

COMMON INTEGRATION STYLES AND ORACLE FMW 11G

SOA differs from previous integration paradigms due to its explicit use of standards. Oracle SOA Suite technologies enable organizations to expose functionalities from systems as standards-based services that can be consumed by other applications or business processes. For most organizations, integration can be divided into one of four different categories:

- Application Integration
- Process Automation/Integration
- Event Driven Integration
- Data Integration

Following an explanation of the integration types and the various Oracle FMW 11g products is an explanation drawing on use case examples of how the Oracle products can be used to implement each type of integration. A discussion of governance and applicable Oracle technologies concludes the white paper. By the end of the paper, not only will you understand how Oracle FMW 11g can be used for different integration scenarios, but you will also have a high level understanding of the Oracle SOA Suite 11g product stack, and its reference architecture.

INTEGRATION CATEGORIES

There are four general types of integration, which are:

Application Integration is the process of bringing data or a function from one application program together with that of another application program. An example of Application Integration would be integrating HR, Financial, or CRM systems with legacy applications, partner applications, or each other.

Process Integration/Automation/Workflow focuses on integrating and automating business processes. Even today most organizations still have many business processes and workflows that are executed and managed manually. Often these processes span across many business units and systems. Optimizing and automating these business processes will lead to improved efficiencies and lower costs.

Event Driven Integration can be thought of as an asynchronous publish and subscribe model or pattern. Systems trigger events by publishing them and other systems that need to take action on these events subscribe to them. Event Driven Integration differs from other integration types because publishers are completely unaware of the subscribers and vice versa. Event Driven Integration can contain both simple and complex event processing. Complex Event Processing is when multiple events occur that trigger other events. Complex Event Processing involves processes that are typically non-linear, dynamic, and continuously influenced by outside events.

Data Integration by definition involves combining data residing in different sources and providing users with a unified view of the data. One example of data integration would be an Extract Transform and Load (ETL) process that pulls information from source databases and loads a data warehouse. Basically Data Integration is about moving the data to where it's needed. To accomplish this software needs to address the distribution, consolidation and synchronization of data.

ORACLE FMW INTEGRATION TECHNOLOGIES

Before mapping Oracle FMW 11g products to each of the Integration Categories, I want to give an overview of the technologies.

Oracle BPEL Process Manager enables enterprises to orchestrate disparate applications and Web Services into integration processes and business

processes. BPEL can create and manage both short and long running multi-step processes. BPEL is used for integration, to build services for a Service Oriented Architecture (SOA), or to manage workflow tasks. BPEL is often used for Application Integration, Process Integration and can play a role in Event Driven Integration. More specifically, BPEL BPM can be used to:

- Build end-to-end integration solutions
- Build services for a Service Oriented Architecture
- Expose legacy applications as services
- Manage workflow processes

Oracle Service Bus is a highly scalable, lightweight platform designed to connect, mediate, and manage interactions between services, legacy applications, and packaged applications. It's commonly used for configuration-based, high volume Event Driven Integration and Application Integration.

Oracle Business Activity Monitoring is a software solution for building interactive, real-time dashboards and proactive alerts for monitoring business processes and services.

Oracle Business Rules enables business analysts and non-developers to define and modify business logic without programming changes.

Oracle Business Process Management is a set of tools for creating, executing, and optimizing business processes. BPM supports the complete business process lifecycle and enables collaboration between business and IT to automate and optimize business processes. With BPM, business analysts can model business processes and easily transition the process flows to IT for implementation. Once the business processes have been modeled, users can run simulations to identify areas for process improvement. BPM is primary tool in the Process Integration category and is targeted at workflows that contain human interactions.

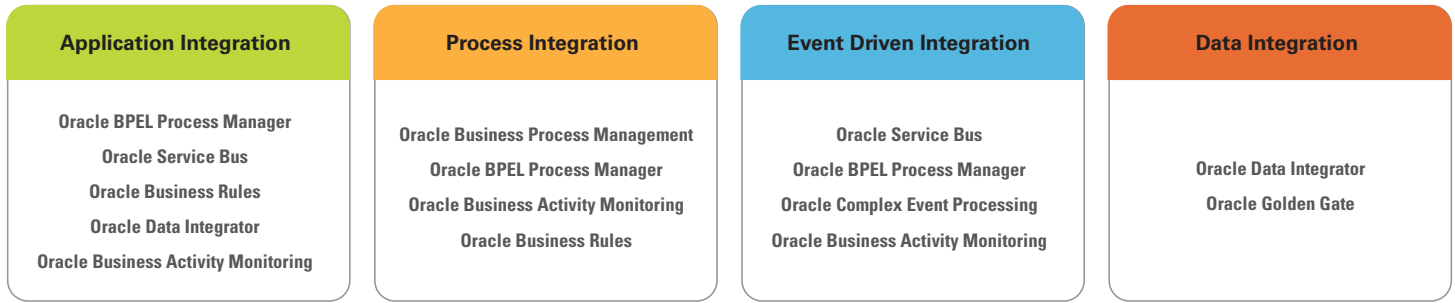
Oracle Data Integrator is a data integration platform that covers all data integration requirements from high-volume, high-performance batch loads, to event-driven, trickle-feed integration processes. ODI enables extract, transform and load (ETL/E-LT) for handling large data payloads.

Oracle Complex Event Processing is a declarative environment for developing event processing applications to improve the effectiveness of business operations. CEP can process multiple event streams to detect patterns and trends at real time and provide enterprises the necessary visibility via monitoring to capitalize on opportunities or mitigate developing risks. CEP features include pattern matching and user-defined windows for event evaluation.

Oracle Golden Gate is a high-performance bidirectional data replication solution that provides real-time data integration and data delivery between heterogeneous systems. It helps organizations eliminate downtime caused by both unplanned and planned outages. According to Oracle, Golden Gate is especially effective for:

- Real-time data warehousing – Provide continuous, real-time capture and delivery of the most-recent change data between OLTP systems and data warehouse. GoldenGate integrates with ODI EE and other ETL/E-LT solutions.
- Disaster recovery and data protections. Create and maintain an immediate failover with up-to-the-minute data to minimize recovery time for mission-critical systems – deploy with Oracle Database or in non-Oracle environments.
- Query offloading. Ensure high performance for productions systems while still supporting necessary read-only activities by replicating data between heterogeneous sources and targets.

ORACLE FMW INTEGRATION TECHNOLOGIES



APPLICATION INTEGRATION

Application integration is a very common scenario that is often driven by the addition of new systems or acquisitions and mergers. The characteristics or motivation behind legacy application integration include scenarios where:

- Integration work must be done as quickly as possible
- Legacy applications will change very little or not at all – this is because there may be no legacy resources available, limited knowledge or access to legacy system source code
- Legacy applications have limited or no Web Services capabilities
- There are routing requirements
- There are transformation requirements

Integration with legacy systems using products like BPEL and OSB is accomplished through adapters or connectors. A logical architecture of legacy systems integration would look similar to Figure 1. The adapters access the legacy system using existing legacy system integration methods and API's. The primary out of the box adapters for integrating with legacy systems are:

- Database adapter – run SQL queries, select, insert, update, delete, you can also call Stored Procedures
- Queuing adapters – JMS, AQ, and MQSeries
- File adapter – read, write files. You can also poll a directory so when a file arrives it can trigger a process
- FTP adapter

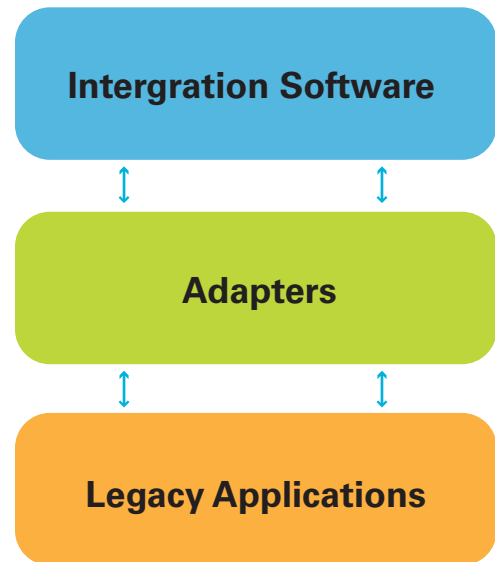


Figure 1

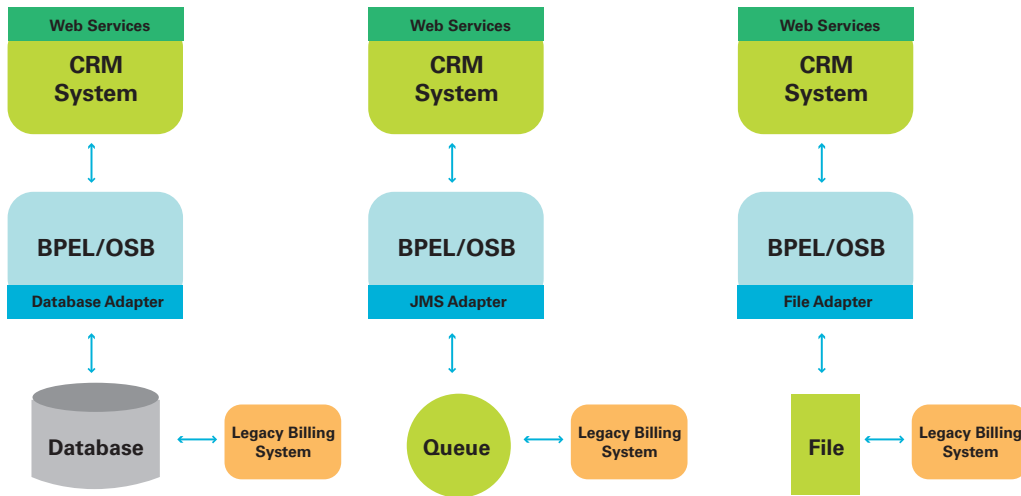


Figure 2

An example of legacy system integration could be a package application, service, or partner application that needs to integrate with a legacy system.

Figure 2 shows a CRM system that needs to exchange transaction data with a legacy billing system. The billing system has API's that can be called by a database stored procedure, or integration can be accomplished through database staging tables. BPEL can be used with the out-of-the-box Database Adapter to implement the interface and process the transaction on the legacy Billing System. The same outcome can be accomplished using other adapters if the legacy system supports queuing or file-based API's. These three adapters are very commonly used to interface with legacy systems.

Another common scenario arises when information from a legacy system is needed by several systems across the enterprise. Many of these systems support Web Services and you want to provide access to the legacy system as a set of services. Examples could be things like customer information lookup, order history, or order entry. Use BPEL or the Service Bus to wrap the legacy system and provide access via web services.

MULTI-SYSTEM INTEGRATION

The characteristics of a multi-system integration use case are similar to the Application Integration, except that here integration requires the orchestration of information from many different systems, services, or data sources. Often business rules will drive the process flow.

Multi-System Integration Characteristics:

- Changes in integration driven by the addition of new systems or acquisitions and mergers
- Integration requires information from multiple systems and/or services
- Business rules drive process flow
- Routing requirements
- Transformation requirements

The high level logical architecture in Figure 3 shows the integration layer connecting to many different systems and services as part of a given integration process. Business rules may need to be applied to make decisions or perform calculations. Information from the different sources can be combined and transformed to form to a canonical model or target system formats.

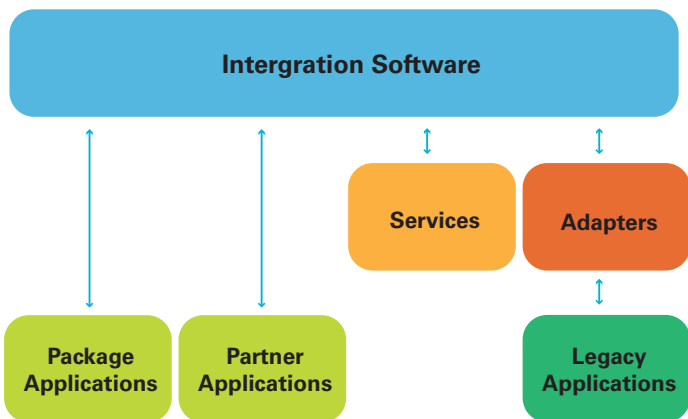


Figure 3

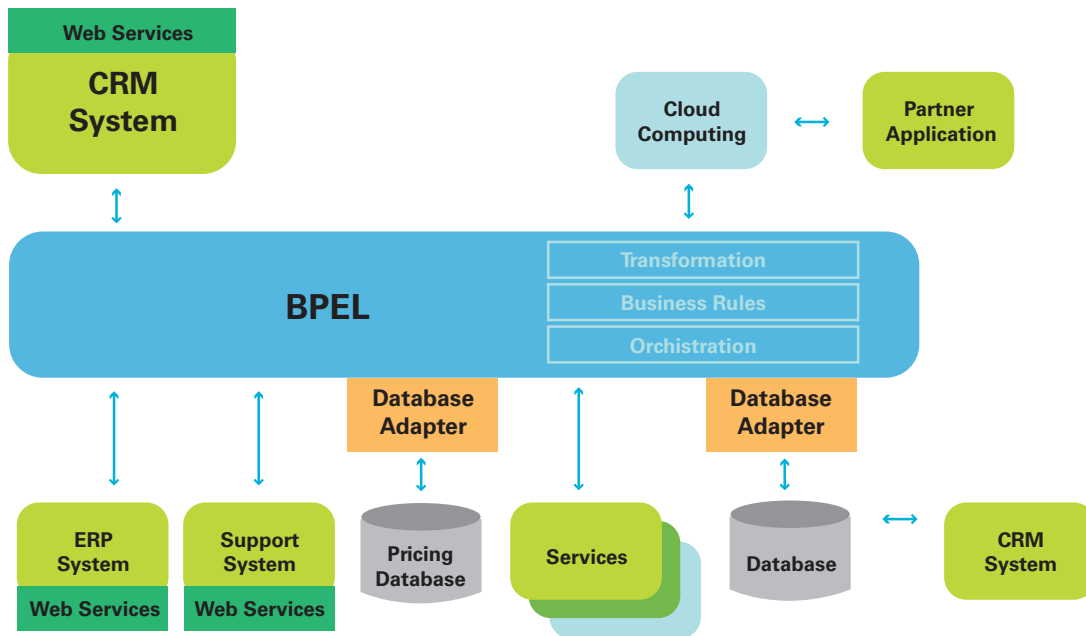


Figure 4

An example multi-system integration might involve a partner application that needs customer history information to complete transaction processing. To complete the integration process, information from several sources needs to be combined and transformed into the partner application format. Figure 4 shows an example of a CRM system making a request to a customer history service that is implemented in BPEL. The BPEL process orchestrates the gathering of information from several sources. For example, customer information from the CRM system, order history from the ERP, customer support history, status and pricing information from a database, information from various services, and billing history from a legacy system. BPEL then combines the information, applies business rules, and transforms the data into the required format before calling a service to complete the transaction on the partner application.

Even though this example looks complicated it can be easily implemented using tools like the BPEL Process manager, Oracle Service Bus, and Oracle Business Rules Engine.

The benefits for using Oracle Fusion Middleware 11g integration tools for application and multi-system integration projects are flexibility, speed and reuse. Oracle SOA Suite, in particular, can create provide for integration between heterogynous systems and provide a foundation for services based integration or a full blown SOA.

PROCESS INTEGRATION

Process integration focuses on automating business processes.

Characteristics for this type of integration are:

- Key business processes span functional units, systems, and applications
- Multiple human touch points and handoffs
- Unmet SLAs
- Little insight into process status (e.g., where you are at in a given instance of a process and how long it took to complete a given step)
- A desire to do more with less staff

The most commonly used products are the BPM and BPEL products, because both can be used to implement business processes containing human workflow steps.

In the high-level architecture shown in Figure 5 the BPM layer has to coordinate and interface with process participants. The BPM layer will also use integration components to connect to existing assets using WebServices and Adapters/Connectors, similarly to application integration.

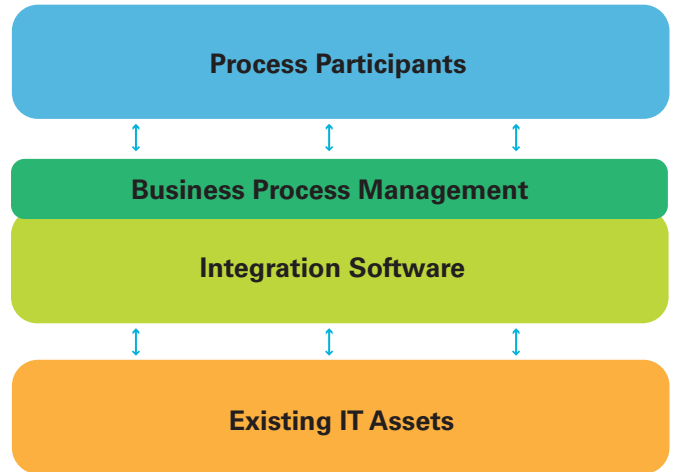


Figure 5

The order entry workflow example shown in Figure 6 helps explain process integration. In this example the customer enters an order through an Order Entry Portal. Throughout the workflow there are several different decision points. Examples are; taking action based on a customer’s status, what to do with an out of stock item, and processing order items that are eligible for special pricing. The order process spans three different business units: customer service, sales and marketing, and shipping. There are several steps along the way that require a business person to take action on the order.

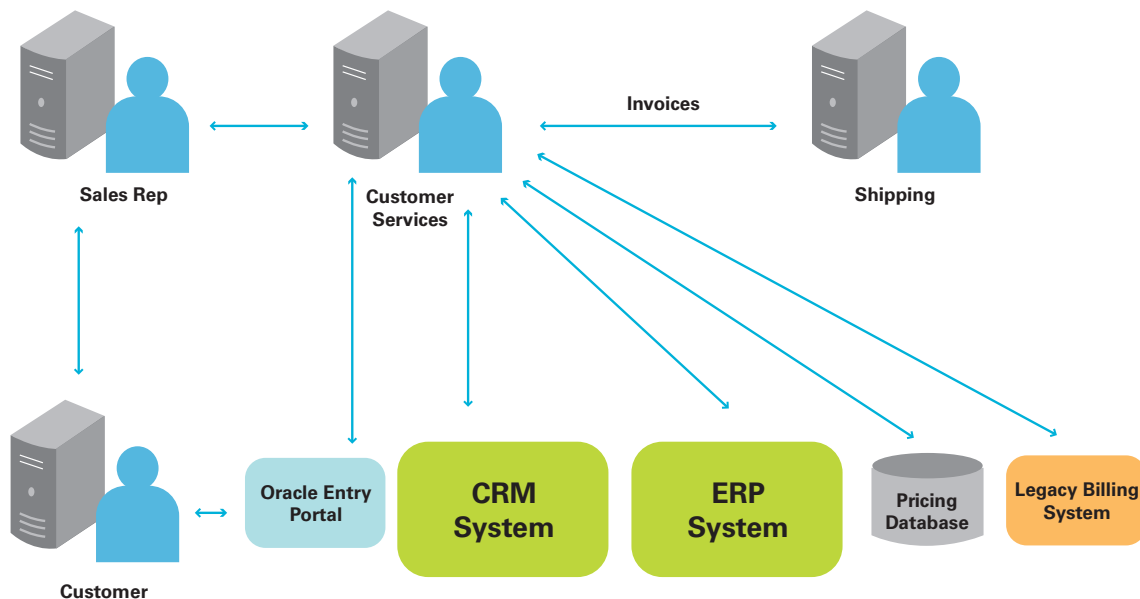


Figure 6

Without business process management many of the steps for order processing would have to be done manually. To complete the order process users will have to access several different systems for data lookup and data entry. Handoffs between users are usually done through email or some type of paper based process.

This approach is inefficient, is prone to error, and lacks clarity, as management has very little visibility into key business processes. It also has a big impact on business agility, since manual business processes that span multiple business units make it difficult to launch new business initiatives and take advantage of changing market conditions.

ORDER ENTRY WITH PROCESS INTEGRATION

With BPM, many of manual steps in the workflow can be automated, reducing the number of human touch points. BPM provides a worklist user interface where users can go to see and manage all of their assigned tasks. This helps improve efficiency because all of the information can be displayed in one place and users will no longer have to use several different systems to complete a given task.

In the automated example shown in Figure 7 – Order Entry with Process Integration, the customer places the order through a portal. BPM receives the order and uses integration components to automatically look up the customer information and check inventory. At this decision point, if the items are in stock then BPM will retrieve pricing information and evaluate if the order qualifies for special pricing. If so, the order gets sent to customer service for approval. Everything up to this point has been automated.

Using BPM's worklist UI, Customer Service reps can view all of the relative order information from various systems in one place and either approve or reject the special pricing. BPM will then route the order to the shipping department. When the item is shipped, the user will release the order and the system will automatically update inventory, billing system and send out an email to the customer providing them with the shipped date and tracking information.

BPM applies business rules to make decisions on the flow of a process. For example, if an item was out of stock there would be a different set of steps required to complete the order.

Overall, BPM improves efficiency, visibility, and agility. This translates to improved customer service levels, enhanced productivity, reduced costs, and increased competitiveness.

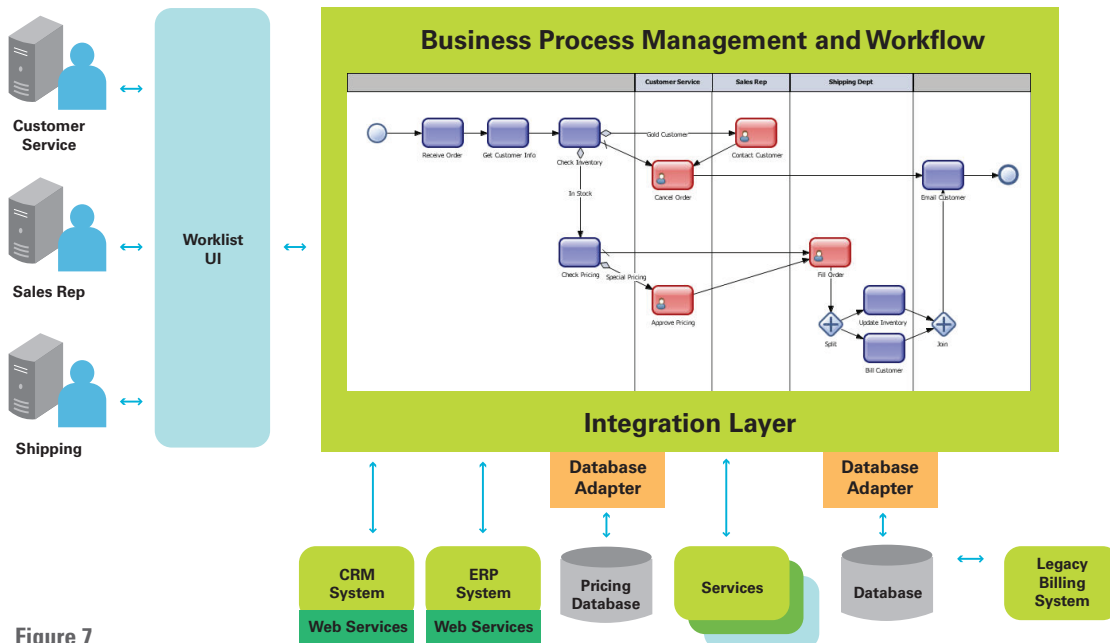


Figure 7

EVENT DRIVEN INTEGRATION

There are two types of event processing, simple and complex. Complex event processing (CEP) can be applied to solve business challenges in fraud prevention, risk management, and compliance verification. With complex event processing:

- Multiple systems independently need to take action on a given event or events
- Processes are unpredictably non-linear, which means that timing and sequence cannot be defined in advance, so you don't know when the events are going to occur and you don't know the sequence in which they will occur. When specific processes occur, specific actions need to be taken.
- Processes are dynamic (they change all of the time)

In the logical architecture shown in Figure 8, events can come from diverse sources, such as people, applications, data sources, networks, and devices. These events are streamed through the integration layer where event processing evaluates the data to detect patterns in real time. Once a pattern is matched, CEP can generate its own events, feed monitoring software, or perform some downstream action by integrating with existing IT assets or services.

The financial industry has many business processes that can use CEP to help monitor for fraudulent activities. In Figure 9, the insurance, banking, annuity, mutual fund, and the CRM systems can all feed events into the event stream. Oracle's CEP software can correlate events, apply pattern matching, business rules and event decisioning in real time, looking for fraudulent activities. CEP can constantly correlate balance changes in banking accounts, surrendering of insurance policies, free-looking of annuity policies, and the buying or selling of securities. If fraud is detected, CEP can take action by sending notifications or executing transactions on internal or partner applications.

According to Oracle, CEP:

- Enables real-time situational awareness for faster decision making and better customer service
- Decreases costs and improves compliance by identifying and proactively responding to business threats such as fraudulent activities
- Improves operational efficiencies by providing immediate insight into things like supply chain and resource utilization

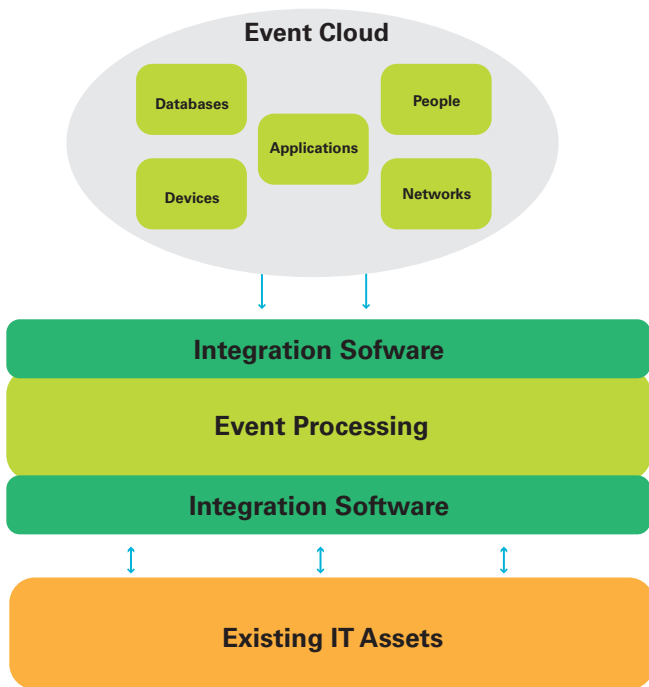


Figure 8

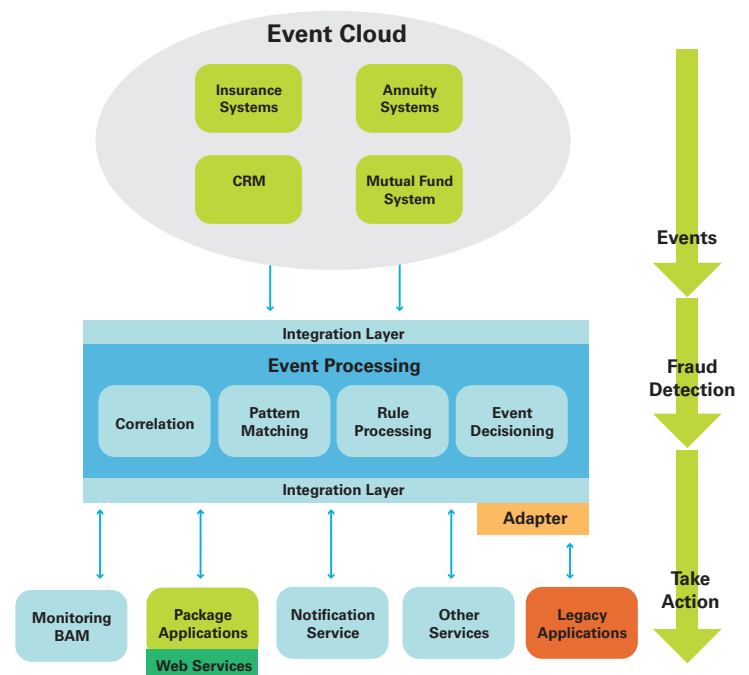


Figure 9

DATA INTEGRATION

Data integration is driven by the need for:

- Reporting on data from multiple sources
- Reporting on operational systems without impacting performance
- Continuous real time data synchronization between multiple data sources
- Data integration and delivery between heterogeneous systems
- Combined data sources to create dimensional data warehouses

In the logical architecture shown in Figure 10, the data integration layer combines data residing in many different data sources to populate data warehouses and other data sources. These new data sources are used by existing IT assets like reporting systems and applications.

The data integration layer is configured to pull data from multiple sources. It manages the real-time distribution, consolidation, and data synchronization needed to create and maintain data warehouses and synchronized data sources.

Oracle Golden Gate, as shown in Figure 11, is a high performance solution for real-time data capture, transformation, and delivery.

According to Oracle, it can be configured to support:

- Disaster Recovery - by maintaining a failover sites with up-to-the-minute data.
- Data Distribution - by synchronizing data from distributed applications in real-time.
- Query offloading - by replicating and synchronizing data from high performance production systems to reporting data sources.
- Real-time Business Intelligence and data warehousing – by providing real-time capture and delivery of changed data between OLTP systems and data warehouses.

Overall, high-performance, real-time data integration leads to enhanced decision making, and improved system performance and scalability.

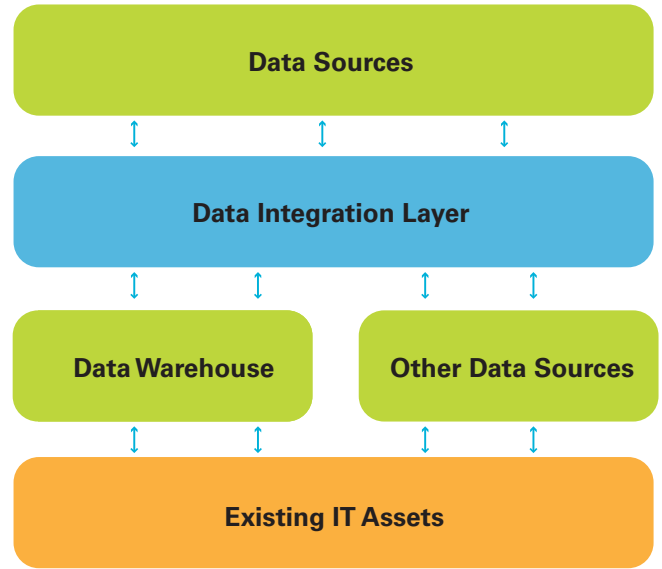


Figure 10

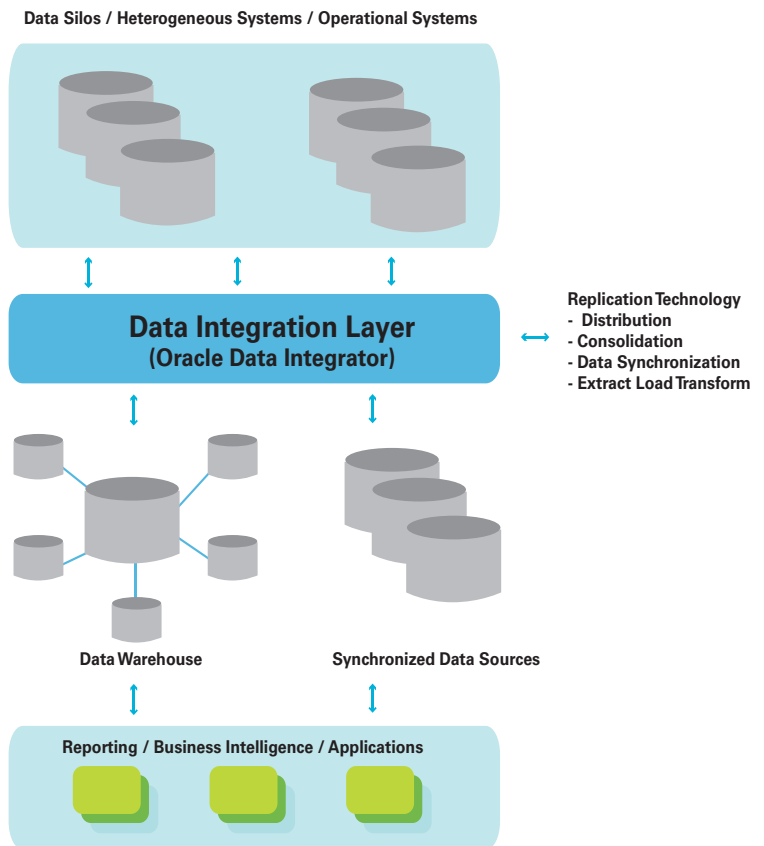


Figure 11

OTHER FMW PRODUCTS AND INTEGRATION

Additional Oracle products available for integration implementations are Oracle Business Rules and Oracle Business Activity Monitoring. Both of which can be used across all of types of integration to help support and enhance the integration effort.

ORACLE BUSINESS RULES

Oracle Business Rules enables you to separate rule logic from underlying application code. This allows more agile rule maintenance and empowers business analysts with the ability to modify rule logic without programmer assistance and without interrupting business processes.

Oracle Business Rules components include:

- Facts and Bucketsets
- Rulesets and Rules
- Decision Tables
- Decision Functions

Business rules are typically statements that describe business policies or describe key business decisions. A loan approval process is one good example of where a business rule could be used. The rule could be setup at key decision points in a workflow for automatically approving a loan or requiring the loan to go through a manual approval process. The workflow could be setup based on an approval status which is derived from a business rule or set of business rules. The business rule can incorporate elements such as customer status, credit score, income, promotional dates, etc. Once in place, the business rule can be modified and maintained by business analysts without programmer assistance. This same rule can be invoked by multiple business processes and services thought the enterprise.

ORACLE BUSINESS ACTIVITY MONITORING (BAM)

BAM provides a collective interface to monitor all sources of information and provides real-time insight into business processes. BAM's message-based, event-driven architecture is specifically designed for real-time monitoring, analytics, and reporting. BAM provides open integration that allows the use of a variety of mechanisms to capture real time business events.

Source data can include:

- Databases
- Messaging systems based on JMS, Oracle AQ, MQ Series,
- ETL solutions like Oracle Data Integrator (ODI)
- Web services.

BAM filters and correlates data events to create graphical views and allows users to create personalized dashboards.

GOVERNANCE

When discussing integration, another topic to cover is governance. When researched online, you'll find that governance in general has to do with making decisions that define expectations, grant power, or verify performance. When applied to Information Technology projects it's safe to say that most organizations have some form of governance already in place.

There are processes to:

- Prioritize projects.
- Define how projects are implemented.
- Monitor solutions after they have been deployed to ensure systems behave appropriately.

Governance is especially important for SOA-based integration projects or projects that use integration software for creating and exposing standards-based services. This is because integration that exposes services creates a more abstract and distributed environment than traditional integration approaches. SOA governance can be viewed as the combination of people, policies, processes, and technologies applied to support an organization's SOA objectives.

It's all about establishing control over an environment and addresses:

- What services should be built.
- How services should be built (what's the right way).
- Understanding what services have been built and how to use them.
- How to ensure that services are working/behaving appropriately.

To be successful with governance, it's best to start out small and grow your governance within the organization over a period of time. There is no need to define and implement too much change all at once. Governance should be implemented iteratively and constantly revisited and refined.

ORACLE SOA GOVERNANCE PRODUCTS

Tools, such as the Oracle Enterprise Repository and Oracle service Registry, provide a great place to start implementing a governance model because they help you visualize what services exist. Additionally, Oracle Web Services Manager help execute governance by enforcing security policies for services, and Oracle Enterprise Management Pack for SOA will monitor performance of those services.

Overall, Oracle provides all the tools needed to control a service lifecycle.

CONCLUSION

There are four basic integration categories:

- Application Integration
- Process Integration
- Event Driven Integration
- Data Integration

Oracle Fusion Middleware provides a complete set of product to implement integration solutions across all of the four categories. Understanding the integration categories will help you understand what product mix can be used to implement enterprise wide integration solutions. And, when considering enterprise-wide integration initiatives, remember that governance is an important part of any integration project. Governance is especially important for SOA and is most successful when constantly revisited and refined.