

Technical White Paper by RJ Lee Group: Federated Architecture Approach

1 Federated Architecture Concept

RJ Lee Group (RJLG) concept design for a Federated system to support a federated architecture, whether database or application, is the union of independent disparate systems under a single cohesive view. There are two main types of systems within a federated portal architecture, providers and consumers. The providers are systems that publish data and application services that may be specific to a singular business domain. The consumer system aggregates the content of the remote providers into a single view with transparent boundaries.

Federated systems can also be organized in hierarchical structure where a consumer of one set of remote services can serve as a provider to another consumer. In real terms this would be like System 1 consuming remote data services from System 2 and providing those same services to an Enterprise Federated System.

The following figure illustrates the high level conceptual architecture for the federated concept. The Centralized View of Data and Applications serves as the web portal¹ as the centralized users view to consume services (data or applications) from a provider system such as System Domain 1, System Domain 2, System Domain 3 and other systems or domains. It's important to note that the data and applications from the provider systems remain physically separated (distributed / non-centralized) and are virtually integrated / centralized within the federation that gives the end user a single point of entry to access data and applications throughout the enterprise.

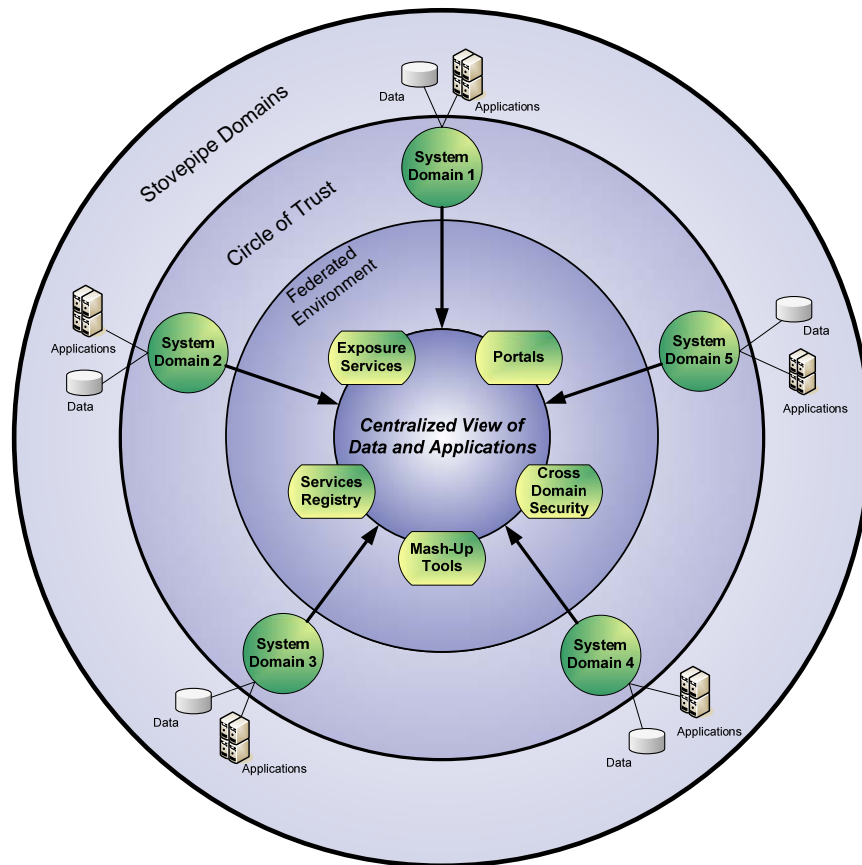


Figure 1: High Level Conceptual Federated Architecture

¹ A web portal will provide an amalgamation of software applications that consolidate, manage, analyze and distribute information across and outside the enterprise (including business intelligence, content management, data warehouse/marts and data management applications). A web portal is a web application. A portal server is an application having a portal and its management capabilities. In traditional web applications, a separate management section for managing the web application exists. However, a portal server provides the application as well as the management section. In other words, a portal server is an application deployed inside an application server. Whereas, an application server is a system that provides the execution environment that is at the core of network computing or web-based architectures, providing a full set of services.

Technical White Paper by RJ Lee Group: **Federated Architecture Approach**

2 Features provided by a Federated Portal Architecture

There are many features that may be centralized in a federated architecture; a few of them are user security and system metadata. The central consumer of the remote services may own user, authentication, and authorization information. Otherwise, there must be an agreed upon messaging layer where this and other security data is shared. This data can, however, be hierarchically distributed across the provider / consumer chain. While basic user information would still want to be centrally located, the exact roles and permissions assigned to users may be stored at each federation member location.

Metadata² describing the data structures (*i.e.*, Ontologies³), services, and topology of the remote systems may be stored in a centralized location or shared through a common messaging layer. This is the central concept behind the Semantic Web⁴ which is an emerging standard for transforming physically disparate data into a globally linked database whose data is published in a repurposable form allowing data and business logic to be reused across different domains to perform a variety of specialized tasks, increasing the value of the data and applications that make up the enterprise and beyond.

2.1 Portal Standards and Operation

There are two well accepted portal standards: WSRP (Web Services for Remote Portals⁵) and JSR 168 (Java Portlet specification⁶). While WSRP and JSR 168 are common standards defining portals and portlets, they are not competitors. Each standard has its strengths and weaknesses but truly have their own focus. The WSRP specification defines a portal standard centered around SOAP and HTML communications over HTTP, and is the general standard that's independent upon the technology used (e.g., Java, Apple, .NET). The JSR 168 standard extends the J2EE web component standards to make an easy to use, Java based, portal architecture.

The two portal standards work well together and most popular application servers support both standards. As well, portlets developed to adhere to the JSR 168 standard can be exposed as WSRP portlets. Applications that have previously been written as J2EE compliant applications can be ported to JSR 168 portals / portlets and subsequently consumed by a central federation aggregate system (e.g., application services).

RJLG promotes web application portal implementations following the WSRP standard, but JSR 168 (or higher specification) can be used if the web application was written in Java.

² Metadata is "data about other data". An item of metadata may describe an individual datum, or content item, or a collection of data including multiple content items and hierarchical levels, for example a database schema. In data processing, metadata is definitional data that provides information about or documentation of other data managed within an application or environment. The term should be used with caution as all data is about something, and is therefore metadata.

³ Ontology (in information or computer science) is a formal representation of a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to define the domain.

⁴ Semantic Web is an evolution of the World Wide Web in which information is machine processable (rather than being only human oriented), thus permitting browsers or other software agents to find, share and combine information more easily.

⁵ Web Services for Remote Portlets (WSRP) is a standard for content aggregators, such as Web portals, to access and display content sources (*i.e.* portlets) that are hosted on a remote server. WSRP is a technology agnostic protocol designed for accessing remote Portlets in a standard manner. The WSRP specification defines a web-service interface for interacting with interactive presentation-oriented web services.

⁶ The Java Portlet Specification defines a contract between the portlet container and portlets and provides a convenient programming model for portlet developers. The Java Portlet Specification V1.0 was developed under the Java Community Process as JSR 168.

(Sources: <http://en.wikipedia.org/wiki/>, <http://www.oasis-open.org/>, <http://www.jcp.org>)

Technical White Paper by RJ Lee Group:

Federated Architecture Approach

2.2 Data Standards and Operation

Data standards within a Federated system shall use XML⁷ document exchange to enable interoperability with XML schema definitions and metadata. These XML-based specifications will enable end-to-end, vertical & horizontal information integration using SOAP-formatted XML envelopes and have their interfaces described by WSDL.

XML Web Services will be used to support interoperable System-to-System interaction over the federated network to define a set of business procedures running within the federated network, serving interoperability and as common gateways. Common Web Service specification standards include SOAP⁸, WSDL⁹, UDDI¹⁰, WS-Security¹¹, WS-ReliableMessaging¹² and WS-Reliability¹³.

2.3 Security Standards and Operation

Security is often considered one of the most challenging and concerning aspect by Stakeholders with the integration and centralization of sensitive or private data. RJ Lee Group's security approach within a federated architecture solves and addresses these challenges and concerns. The three primary components of Federated Security include Service Providers, Identity Providers and Policy Agents. Service Providers manage user accounts located in remote LDAP¹⁴ directories. Identity Providers issue user credentials and verifies that the issued credentials are valid. Policy Agents protect content on web and application servers from unauthorized intrusions.

The two primary technology standards used in federated security are Public Key Infrastructure (PKI¹⁵) and Security Assertion Markup Language (SAML¹⁶). However, these technologies are not mutually exclusive and each address security issues in various ways. PKI uses digital certificate as credentials to identify a user. Application hosting environments within an organization can use the certificates to determine if the users are valid and to acquire the users' IDs. Users can be authenticated using PKI to systems independently of other systems. PKI can be a building block toward a federated architecture.

SAML is an XML standard for exchanging authentication and authorization data between security domains. Identity management is maintained by identity providers that contain both authentication information and authorization information (e.g., SAML 2.0). The application hosting environments will rely on the identity providers to authenticate the users and will request the authorization information from the identity providers. A SAML Token is used for session authentication. PKI certificates are also used for authentication as well as denying user entry by revocation of the user's certificate. SOAP over HTTPS communications are used between the application hosting environments and the identity providers.

⁷ Extensible Markup Language (XML) is a W3C-recommended general-purpose markup language that supports a wide variety of applications. XML languages or 'dialects' are easy to design and to process. XML is also designed to be reasonably human-legible, and to this end, terseness was not considered essential in its structure. XML is a simplified subset of Standard Generalized Markup Language (SGML). Its primary purpose is to facilitate the sharing of data across different information systems, particularly systems connected via the Internet.

⁸ Simple Object Access Protocol (SOAP) is an XML-based, extensible message envelope format, with "bindings" to underlying protocols (e.g., HTTP, SMTP and XMPP).

⁹ Web Services Description Language (WSDL) is an XML format that allows service interfaces to be described, along with the details of their bindings to specific protocols. Typically used to generate server and client code, and for configuration.

¹⁰ Universal Description, Discovery and Integration (UDDI) is a protocol for publishing and discovering metadata about Web services, to enable applications to find Web services, either at design time or runtime.

¹¹ Web Services Security (WS-Security) defines how to use XML Encryption and XML Signature in SOAP to secure message exchanges.

¹² Web Services Reliable Messaging (WS-ReliableMessaging) is a protocol for reliable messaging between two Web services.

¹³ Web Services Reliability (WS-Reliability) is an OASIS standard protocol for reliable messaging between two Web services.

¹⁴ Lightweight Directory Access Protocol (LDAP) is an application protocol for querying and modifying directory services running over TCP/IP.

¹⁵ The Public Key Infrastructure (PKI) is a set of hardware, software, people, policies, and procedures needed to create, manage, store, distribute, and revoke digital certificates. In cryptography, a PKI is an arrangement that binds public keys with respective user identities by means of a certificate authority (CA).

¹⁶ Security Assertion Markup Language (SAML) is an XML-based standard for exchanging authentication and authorization data between security domains, that is, between an identity provider (a producer of assertions) and a service provider (a consumer of assertions). SAML is a product of the OASIS Security Services Technical Committee.

(Sources: <http://en.wikipedia.org/wiki>)

Technical White Paper by RJ Lee Group: Federated Architecture Approach

3 Approach

The following figure illustrates the high level technical architecture for the federated approach. The Centralized View of Data and Applications serves as the web portal as the centralized users view to consume Exposure services (data or applications) from a provider system such as System Domain 1, System Domain 2, System Domain 3 and other systems or domains. It's important to note that the data and applications from the provider systems remain physically separated (distributed / non-centralized) and are virtually integrated / centralized within the federation that gives the end user a single point of entry to access data and applications throughout the enterprise.

The WSRP standards shall be followed so the web application service providers can be consumed in the portal. Two key architectural components are: Service-Oriented Architecture (SOA¹⁷) and portlets¹⁸. SOA offer building blocks from each domain that can be aggregated and repurposed to create portlets that offer domain specific functionality. These portlets are then aggregated together to form a portal which offers consolidated access and cross-cutting functionality such as authentication/authorization. A Federated SOA implementation shall leverage SOAP Web Services and other mature remote service protocols. The following figure illustrates the technical level federated conceptual architecture.

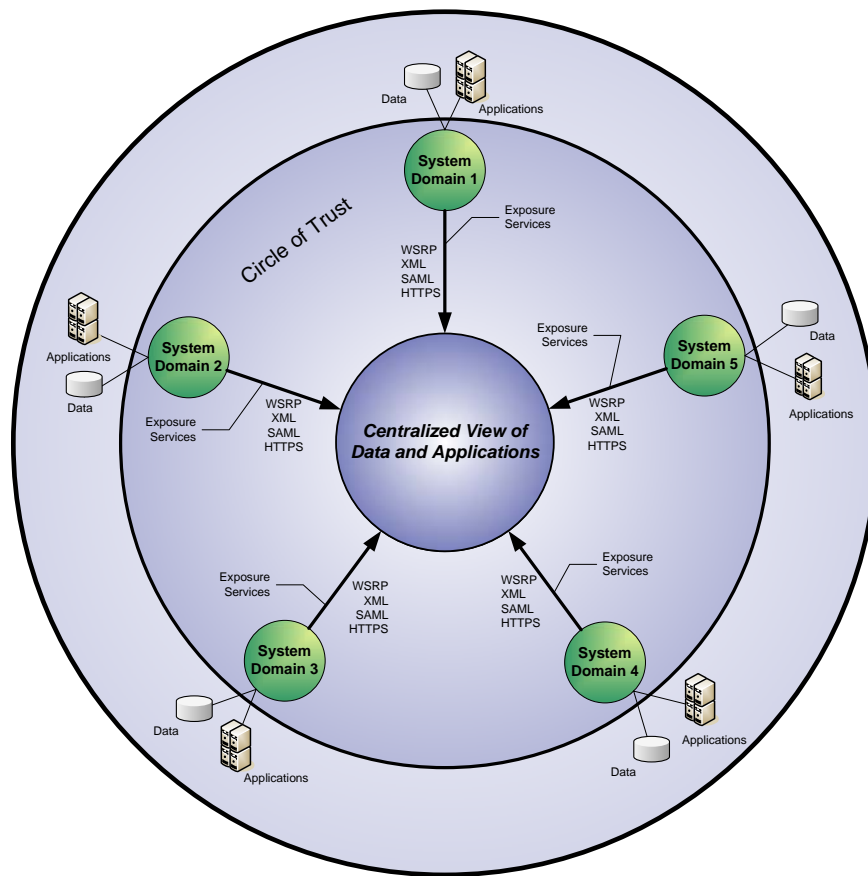


Figure 2: Technical Level Conceptual Federated Architecture - Application & Data Services

¹⁷ Service-oriented architecture (SOA) describes a software architecture that defines the use of loosely coupled software services to support the requirements of business processes and software users. Resources on a network in an SOA environment are made available as independent services that can be accessed without knowledge of their underlying platform implementation.

¹⁸ Portlets are pluggable user interface software components that are managed and displayed in a web portal. Portlets produce fragments of markup code that are aggregated into a portal page. Hence a portlet (or collection of portlets) resembles a web-based application that is hosted in a portal.

Technical White Paper by RJ Lee Group:

Federated Architecture Approach

3.1 Proof of Concept and Prototype Development

RJ Lee Group's federated architecture methodology is executed in an iterative approach starting with small proof of concepts that build toward a functional prototype federated system. RJ Lee Group's HP Blade System (hardware platform) located at RJLG EverGreene Technology Park facility in PA is used as the web portal server for the prototype. Initial prototyping efforts begin in RJLG's test environment utilizing existing hardware and software, or specific hardware/software can be used per stakeholder requirement.

The federated prototype starts with a selected system consumed by the web portal such as consumption of System Domain 1, and then phase in to consume a 2nd provider such as System Domain 2. Once the initial federated environment and prototype/pilot system is established and any issues resolved, other systems are migrated to the federated environment.

Although it is expected the existing RJ Lee Group hardware and software can be used for initial prototyping/pilot efforts, additional hardware and/or software may be required by either RJ Lee Group or the other systems to support requirements of a federated architecture. Determination will be made during the inception (scope/requirements) and elaboration (analysis/design) phases.

3.1.1 Key Objectives

Implement a federated architecture as a functional prototype/pilot system for the Stakeholder.

- Single sign-on thru web portal.
- Establish trust relationships with all provider systems adhering to federated security and identity management standards.
- Domain centralization user perception but systems/applications are physically distributed and owned/maintained by domain areas.
- Publish open specifications to enable end-to-end, vertical and horizontal information integration.
- Link computational resources and promote their reuse to help Stakeholder respond more quickly and cost-effectively to changing conditions.

3.1.2 Next Steps

- Define scope of federated prototype/pilot system.
- Define scope of federated production system.
- Define milestones and determine schedule.
- Develop detailed project iterative plan.
- Begin project to implement prototype/pilot system – inception, elaboration, construction, transition phases.