

WHITE PAPER

CentraSite: An Integrated SOA Registry and Repository

Sponsored by: Fujitsu and Software AG

Sandra Rogers
February 2006

The Business of Managing Services

There are many benefits to implementing a service oriented architecture (SOA), such as the efficiency gained in utilizing shared resources, the flexibility of composing systems with model-based constructs, and the ability to incorporate centralized policies and oversight mechanisms. However, as services proliferate and the applications that consume them get more complex, the need to better manage and govern these assets increases dramatically. Such efforts will inevitably help organizations better capitalize on what SOAs have to offer and also mitigate some of the risks inherent in a highly modular and distributed computing environment.

Metadata at the Foundation of SOA

The key construct and advantage behind SOA is in leveraging informational or procedural services that fundamentally focus on their core value propositions, abstracted from any specific implementations. In creating this environment, each element of the system can stand apart and be reused in a variety of ways. However, the context behind these system artifacts and their critical relationships needs to be documented and exposed to all other elements of the system. These entities need to be managed throughout their life cycles and associated, changed, and versioned appropriately.

According to IDC research, one of the primary reasons that organizations believe they are not reaching satisfactory levels of system reuse is the lack of knowledge regarding what is actually available for reuse. By creating a mechanism to help business and IT resources identify and expose specific data views or functional processes, IT organizations can contain duplicative efforts in creating and enforcing aspects of policy, overall change control, and maintenance.

Many organizations, in their enthusiasm to take advantage of Web services, have been building up cadres of services with configurations in a somewhat chaotic fashion. Most of these enterprises have relied on the manual efforts of IT personnel to track and communicate additions and changes to available services among known parties involved with developing, sharing, deploying, and maintaining these system assets. Human intervention and organizational dynamics are usually involved in searching, discovering, and negotiating service usage scenarios. Some companies have elected to build their own rudimentary solutions to capture the most basic provisioning information. However, as the volume and complexity of SOA-based systems grow, so does the need to create more robust environments to manage them.

Governing SOA Within the Context of Change

Applications formed in true SOA fashion contain a host of fragmented parts, each of which needs to be managed. A process layer may define workflow execution rules, procedural services may contain discrete instructional processes in varied native languages, and user interface dimensions may be contained in explicit code or exposed via a portal or Web browser. Previous generations of applications had all these elements "tightly coupled" and bound. SOA, on the other hand, relies on metadata to define, track, and associate all these piece parts together in a more dynamic fashion.

The need for more granular control and comprehensive visibility across the network of services and processes is an obvious imperative. Services are meant to be composed together; therefore, each individual component can have a significant impact on others. The need to track *related* sets of services and their underlying elements, in context to the solutions they can form, is critical. For such a dynamic environment, capturing a significant amount of metadata at all stages of a process and a given service is essential.

Development concepts from the traditional application life-cycle domain need to be enhanced, extended, and more important, exposed to a wider base of potential consumers. In a shared environment, varied user and system roles and views will need to be considered, allowing tiered and secure access to information and policy control points. Managing a multiparty collaborative process for additions, deletions, and change management is an overt requirement. However, automated alert mechanisms, impact and dependency analyses, and comprehensive reporting on the status of services, and their unique interrelationships, should be well incorporated into the overall process.

Creating a virtual centralized mechanism to store and share relevant service-based information is core to this vision. One industry-supported methodology for managing this complex environment has taken form in registry and repository-based software.

IDC believes some of the benefits to such software include:

- ☒ Better organization and management of a structure of services over time
- ☒ Increased levels of oversight and coordination of service development and usage
- ☒ Visibility into available services and critical relationships between services and supporting componentry
- ☒ Prevention of building redundant or contradictory services and components (thereby saving both developer time in the short run and change management time in the long run)

The Integrated SOA Registry and Repository

Directories for systems management and security policy have been in use for years now. This concept is now expanding in scope and intended functionality, moving beyond a standard directory or registry with the ability to capture and navigate among multiple sources of service-oriented information. It should allow an individual, or a connected system, appropriate exposure and navigation to relevant system-level and business-level semantics necessary to form solutions from the underlying assets.

Utilizing a combined registry and repository service is an emerging practice for managing the proliferation of services and systems artifacts involved in SOA implementations. Administering these elements in a unified fashion, and allowing for a synchronized federated environment of multiple registries and repositories, is an ultimate goal.

A services registry provides a foundation for cataloging and classifying services and other components. It presents a unified environment for publishing, discovering, and retrieving metadata related to these system elements. A registry typically stores links or pointers to auditable service information and artifacts that reside outside the registry in some type of database or repository that contains more extensive or domain-sensitive configuration parameters.

Some of the primary tasks performed with the assistance of a services registry and repository include:

- ☒ **Identify services** — Service description and origination, processing specifications, communications protocol and device orientations, publisher assertions, binding instructions, registry/repository pointers or APIs
- ☒ **Search and locate** — content discovery and query against classifications
- ☒ **Service life-cycle tracking** — document service across stages of life cycle — development, test, approval process, migration, retirement, and deprecation — including support for versioning, check-in/checkout, and approval processes
- ☒ **Dynamic service invocation or binding** — provide directions and explicit protocol instructions
- ☒ **Classification** — catalog, navigate, and manage relationships among metadata and content
- ☒ **Associate services and other system artifacts** — describe dependency relationships and group sets of services with associated artifacts with an extensible taxonomy
- ☒ **Provisioning** — extend shared repository to capture and store multiple data types (including unstructured content and images) including instances of services schema, artifacts, process design models, documentation, etc.
- ☒ **Change notification services** — event-based change notification

- ☒ **Validation and transformation** — for adherence to defined standards, exceptions, and policy instructions
- ☒ **Administration** — access and rights control to repository contents for varied groups, roles, scenarios, and services descriptions to facilitate enforcement of when and who can change and modify services
- ☒ **Analyses and reporting** — to understand service and artifact relationships and perform impact analyses and parallel conflict resolution

A registry and repository could also reference external locations for where other system artifacts actually reside, such as within requirements management systems, automated test systems, security systems, management systems, and more.

A building role for registries and repositories is to support collaborative development and deployment *processes* within a shared environment — either by integrating into other application and management life-cycle systems or by providing direct functionality to address these needs. Incorporating an automated governance process with change control can save significant manual efforts and reduce the likelihood of errors or inadvertent exclusions. Additional reporting and analyses capabilities, such as tracing service lineage and performing subsequent change and versioning impact test scenarios, are also important considerations.

The need to support distributed environments is a critical and defining part of an SOA. Most enterprises acknowledge that it would be challenging to create and enforce use of a physically centralized SOA repository; thus, support of federated repositories is an important consideration. Specific elements of the Universal Description, Discovery, and Integration (UDDI) 3.0 specification provide a standardized mechanism for extracting information from multiple registries and repositories. Extending beyond these basics is the need to manage and synchronize federated registries and repositories and their contents.

The Importance of Standards and Model Extensibility

Industry standards continue to evolve from an SOA registry perspective; however, at any given point in time, multiple versions of standards, or a lack thereof, may need to be accommodated. Maintaining a resilient yet extensible metadata model is advisable because it allows the needed flexibility to not be bound to explicit schema.

IT and business groups and industry consortiums such as the W3C and OASIS have rallied significant vendor and enterprise support to address the growing need for interoperability standards. XML has served as a common foundation for most of these syntactical schema, including basic definitions for Web services description and communications protocols in the form of Web Services Description Language (WSDL) and Simple Object Access Protocol (SOAP), respectively.

UDDI is an important specification. This protocol continues to evolve and build industry support for tackling the Web services service registry and discovery space. Other directory and registry protocols do exist, however. Therefore, a federated view across registry and repository technologies is often desired. To standardize completely across the entire constituent parts of an enterprise-scale SOA

environment would be a difficult endeavor, so it is judicious to prepare for a hybrid environment. Repositories should provide interfaces to function with various tools (e.g., development and modeling, process automation, performance monitoring) to bring knowledge elements into the repository and vice versa.

Along with varied technical schema, reference models in an SOA address varied functions. These models help semantically define systems performance and security rules, service and component processing instructions, system (SLA) and business requirements, and contractual agreements. The ability to support, or minimally associate, elements from varied models is critical to ensuring that information is interpreted consistently throughout the system.

The Meaning of Management: Design-Time Versus Run-Time Domains

The term "management" tends to be overused in the IT industry. When speaking of management, one must first consider and ask the following questions: What types of system elements and policies are being addressed? Where are they defined and stored? Where and how are they enforced? Where and how are they monitored and audited?

A growing number of Web services and SOA-based software entrants can produce configuration-based systems for assembling constraints and metadata, but only a portion are actively engaged in an automated, run-time production environment. Some organizations will seek to manage policies for services in a centralized, albeit virtualized, capacity where the enforcing systems will extract the necessary information at run time. Varied opinions exist regarding the ultimate role of a registry/repository-oriented solution when combined with other aspects of services mediation. IDC expects further integration and convergence between design-time and run-time technologies to unfold over time.

CentraSite: An Integrated Registry and Repository for SOA

Fujitsu and Software AG have teamed up to create what they refer to as an *SOA registry and repository* to address the growing need to manage highly distributed service-based system assets. The companies have introduced a common product, CentraSite™, that both organizations' business process and integration software can interface to and populate with metadata. The solution is based on industry standards; therefore, the product is also designed to be leveraged by other systems in one's enterprise. Figure 1 provides an overview of the CentraSite™ solution.

CentraSite is targeted to support the evolving life-cycle management of services with the following features:

- ☒ **UDDI-based registry** — to publish, categorize, and discover services and their technical operational specifications

- ☒ **Service-related metadata management and storage** — to provide a centralized mechanism for storing, versioning, and managing SOA artifacts such as service profiles, process definitions, and policies

- ☒ **Multiformat data support** — for any binary data format
- ☒ **Query facilities** — menu-driven query facility with xQuery language support
- ☒ **Tiered access security** — to facilitate access and authorization at the element level, based on specific user, group, function, or scenario (The CentraSite repository is reported to be secured via SSL and designed to interact with LDAP, Active Directory, and Local Operating System security protocols.)
- ☒ **Usage and lineage reporting facilities** — to track status and use of services and, in particular, perform lineage and impact analyses to any additions or changes (Reports can be designed in Eclipse, or one can utilize a selection of predefined reports.)
- ☒ **Eclipse-based tooling** — for interoperability and integration with other system infrastructure and tools
- ☒ **CentraSite control user interface** — Ajax-based, textual interface to navigate and perform human-based service discovery and analyses, including impact analyses

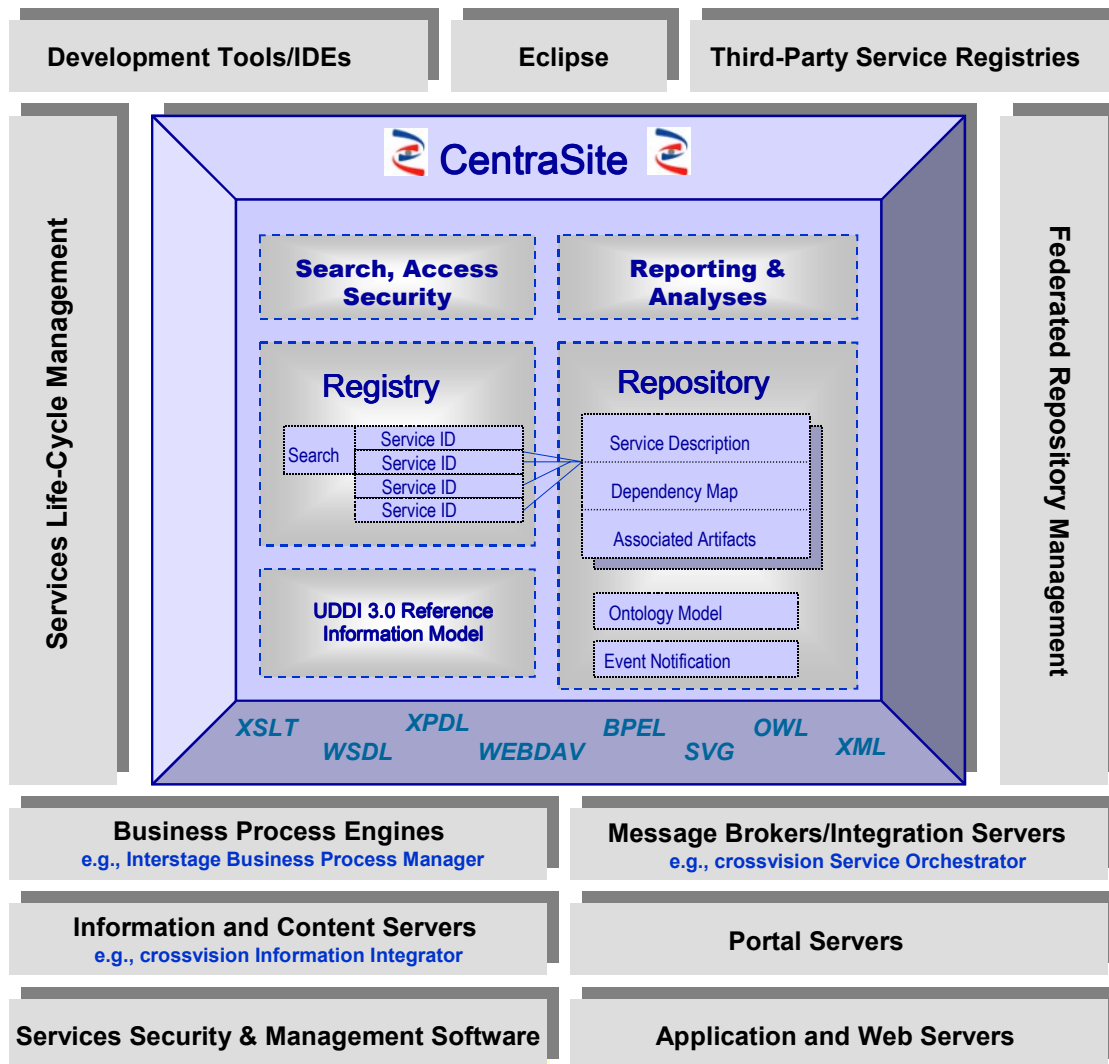
The CentraSite product is just one part of an ongoing technical and marketing partnership started by Fujitsu and Software AG in 2004. The product was developed, leveraging the firms' complementary strengths in application infrastructure, including metadata modeling, process and workflow solutions, and information management.

The repository utilizes an integrated Web-based Distributed Authoring and Versioning (WebDAV) interface. WebDAV is a standard protocol used for collaborative authoring and Web file management. It can be used for such items as namespace management (for nested collections or directories), file properties (such as last-modified, content length, or content type), and overwrite protection (persistent locking) to facilitate access for document-level versioning. It is the standard API used to store and retrieve development assets from Fujitsu's and Software AG's suite products.

Fujitsu and Software AG have designed CentraSite to support a variety of industry standards and Web services protocols and ensure WS-I compliance. XML is its main storage mechanism, supporting varied XML schema; XQuery is utilized as the primary search method, extended by support of the XQJ standard; and the JAXR API allows for integrated tooling with the repository. For integration with complementary service and system management technologies, CentraSite provides an API based on the Java Management Extension (JMX) standard. This interface is intended to extend integrated access to its administrative functionality and maintenance functions such as monitoring of activity, performance and configuration management, as well as backup and restore capabilities.

FIGURE 1

The CentraSite™ Solution



Source: Fujitsu and Software AG, 2006

The following products are CentraSite-enabled:

- ☒ **Fujitsu's Interstage Business Process Manager** — for business process modeling, automation, integration, monitoring, and optimization (version 7.1 and beyond)
- ☒ **Software AG's crossvision Information Integrator** — for organizing, managing, aggregating, transforming, and filtering business data (version 2.2 and beyond)
- ☒ **Software AG's crossvision Service Orchestrator** — for Web services and XML-based message routing, schema validation and transformation, and sequence workflows (version 2.1 and beyond)

These software products automatically ship with the core CentraSite technology, providing automated capabilities to populate the service registry and repository, along with integrated tooling to incorporate services managed by CentraSite into each respective system. Although these products do not depend on the CentraSite technology to operate or function, the environments will provide efficiency for users because of the integrated presence and the underlying metadata model optimized for them.

By extending the traditional UDDI schema with Fujitsu's and Software AG's product metadata models, the companies have taken what would have been a generic tool and facilitated its use and interoperability among these products out of the box. Any Web services interfaces, process models and orchestrations, or other system artifacts are automatically published into CentraSite to enable visibility of the specific assets, including WSDL files, BPEL sequence code sets, XSLT style sheets, XPDL process models, SVG diagrams, OWL information models, and other XML-based elements.

Looking Ahead to the CentraSite Road Map

Both firms are actively working on CentraSite-enabling other elements of their respective product lines; for example, Software AG's crossvision Legacy Integrator, which provides bidirectional legacy to service messaging, integration adapters, wrapping facilities, and multilegacy platform support. Legacy systems are important participants in today's SOA environments; thus, optimizing these systems to interact with registries such as CentraSite should add to greater levels of system extensibility.

Another example is Fujitsu's Interstage Contentbiz, an enterprise content integration solution focused on unifying access to multiple information sources and source types. Providing cohesive views across burgeoning amounts of information captured from various systems is imperative to most enterprises for both operational and reporting requirements. Most C-level executives report that increased visibility for more informed decision making is imperative and often a top intended IT initiative.

The CentraSite technology road map is slated to support the following expanded features:

- ☒ **Grouping and managing services as sets** — for example, the ability to process a set of services through varied stages of the life cycle
- ☒ **Support for UDDI 3.0** — critical aspects of the updated spec include support for multiregistry keys, digital signatures, and change notification
- ☒ **Customizable classifications** — to define additional taxonomic classifications for unique file requirements
- ☒ **Customer-extensible metadata model** — to incorporate and search against additional service attributes
- ☒ **Enhanced search, user interface, and dynamic reporting** — incorporating user-defined extensions and wizard-like interfaces
- ☒ **Event notifications** — to capture and invoke event messages that alert users to changes in the registry and repository

- ☒ **Graphical views** — to facilitate navigation and association among services
- ☒ **Standard service governance process** — to guide users in the management of the life cycle of a service (Subsequent capabilities will eventually be built to customize user-defined processes.)
- ☒ **Market availability and support as a standalone product**
- ☒ **Extended semantic information modeling**
- ☒ **Replication and federated registry support**
- ☒ **Prebuilt content and integration with partners' technologies**

Challenges and Opportunities for CentraSite

The services infrastructure market is a highly contested emerging market, with new entrants surfacing every day from all walks of the IT ecosystem. CentraSite should experience competition as well as partnering opportunities with incumbent and emerging services management, as well as other registry and repository vendors. This space could additionally be impacted by open source industry initiatives. It will be imperative for the product to adhere to the latest standards in a timely fashion and support even further heterogeneous environments (i.e., ebXML, other vendor's products).

The idea of an embedded registry and repository available for application solution providers that have not built their own and need a robust mechanism to compete in today's market should be enticing. The ability to support this technology and have other ISVs incorporate it into their respective products will provide enormous value to consumers as they seek to standardize on a common interface and taxonomical orientation for services and related schemas, models, and supporting collateral. Fujitsu and Software AG have OEM partnering experience from prior initiatives involving their respective business process technology and information management technologies; thus, it should be a natural evolution for those same partners to add CentraSite capabilities.

IT service organizations, such as systems integrators, may also be keen to leverage independent SOA repository capabilities with their SOA deployments. This base of potential users will look for advanced features, in particular, as well as the ability to integrate the technology and capitalize on extended metadata models to customize the repository as needed. As Fujitsu and Software AG build a community of CentraSite users and partners, it will be imperative to not assume use with or dependencies on any of their other products.

One key challenge, and opportunity, for this joint team will be educating enterprises on best practices in building system policies and business taxonomies in an abstracted manner to make the best use of a services infrastructure and repository. Fujitsu and Software AG have taken a critical first step by extending a standard meta model to support additional systems requirements derived from their own business process and information integration technologies. Further opportunities exist that will require a much more domain-specific approach — with industry-specific expertise or sophisticated business process acumen. Both of these avenues require that the companies work closely with channel partners that have particular industry presence and market leverage.

A large portion of the value in SOA seen by large enterprises comes directly from the ability to more readily repurpose and extend existing systems. Thus, the importance of legacy and host systems to many organizations will require any solution to adapt across varied informational entities and provide capabilities to handle transformations effectively. Software AG and Fujitsu should use their complementary experience to bring this additional dimension to the table.

Ideally, SOA futurists envision that the composition and binding of one element to another should be done dynamically at run time, and thus each participant needs to be located, identified, and utilized appropriately based on system and business policies. To prepare for this real-time dimension of SOA, Fujitsu and Software AG need to ensure that the services in CentraSite are available and accessible to all industry-standard process and service management engines.

Conclusion

Organizations that are able to quickly react and change to capitalize on market opportunities are best positioned in an increasingly challenging economic and political climate. The velocity and complexities of such volatile business, and ultimately supporting systems dynamics, continue to drive greater degrees of automation, control, and visibility at detailed levels. Additional requirements to meet regulatory compliance mandates implore both business and IT executives to preserve an accurate picture of a systems architecture and its constituent parts at any given time.

A key benefit behind the highly virtualized and distributed nature of SOA is the flexibility to handle change. This added nimbleness creates a heightened need to automate the governance of what can be a volatile and complex environment, especially as shared entities are extended outside the domain of physically constrained control.

We repeatedly hear enterprises note that they do not want to be the integrators — they would prefer that software vendors' information models be interoperable and semantically linked. Although based on standards and open for use with other products, CentraSite technology is initially targeted to leverage Fujitsu's Interstage and Software AG's crossvision in an SOA construct. It is one step in this type of vision of cross-system and vendor harmonization. The product road map outlines a steady progression to provide customizable modeling to address organizations' evolving system environment and business requirements. This capability will be critical to creating an environment that can live beyond initial development efforts.

Copyright Notice

External Publication of IDC Information and Data — Any IDC information that is to be used in advertising, press releases, or promotional materials requires prior written approval from the appropriate IDC Vice President or Country Manager. A draft of the proposed document should accompany any such request. IDC reserves the right to deny approval of external usage for any reason.

Copyright 2006 IDC. Reproduction without written permission is completely forbidden.