

Oracle 9iAS InterConnect Version 4.1

An Oracle White Paper
May 2001

INTRODUCTION.....	3
FEATURES	3
Clean Separation of Integration Logic from Integration Platform.....	3
Integration Logic	4
Integration Logic Drives Integration Platform	4
Unique Integration Methodology	4
Business Process Collaboration Using Oracle Workflow.....	6
Rapid and Reusable Integration Specification through iStudio.....	7
Complete Lifecycle Support	8
Based on Robust Oracle Infrastructure	8
Prepackaged Adapters	9
Agent/Bridge Combination	9
Extensible through SDKs.....	10
Standard Messaging Middleware Services	11
Key Value Added Features	11
Flexible Deployment	12
9iAS InterConnect as a Comprehensive Toolkit.....	12
CONCLUSION.....	13
Figure 1: 9iAS InterConnect Metadata and Data Flow.....	4
Figure 2: 9iAS InterConnect Hub-and-Spoke Model.....	5
Figure 3: 9iAS InterConnect Adapter Architecture.....	10

Oracle 9iAS InterConnect 4.1

INTRODUCTION

9iAS InterConnect (formerly Oracle Applications InterConnect or OAI) is a comprehensive application integration framework that enables seamless integration of enterprise software. It is built on top of Oracle's robust integration platform and leverages its underlying services. It is designed to integrate heterogeneous systems, be it Oracle Applications, non-Oracle applications, or 3rd party messaging oriented middleware (MOM). This integration can be deployed either within an enterprise or across enterprise boundaries through the Internet.

The technical design goals for 9iAS InterConnect are as follows:

1. Elevate the integration problem from a technical coding exercise to a functional modeling exercise thereby reducing (eliminating, in the best case) the programming effort normally associated with integration.
2. Develop and expose an integration methodology that promotes reuse and reduces the complexity and management issues that arise over the software lifecycle.

The following features address these goals.

FEATURES

Clean Separation of Integration Logic from Integration Platform

9iAS InterConnect separates the integration problem into two discrete components - high-level integration logic and low-level platform services.

Integration Platform

Integration Platform consists of the integration infrastructure that comes bundled with Oracle9iAS, namely, the **Oracle database with Advanced Queuing (AQ)** and Oracle Workflow. In addition, 9iAS InterConnect provides application and protocol **adapters** which fall into this category. The integration platform provides the requisite infrastructure necessary for integration.

Integration Logic

Integration logic consists of the business rules and transformation logic necessary to integrate applications and heterogeneous systems. **iStudio** is an integration specification design tool that allows you to model this integration logic. The results are then stored in the 9iAS InterConnect **repository** as metadata.

Integration Logic Drives Integration Platform

Integration using 9iAS InterConnect is a two step process. During **design time**, integration logic gets modeled in iStudio and captured in the repository as metadata. At **runtime**, the underlying platform treats this metadata as runtime instructions to enable the conversation among participating applications.

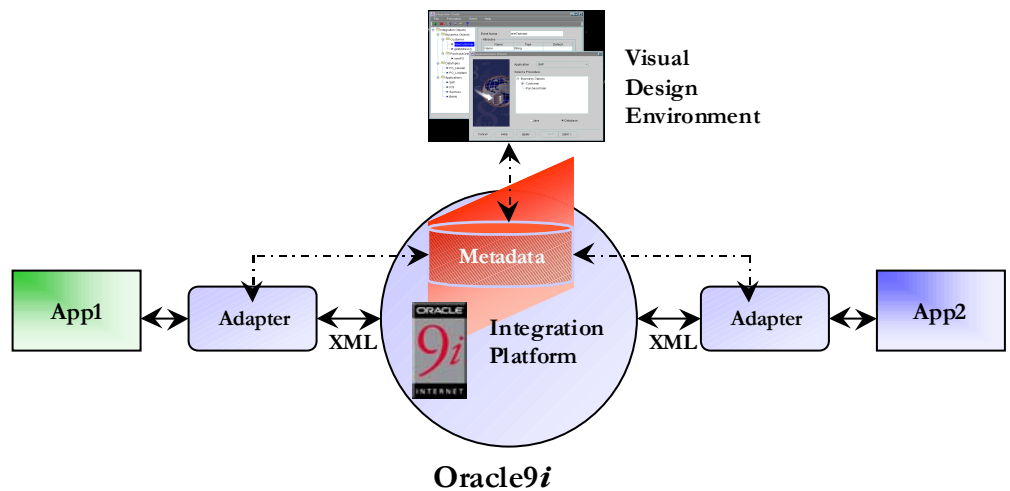


Figure 1: 9iAS InterConnect Metadata and Data Flow

Unique Integration Methodology

iStudio, the design time component of 9iAS InterConnect, exposes an integration methodology that eliminates the complexities of point-to-point custom integration solutions. The integration methodology is based on a **hub-and-spoke** model and works as follows:

An **integration point** is defined as an “event” that triggers communication between two or more participating applications in the integration scenario. Examples of integration points are “Create Customer,” “Cancel Purchase Order,” “Get Item Info,” etc. The **common view** consists of a list of such integration points, each with its own associated data. Applications participate in the integration by binding to one or more of these common view integration points. In the context of each binding, applications have their own **application view** of data that needs to be exchanged. Each binding involves **mapping** (also known as **transformations**) between the application view and the common view in the

context of the integration point. In this model, the application views are at the **spokes** and the common view is the **hub**, hence the name.

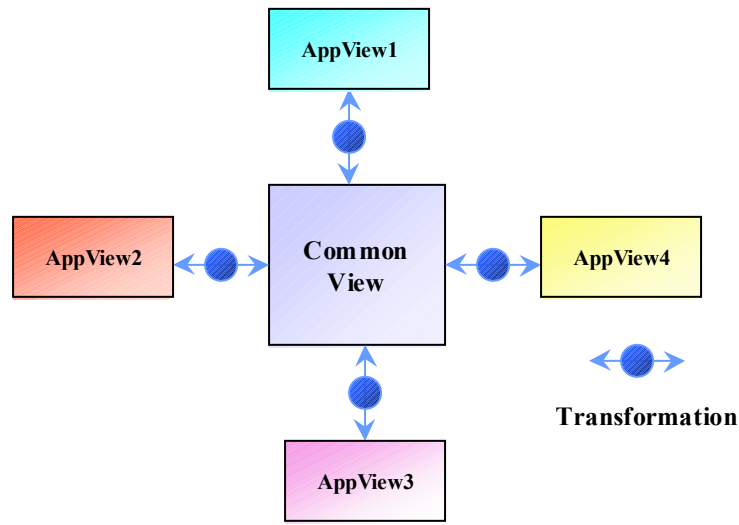


Figure 2: 9iAS InterConnect Hub-and-Spoke Model

This hub-and-spoke model has the following advantages:

1. *Loosely coupled integration.* Applications integrate to the common view, not with each other directly. This dramatically reduces the number of integration interfaces.
2. *Easy Customization.* Changes in application views due to application upgrades are localized. The upgraded application's changes need be reflected only through changes in its application view and mappings to the common view. In other words, only the changed application's spoke needs to be remapped to the hub. The other spokes and their relationship with the hub remains unchanged.
3. *Easy Extensibility.* Applications can be added or removed from the integration scenario surgically without affecting the rest of the applications. For example, if a new application needs to be added to the integration scenario, it must define its application view(spoke component) and map that to the common view(hub) on a per integration point basis. This exercise does not affect the other applications in the integration.
4. *Enhanced Reusability.* This is best explained through an example. If you want to integrate Oracle's iMarketing CRM module to SAP, the integration would be from iMarketing to common view to SAP. Now, if there is a requirement to integrate iMarketing to Peoplesoft, then the iMarketing to common view

integration can be reused. Only the common view to Peoplesoft integration need be built.

Business Process Collaboration Using Oracle Workflow

9iAS InterConnect 4.1 is integrated with Oracle Workflow 2.6 to explicitly capture business processes that drive communication between two or more applications. Until this release, the business process definitions were implicit in the message definitions through iStudio. With 9iAS InterConnect 4.1, you can leverage the robust business process definition and execution capabilities of Oracle Workflow to manage your organization's integration related business processes. Following are some common business problems that can be addressed using this feature:

1. *Error Management.* If there is a problem in a conversation between two or more applications, the errors arising from this problem can be centrally managed and appropriate remedial actions can be defined. For example, it may be a requirement to keep the data of an order entry system in synch with a backend ERP system. Consider the case where a new purchase order is created in the order entry system but the ERP system is down at the time the purchase order is created. At a later time, the ERP system comes back up and an attempt is made to create a corresponding new purchase order through messaging using 9iAS InterConnect. This attempt fails. To deal with this scenario, the integrator can utilize Oracle Workflow to send a compensating message to the order entry system to undo the creation of the purchase order and notify the user who created the order.
2. *Human Interaction.* In previous versions, the conversation between two or more applications was based purely on messaging. Now, human interaction can be added to better capture business processes. In the example above, 9iAS InterConnect and Oracle Workflow can be used to model the following:
 - For every purchase order that is over \$50,000, send a notification to a named approver, and wait for approval. If approved, send the message to the ERP system, otherwise send a message to the order entry system to rollback the order creation.
3. *Message Junctions.* Fan-in and fan-out of messages can be effectively modeled using 9iAS InterConnect and Oracle Workflow. Fan-in involves combining two or more messages into one. Fan-out involves splitting one message into two or more. For an example of fan-in, consider the following. A global organization has a centralized Human Resources ERP application in the United States. Each country has one or more local systems that capture local employee information. If a new employee joins

the Japanese branch of this organization, data is entered into a local HR application and a local Benefits application. Each entry launches a message for adding this information to the centralized system. However, the centralized system needs data from both systems combined and will only commit the data if it was entered successfully in both the local systems. Using Oracle Workflow, this process can be modeled so that 9iAS InterConnect routes messages from both local systems to Oracle Workflow, Oracle Workflow waits until it receives both messages, combines the data, and launches a single message to be delivered by 9iAS InterConnect to the centralized HR system.

4. *Stateful Routing.* 9iAS InterConnect provides extensive support for stateless routing through event based and content based routing features. Using Oracle Workflow, you can now do stateful routing. In other words, the decision to route can be based on the state of the business process in addition to just the event or the content of the message.
5. *Composite Services.* Using all of the above examples, an internal (organization focused) or external (customer/partner focused) service can be built through a well defined set of business processes involving communication between two or more applications. For example, a brick-and-mortar retail company could provide an on-line procurement service to their customers. Behind the user interface would be several business processes controlling communication across several internal applications to deliver a robust and performant service to the customer.

Please refer to the 9iAS InterConnect User's Guide for more information on this functionality.

Rapid and Reusable Integration Specification through iStudio

iStudio is 9iAS InterConnect's design time integration specification tool targeted at business analysts to help them specify the integration logic at a functional level, not at a technical coding level. It exposes the integration methodology described above through simple wizards. It drastically reduces (eliminates, in the best case) the need for writing code to specify the integration logic, thus reducing the total time required to complete your integration.

In addition, with this release, iStudio is integrated with Oracle Workflow tools, namely Workflow Builder and Workflow Home Page. This tool integration provides a seamless business-process collaboration definition capability in addition to the core 9iAS integration specification features.

Since iStudio is a multi-user tool with fine grained locking for all 9iAS InterConnect "first class objects," multiple users can work simultaneously on the same integration scenario without compromising the consistency of the metadata.

Complete Lifecycle Support

Managing, customizing, and evolving your integration over time is equally, if not more important than creating the integration in the first place. The hub-and-spoke integration model explained above has some obvious advantages that help achieve this goal. In addition, the 9iAS InterConnect repository, which contains all the integration logic, provides extensive services for managing changes over time. The repository provides **fine grained versioning** of all 9iAS InterConnect “first class objects” such as events, messages, data types, etc. The following features highlight some of the important aspects of versioning that help with the lifecycle support:

1. *Basic Versioning.* New versions of first class objects such as messages can be created to address changing integration needs. Different versions of the same object can co-exist in the repository. This has two advantages. One, it eliminates the need for an expanded namespace to address modifications. And two, it allows related entities to be grouped together for easier management.
2. *Multiple Active Versions.* Multiple versions of the same message can be active in the same integration scenario simultaneously. This can help transition your integration incrementally without resorting to a “big bang” approach. For example, if an application’s purchase order definition (its application view of purchase order) needs to change, a new version of the CreatePurchaseOrder can be created and activated for that application. Once this metadata is created, the application can smoothly transition from sending/receiving messages based on the old definitions to the new one.
3. *Migration Support.* Different versions of metadata can be migrated across repositories on a first class object basis. This feature gives you fine grained control of what’s in your development repository versus your production repository.
4. *Consistency Control.* 9iAS InterConnect detects and flags metadata conflicts, thus preventing accidental overwriting of metadata. This helps maintain consistency of metadata in the repository.

Based on Robust Oracle Infrastructure

All components of 9iAS InterConnect are written in pure Java and utilize proven Oracle infrastructure to deliver a robust, reliable, scalable and performant integration solution. In particular, **Advanced Queueing (AQ)** in the **Oracle 8i/9i database** provides the standard **Java Message Service (JMS)** interface for messaging. It also provides configurable message retention, auditing, and tracking support. for messages. This technology makes up the messaging backbone for 9iAS InterConnect. In addition, **Oracle Workflow** is utilized by 9iAS InterConnect for business process collaborations. All these components, together with the 9iAS InterConnect **repository**, constitute the **messaging hub** at runtime.

Prepackaged Adapters

Prepackaged adapters help re-purpose your applications at runtime to participate in the integration without any programming effort.

Agent/Bridge Combination

Adapters are 9iAS InterConnect's run-time component and have the following responsibilities:

1. *Application Connectivity*. Connect to applications to transfer data and communicate events between the application and 9iAS InterConnect. The logical sub-component within an adapter that handles this responsibility is called a **bridge**. This is the protocol/application specific piece of the adapter that knows how to communicate with the application. For example, the database adapter is capable of connecting to an Oracle database using JDBC and calling SQL APIs. This sub-component does not know **which** APIs to call, **only how** to call them.
2. *Transformations*. Transform data from the application view to common view and vice versa as dictated by the repository metadata. In general, adapters are responsible for carrying out all the "runtime instructions" captured through iStudio as metadata in the repository. Transformations are an important subset of these instructions. The logical sub-component within an adapter that handles this responsibility is called an **agent**. This is the generic runtime engine in the adapter that is independent of the application the adapter connects to. It "focuses" on the integration scenario based on the integration metadata in the repository. There is no integration logic coded into the adapter itself. Instead, all the integration logic is stored in the repository. The repository contains the metadata that drives this sub-component. In the database adapter example above, this is the sub-component that knows **which** SQL APIs to call (specified through the metadata), but not **how** to call them. All adapters have the same agent code. It is the difference in metadata that each adapter receives from the repository that controls and differentiates the behavior of each adapter.

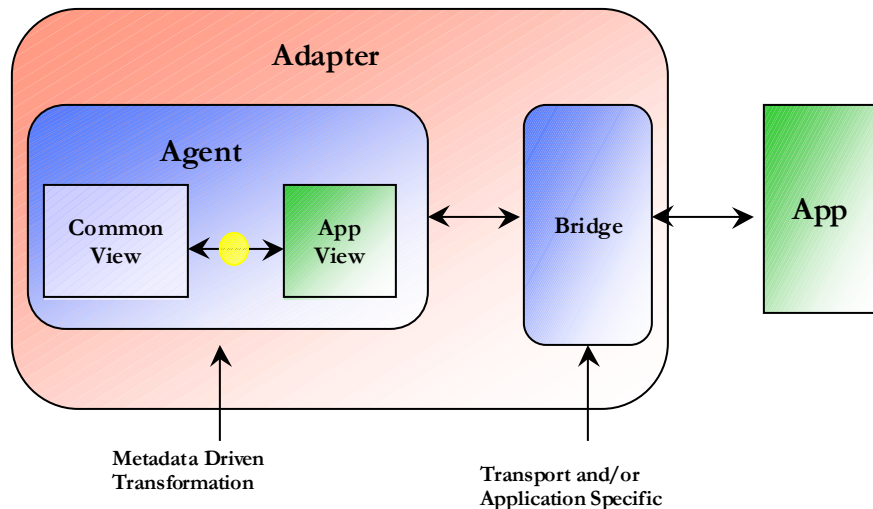


Figure 3: 9iAS InterConnect Adapter Architecture

These adapters can be technology adapters or application adapters. 9iAS InterConnect 4.1 currently packages technology adapters for the Oracle database, AQ, and HTTP/S technology. Access to non-Oracle databases is possible through Oracle's Transparent Gateways. Application adapters include Oracle and SAP R/3 adapters. New adapters are regularly added to this list and released through patches.

Extensible through SDKs

9iAS InterConnect can be easily extended through SDKs to address your custom integration needs. There are two SDKs that give you this flexibility:

1. *iStudio SDK*. With this SDK, you can extend iStudio by writing code in Java to do the following:
 - Add custom transformation functions.
 - Add custom browsers to import API or data structure definitions from application native repositories into iStudio. For example, the database browser (comes standard with iStudio) allows you to import SQL API, table, view, or Abstract Data Type (ADT) definitions from the database to define the **application view** of data for a database based application.
2. *Adapter SDK*. With this SDK, you can write new adapters in Java for applications or protocols not currently supported through 9iAS InterConnect. Specifically, you need to write only the **bridge** sub-component. The **agent** is a

generic engine already written for you and is part of each adapter as described above in the Prepackaged Adapters section.

Standard Messaging Middleware Services

9iAS InterConnect provides all the basic services expected of a messaging middleware platform. Some of the important ones are as follows:

1. *Support for all major messaging paradigms.* Publish/Subscribe, Request/Reply (synchronous and asynchronous) and Point-to-Point.
2. *Guaranteed Delivery of Messages.* All messages are guaranteed to be delivered. They are delivered exactly once and in the order in which they were sent.
3. *Scalability.* Multiple adapters can be instantiated to serve one application. The hub can run in an Oracle Parallel Server environment.
4. *Load Balancing.* Messages can be partitioned based on load between multiple adapters servicing one application. At one extreme, one or more adapters can serve all messages for one application. At the other, one or more adapters can be dedicated per integration point that the application participates in.
5. *Runtime Management.* An Oracle Enterprise Manager based **runtime management console** helps manage the integration scenario and components at runtime. This console allows you to start/stop components, monitor message flow, detect performance bottlenecks, and resubmit messages that have been corrected.
6. *Deployment Support.* The messaging hub consists of AQs that must be configured for runtime. IStudio provides configurable parameters for specifying:
 - how many queues should be created
 - what the names of these queues are
 - what queue each adapter uses and for what purpose

Key Value Added Features

9iAS InterConnect goes beyond just providing standard messaging middleware services. In addition, it exposes key features that help reduce the integration effort. All these services are exposed through iStudio and do not require any coding:

1. *Content Based Routing.* Messages can be routed by building business rules based on message content. For example, a procurement system can route fulfillment requests to different fulfillment centers based on originating location.

2. *Cross Referencing.* Keys for corresponding entities created in different applications can be correlated. For example, a purchase order created in a procurement system has a native id X. It then gets routed to a fulfillment system. The purchase order gets created in the fulfillment system with native id Y. X and Y must be cross referenced so that 9iAS InterConnect can correlate communication about this same logical entity in two different systems without the systems knowing about each other's native ids.
3. *Domain Value Mapping.* Code tables can be mapped across systems. For example, a purchase order in a procurement system might have a PO Status field with possible domain values {booked, shipped}. The corresponding field in a fulfillment system might have the domain value set {1, 2}. 9iAS InterConnect allows you to create the following mappings {booked=1, shipped=2} so that it can correlate these values at runtime without the systems knowing each other's domain value sets.

Flexible Deployment

9iAS InterConnect provides a complete framework for e-Business application integration across the Application-to-Application (A2A), Application Service Provider (ASP), and Business-to-Business (B2B) domains. 9iAS InterConnect components can be deployed for all of the above domains as described below:

1. *A2A.* Applications are distributed within a Local Area Network (LAN) or across a Wide Area Network (WAN). 9iAS InterConnect is deployed within the organization to integrate these applications.
2. *ASP.* Applications are distributed across firewall boundaries with some applications residing inside the ASP firewall and others inside the customer firewall. 9iAS InterConnect can be deployed inside either one of the two firewalls or inside both to integrate these applications across the firewalls.
3. *B2B.* This is similar to the ASP model if you replace the ASP with another customer.

9iAS InterConnect as a Comprehensive Toolkit

9iAS InterConnect provides three different levels of service to the end customer based on their integration needs:

1. *Pre-built Integrations.* This includes pre-packaged adapters **and** integration logic captured in the repository through iStudio. For example, Oracle iProcurement to SAP R/3 is a pre-built integration solution available today, through Oracle.

2. *Pre-packaged Adapters Only.* Adapters are pre-packaged but the integration logic needs to be built through iStudio. A variety of adapters come with Oracle 9iAS InterConnect, and new adapters will be released by Oracle and its partners.
3. *SDK.* When access to applications or systems are required, for which no pre-packaged adapter exists, the Software Developers Kit can be used to build new adapters. The new adapters can be used in conjunction with existing adapters to meet the customer needs.

CONCLUSION

Oracle today, is uniquely positioned to provide customers with a comprehensive integration solution. Oracle's e-Business Suite, a complete, integrated suite of CRM and ERP applications hosted on the Oracle 9i platform, give Oracle extensive insight into the functional aspects of integration. Oracle 9i consists of a highly scalable, reliable, feature-rich platform which include all the underlying services necessary for complex integration. Finally, 9iAS Interconnect goes beyond the traditional 'hard-coded' integration methodology and, instead, focuses on integration at the functional level. 9iAS InterConnect consists of a growing set of pre-packaged adapters for popular applications and systems as well as a framework for designing, deploying and managing your integration solutions within and across the enterprise.



Oracle 9IAS InterConnect 4.1
May 2001
Author: Rahul Pathak
Contributing Authors: Gordon Callan

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Oracle Corporation provides the software
that powers the internet.

Oracle is a registered trademark of Oracle Corporation. Various
product and service names referenced herein may be trademarks
of Oracle Corporation. All other product and service names
mentioned may be trademarks of their respective owners.

Copyright © 2001 Oracle Corporation
All rights reserved.