSoft adapter for communication with the PeopleSoft application

Note: Additional adapters can be developed if required using Oracle's supplied Adapter SDK.

How Integration Occurs

Applying integration messages from InterConnect HUB into an application

When InterConnect publishes a message to a destination application that application's adapter receives the message and invokes a "Stub API" within the destination database. This Stub API is passed the data payload contained within the message, it then calls an Integration API which applies specific business logic to the data and integrates into destination application.



Application Table(s)

Integration API

Developer generated PL/SQL procedure containing any business requirements defined within Integration Functional Specification & logic to apply message into destination application

Stub API
InterConnect & Developer generated PL/SQL procedure defined in iStudio.

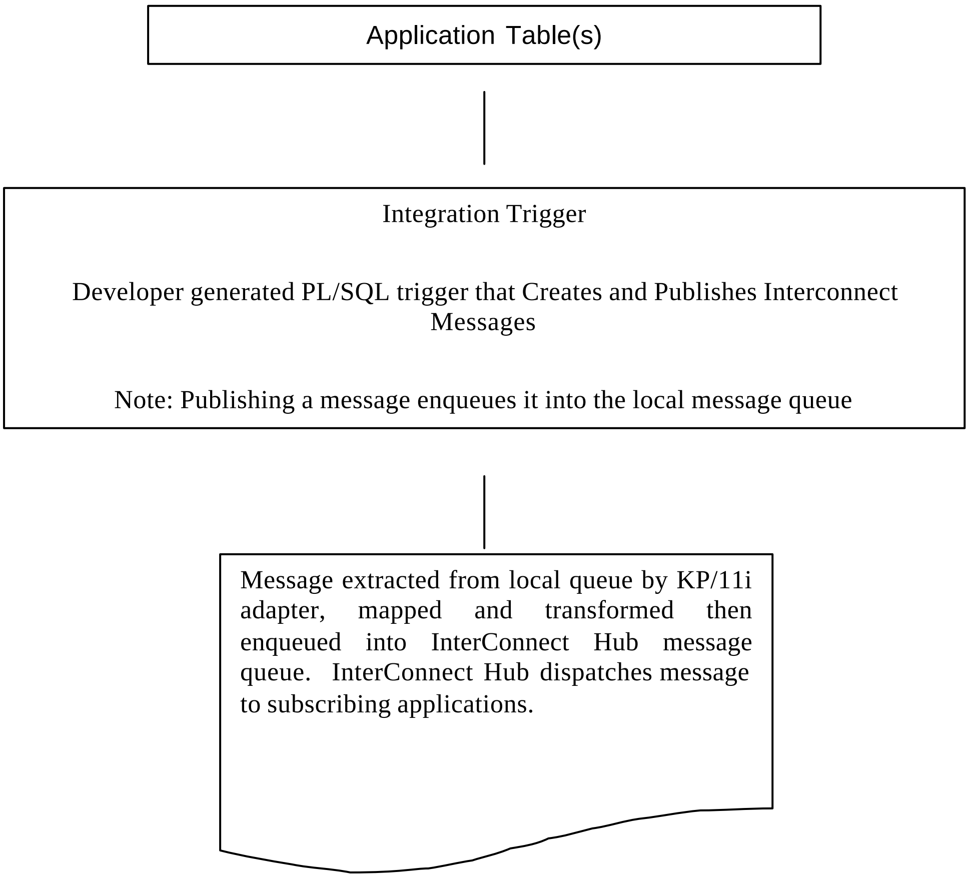
Integration Message received from InterConnect Hub and applied by KP/11i Adapter.

Any exceptions are logged by Adapter and message is retried.

Generating integration messages from Source application

When events occur within a source application (eg. KP or 11i), which require integration into another application (destination application), the messages must be generated within the source then published to InterConnect. The generation and publishing of messages will be accomplished via triggers on appropriate application table(s).

The following diagram details outgoing message generation and publishing:



Application Table(s)

Integration Trigger

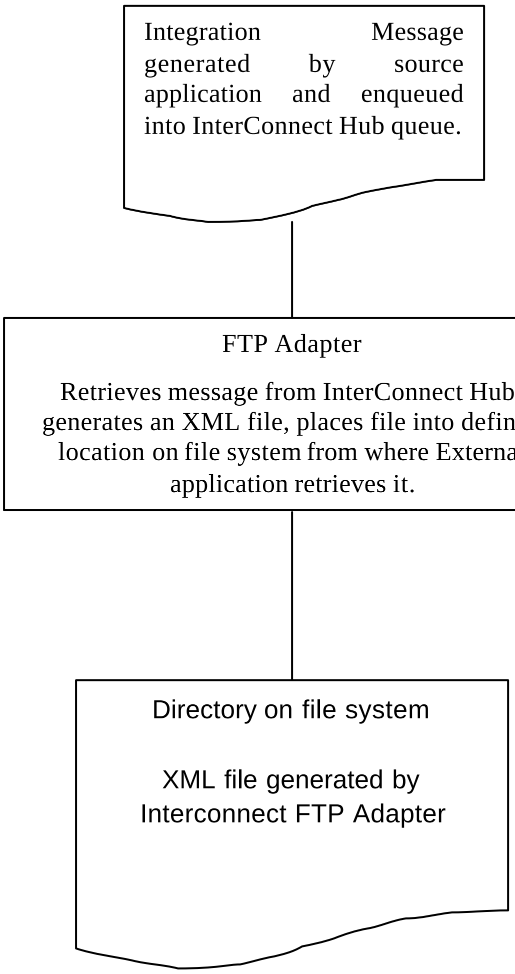
Developer generated PL/SQL trigger that Creates and Publishes Interconnect Messages

Note: Publishing a message enqueues it into the local message queue

Message extracted from local queue by KP/11i adapter, mapped and transformed then enqueued into InterConnect Hub message queue. InterConnect Hub dispatches message to subscribing applications.

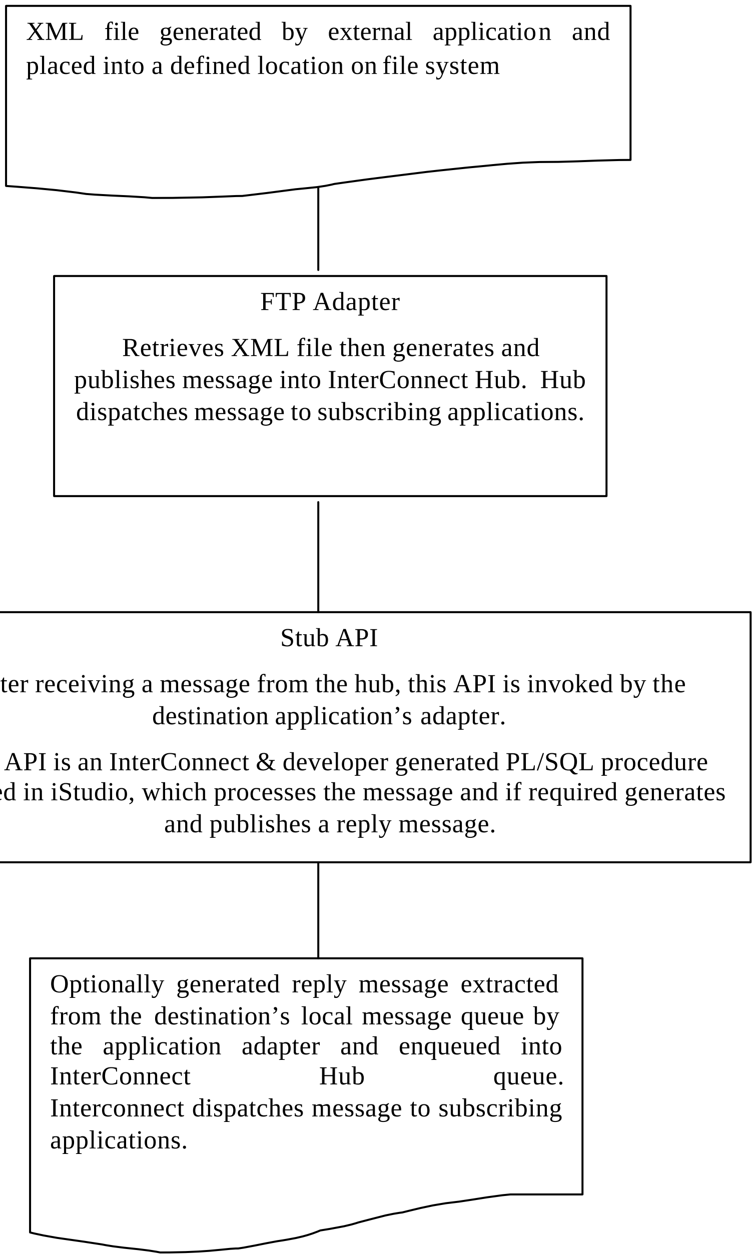
Generating XML output for application into an External application

The integration process will not apply data into an external application, it will however create an XML file containing the integration data and place it within a defined location on the file system. The onus is on the external application to retrieve and apply this file.


Integration Message generated by source application and enqueued into InterConnect Hub queue.
FTP Adapter
Retrieves message from InterConnect Hub, generates an XML file, places file into defined location on file system from where External application retrieves it.
Directory on file system
XML file generated by Interconnect FTP Adapter

How to integrate an externally generated XML file

When an External application needs to apply data into an application it must format the data into a pre-defined XML format place this file into a defined location on a file system accessible by the InterConnect HUB Server. This location is polled by the FTP Adapter, XML files are converted into Interconnect messages and published to the InterConnect HUB from where they are sent through to the destination application's adapter for processing. Within the destination application the adapter invokes a Stub API specific to the message type, processes the message and if required produces a message in reply. The adapter dispatches any reply message to the InterConnect hub, which then passes the message onto any subscribing applications.



XML file generated by external application and placed into a defined location on file system

FTP Adapter

Retrieves XML file then generates and publishes message into InterConnect Hub. Hub dispatches message to subscribing applications.

Stub API

After receiving a message from the hub, this API is invoked by the destination application's adapter.

The API is an InterConnect & developer generated PL/SQL procedure defined in iStudio, which processes the message and if required generates and publishes a reply message.

Optionally generated reply message extracted from the destination's local message queue by the application adapter and enqueued into InterConnect Hub queue. Interconnect dispatches message to subscribing applications.

Development Requirements

A Business analyst must define components of the integration using iStudio, where as other components must be coded by a developer.

Defining the InterConnect Repository

Both Business analysts and Developers use the JAVA GUI tool "iStudio" to create an integration repository, which defines:

- Every integrated business object within KP, 11i and an External application including the objects structure. Known within InterConnect as the objects "Application View".
  eg. KP/Exchange Rate and Bank/Exchange Rate
- Each associated pair of KP/11i/External business objects must have a common InterConnect structure defined. Known as the "Common View", to which each application interfaces ensuring application independence.
- Mappings for each KP, 11i and External object to their relevant "Common View", including any data transformations.
  eg. KP/Exchange Rate values Date and Time map to Common View "DateTime" with a transformation to concatenate with separator of space.
- Each event within KP, 11i and External applications that trigger an integration transaction
  eg. Create_Exchange_Rate
- Each event has a Stub procedure defined, which contains logic for applying integration data
- Which events an application subscribes to and publishes
- InterConnect can be configured for these optional features:
  - Cross-referencing – by maintaining cross-reference tables, Interconnect is able to seamlessly transpose primary key values between applications.
  - Domain Value Mapping – code tables can be mapped across applications.
    eg. Code value for city "Auckland" within External application is "109" and KP is "AK"
  - Content-Based Routing – routing messages based on its content to specific applications.

InterConnect Supported Message Models

Integration messages sent between KP, 11i and External applications can be configured within InterConnect to occur either synchronously or asynchronously. Only asynchronous messages will be used within this solution to ensure application independence.

There are also two types of message supported by InterConnect:

- **Publish/Subscribe** – asynchronous only. An application places a message into the Hub the other application receives the message and applies it.
- **Request/Reply**– choice of synchronous/asynchronous. An application places a message into the Hub the other application receives the message, applies it, then produces a reply message and places this on the Hub. The originating application receives the reply and performs appropriate tasks based on its contents.

Both message types can co-exist within the integration solution as they are specified at the event. As both types support asynchronous communication either can be employed, the decision on which to use lies with the Business analyst.

Components defined by Business Analyst:

- Applications eg. KP
- Business Objects eg. Currencies
- Events eg. Create_Currency

- Structural mappings between KP/11i and external objects
  eg. KP/Currency to Bank/Currency
- Data transformations between mappings
  eg. Bank/Exchange_Rate DateTime expand and map separately to KP/Exchange_Rate Date and Time
- Each event an application publishes
- Each event to which an application subscribes
- Cross-Referencing

Components defined by Developer:

- Within iStudio define the Stub API for each event to which an application subscribes. This API is a standard Oracle procedure written in PL/SQL, which receives data from a message and calls an appropriate Integration API.
- The Integration API applies any integration specific business logic to the received data before apply to destination application.
- An application must have the ability to seamlessly generate messages for each event it publishes. These messages will be generated via standard Oracle Insert/Update/Delete Triggers placed on appropriate tables. These triggers are written in PL/SQL and call InterConnect supplied APIs for generating and publishing InterConnect messages.

Generating the Integration code

The only component of this integration which requires generation is the STUB API's which are exported from the InterConnect repository via iStudio using the "export" option within the file menu.

After completing the repository definition InterConnect Stub API procedures must be exported and applied into KP and 11i. The iStudio product has an export utility, which generates creation scripts for all APIs.
The scripts generated include:

- If an application subscribes (eg. Create_Currency) to a Business object event, a package, with same name as business object, will be generated (eg. Currency). The package contains a procedure for each Event on the business object to which the application subscribes
  Eg. Currency.Create_Currency and Currency.Update_Currency

- If an application publishes (eg. Create_Currency) a business object event, a package, with same name as event, will be generated (eg. Debtor). The package contains a "Generate Message" and "Publish Message" procedure for each event the application publishes on the business object.
  eg. "Currency.Generate_Message_Create_Currency"

Error Monitoring/Alerting

There are a number of points within the processing cycle of an Interconnect message where errors can potentially occur, therefore a robust error management system must be designed to ensure all errors are captured and alerted to the appropriate person(s). This section describes potential errors, a solution for managing these errors and a message management process.

InterConnect Message cycle

1. Message creation – a database trigger or batch procedure generates and publishes the message using standard InterConnect APIs. The publishing process inserts the message into a local message queue where it is dequeued by the applications InterConnect adapter.
2. Transmitting message to InterConnect Hub - the application's InterConnect adapter dequeues the message then performs mapping and transformation tasks as defined in the InterConnect repository. After successfully performing these tasks the modified message is placed into the InterConnect Hub queue, from there InterConnect alerts subscribing applications of its existence.

3. Applying Message into subscribing application – the subscribing application's adapter, after being alerted of the message, dequeues the message from the InterConnect Hub queue and performs mapping and transformation tasks as defined within the InterConnect repository. After successfully performing these tasks data from the message is passed thru an Oracle procedure. This procedure exists within the subscribing database and contains logic necessary to apply the data into the subscribing application.

InterConnect Message cycle - potential errors

1. Message creation

Potential Errors

a. Operator's transaction triggers the generation of an integration message via a database trigger, but the integration trigger fails for some reason..
b. Batch processing procedure triggers the generation of integration message(s) but fails to generate a message(s) for some reason.

Notification Process

a. Any failure that occurs within a trigger will result in the transaction failing and the operator will receive the error. Operator rectifies error and re-submits or contacts the systems administrator if unable to resolve.
b. Any failure that occurs within a batch procedure will result in the posting process to fail and errors will be logged to the standard batch processing error management system. These errors will be identified and resolved by the systems administrator and the batch re-submitted.

2. Transmitting message to InterConnect Hub

Potential Errors

a. Publishing application's InterConnect adapter is unavailable
b. Adapter encounters errors during the mapping/transformation/enqueue process

Notification Process

a. Oracle Enterprise Manager configured to alert (via email) when any InterConnect adapter becomes unavailable
b. Custom developed Monitoring application polls the adapter error logs and emails alerts to defined administration group when error encountered within log.

3. Applying message into subscribing application

Potential Errors

a. Subscribing application's InterConnect adapter unavailable
b. Adapter fails to successfully map and transform message
c. Adapter fails to invoke a call to Oracle procedure or procedure encounters fatal database type errors while processing message

Notification Process

a. Oracle Enterprise Manager configured to alert via email when any InterConnect adapter becomes unavailable
b. Custom developed Monitoring application polls the adapter error logs and emails alerts to defined administration group when error encountered within log.
c. as above.

Installing InterConnect

Installation of Repository

Install the product InterConnect, from the Oracle 9iAS media, into its own ORACLE_HOME. All InterConnect components excluding adapters and iStudio will now exist within the new ORACLE_HOME location. Assuming you already have an Oracle database defined to contain the InterConnect repository the next step will is to run the appropriate scripts which define an oaihub902 schema within this database and all objects which constitute the repository.

Installation of Adapters

The process of installing the database adapter into an application has the following impact:

- Each adapter is installed into either the ORACLE_HOME of the application or Oracle InterConnect ORACLE_HOME the choice is yours. The directories created during installation contain adapter specific executables and '.ini' files
- Post installation tasks involve the creation of a new schema within the application database. This schema contains the Interconnect message queues monitored by the adapter and is where all InterConnect Stub APIs will be installed.
- Adapters are started manually using the supplied start scripts or via OEM.
- With Release 2 of 9iAS it is now possible to have multiple adapters within the same ORACLE_HOME, this can be accomplished by using the supplied "copyAdapter" utility. This functionality may be useful if multiple FTP adapters are required.

Oracle Enterprise Manager

If any of the KP or 11i InterConnect Adapters are unavailable the integration will not complete. Messages are never lost when an adapter is down all that occurs is that:

- Outgoing messages from the application remain queued within its resident InterConnect message queue
- Incoming messages remain queued within the InterConnect Hub's application queue

Oracle's product Oracle Enterprise Manager can be configured to manage all InterConnect components.

- Stop & Start Repository Server(s) and Adapter(s)
- Modify configuration of existing adapters
- Dispatch alerts upon failure to detect repository servers and/or adapters at an individual level
- View adapter log files
- Ability to resubmit certain failed messages

Deploying the Integration

When the GPMS is deployed it will contain the following additional components for integration:

- KP/11i/ External InterConnect Repository – export/import the repository with the InterConnect supplied scripts oaiexport/oaiimport.
  The oaiexport script prepares the repository for deployment and produces a dump file using the standard Oracle "exp" utility.
  The oaiimport removes the existing repository schema, re-creates the schema, performs a standard Oracle "imp" then applies a post installation process.
- Integration database triggers, Stub APIs, Integration APIs and KPAPI views/triggers – these will be applied to KP/11i as part of its standard application releasing process.
- Error Monitoring / Alerting application

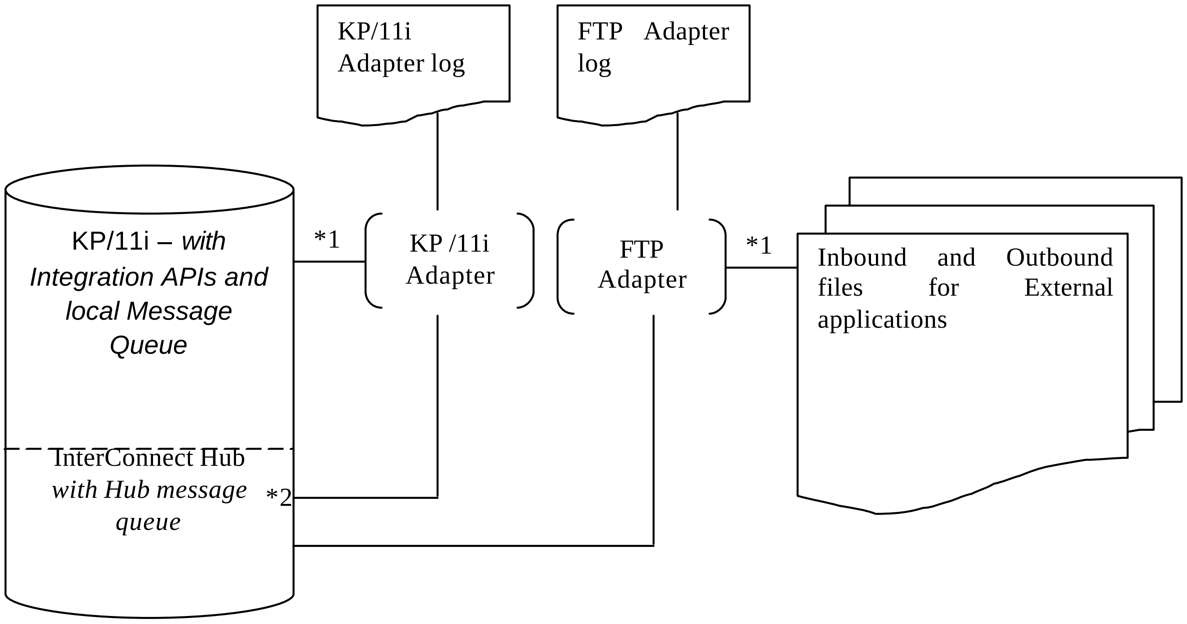
InterConnect Hub Configuration

Installation of the InterConnect product must be completed before deployment of the KP/11i/External integration solution. It is possible to configure the Hub to reside within its own database or co-exist in an existing database. Both options have their merits and the appropriate configuration can be determined and changed at any stage

Combining InterConnect Hub into KP Database

Within this configuration when the KP database is unavailable so is the InterConnect hub. In this situation the following occurs:

1. Messages generated by external applications cannot be transmitted to the hub resulting in the message files remaining queued in its local storage location. The External application continues to function uninterrupted as integration messages are continuing to be produced and queued.
2. The FTP adapter will log errors indicating the inability to communicate with the InterConnect hub. When the KP database becomes available the FTP adapter automatically reconnects with the hub and begins processing messages queued within its local storage location.
3. The KP adapter will log errors indicating the inability to communicate with both the application and hub. When the KP database becomes available the KP adapter automatically reconnects with the application and hub then begins processing both incoming and outgoing messages.


KP/11i Adapter log
FTP Adapter log
KP/11i – with Integration APIs and local Message Queue
InterConnect Hub with Hub message queue
*1
KP /11i Adapter
FTP Adapter
*1
Inbound and Outbound files for External applications
*2

*1 - OutBound Message Service

*2 - InBound Message Service

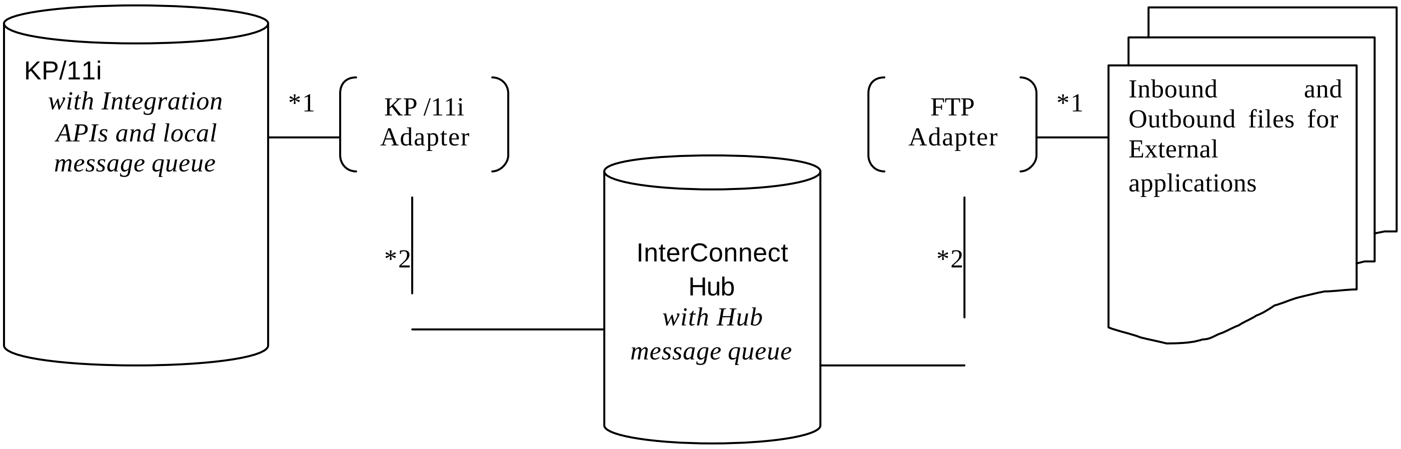
InterConnect Hub configured into own Database

Within this configuration when an application's database becomes unavailable the following will occur:

1. The InterConnect adapter associated with the unavailable application will log errors indicating the inability to communicate with the application. Messages inbound to the application remain queued in the InterConnect hub.
2. The other application continues unaffected, it processes outbound integration messages and dispatches them to the InterConnect Hub. It won't be receiving any inbound messages until the other application becomes available.

When the Hub database becomes unavailable the following will occur:

1. Messages generated by each application cannot be transmitted to the hub resulting in them remaining queued in their local message queue. The application continues to function uninterrupted as integration messages are continuing to be produced and queued.
2. All InterConnect adapters will log errors indicating the inability to communicate with the InterConnect hub. When the hub database becomes available all adapters automatically reconnect with the hub and begin processing messages within their applications local message queue.


KP/11i
with Integration APIs and local message queue
*1
KP /11i Adapter
*2
InterConnect Hub
with Hub message queue
FTP Adapter
*2
*1
Inbound and Outbound files for External applications

*1 - OutBound Message Service

*2 - InBound Message Service

Miscellaneous

InterConnect Message processing sequence

By default InterConnect is configured to processes messages in the order received, this default behaviour can be overridden. It is expected the default will remain for this integration to preserve transaction sequence.
eg. Currency is created before an Exchange rate for that currency is defined.

Guaranteed message delivery

The Oracle InterConnect product guarantees a message is always delivered and delivered only once. The integration product is layered on the Oracle RDBMS technology using Oracle Advanced Queuing, and is configurable for message retention, auditing and tracking.

Oracle Products

- Oracle InterConnect v9.0.2 on 9iAS v9.0.2 media
- This will be installed into its own Oracle 9i database.
- InterConnect RDBMS Adapter installed into IC Oracle Home
- InterConnect FTP adapter installed into IC ORACLE_HOME
- InterConnect AQ adapter installed into IC ORACLE_HOME
- InterConnect PeopleSoft adapter installed into IC ORACLE_HOME
- Oracle Enterprise Manager with repository installed into Hub database
- Oracle InterConnect OEM Server service

Oracle services running on Server

- Oracle Management Server
- OEM Intelligent Agent
- InterConnect repository server
- All InterConnect Adapters

Oracle Products installed on BA and Developer NT/Win2K Clients

- Oracle 9i Client
- iStudio v5
- Oracle OEM Client
- Oracle InterConnect OEM client service

About the Author

Darren Wenham began his involved within the IT industry some 15 years ago after completing a degree at the then Auckland Institute of Technology. He has worked his way through many aspects of the industry as a developer, designer, project manager, trainer and pre-sales consultant. Darren’s exposure to Oracle commenced back in the early 1990s when he was a product specialist responsible for an Oracle application used within a number of Local Authorities throughout New Zealand and Australia. During this period Darren was offered the opportunity to join Oracle as their National Technical trainer, which he eagerly accepted. The following period had Darren training many New Zealand DBAs and developers in Oracle’s products including Database, Tuning, Forms, Reports and others. Back in the late 1990’s Darren was approached and offered an opportunity to architect a substantial E-Commerce application for the Ministry of Economic Developments (MED). As an independent consultant and Senior Architect, Darren spent the following 5 years designing many of MEDs E-Commerce offerings including the “Companies Office”, “Personal Properties Securities Register”, “Intellectual Property Office”, back office systems and others. Darren now works as a Senior Consultant for Theta Systems where he has recently been involved as a technical consultant within an Integration project for a UK based firm.