JEMS: The Open Source Platform for SOA

THE PLATFORM FOR FLEXIBILITY, INTEROPERABILITY, AND CHOICE

Contents

EXECUTIVE SUMMARY

The term “Service Oriented Architecture” (SOA) has been described in the IT industry in various forms over the years. Since 2001, however, widely agreed-upon standards such as Java EE and Web services have enabled a new level of interoperability and SOAs. While many vendors have announced SOA initiatives and platforms, most implementations are dependent upon expensive, closed source platforms. Moreover, while there are a range of open source projects focused on the “piece parts” of SOA, nothing exists to ensure these piece parts interoperate within a cohesive whole.

JBoss directly addresses these issues with the JBoss Enterprise Middleware Suite (JEMS), the Open Source Platform for SOA. The JBoss Professional Open Source model presents JEMS as an affordable, developer-friendly, easily consumed and cohesive SOA foundation. JBoss Application Server (www.jboss.com/products/jbossas) is widely used today to host J2EE Web services and is a key platform for interoperability in a growing number of SOA deployments. JBoss Portal (www.jboss.com/products/jbossportal) and JBoss jBPM (www.jboss.com/products/jbpm) currently support SOA applications requiring a unified application user interface that can deliver process-driven business value.

By incorporating a business rules engine (www.jboss.com/products/rules), JEMS can enable dynamic processing and intelligent routing within business processes, based on service level agreements or other business rules. The SOA capabilities of JEMS will continue to be enhanced with the release of JBoss Messaging in early 2006. JBoss Messaging will be the backbone of JBoss Enterprise Service Bus (ESB), due later in 2006, which will be based on the Java Business Integration (JBI) specification (JSR-208).

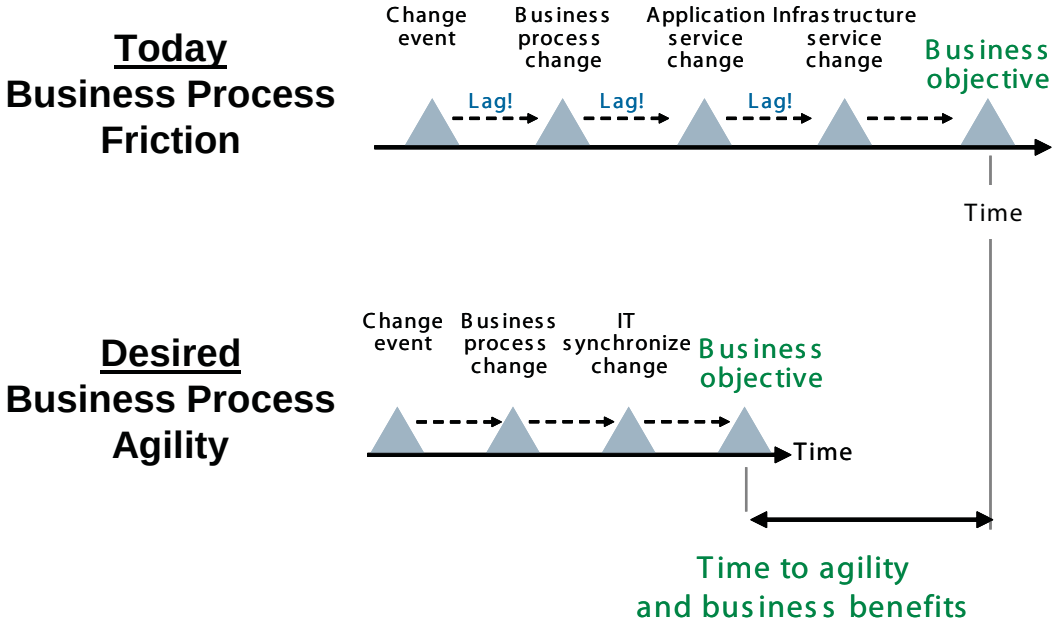
The technical value of JEMS as a SOA platform is amplified by the business power of the JBoss partner ecosystem which rounds out the JBoss SOA strategy and enables our customers to derive the maximum value and choice from their SOA initiatives. As exemplified in the recently announced alliance between JBoss and Microsoft, interoperability and flexibility of choice are of paramount importance to Enterprise IT organizations. (See press release “JBoss and Microsoft Outline Interoperability Goals”, September 27, 2005 www.jboss.com/partners/microsoft)

Bottom-line: Until now, enterprises looking to implement and realize the benefits from SOA have had to choose between pricey, monolithic, proprietary platforms or an assortment of open or mixed source components that need to be cobbled together. In contrast, JEMS delivers the only cohesive suite of market-leading open source middleware products that can be used alone, mixed and matched, or as a whole to build and deploy applications, business processes and Web services. JBoss raised to the power of the JBoss partner ecosystem enables JBoss customers to realize the maximum benefit from developing and deploying SOA solutions that utilize the JBoss Enterprise Middleware Suite (www.jboss.com/products).

CUSTOMER REQUIREMENTS AND DRIVERS FOR SOA

Enterprises are realizing that there is a better way to architect their IT infrastructure and applications. Today, most enterprises experience friction in their business processes. They also experience friction and delay in implementing IT changes to support new improvements to business processes.

Figure 1: Customer Requirements – Speed & Flexibility



Change events, such as a new partner or supplier coming on board, drive business process changes as shown in Figure 1. Business process change drives application service change which, in turn, can drive infrastructure service change. Finally, at the end of this chain of events, a new business objective may [or may not] be met, such as integrating a new supplier into an enterprise supply chain. Typically, there are time lags and delays at each step along the way to achieve a business objective. These lags and delays result in lower productivity and business process friction.

THE SOA SOLUTION

Enterprises are looking to eliminate as much delay and business process friction as possible, and SOA is a path to do this. SOA enables more flexible and reusable services that may be reconfigured and augmented more swiftly than traditional stove-pipe applications. Hence, SOA can accelerate time-to-business objective, resulting in better business agility.

Achieving this business agility is a key tenet of improved competitiveness. Enterprises are looking to SOA to maximize return by reducing complexity and cost of change. SOA can mitigate the risk of technological and business change since SOA platforms offer standards-based services that can be reused. SOA increases business agility and responsiveness by increasing reuse of components and services, reducing new code creation and associated cost. Finally, enterprises are looking to improve business performance including customer satisfaction and improved value chain execution.

SOA is an approach for building distributed systems that deliver application functionality as loosely-coupled services. SOA provides a standard way to represent and interact with application functionality by leveraging open standards. This is critical to improve interoperability and integration across an enterprise and value chain. Standards also reduce business process friction by enabling the reuse of services. Developers can create new applications from existing components more quickly than building functionality variations from scratch. SOA allows the developer to focus on application assembly which speeds time to implementation.

SOA Changes IT and Vendor Focus

Traditional application development favors monolithic architecture and structure. These applications were focused on functional automation and were designed to last. Cost reduction remains the primary motivator of this style of application development. Traditional application development drives long development cycles and is code-oriented. These long development cycles come about partly due to the fact that it is difficult or impossible to reuse tightly coupled functionality in this development paradigm.

Figure 2 highlights key features of traditional application in the left column. Figure 2 also contrasts traditional application development with SOA in the right column. SOAs are application and business process assemblies of business functionality. Hence, SOA presents a loosely coupled, agile and adaptive infrastructure and application architecture. SOAs are built using interactive and iterative development processes.

Figure 2: SOA Changes IT and Vendor Focus

Traditional Applications	Service-Oriented Architecture
Designed to last	Designed to change
Tightly coupled	Loosely coupled, agile and adaptive
Integrated silos	Composed of Services
Code-oriented	Process-oriented
Long development cycle	Interactive and iterative development
Cost-centered	Business-centered
Middleware makes it work	Architecture makes it work
Favors homogeneous technology	Favors heterogeneous technology

THE JBOSS SOA VISION

JBoss's vision for SOA is to make **JEMS the leading, *interoperable* Open Source Platform for SOA.** The market focus for JEMS is, in many respects, on the unserved market; that is, those organizations that have been forced to roll their own SOA infrastructure due to the prohibitive and high cost of today's commercial SOA platforms. JBoss is focused on addressing this unserved market with a cost-effective platform (JEMS) paired with a vibrant partner ecosystem which magnifies the JBoss value – we refer to this effect as "**JBoss Raised to the Power of Partners**".

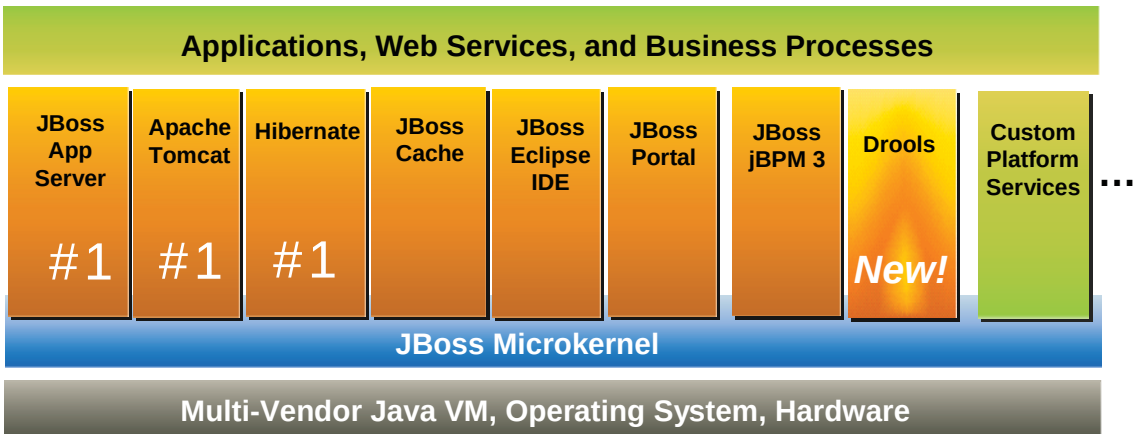
JEMS is an easily consumed SOA platform at an irresistible price that developers and businesses love. JEMS is comprised of plug-and-play modular components that enable *any* application and business process to *interoperate* with any other regardless of where its hosted. The Professional Open Source model enables a greater degree of openness and flexibility than traditional, closed source "Super Platforms". The transparency of the Professional Open Source model enables JBoss and its partners to deliver high quality services and support. This openness of architecture and Professional Open Source model makes JEMS attractive and safe for partners.

JEMS – THE OPEN SOURCE PLATFORM FOR SOA

The JBoss Enterprise Middleware Suite is comprised of such category leaders as JBoss Application Server, Apache Tomcat and Hibernate. Newer products that are rapidly establishing leadership positions in the open source community based on customer and partner adoption include JBoss Portal, JBoss jBPM, JBoss Cache and JBoss Eclipse IDE. Together, this middleware suite offers the foundation for SOA and a path to greater business agility. Its modularity enables enterprises to standardize on JEMS at their own pace. Many companies include parts of JEMS such as Hibernate, JBoss Cache and JBoss Portal to enhance existing application and SOA deployments. As companies discover the benefits of Professional Open Source-backed middleware, they can add other JEMS products for new workloads or migrate existing work as they see fit.

Figure 3 illustrates JEMS – The Open Source Platform for SOA and its component products. The Drools project joined JBoss in October 2005 and will become a JEMS product, likely named JBoss Business Rules, by early 2006. Drools is the leading open source business rules engine and adds important SOA capability to JEMS. Rules engines enable enterprises to separate complex business decision making logic from presentation and process logic, making it easier to add and maintain business policies.

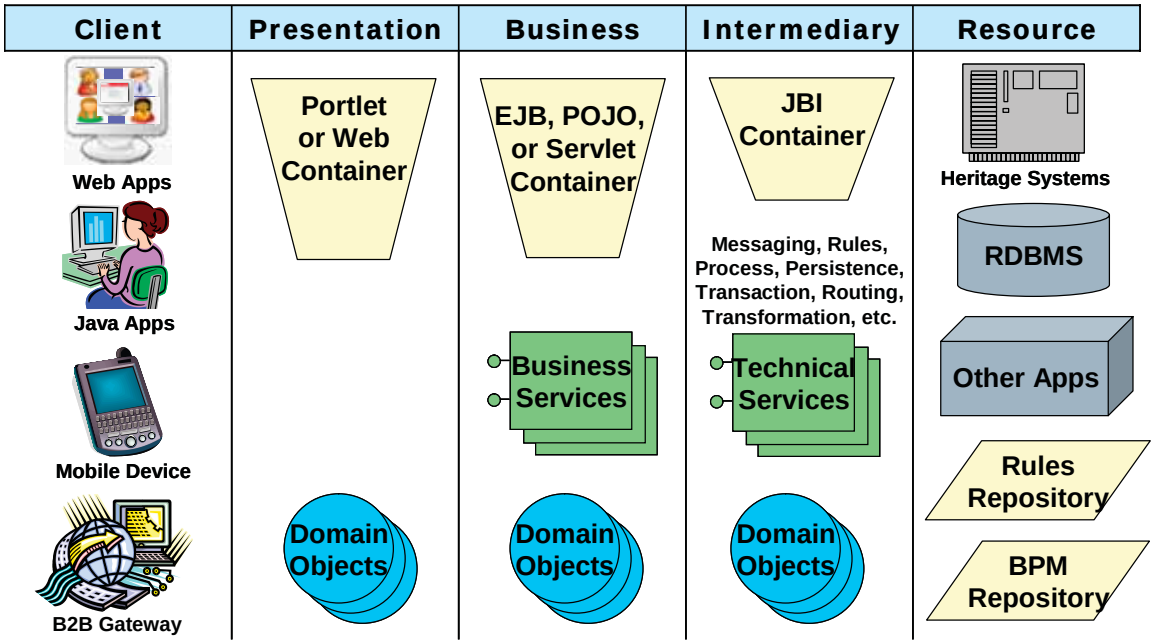
Figure 3: JEMS – Open Source Platform for SOA



SOA Tiers and Components

Figure 4 illustrates the primary logical tiers from an application perspective within a SOA. The three middle tiers (Presentation, Business, and Intermediary tiers) primarily exist to connect Clients to Resources and to do so in a manner that is efficient, scalable, and minimizes the effect and scope of changes. Within the system, there are three basic elements: Domain Objects (which for JEMS are typically Java objects), Business Services, and Technical Services.

Figure 4: Logical SOA Tiers and Components



The ***Client Tier*** manages interactions with the user. It renders HTML, presents application data, intercepts user input and may even do rudimentary application-specific range and syntax checking. In a J2EE application, this tier typically executes in a standalone Java application, Web browser, mobile device, or B2B gateway.

The ***Presentation Tier*** essentially provides different ways for clients to interact with components in the Business Tier such that the Business Services may be shared among multiple applications. It handles exceptions that occur during invocation of the Business Services and also transforms the data in the Domain Objects to other formats required by the different clients in the Client Tier. In essence, the Presentation Tier provides whatever mechanism the client needs to interact with the Business Tier. Thus, while the Presentation Tier may have to change in response to new client requirements or devices, the business logic and Business Services can remain unchanged.

Domain Objects exist in all three middle tiers because their job is to transfer data between system components. For example, the Technical Services in the Intermediary Tier mediate the transfer of data from the Resource Tier to the Business and Presentation Tiers. For example, for reading and writing data, the Persistence Service in the Intermediary Tier interacts with a Domain Object within the Business Tier or Presentation Tier and provides indirect access to the data stored in the backend resource. Changes to the data are then coordinated through the Domain Object and stored in the corresponding repository in the Resource Tier by a corresponding Technical Service.

The ***Business Tier*** is responsible for implementing ***Business Services*** and making them available as service-oriented interfaces to the Presentation Tier. The Business Services leverage the Domain Objects and the Technical Services to implement the business logic. Note that the Business Tier Domain Objects are the distributable data representation for communicating with the Presentation and Intermediary Tiers.

The ***Intermediary Tier*** provides ***Technical Services*** such as rules and process/workflow engines, query, transformation, and persistence. In general, if the application requires access to external systems or any third-party component, this tier provides adapters to those resources via Technical Services. These services act as "abstraction layers," such that a change to, or replacement of, a component in the Resource Tier does not affect components in the Business Tier. The Java Business Integration (JBI) container standardizes the platform for some of the SOA-related Technical Services.

The ***Resource Tier*** is where the shared enterprise resources such as database systems, business rules repository, BPM repository, and so forth, reside. These resources can be accessed from the Technical Services in the Intermediary Tier. Some examples of Technical Services accessing resources are: a) the persistence service reading and writing rows to/from the database, b) a process/workflow service initiating and executing businesses processes, c) a rules service executing business rules, and so forth.

JBoss SOA Use Cases

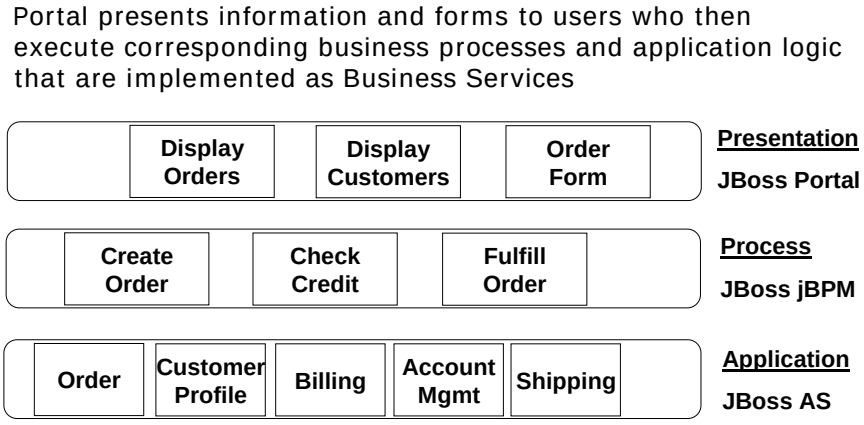
Enterprise customers are deploying SOA implementations on JEMS in increasing numbers. These customers are tired of architecting their SOAs around expensive proprietary license products. They look to open source platforms like JEMS to relieve financial and technical pain associated with monolithic super platforms.

Many of these enterprises, however, have significant investments in these super platforms. JEMS takes this fact into mind by enabling them to take advantage of the plug-and-play nature of JEMS and its open and interoperable value proposition.

Multi-Tiered User Example

Order processing represents a good example of SOA where shared components across many business processes can be reused. Some of these functions that may be represented as services include: a) order form and customer record presentation through JBoss Portal, 2) creating an order, checking credit and fulfilling the order; all services orchestrated by JBoss jBPM, 3) and application logic and data access can be presented as services built using EJB3 and standards-based Web services. As shown in Figure 5, JEMS products provide a range of SOA capabilities across the application, process and presentation layers.

Figure 5: Multi-Tier SOA Use Case - Processing an Order



Customer – Financial Services

Financial services companies are building their SOAs on JEMS. Cost benefits and agility are drivers for the decision to leverage JEMS in their infrastructure. Some are using JEMS as both an application and an integration foundation and take advantage of advanced features such as JBoss Cache and clustering for SOA scalability and performance. Many of these companies realize that refactoring old applications into the SOA model readies the organization for a more agile future. They create pools of coarse grained services using stateless session beans as endpoints. Some of the services are presented as Web services and registered in UDDI directories for third party consumption. Focused units of functionality are the key for agility and reuse while leveraging JBoss Application Server's interceptor capabilities to handle cross-cutting concerns such as security.

Customer – Insurance Company

Insurance firms are also building SOAs on JEMS. In some of these cases, the cost and complexity of traditional EAI drove a different approach. Some companies are adding additional Oracle or SAP applications which need to be integrated into their business processes. These applications drive dozens of service endpoints that need to be defined. Services are either stateless session EJBs or Web services depending on performance considerations and third party access requirements. Leading enterprises are deploying JEMS as the SOA fabric interoperating with mainframe and distributed custom applications along with Oracle or SAP.

Customer – Media Company

Media firms seek competitive advantage using SOA and JEMS. In this industry, JBoss sees duplication, responsiveness and competitiveness as pain points using monolithic architectures. JBoss Application Server has been deployed as the SOA services hosting environment for enterprise portal consumption. These services are exposed as light-weight Web services. The enterprise portal drives many thousands of interactions per hour driving page creation from the SOA JEMS platform. Many partners and customers access these enterprise portals.

JEMS Support of SOA Standards

JEMS supports the key Java, XML and Web services standards that have been widely adopted by most of the industry to build SOA infrastructure. JEMS supports these standards to enable ISV, partner and enterprise interoperability and integration.

Java and Web services standards supported include:

Java and Web services standards supported include:	
» EJB	» WS-I Basic Profile
» JSP and Servlet	» WS-I Attachments Profile
» JMX (Java Management Extension)	» SOAP
» JMS: Java Messaging Service	» UDDI
» JTA: Java Transaction API	» JAX-RPC: Java API for XML-based RPC
» CORBA	» SAAJ: SOAP with Attachments API
» J2EE Connector Architecture	» JAXR: Java API for XML Registries
» JDBC	» Java Portlet Specification (JSR-168)
» JNDI (Java Naming and Directory Interface)	» BPEL (Business Process Execution Language)
» JAAS: Java Authentication and Authorization Service	» WSRP (Web Services Remote Portlets) » ON JEMS ROADMAP
» JAXP: Java API for XML Processing	» JBI (Java Business Integration) » ON JEMS ROADMAP

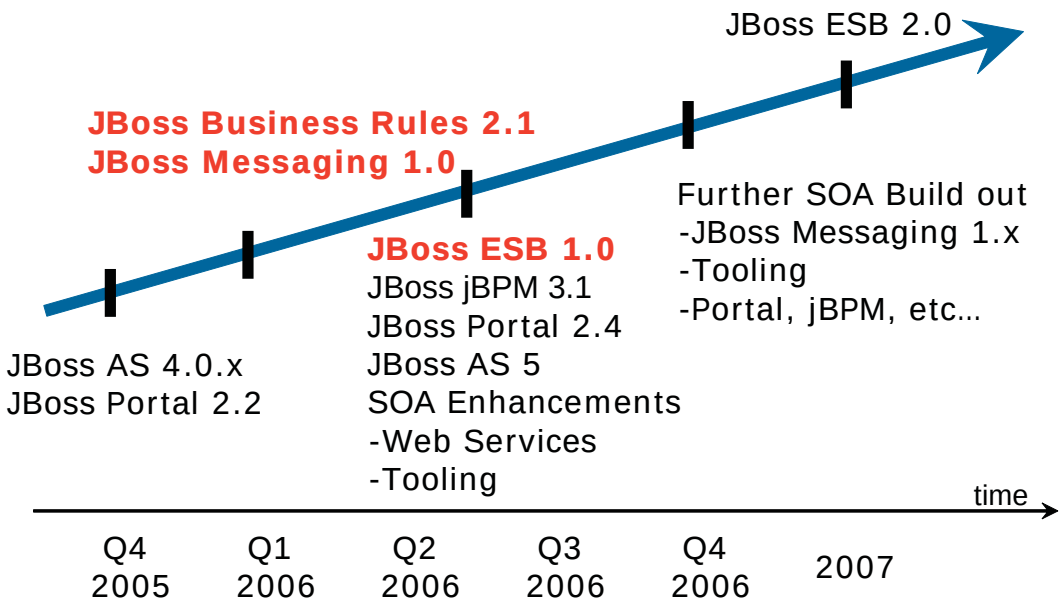
JEMS SOA Roadmap

JEMS is being used today as the open source platform for SOA across a range of implementations, spanning simple service publication up through significant business process management and interaction scenarios including support for hundreds of services and millions of transactions per day. Today, JEMS provides the necessary capability and products for service development, publication, and hosting.

As illustrated in Figure 6, JBoss plans to enhance its Open Platform for SOA by introducing JBoss Messaging. JBoss Messaging will be a fully compliant JMS 1.1 implementation. Also, JBoss Messaging will be available as a standalone product, as well as integrated with JBoss Application Server (replacing the currently bundled JBossMQ). Other areas of focus for JBoss Messaging include improved high availability and load balancing features, improved performance, and it will be designed to be the backbone of JBoss ESB.

The Drools project, a leading open source business rules engine, will be federated into JEMS by early 2006 with a likely product name of JBoss Business Rules. In 2006, JBoss Portal will be enhanced with support for WSRP. JBoss jBPM will be integrated into JBoss Portal to enable embedded workflow in portal-based application presentation. Later in 2006, the first release of JBoss ESB will be available. This release will be in tandem with the new micro-container-based JBoss Application Server 5. Other 2006 SOA enhancements include additional Web services standards support and JBoss Eclipse IDE tools support for SOA construction.

Figure 6: JEMS Open Source Platform for SOA Roadmap



NOTE: Dates are meant to illustrate JBoss strategic intent. Exact delivery dates may change.

Enterprise Service Bus (ESB) and SOA

As we have seen, enterprises are building SOAs on JEMS. To further facilitate these efforts, JBoss is building an Enterprise Service Bus product targeted for mid-2006. JBoss supports the Java Business Integration (JBI: JSR-208) direction and views this as a standard foundation for SOA. The JBoss ESB will enable collaboration between different technologies including other JEMS components such as JBoss jBPM as well as third party products and plug-ins.

As many players in the industry rally around this standard, JBoss expects to see a wide variety of third party components, both open source and traditional, expand the integration ecosystem and provide customers with greater choice. Opportunities for partners will abound in leveraging the high quality, cohesive JEMS suite and JBoss ESB to solve enterprise business process automation and integration problems.

JBOSS RAISED TO THE POWER OF PARTNERS

SOA is about improving business execution and IT's ability to respond to business change and opportunities. To fully maximize a SOA, an enterprise will need to draw on resources beyond any single IT vendor. For IT vendors, JBoss's Professional Open Source model offers the best of the open source community along with a cohesive and high quality SOA platform around which to build their solutions. JEMS is a safe choice for JBoss partners since JBoss does not compete with SOA professional services providers and independent software vendors.

Figure 7: JBoss Certified Partners

Advantages of Using JEMS for SOA – Large Partner Ecosystem

JBoss has built a large ecosystem of partners to extend, complement, and support JEMS with additional software, hardware and professional services. These JBoss Certified Partners add significant value to JBoss deployments and help enterprises build SOAs to improve business execution. Figure 7 illustrates a range of JBoss Certified Partners. These include nearly all of the system providers, many solution providers, and services companies.

Key areas of the solutions stack JBoss Certified Partner's build businesses in include:

- Platforms & systems providers - E.g., Azul, Dell, HP, NEC, Novell, Sun, Unisys
- Middleware and solution partners - E.g., Arjuna, ATG, DataDirect
- Interoperability with existing solutions
- Strategically focused on interoperability alliances such as with Microsoft
- SOA services providers - E.g., Capgemini, HP, Novell, Unisys

SOA Solution Example - HP – JBoss Partnership

JBoss and HP partner to provide a complete SOA solution ranging from the JEMS Open Source Platform for SOA, HP's OpenView, to HP Services for SOA. HP is a key partner that provides SOA and integration services, including business assessments, building on JEMS. HP is a JBoss partner in the following dimensions:

- Premier Technology Partner
- Premier Systems Integrator (this is where JEMS-based SOA services come in)
- JBoss Authorized Service Partner

Additionally, JBoss Application Server is a key component of the HP Linux Reference Architecture which is HP's open source solution stack framework.

HP SOA Services for JBoss

HP has a complete services offering for SOA built on JEMS. These services span the complete SOA lifecycle from initial envisioning and business assessments to management of deployed implementations. Figure 8 describes HP's startup, business assessment and architecture services of SOA development, as well as development, deployment and management services.

Figure 8: Portfolio of HP SOA Services for JEMS

HP Service	Description
SOA Envisioning	» Intended for large enterprises to develop an understanding of Service-Oriented Architecture (SOA) concepts, and to identify the benefits and potential impacts on the enterprise.
SOA Assessment	» Utilizes the HP SOA Agility Assessment approach to help customers develop a comprehensive roadmap to guide the adoption of SOA across their enterprise.
SOA Governance & Architecture	» Establishes the SOA Architecture Program Office, to oversee enterprise architecture and the SOA governance model as the enterprise is transformed.
SOA Enablement	» Based on learnings from the Governance and Architecture service, SOA Enablement prepares the infrastructure for the implementation of Service-Oriented Architecture.
SOA Service Development	» Helps the customer define, develop and deploy SOA business and IT services across the enterprise, line of business, or department, or at a project level.
SOA Software Development	» Provides for volume and scale in development and delivery of business and IT services through the SOA adoption. Ensures maximized productivity from customer development teams by utilizing global software development capability.
SOA Management	» For enterprises looking to gain control of their SOA adoption, including life cycle management, services management, monitoring, auditing, analysis, service level agreements and policies.

WHY JEMS FOR YOUR SOA?

SOA is about improving business and IT execution. With strong partners with business and technical professional services, software and hardware, JBoss is well positioned to help enterprises realize their potential with SOA, and the underlying technology – JEMS – is a critical part of the SOA puzzle. JBoss rounds JEMS out with high quality technical support and services with the JBoss Subscription to speed enterprises along to their SOA-enabled business goals.

Effectiveness, consumability, ease-of-development and deployment, and cost are key tenets for a successful SOA platform. Built with the Professional Open Source model resulting in high customer satisfaction, JEMS is the market's ***Only Open Source Platform for SOA***. JEMS provides mass-market economics for an enterprise SOA platform.

The JEMS architecture is built on a compact microkernel that provides base services and a foundation for plug-and-play services. In fact, the JBoss Application Server is a collection of services like EJB, Tomcat, JMS, Web Services, and Security plugged into the microkernel. One of the key benefits is that users and customers can replace and add components on a dynamic basis. Moreover, JBoss AS was the first J2EE 1.4 certified open source application server. J2EE 1.4 standardized the Web Services layer and assured interoperability with XML, JAX-RPC, SOAP, WSDL, and UDDI. With this foundation, customers can build a complete web services architecture.

The Professional Open Source business model is designed to let customers and partners expand to huge deployments. Since there is no per-CPU charge, customers can scale up and out to meet their business needs – whether it is a large central grid or a distributed store network of 3,000 locations.

JBoss does not lock customers into a closed, proprietary, and expensive implementation. All JBoss products are based on industry standards that assure interoperability. Financial services, media and insurance companies are just some of the types of enterprises that are building their SOAs on JEMS. If you are planning or looking to fine tune your SOA initiative, we encourage you to learn about our market leading products (www.jboss.com/products) and see for yourself how they can satisfy your SOA platform needs.