Integration Architecture Capability Case Model

As of April 2004

Lawrence Livermore National Laboratory
Livermore, California 94550
UCRL-TR-203567

R.W. Watts

Disclaimer

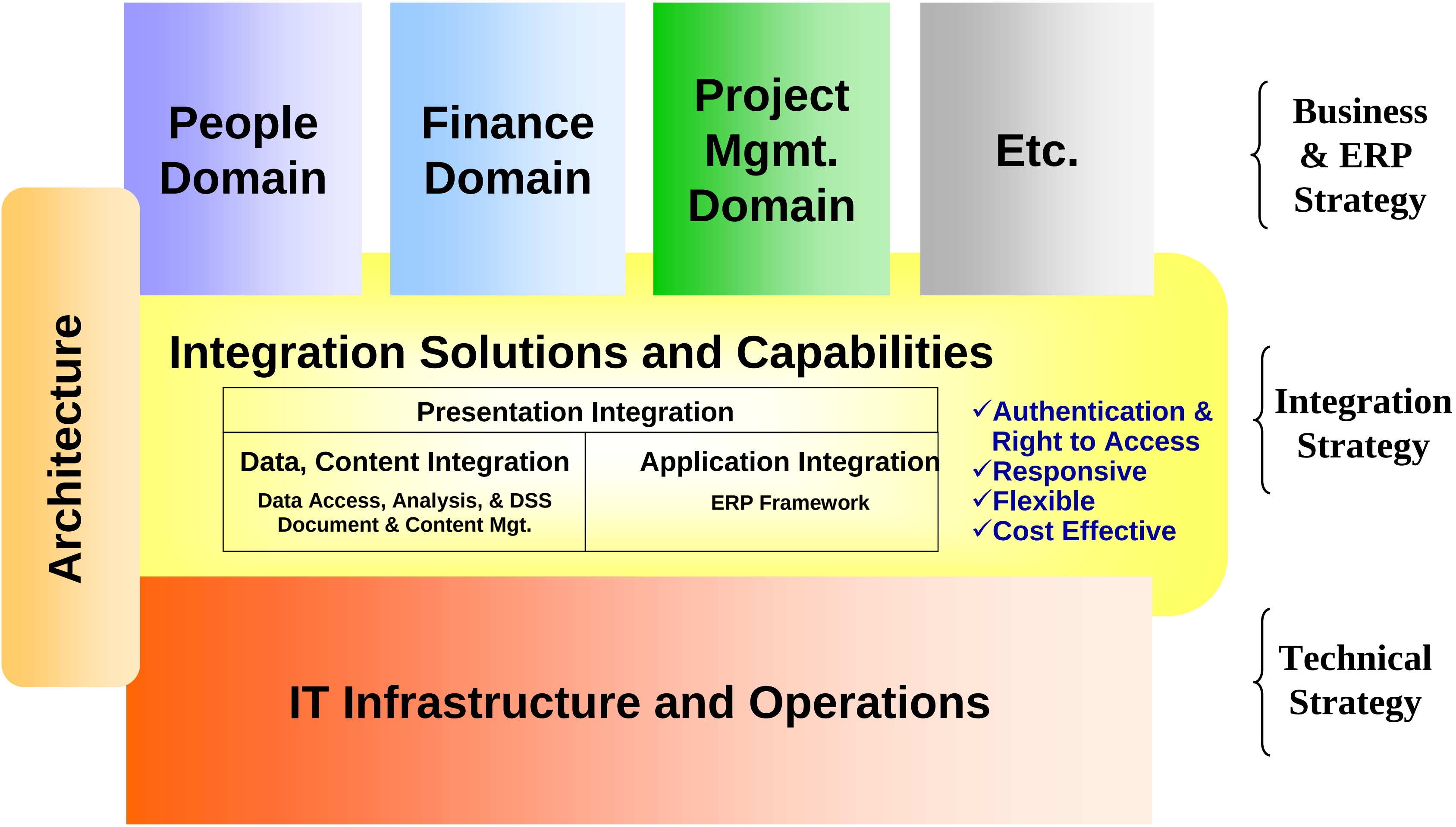
This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

Information Integration Architecture Capability Case Model

- The LLNL Administrative Information Systems (AIS) Department is developing and refining strategies to address information and system integration.
- The “Framework for Integration of LLNL Business Operational Information” white paper authored by R. Watts (UCRL-AR-202823) identifies foundational principles for an information integration framework and provides a context for the Capability Case Model.
- AIS IT managers and staff are developing the Capabilities Case Model as a tool to identify the basic elements of an integration architecture. The model has been useful in clarifying integration approaches.
- This effort is a work in progress. The Capability Case Model is a working document and will be periodically revised, amended, and defined in greater detail as the architecture evolves.
- The Capability Case format was adopted with permission from Topquadrant, Inc. for LLNL internal use.

An information integration strategy and architecture will become foundational for all business domains



People Domain

Finance Domain

Project Mgmt. Domain

Etc.

Architecture

Integration Solutions and Capabilities

Presentation Integration	
Data, Content Integration Data Access, Analysis, & DSS Document & Content Mgt.	Application Integration ERP Framework

- ✓Authentication & Right to Access
- ✓Responsive
- ✓Flexible
- ✓Cost Effective

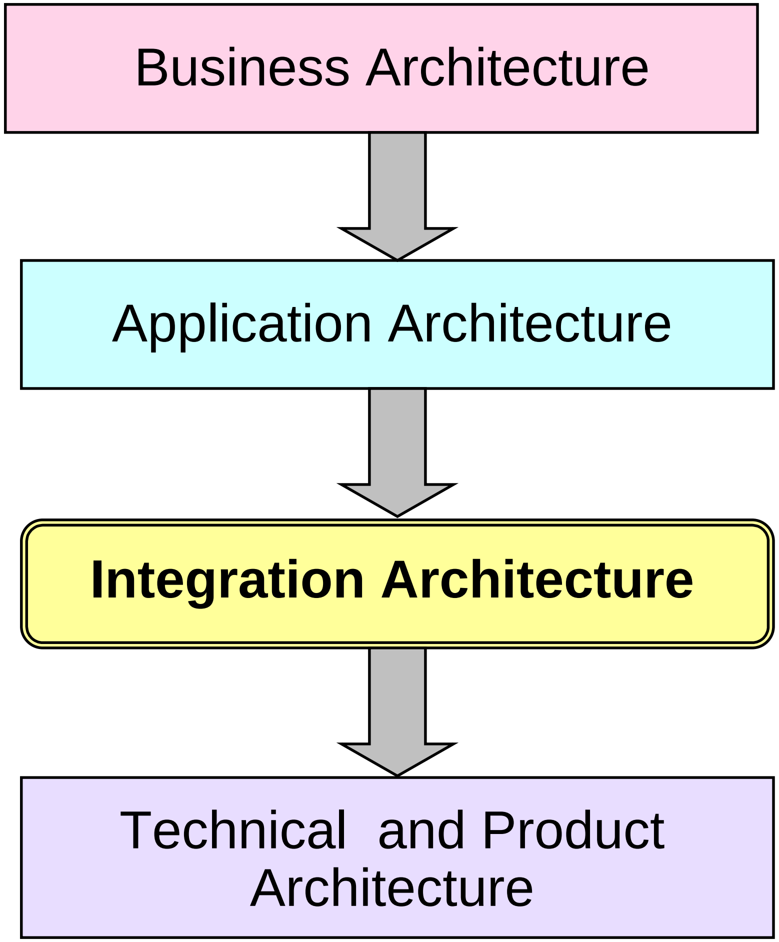
IT Infrastructure and Operations

Business & ERP Strategy

Integration Strategy

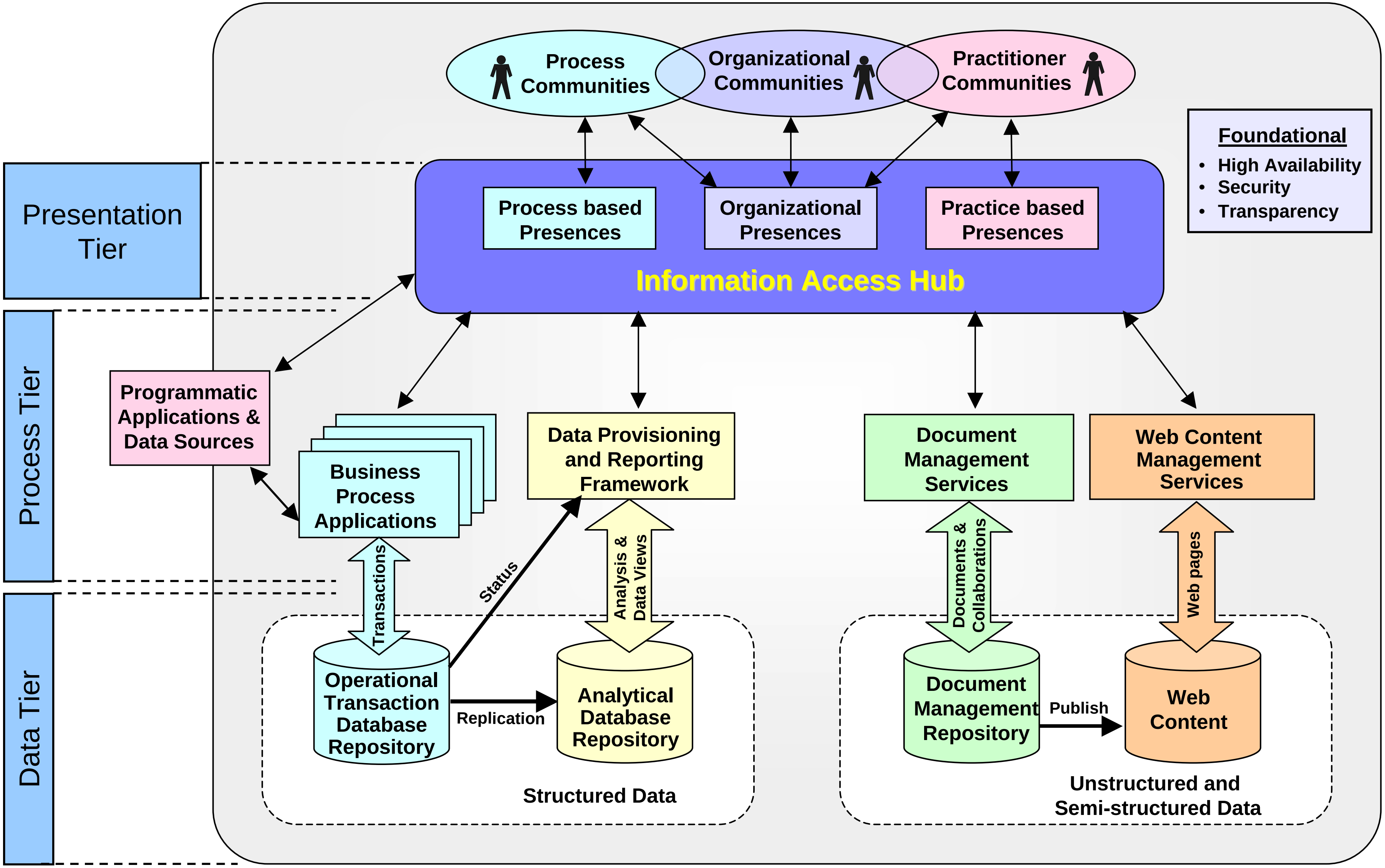
Technical Strategy

A layered IT architectural is a key component of IT strategy



- An integration architecture is one layer in an over arching IT architectural strategy
- The integration architecture provides a basis for the integration of the various data, process, and presentation elements of the application architecture
- The integration and application architectures form the requirements for the technical and product architectures
- Conceptually, integration architecture requirements can be defined as is a set of capabilities and, at a high level, be described using a Capability Case format

*Elements for information integration **


Process Communities
Organizational Communities
Practitioner Communities
Foundational
• High Availability
• Security
• Transparency
Presentation Tier
Process based Presences
Organizational Presences
Practice based Presences
Information Access Hub
Process Tier
Programmatic Applications & Data Sources
Business Process Applications
Data Provisioning and Reporting Framework
Document Management Services
Web Content Management Services
Transactions
Status
Analysis & Data Views
Documents & Collaborations
Web pages
Data Tier
Operational Transaction Database Repository
Replication
Analytical Database Repository
Document Management Repository
Publish
Web Content
Structured Data
Unstructured and Semi-structured Data

* Reference the “Framework for Integration of LLNL Business Operational Information” white paper by R. Watts (UCRL-AR-20823)

*Integration Architecture Capability Matrix **

□ → **Signifies capability has been documented in a capability case**

		Business Architecture Pattern *					Foundational Capabilities
		User to Business (Transactional)	User to Data (Analytical)	Bus. to Bus. (Sys. to Sys.)	User to User (Interaction)	Org. to User (Informational)	
Presentation Tier		• **eWorkplace**	• **eWorkplace**		• **eWorkplace**	• **eWorkplace**	• **Application Roles and Authorization** • **Application Security**
Process Tier	**Federated** (loose coupling)	• **Business Process Orchestration** • Workflow	• **Enterprise Rptg.** • **Decision Support (static & dynamic)**	• **Synchronous Request & Reply** • **Asynchronous Publish & Subscribe**	• Email • Calendaring • Document Mgt • Distributed Authoring and Versioning • Instant Messaging • On-line meeting environments	• Messaging Facility	• Browser and HTML based Webtop stds.
	Monolithic Homogenous (tight coupling)	• Proprietary	• Application Specific Reporting	• Inter-process Communication • File Transfer			
Data Tier	**Structured** (data)	• Transactional Relational Database	• **Data Repository Architecture** • **ODS, EDD, Data Mart**	• **Data Repository Architecture** • **ODS, EDD, Data Mart** • **Extract, Transform, Load**			
	Unstructured (content)	• Document Mgt	• Information Hubs • Taxonomy • Ontology/RDF • ebXML	• Files • ebXML	• Email Messages • Document, Content Repository	• Messages • Reference Repository (policies, etc.)	

*** Reference the "Framework for Integration of LLNL Business Operational Information" white paper by R. Watts (UCRL-AR-20823)**

Data Tier Capability Cases

		Business Architecture Pattern					Foundational Capabilities
		User to Business (Transactional)	User to Data (Analytical)	Bus. to Bus. (Sys. to Sys.)	User to User (Interaction)	Org. to User (Informational)	
Presentation Tier							
Process Tier	**Federated** (loose coupling)						
	Monolithic Homogenous (tight coupling)						
Data Tier	**Structured** (data)	• Transactional Relational Database	• **Data Repository Architecture** • **ODS, EDD, Data Mart**	• **Data Repository Architecture** • **ODS, EDD, Data Mart** • **Extract, Transform, Load**			
	Unstructured (content)	• Document Mgt	• Information Hubs • Taxonomy • Ontology/RDF • ebXML	• Files • ebXML	• Email Messages • Document, Content Repository	• Messages • Reference Repository (policies, etc.)	

Integration Architecture Capability Case

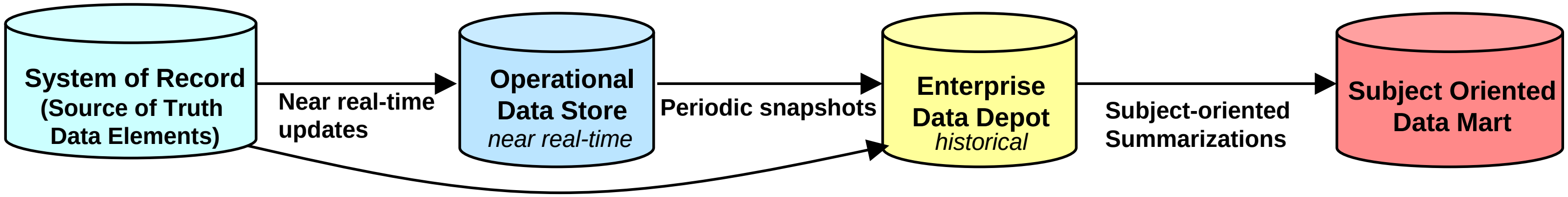
User to Data and System to System:
Data Repository Architecture

Intent

Data architecture consists of data modeling, design and implementation strategies aimed at minimizing data redundancy, inaccuracies, and inconsistencies and establishing common data semantics. Data architecture strategies establish the data building “blueprint” which will be used by all applications. The data architecture is a conceptual design model. It outlines the roles and behaviors of the various data stores necessary to support a variety of applications. The data architecture will specify information integration and summarization relationships. Although the goal is to minimize redundancy, summarizations are specified in order to improve performance and application reliability through controlled redundancy. The data architecture consists of four core data repositories: the system-of-record (SOR) database, the operational data store (ODS), the enterprise data depot (EDD) repository and subject-oriented data marts. All data of enterprise integration value will be loaded and integrated into the EDD.

Solution Stories

All applications utilize the appropriate core data repositories: SOR database, ODS, EDD or subject-oriented data marts.
All applications are designed to conform with the data architecture. An exception process will govern appeals to create non-integrated repositories for cases where the data has localized value.



System of Record **(Source of Truth Data Elements)**

Near real-time updates

Operational Data Store *near real-time*

Periodic snapshots

Enterprise Data Depot *historical*

Subject-oriented Summarizations

Subject Oriented Data Mart

System of record (SOR) applications maintain data in their associated SOR database. All update (create, update and delete) transactions that change the state of data are made real-time through the SOR application logic and recorded in the SOR database. Updates to the SOR database trigger near real-time update transactions to be applied to the Operational Data Store (ODS). The ODS is used by applications needing read access to data maintained in SOR databases. State changes to data stored in the ODS may trigger events that the event-based publish and subscribe service will distribute to subscribing applications.   An extract, transform and load layer will perform periodic snapshots of data stored in the ODS and load them into the EDD. The periodicity of the snapshots will be depend on the update frequency of the data in the SOR database.   As an example, ledger data would be loaded on a weekly basis, while property, people and facility data would be daily loads. Data loaded into the ODS will be normalized, rationalized and given the enterprise consistent fiscal and calendar year time dimensions. Subject-oriented data marts will be created by creating summarized data structures, star schemas, etc., tailored to support a specific community’s decision support needs.

Benefits

A consistent data architecture will enable efficient data provisioning with data repositories and data structures tailored for the specific needs of all applications from SOR to DSS. The data architecture is also a required element to enabling integrated, consistent, and timely data aggregated from multiple SORs.

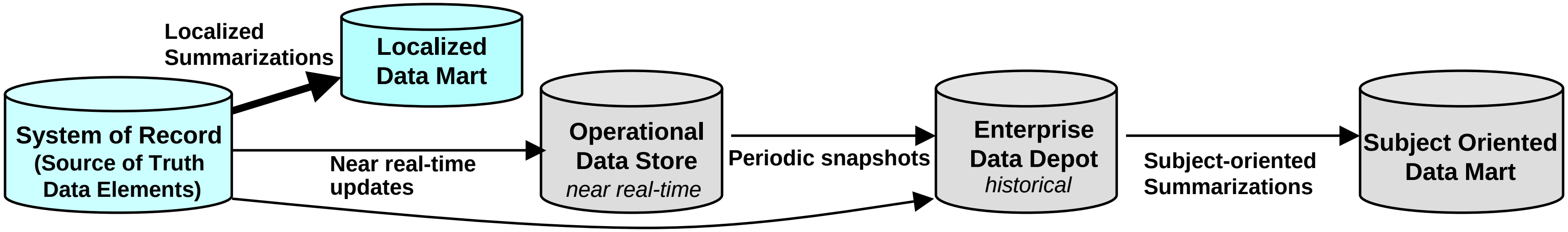
Products

Oracle RDBMS, Librados, PL/SQL, SQL, Appworx,

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case	***System to System:*** **Data Repository Architecture – Exception Case**
Intent	This is an exception case to the enterprise data repository architecture for cases where the value of the data has been determined to be localized in scope. Localization, means the data can be isolated (not integrated) because it provides no potential enterprise data integration value. A governing board (to be determined) will rule on the enterprise integration value of the data. If the data has been judged to have localized value, a Localized Data Mart (LDM) will be created to satisfy the decision support system (DSS) needs of that work group.
Solution Stories	Work group members access a static or dynamic DSS application which utilizes a LDM which has been loaded via localized summarization ETL processes which load data from the system of record (SOR) database.


Localized Summarizations
Localized Data Mart
System of Record (Source of Truth Data Elements)
Near real-time updates
Operational Data Store near real-time
Periodic snapshots
Enterprise Data Depot historical
Subject-oriented Summarizations
Subject Oriented Data Mart

SOR applications maintain data in their associated SOR database. All update (create, update and delete) transactions that change the state of data are made real-time through the SOR application logic and recorded in the SOR database. Periodic updates will build summarized data structures or star schema tables to be used by a local DSS application.

Benefits	Extends the Enterprise Data Repository Architecture by providing support for data with localized value. This is an important benefit as it enables the enterprise data model to remain simple and concise with only data with real enterprise integration value or potential value.
Products	Oracle RDBMS, Librados, PL/SQL, SQL, Appworx,

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

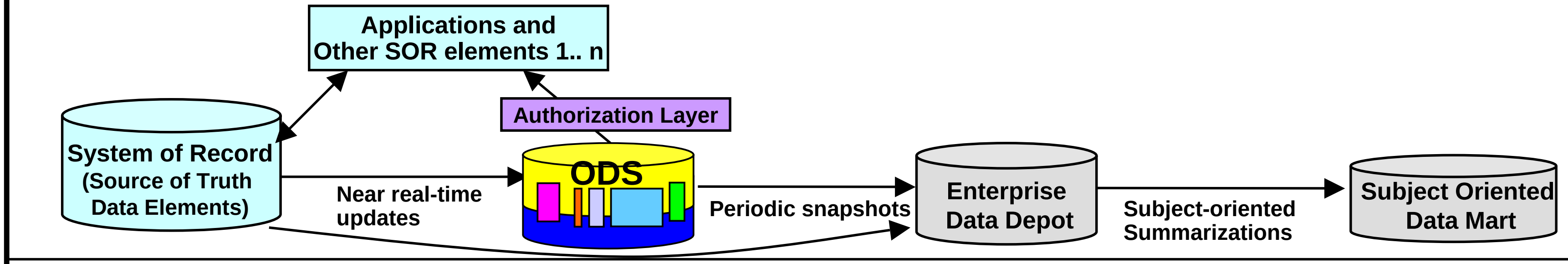
User to Data and System to System:
Operational Data Store (ODS)

Intent

The Operational Data Store (ODS) is a core component of the data repository architecture model, which consists of four core data repositories: the system-of-record (SOR) database, the operational data store (ODS), the enterprise data depot (EDD) and subject-oriented data marts (SODM). An ODS is subject-oriented, such as the people, financial, or facility domains. The intent of the ODS is to create a data repository that collects subject-oriented data from the various SORs deemed as the source of truth for that data element. Applications are simplified as they need only interface to the ODS instead of the multiple SOR DB's. The ODS is tuned for rapid and scalable access. The ODS is the basis for the event-based data subscription model by which subscribing applications can subscribe and receive near real-time updates on data element state changes (ex. employment status). The intent of the ODS is to reflect current state information about the subject of the ODS. The ODS supports both business sensitive and non-sensitive data. The authorization layer provides authorized access to authenticated individual requests and is a key element of the overall ODS architecture.

Solution Stories

Source application updates in the SOR DB's trigger near real-time updates to the ODS. Applications access (read-only) the ODS as part of their business processing through the authorization layer (handles sensitive data authorization).



The ODS is a key integration hub as it tracks changes to core data entities and makes these changes or the entire data entity available to querying applications. The ODS is used by applications needing read access to data maintained in SOR databases. State changes to data stored in the ODS may also trigger events that the event-based publish and subscribe service will distribute to subscribing applications. The ODS does not contain historical or other time-based information. Tracking the time dimension of data is the role of the EDD. The ODS analysis and design process utilizes the EDAP process to determine the appropriate data to be added to the ODS, the SOR DB's for each data element and the update frequency of the each data element.

Benefits

The creation of subject-oriented ODSs, provides application developers with a consistent repository and interface for which they can access current state information and shields them from the complexities of its creation.

Products

Oracle RDBMS, Librados, PL/SQL, SQL, Appworx, Oracle Single sign-on, Authorization engine

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

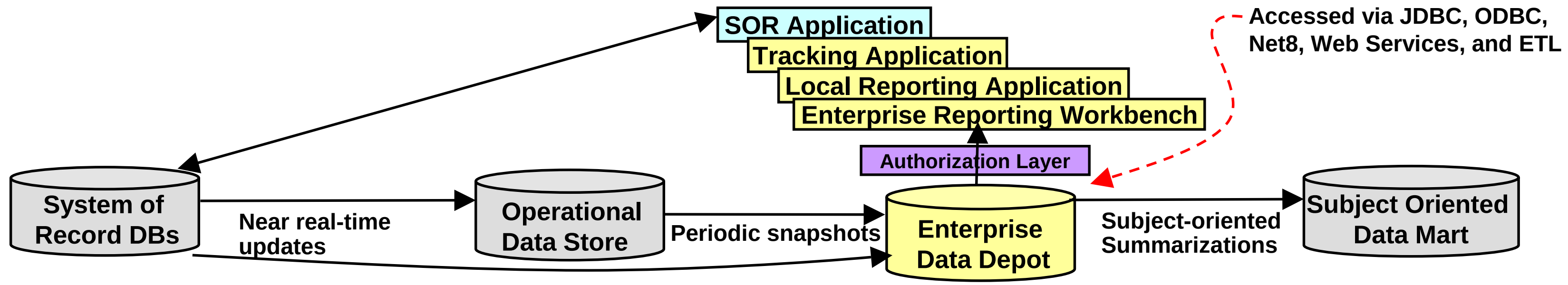
System to System:
Enterprise Data Depot

Intent

The data architecture consists of four core data repositories: the system-of-record (SOR) database, the operational data store (ODS), the enterprise data depot (EDD) repository and subject-oriented data marts (SODM). The EDD is a set of normalized and time synchronized (fiscal and calendar) subject tables. The EDD's subject tables provide a current and historical record of the data entities through a snapshot loading process. While the ODS and SODMs are subject-oriented, the EDD strives for breadth of interrelated business data. The periodicity of the snapshots is tied to the update frequency of the data sources. Although currently loaded directly from SOR databases, the EDD will be loaded from subject-oriented ODS's as they become available. The EDD is designed for ease of reporting and data integration. The EDD is the data source for the Data Warehouse Reporting application. The EDD supports JDBC, ODBC and Net8 drivers to allow applications direct read-only access to the EDD subject tables. A web-site provides EDD meta data information designed to help application developers utilize the EDD tables and API's.

Solution Stories

The Data Warehouse provides enterprise reporting services utilizing the EDD. SOR, tracking or local reporting applications access the EDD. The EDD extract, transform, and load processes trigger subject-oriented Data Mart summarizations.


SOR Application
Tracking Application
Local Reporting Application
Enterprise Reporting Workbench
Accessed via JDBC, ODBC, Net8, Web Services, and ETL
Authorization Layer
System of Record DBs
Near real-time updates
Operational Data Store
Periodic snapshots
Enterprise Data Depot
Subject-oriented Summarizations
Subject Oriented Data Mart

EDD periodic snapshots from the ODS or SOR databases will populate the EDD tables adding the time dimension and institutional attribute relationships. Institutional attributes are the key dimensions necessary for cross entity joins. Key institutional attributes are person, account, facility and payroll account by year. EDD snapshots are scheduled based upon the update frequency of the data entity and are triggered by flags set in the data availability table. The EDD serves as a data source for a variety of applications which access the EDD via industry standard database interfaces. The EDD will contain business sensitive and non-sensitive information. EDD access to sensitive information will be through a web service interface while non-sensitive data can be accessed using JDBC, ODBC and Net8 database drivers. Applications can either directly access the EDD in a real-time manner or may chose to extract the data from the EDD for local processing and storage. Upon successful completion of EDD load processes, summarizations may be triggered to population SODMs.

Benefits

The EDD provides a one-stop, quality assured repository for a broad scope of applications needing to access enterprise information. The EDD provides information integration and history.

Products

Oracle RDBMS, PL/SQL, SQL, Appworx, JDBC, ODBC, Net8 drivers, web services

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

User to Data:

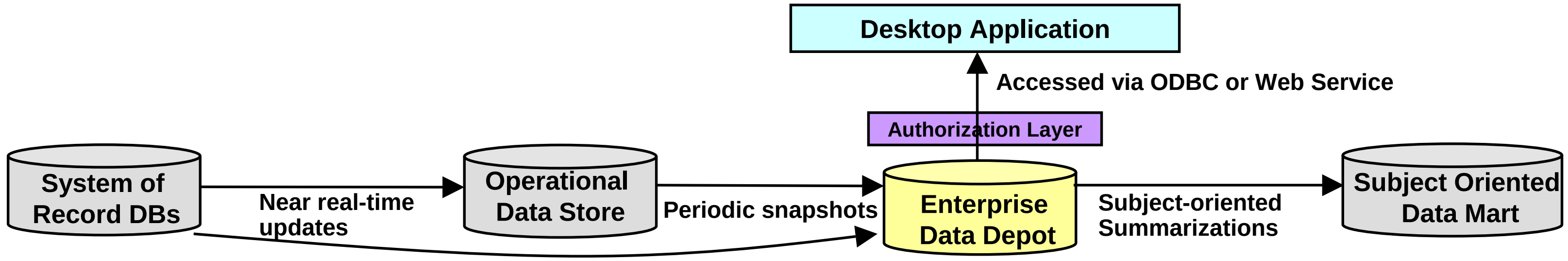
Enterprise Data Depot

Intent

The data architecture consists of four core data repositories: the system-of-record (SOR) database, the operational data store (ODS), the enterprise data depot (EDD) repository and subject-oriented data marts (SODM). The EDD Depot is a set of normalized and time synchronized (fiscal and calendar) subject tables. The EDD's subject tables provide a current and historical record of the data entities through a snapshot loading process. While the ODS and SODMs are subject-oriented, the EDD strives for breadth of interrelated business data. The periodicity of the snapshots is tied to the update frequency of the data sources. Although, currently loaded directly from SOR databases, the EDD will be loaded from subject-oriented ODS's as they become available. The EDD is designed for ease of reporting and application data integration. The EDD is the data source for the Data Warehouse Reporting application. The EDD supports JDBC, ODBC and Net8 drivers to allow applications direct read-only access to the EDD subject tables. A web-site provides EDD meta data information designed to users understand and utilize the EDD tables via ODBC connections.

Solution Stories

Desktop user application written in Excel or MS Access accesses the EDD to obtain a specific subset (ex. All AIS accounts) information to be downloaded locally and manipulated in the desktop application.



A desktop user will work with the EDD support team to establish a working ODBC connection on their desktop machine. The EDD consultant may advise the desktop user on how to get the needed information from the EDD tables. Once the appropriate queries are constructed and configured into the local desktop application (Excel or MS Access), the user then triggers the queries on an as needed basis. EDD requests for business sensitive data must be made through a web service call.

Benefits

Users access the same quality assured data in the EDD that is used by the Enterprise Reporting Workbench and other reporting applications. One stop location for Enterprise data.

Products

Oracle RDBMS, PL/SQL, SQL, Appworx, ODBC, Web Services

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integrated Architecture Capability Case

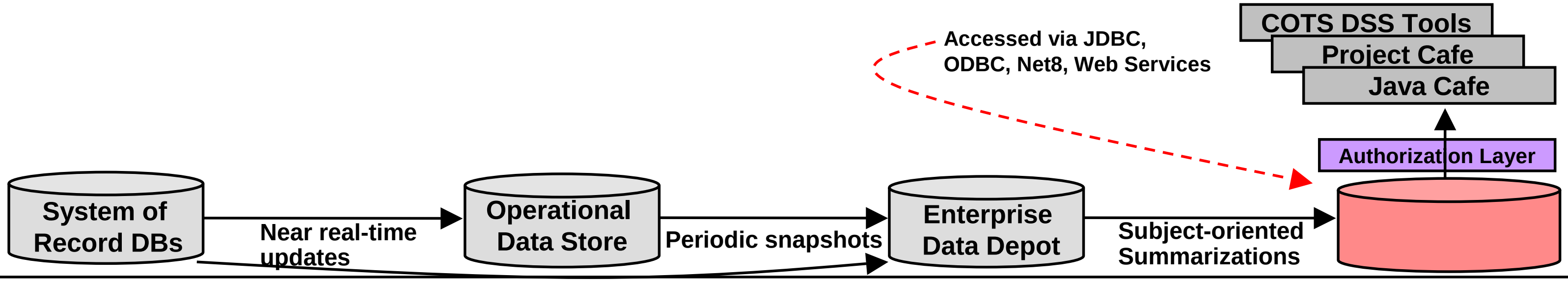
User to Data: Subject-Oriented Data Mart

Intent

The data architecture consists of four core data repositories: the system-of-record (SOR) database, the operational data store (ODS), the enterprise data depot (EDD) repository and subject-oriented data marts (SODMs). A data mart is a strategy employed to provide enhanced querying capabilities on a particular data domain (subject) or a combination of domains. SODM are created to support decision support systems (DSS) which require fast access and the establishment of dimension relationships to support drill down reporting. Data relationships (dimensions) are explored during the analysis phase to determine which relationships are valuable to the particular DSS capability being established. Typical data structures include summarizations from a number of subject areas (ex. costs, effort, planning) across time dimensions (fiscal periods) and star schema designs utilizing fact and dimension tables. The intent of the SODM is to support DSS applications. Although designed for a particular DSS application, the SODM, like the EDD, supports JDBC, ODBC and Net8 drivers to allow applications direct read-only access. An SODM may contain sensitive and non-sensitive data and thus requires an authorization layer to control access

Solution Stories

A user accesses a DSS application to query information. The DSS application accesses a particular SODM to query the data.



The EDD and the SODM's may reside in the same physical database as both are designed for efficient query access with little update activity. Following the successful quality assurance checks of the EDD subject data loads, SODM summarizations are executed creating summarized structures, fact and dimension tables. The existence of an authorization layer is required for SODM containing sensitive information and will likely require the use of web services protocols rather than the traditional database drivers. Although designed initially to support a particular DSS application, SODM may be used by other applications.

Benefits

The SODM provides the specialized, optimized data structure that is the critical component of the DSS application.

Products

Oracle RDBMS, PL/SQL, SQL, Materialized views, Appworx, JDBC, ODBC, Net8 drivers, web services, authorization services

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

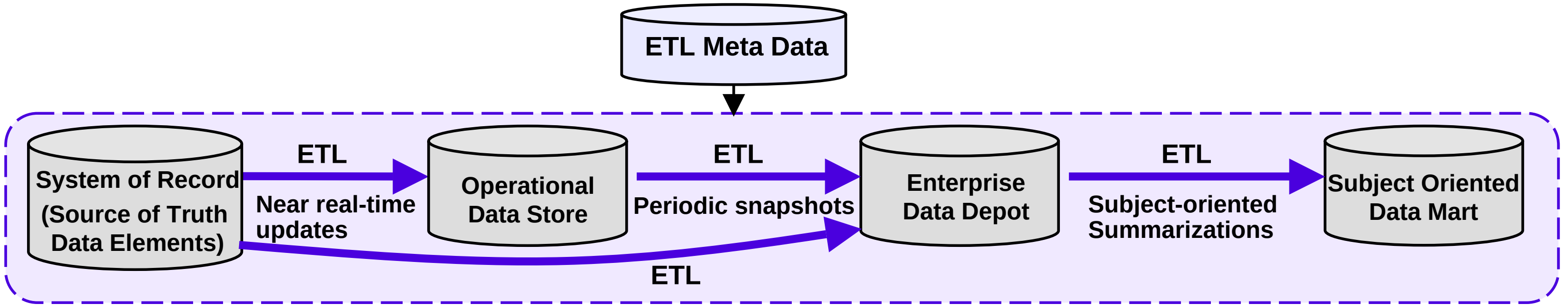
System to System:
Extract, Transform and Load (ETL)

Intent

The data repository architecture consists of four core data repositories: the system-of-record (SOR) database, the operational data store (ODS), the enterprise data depot (EDD) repository and subject-oriented data marts. Key to establishing these core data repositories is the ETL software layer which enables their loading. The ETL can be invoked either through an event-based messaging or a time schedule. A meta data repository will be used to track transformations throughout the data architecture. The ETL layer is responsible for processes associated with loading data between repositories. Having developers with ETL skills is critical due to the significant ETL technology learning curve.

Solution Stories

The SOR application updates the SOR database via a an update transaction which causes a near real-time update message to be sent to the ODS. On a daily interval, the EDD loads information into the EDD adding the time dimension via periodic snapshots. Following the EDD periodic snapshots which load data such of people, effort and ledger data into the EDD, subject-oriented summarizations are executed loading star schema tables and other summarized data mart structures for DSS reporting



The creation of an ETL link between the SOR DBs and the ODS requires a significant amount of analysis to determine the appropriate data to be contained in the ODS, data models, data structures, data formats, data definitions, data SOR applications, data transformations, timings, etc. This analysis is accomplished as part of an enterprise data alignment process. The near real-time synchronization of the ODS with various SOR applications (People for instance has multiple SOR's) requires a message-based ETL tool. Upon a state change, the SOR will trigger a message to update the ODS. There are a variety of approaches including sending the payload in the initial message or sending a message that triggers the ODS to pull the change which need to be explored to determine the optimal approach. Once the ODS has been updated with the latest transaction, the ODS may also established an event in the event-based published and subscribe layer, which then notifies all subscribing applications of the event. For cases where near real time is not required, the ETL capability will have the ability to do batch updates to the ODS. The EDD updates typically use time and flags as triggers. An example is loading the Ledger data on Monday nights after 5pm when the ledger available flag has been posted. Subsequent subject-oriented data mart loads are typically processes that follow EDD loads.

Benefits

An ETL layer is required to enable populating the data repositories with properly formatted, integrated, and timely data to support decision making and transactions. Standardization on ETL tools also reduces development costs

Products

PL/SQL, SQL, Appworx, Librados

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Process Tier Capability Cases

		Business Architecture Pattern					Foundational Capabilities
		User to Business (Transactional)	User to Data (Analytical)	Bus. to Bus. (Sys. to Sys.)	User to User (Interaction)	Org. to User (Informational)	
Presentation Tier							
Process Tier	**Federated** (loose coupling)	• **Business Process Orchestration** • **Workflow**	• **Enterprise Rptg.** • **Decision Support (static & dynamic)**	• **Synchronous Request & Reply** • **Asynchronous Publish & Subscribe**	• Email • Calendaring • Document Mgt • Distributed Authoring and Versioning • Instant Messaging • On-line meeting environments	• Messaging Facility	
	Monolithic Homogenous (tight coupling)	• Proprietary	• Application Specific Reporting	• Inter-process Communication • File Transfer			
Data Tier	**Structured** (data)						
	Unstructured (content)						

Integration Architecture Capability Case

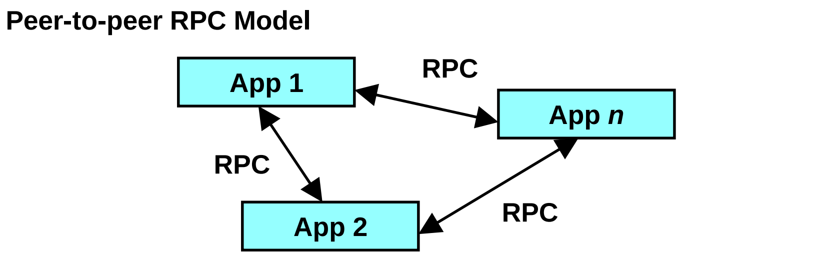
User to Business & System to System:
Business Process Orchestration (Federated Workflow)

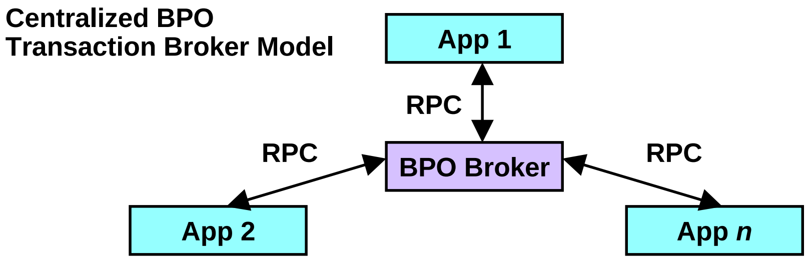
Intent

Business Process Orchestration (BPO) consists of the capability for many applications to combine or orchestrate their application-specific business processes (i.e. stateful, long-running user-to-business interactions) into a single, integrated business process. It is a core capability needed to support our strategic services oriented architecture. The intent is to integrate application workflow processes with minimal development effort, thus promoting loosely coupled application architectures and increasing application longevity. The intent is to also enable inter-application business process design, monitoring and reporting by implementing a single, integrated BPO solution. It would also be ideal for the BPO solution to manage business processes across applications running different technology stacks.

Solution Stories

The orchestration of *m* workflows across *n* discrete applications


Peer-to-peer RPC Model


Centralized BPO Transaction Broker Model

First approach: Standalone applications typically store business process state in their local database schema in a variety of formats. Utilizing a peer-to-peer RPC-based workflow messaging system will allow the systems to leave existing business and persistence logic unchanged while tying their business processes together via a business process execution protocol in conjunction with an authorization and transaction management choreography protocols. The transaction management and security schemes are non-trivial, but are required components of this architecture.

Second approach: Applications would manage only application data, while a centralized BPO broker manages the business process state for the inter-application workflow. When a workflow-related event is fired at the application level, it messages the BPO broker who then asynchronously notifies subscribing clients of the workflow transition. The central broker maintains a central system of record for the entire workflow transaction, thus enabling a central location for workflow definition, querying, analysis and monitoring.

Benefits

Allows integration of business processes and heterogeneous application architectures/technology stacks. Provides a common toolset for designing, implementing, monitoring and inspecting of workflows. Adopting a common BPO solution quickly builds developer proficiency in that solution, thus minimizing training time and costs.

Products

InterSystems Ensemble, Dralasoft Workflow Engine, Librados+Collaxa

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

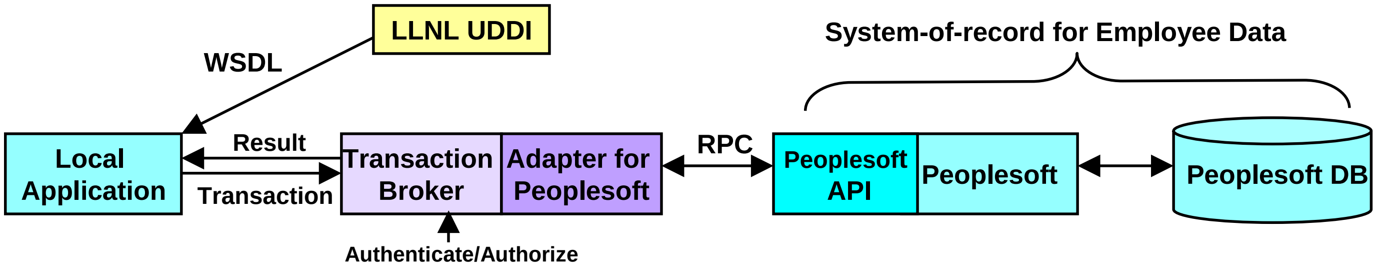
Systems to System:
Synchronous Request/Reply

Intent

Synchronous request/reply application-to-application integration consists of the capability for one application to digitally interact with another application to accomplish work in near real-time. It is a core capability needed to support our strategic services oriented architecture (SOA). The intent is to enable automation of end-to-end business processes by creating workflow applications that utilize standard back office services, rather than duplicating the business logic and data. The intent is to also improve data quality and consistency by designating specific systems as the system-of-record for specific institutional data and to develop digital services that enable other applications to update that data thru the SOR.

Solution Stories

Local applications ability to update employee transactions to the backend PeopleSoft system.



A local application would like to update an employee’s address in the institutional Peoplesoft system. The application checks with the LLNL UDDI repository to obtain the WSDL description of the employee address update service, including where the service is located and how to invoke it (parameters, method names, etc.). It then makes a synchronous call (XML/SOAP/HTTPS) to the service which is managed by a transaction broker, passing it the XML payload for the transaction, and then blocks and waits for a response. The transaction broker invokes its Peoplesoft adapter to establish an RPC based synchronous connection to the Peoplesoft API for updating addresses and sends it the relevant transaction data. The process then blocks while waiting for a reply. The Peoplesoft application edit checks the transaction data, updates the database, and notifies the adapter that the transaction was successful. The adapter communicates the successful completion of the transaction back to the local app.

Desired attributes of this capability include the ability to dynamically discover what services are available, where they are located, and how to interface with them (eg parameters, methods, authentication, authorization, etc.); transaction management services that insure integrity and performance; logging; metrics; transformation services; loose coupling; and the availability of a core set of institutional business services.

Benefits

Programming and architectural flexibility and efficiency. Use of available services without required prior knowledge of what they are and how they work. Dynamic composition of business processes to reflect specific requirements.

Products

Librados, Oracle Integration Suite, J2EE/JCA/JMS Open Source Integration Suite.

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

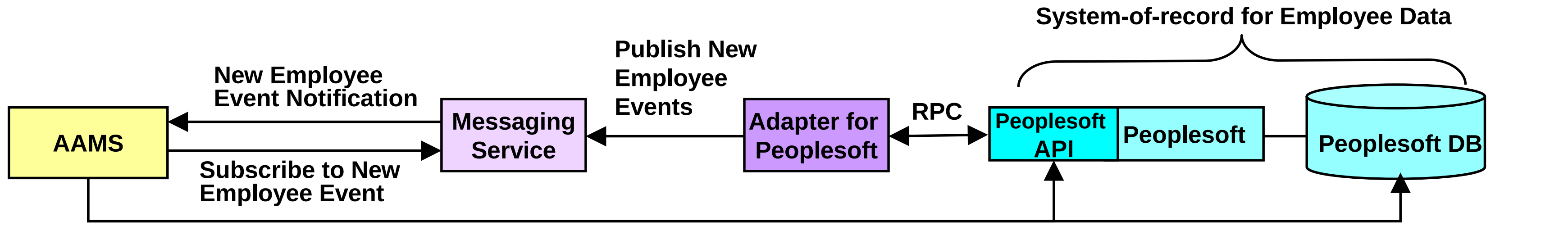
System to System:
Asynchronous Publish/Subscribe

Intent

Asynchronous publish/subscribe application-to-application integration consists of the capability for an application to digitally subscribe to a service to be notified of specific events in near real-time. It is a core capability needed to support our strategic services oriented architecture. The primary intent of this capability is to enable applications to be notified in near real time of events that they need to respond to rapidly.

Solution Stories

The CSTI AAMS system creates OUNs and PACs for users so they can access institutional applications. Currently there is at least a 24 hour lag time from when an employee gets hired to when AAMS creates their OUN and PAC. AAMS would like to be notified in near real time when a new employee is hired so they can create the OUN and PAC for the employee and enable the employee to access institutional applications like email and calendaring as soon as possible.



System-of-record for Employee Data
New Employee Event Notification
Publish New Employee Events
AAMS
Messaging Service
Adapter for Peoplesoft
RPC
Peoplesoft API
Peoplesoft
Peoplesoft DB
Subscribe to New Employee Event

AAMS would like to be notified when a new employee starts at the Lab. AAMS makes an asynchronous call to the Java Messaging Service (JMS) and subscribes to the new employee event. The Peoplesoft application publishes all new employee events asynchronously to the JMS queue with the appropriate new employee event topic identifier. When the JMS service receives the new employee event, it notifies the AAMS application of the event and passes it the new employee message. The AAMS application sends back a reply that the message was received. The JMS service then deletes the message from the queue.

Prior to the deployment of this integration path, agreement has to be reached between the IT providers on the content, structure, and meaning of the new employee event message (probably an XML document). Also, the integration infrastructure has to have the capability for authenticating and authorizing the AAMS application to insure it has the authority to see the new employee message data. Will also require audit trails and logs.

Desired attributes of this capability include the ability to discover what events are available, where they are located, and how to interface with them (eg parameters, methods, authentication, authorization, etc.); transaction management services that insure integrity and performance; logging; metrics; transformation services; loose coupling; and the availability of a core set of institutional business event notifications.

Benefits

Programming and architectural flexibility and efficiency. Use of available services without required prior knowledge of what they are and how they work. Dynamic composition of business processes to reflect specific requirements.

Products

Librados, Oracle Integration Suite, J2EE/JCA/JMS Open Source Integration Suite.

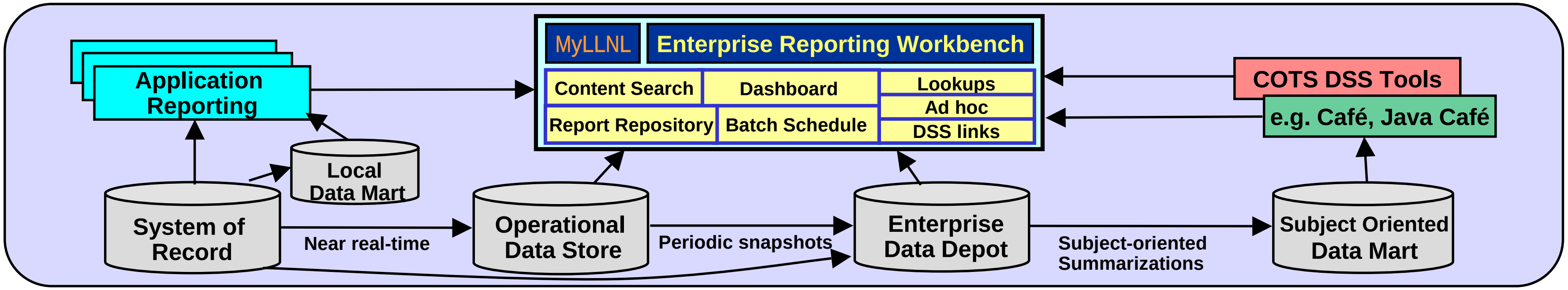
Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

User to Data:

Integrated Reporting Framework Overview



Enterprise Reporting Workbench

Application Reporting

Usability:

- Target: Process participants
- Requires domain knowledge
- Learning curve varies

Source: System of record w/real-time access or localized data mart

Reporting types:

- Optimized for real-time process-specific reporting
- Pre-defined reports
- Workflow reporting
- Exception reports
- Real-time dashboard
- Batch or on-demand
- Web-based

Analytical Reporting

Usability:

- Target: General analyst roles and the enterprise user
- Requires domain knowledge
- “Zero” learning curve

Source: Near real-time ODS, periodic snapshot cross-domain data structures, integrated dimension data

Reporting types:

- Optimized for periodic general reporting, simple ad hoc reporting & lookups
- Pre-defined reports
- Snap shot dashboard
- Batch or on-demand
- Web-based, graphical, tight Excel integration

Static Decision Support

Usability:

- Target: Specific analyst roles
- Requires domain knowledge
- “Zero” learning curve

Source: Summarized data mart optimized for specific use

Reporting types:

- Optimized for specific analyst communities (e.g. RAs, Project Mgt., etc.)
- Pre-defined reports
- Pre-defined drill paths
- Web-based, graphical, tight Excel integration

Dynamic Decision Support

Usability:

- Target: Highly specialized analysts
- Requires expert-level domain knowledge
- Extensive learning curve

Source: Star schema or dimensional Data Mart optimized for specific use

Reporting types:

- Optimized for complex ad hoc analysis
- Dynamic drill paths
- Slice and dice
- Sophisticated GUI, graphical, tight Excel integration

As of 2/27/2004

Integration Architecture Capability Case

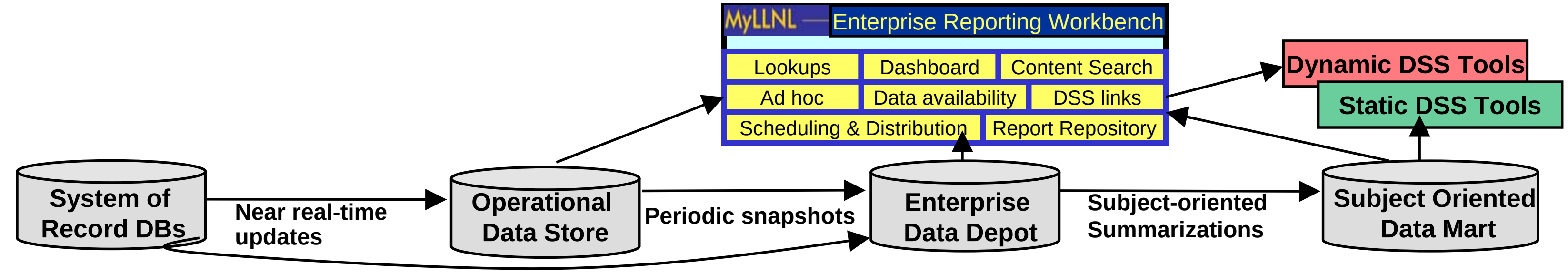
User to Data: Enterprise Reporting Workbench

Intent

The Enterprise Reporting Workbench (ERW) provides the user with a single interface to access enterprise reporting capabilities. The ERW utilizes the MyLLNL interface to deliver web-based reporting capabilities, dashboard controls, content search, report scheduling and distribution, report repository and links to community-based DSS applications. The ERW provides a common framework for report publishing which allows developers of DSS reports to plug these reports into the scheduling and distribution and report repository functionality. ERW is designed to be report writer agnostic. Lookups will utilize the operational data store (ODS) databases where available. The user will be able to select among a number of subject-oriented dash board reports. Users can select links which will direct them under auspices of single sign-on to the select DSS application. All reports that have been run, are available for viewing/downloading from the Report Repository. The users will be able to view the data availability status of all data domains in the ODS, Electronic Data Depot (EDD) and subject-oriented data marts (SODM).

Solution Stories

User has an enterprise reporting need and accesses the Enterprise Reporting Workbench as tab on the MyLLNL interface. Upon review of their dashboard, the user selects the appropriate reporting option to understand the dashboard alarm. User sets up a series of scheduled reports to track information following the update frequency (daily, weekly) of the data.



Users access the Enterprise Reporting Workbench via the MyLLNL interface and obtains access to a wide variety of enterprise reporting services. Based upon the service selected the user may be transparently redirected to a DSS interface using single sign-on. Once in the DSS application, the user conducts drill down analysis within a specific subject area. If the user selects a lookup, the ERW will access the appropriate subject ODS or EDD table if an ODS does not exist for this subject. The current Data Warehouse reports, will be available for execution and scheduling. These reports run off the current EDD tables. Any report that has been executed by a user is available to that user in their report repository sandbox (contains reports they can view). Although not depicted for diagram simplicity, sensitive data will be delivered utilizing the authorization layer.

Benefits

There are two primary benefits of the ERW. First, from the user perspective, there is a one-stop shop for enterprise reporting. From an IT perspective, economies of scale can be achieved by leveraging common reporting strategies and frameworks.

Products

Oracle 10G, Oracle Reports, Jasper Reports, Java (J2EE), COTS and In-house DSS

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case

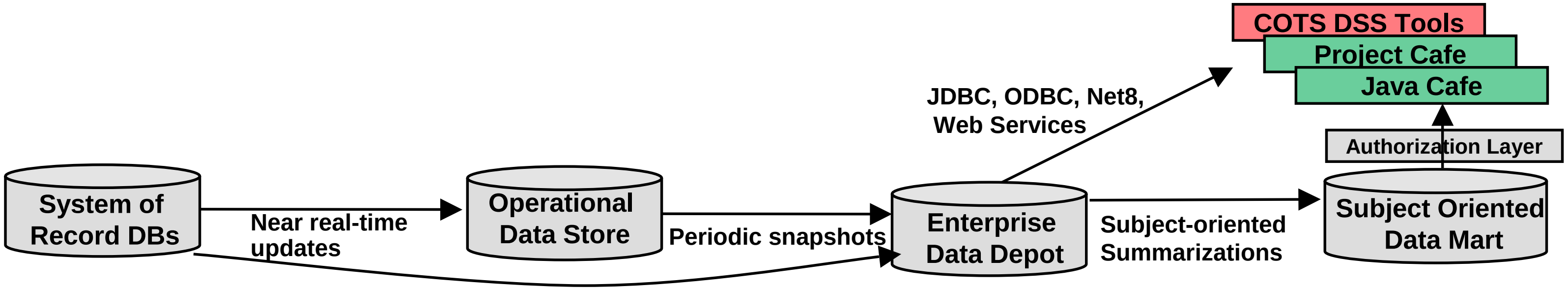
User to Data:
Decision Support (DSS)

Intent

Decision support capabilities are developed to support a particular business community. A business community such as resource managers or facility managers, utilize information that is germane to the business processes that they utilize to conduct their business activities. DSS are applications that are tailored to support the analysis activities of a specific community. Decision support systems are frequently thought of in terms of a “slice and dice” interface sitting on top of a subject-oriented data mart (SODM). These DSS applications support drill down analysis with either a dynamic drill down interface built upon a star schema-based SODM or a pre-defined drill down based interface built on a specialized summarized SODM. The choice of dynamic versus pre-defined DSS should be made based upon community need. Users of dynamic DSS applications require extensive domain and DSS tool knowledge. Dynamic DSS applications are suitable when the analysis paths are volatile. Users of pre-defined DSS applications require cursory domain knowledge and basic web navigation skills. Pre-defined DSS applications are suitable when the analysis paths can be defined.

Solution Stories

A user needs to conduct a data analysis of a complex, unstructured problem associated with particular domain. A user is conducting routine analysis associated with one or multiple domains.



The user will launch their communities’ DSS tool from their communities’ portal workbench tab. Utilizing the DSS application, the users will conduct queries, review data, prepare reports and graphs, and may download data to Excel for local analysis. Various output file formats will be supported. DSS application creation will be conducted as a full software development lifecycle project. Analysis of the communities’ requirements will help determine what type of DSS tool is needed to support these needs. Creation of the SODM, writing the subject-oriented summarizations, creation of DSS tool specific data domain mappings (universes, etc.) and writing of specific reports are some of the key activities necessary to implement a business community DSS. DSS applications dealing with sensitive data must interact with the authorization layer and may restrict the product choices available.

Benefits

DSS applications provide users with tailored analysis tools which can dramatically improve the productivity of a job function.

Products

DSS COTS tools (Cognos, Business Objects, Oracle Discover), in-house DSS tools (Java Café)

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case	***User to Business, User to User, User to Data:*** **Document Management**
Intent	
Solution Stories	
Benefits	
Products	

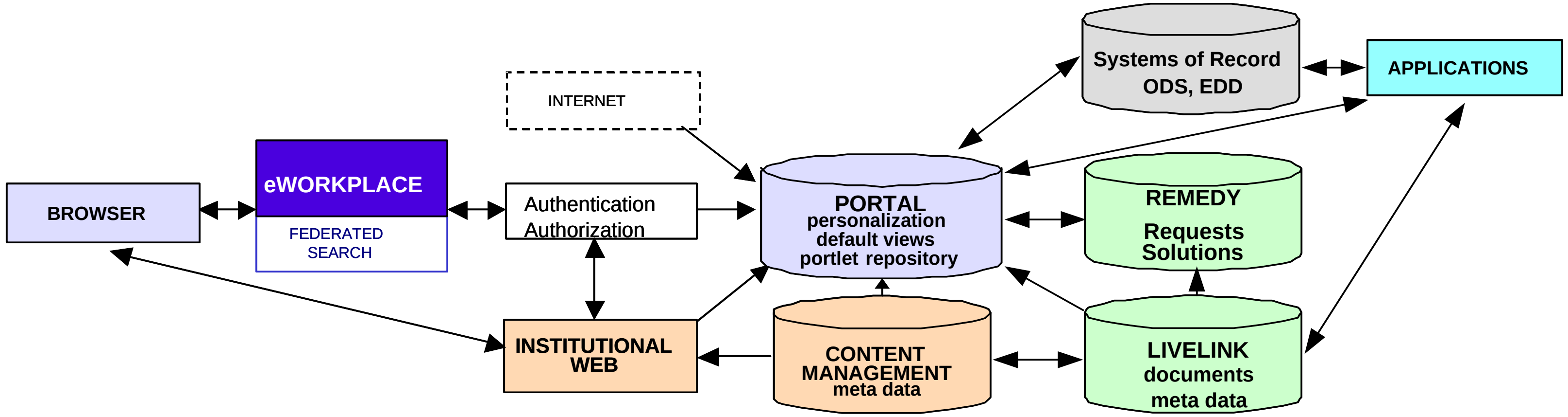
Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Presentation Tier Capability Cases

		Business Architecture Pattern					Foundational Capabilities
		User to Business (Transactional)	User to Data (Analytical)	Bus. to Bus. (Sys. to Sys.)	User to User (Interaction)	Org. to User (Informational)	
Presentation Tier		• **eWorkplace**	• **eWorkplace**		• **eWorkplace**	• **eWorkplace**	
Process Tier	**Federated** (loose coupling)						
	Monolithic Homogenous (tight coupling)						
Data Tier	**Structured** (data)						
	Unstructured (content)						

Integration Architecture Capability Case	*User to Business, User to Data, User to User and User to Organization:* **eWorkplace**
Intent	The eWorkplace delivers integrated static and dynamic content to a personal customizable web interface. Based on the end user's profile, this integrated view uses Oracle Portal, the Portlet Repository, central authentication and authorization, content management tools, and federated search and taxonomy tools to display static and dynamic content which includes business and non-business specific operational transaction data, analytical data, web content, and the document repository.
Solution Stories	The end user accesses the eWorkplace, either unauthenticated or authenticated, to access all appropriate information and electronic tools to accomplish their daily work tasks.



An unauthenticated user accesses the eWorkplace to view the default interface including federated search, taxonomy, utilities, application interfaces, institutional workbenches, and static content. The user authenticates to the eWorkplace to view and interact with personalized content, operational information, applications, and workbenches that are applicable to their role(s). Using advanced and integrated search tools, the user retrieves content and inferred content that reflects application roles and responsibilities. The authenticated user customizes the presentation and content of their eWorkplace.

Benefits	Improved user experience and efficiency due to the personalized, customized interface and federated search. Reduction in training due to common look and feel. Ease of access to job-related information and applications through access to integrated information and personalized content. Flexibility for reuse of portlets and portlet development methodology.
Products	Oracle Portal, OAS, 10G, Java/JSP/Servlets, Javascript, HTML, Verity, Taxonomy, collaboration tools, IE6, Mozilla, Remedy, Livelink

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Foundational Capability Cases

		Business Architecture Pattern				
		User to Business (Transactional)	User to Data (Analytical)	Bus. to Bus. (Sys. to Sys.)	User to User (Interaction)	Org. to User (Informational)
Presentation Tier						
Process Tier	**Federated** (loose coupling)					
	Monolithic Homogenous (tight coupling)					
Data Tier	**Structured** (data)					
	Unstructured (content)					

Foundational Capabilities

- **Application Roles and Authorization**
- **Application Security**

- Browser and HTML based Webtop stds.

Integration Architecture Capability Case

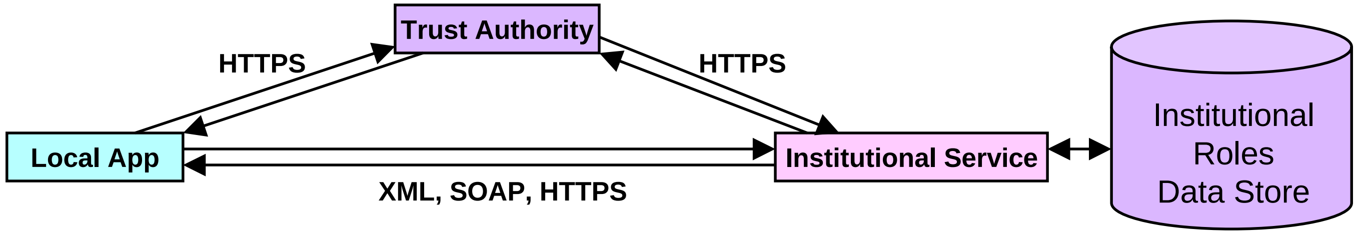
Foundational Capability: Security Model for Heterogeneous System Integration

Intent

Synchronous and asynchronous request/reply application-to-application integration in which sensitive data is accessed or transformed requires the capability for one application to securely interact with another application. It is a core capability needed to support our strategic services oriented architecture. The intent is to centralize a trust mechanism such that services can be invoked by a restricted set of users and systems in an auditable way.

Solution Stories

Applications have the ability to securely identify themselves so that system interactions are auditable and able to be restricted.



A local application would like to update an employee’s address in an institutional system. The application sends a request to the institutional trust server asking for a token to include or encrypt its message to the institutional system. The institutional trust server then issues a challenge to the local application to prove its identity. Upon completion of a successful challenge, the local app receives a token from the trust authority. The updated message is then sent to the institutional service with the token. The institutional service then uses the token to query the trust server as to the authenticity of the message. The trust authority responds with a message indicating the authenticity of the token.

With a non-repudiation mechanism in place, a user identifier/credential can be sent along with the message indicating which user is invoking the service call. With an institutional roles store, the institutional service can query such a store to determine if that specific user has the ability to invoke the institutional service in question. In a less fine grained model, a system identifier can be sent with the message and a institutional systems definition can be queried to determine authorization.

Benefits

Ability to restrict access to sensitive services based upon user authorization levels. Potential for the aggregation of access control levels to a centralized mechanism.

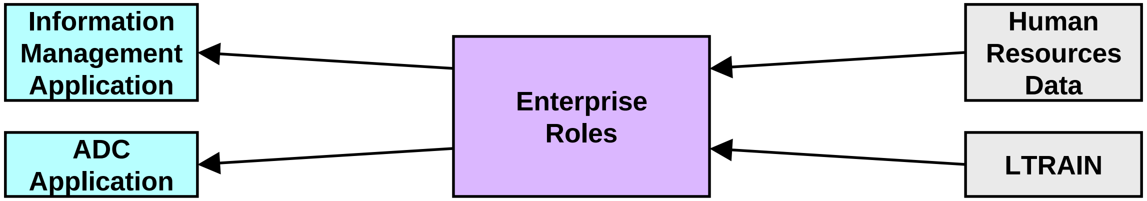
Products

SAML, Digital Signatures, Java Cryptography API

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case	*Foundational Capability:* **Enterprise Roles and Application Authorization Integration**
Intent	To provide an information service and architecture that efficiently and securely grants and revokes appropriate access to business information and applications. An enterprise roles and application authorization solution will shape the user experience by simplifying access to laboratory systems (i.e., via single sign-on) and by streamlining the procedures for granting, revoking and managing authorization. Managing authorization information is as important to customer service as it is to security and offer the potential for streamlined processes and clarified responsibilities.
Solution Stories	Laboratory employee lifecycle encompassing hiring, job change, retraining and ultimately termination.



Laboratory employees' access to applications are governed by application roles. Many application roles are derived from institutional or organizational roles that apply to employees immediately upon hire or reassignment. Processing a change in an employee's employment status needs to reflect the ability to perform job functions within systems immediately upon change. By providing application authorization based upon an enterprise roles architecture allows the employee to complete business processes leveraging applications without downtime. Employees are also trained to accept new responsibilities and are required to maintain their training to complete assigned business processes. Providing training data to the enterprise role repository will enable a secure and auditable grant and revoke scheme for application authorization. Consequently an employee that fails to complete required training refreshers will be unable to work on applications that require the requisite knowledge, while newly trained employees will be able to access applications needed for their new assignments following the training session. The Human Resources system will provide employment status information to the enterprise role repository, securing the application access to systems immediately on termination.

Benefits	Increased employee productivity by eliminating downtime between access list updates, improved application and business process security.
Products	

Note: Capability case diagrams reflect logical, not physical relationships.

As of 2/27/2004

Integration Architecture Capability Case	
Intent	
Solution Stories	
Benefits	
Products	

Note: Capability case diagrams reflect logical, not physical relationships.

As of