

A large, abstract graphic on the left side of the page, consisting of several overlapping, curved, semi-transparent shapes in shades of gray, creating a sense of depth and movement.

# **USING OPEN-SOURCE SOA IN A GOVERNMENT ENTERPRISE DEPLOYMENT**

Reducing costs, lowering risks, improving security, and increasing agility

White Paper  
May 2009

## Table of Contents

<b>Introduction</b> .....	<b>1</b>
<b>An Open-Source Introduction</b> .....	<b>3</b>
General benefits of open source .....	3
Sun's approach to open source .....	4
<b>Business Drivers for Open-Source SOA</b> .....	<b>9</b>
<b>Modular, Open-Source SOA</b> .....	<b>12</b>
<b>Conclusion</b> .....	<b>15</b>
<b>More Information</b> .....	<b>16</b>

## Chapter 1

# Introduction

Traditional IT infrastructure can be described as “silos of information.” Applications and vital information is distributed amongst disparate systems such as intelligence, sensor, radar, weather and transportation infrastructure data where information is locked inside of one application and virtually locked out of all others which makes it impossible to communicate with each other. Historically, there hasn’t been a standard way to effectively share data amongst enterprise or agency applications short of custom integration efforts, where each change is an expensive and time-consuming endeavor.

Over the past decade many organizations have integrated their enterprise applications themselves, causing IT software infrastructures to evolve into what can be described as “accidental architectures.” Consolidations and changes within agencies have further complicated the landscape, with many organizations hosting multiple deployments of overlapping application software functionality (multiple CRMs, multiple directory services, and so on).

Many agencies are turning to a service-oriented architecture (SOA) approach to use their existing IT assets in new ways. Application functionality is made available as a service, available to all applications that use that particular capability. Managers know that an SOA is a key technology in business process integration, and can help increase agility, improve situational awareness, and increase responsiveness.

An SOA uses standards-based, Web Services APIs. Rather than taking a “rip and replace” approach — eliminating existing software assets such as legacy applications and replacing them with new services — an SOA can extend existing software applications and data repositories by wrapping them with appropriate Web Services interfaces and exposing them as services to build composite applications. The services can be reused to create new applications more quickly and easily than traditional methods.

More importantly, these services create an application that is more aligned with agency goals and requirements. The same data that was reserved for one type of application can instead be used across SOA applications. Business processes are no longer dependent on specific applications. Instead, they are assembled from proven, reliable components that are used in other applications. Services can be upgraded, and the new functionality will be available in all applications that use it. This can help reduce costs and speed deployment and upgrade times.

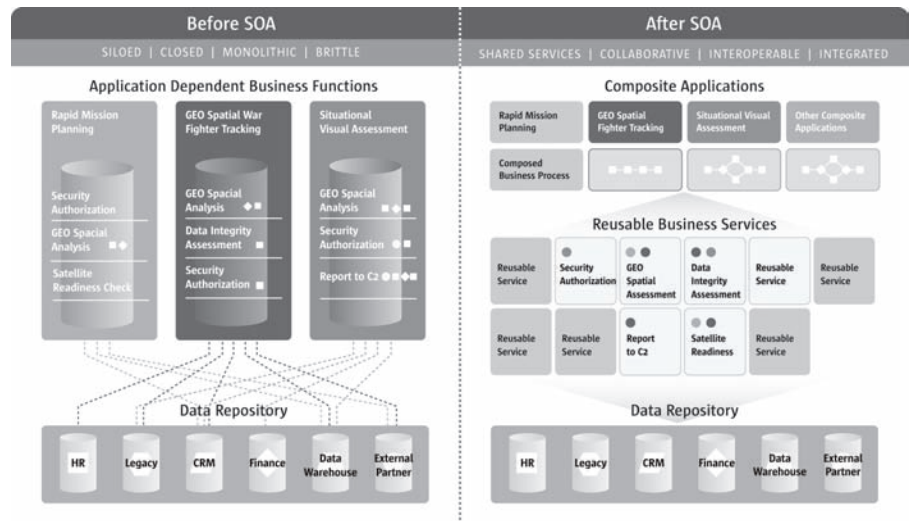


Figure 1. An SOA can help increase enterprise agility and lower costs.

Agencies choose SOA to increase agility, simplify their IT infrastructure, and reduce cost pressures. This can be difficult with solutions that are delivered under a “closed source” model. With proprietary or closed source solutions —

- There is virtually no transparency into the vendor development process. The software upgrade process can be complex and risky
- Vendors completely control the SOA platform. With vendor lock-in, customers and partners are unable to influence platform direction, and prices can be easily increased. This is more likely as the number of enterprise software vendors is reduced.
- Innovation can not easily be incorporated. It is difficult for external developers and customers to include value-added differentiation.

Sun offers a modular, open-source approach to SOA solutions, which provides significant benefits when compared to closed, proprietary platforms. Sun provides the combination of business integration expertise combined with the assurance of enterprise-class support to deliver a modular, open-source based SOA platform that is flexible, lowers risk, and reduces costs.

This paper provides a brief overview of the benefits of open source and the business drivers behind its market momentum. The paper also describes Sun’s modular, open-source approach to an SOA.

## Chapter 2

# An Open-Source Introduction

**Myth:** Few organizations are using open-source applications.

**Fact:** According to industry analyst Gartner, 85 percent of companies are using open-source software, and the remaining 15 percent are expected to do so within the next year.<sup>1</sup>

It is worthwhile to consider the definition of the term open source. A widely recognized definition of open source is posted on the Open Source Initiative's web site:

“Open source is a development method for software that harnesses the power of distributed peer review and transparency of process. The promise of open source is better quality, higher reliability, more flexibility, lower cost, and an end to predatory vendor lock-in.”<sup>2</sup>

Communities are areas where groups of individuals or agencies can collaborate in the development of the source code in an open and transparent way. Communities also include those who support and use open-source software. There are many reasons why open-source software has been widely adopted in enterprise environments and large organizations.

## General benefits of open source

There are a number of advantages that open-source code offers over proprietary (closed source) solutions.

- **Transparent access to code:** Community-developed software enables everyone to see what is being built and what features will be included as early as possible. Insight into proprietary software is usually very late in the life cycle, and typically no sooner than beta test. Developers and customers do not need to wait for a vendor to publish a roadmap or wait for a product launch to know what is being developed, enabling them to make better and faster decisions relating to their software infrastructure.
- **Lowers barriers to adoption:** Open-source software offers users the right to use the software in any way they wish. Proprietary and closed-source development, with its high license fees, discourages developers and customers from adopting a product or technology. Open-source development removes these barriers by encouraging participation and allowing anyone to get the source and try it out. Most open-source communities provide binaries on a periodic basis so users can easily try it.

1. <http://news.zdnet.co.uk/software/0,100000121,39554840,00.htm>

2. [www.opensource.org/front](http://www.opensource.org/front)

- **Prevents vendor lock-in:** Proprietary interfaces and components can help vendors lock-in customers. The source code for open-source software is freely available and supported by many in the community, and uses standardized interfaces. If an open-source vendor goes out of business or tries to increase their fees for the commercial product, other vendors can and will step in to meet the needs of the customers at a competitive market price.
- **Opportunity for early feedback and collaboration:** Freely available access to source and binary code not only allows developers and customers to try it out, but also enables users to provide feedback during development. This provides a much tighter feedback loop than closed-source software, improving overall quality and functionality. Each open-source project benefits from a large community of users who download the software, try it out, and provide feedback and suggest changes. Rather than depending on a single in-house team, open-source software design is the product of many great minds collaborating across the globe, which improves the functionality of the product and accelerates the development time. Proprietary software, by contrast, allows customer feedback late in the development cycle — beta testing is typically only for fixing high priority bugs, not feature feedback.
- **Lowers the cost of talent:** Community development, adherence to standards, and lower barriers to adoption help increase the number of developers who become aware of, and proficient in, the use of the product or technology — reducing upward pricing pressure.
- **More secure and stable software:** Open source is inherently more secure than proprietary solutions because everyone has access to the code and nothing is hidden. Bugs and vulnerabilities are found more quickly and can be addressed efficiently by the entire community, without dependency on a proprietary vendors release cycle. This means that bugs or security holes can be observed, found, and fixed by all. It is through peer review and use that the resulting product is both more secure, tested, and stable than many closed-source products. A good way to look at it is if the ‘Trojan Horse’ would have been glass, it never would have gotten into the castle.
- **Pay at the point of value:** Non-open-source products often require that they be purchased before they can be used or evaluated. This means that the software must be purchased before the customer sees value in the product. Open-source software allows for evaluation and initial proof of concept development. There are no restrictions on when open-source software can be used. Only when a customer sees value — such as for support, integration, or additional functionality — is payment required.
- **Better quality/value:** Because customers are only required to ‘pay at the point of value,’ customers are not stuck with an inferior product, just because they have already paid for it. Hence, the overall quality is better because they make choices based on functionality rather than obligation.
- **Lowers risk:** Open-source software lowers risk for a customer by providing early visibility into roadmaps and access to a larger pool of talent at a low cost, preventing vendor lock-in, and paying at the point of value.

“Success in open source requires you to serve:

1. Those who spend time to save money
2. Those who spend money to save time.”

**Marten Mickos**<sup>4</sup>  
CEO, MySQL

## Sun’s approach to open source

In 2005 Sun came to the conclusion that open source was the future of the software industry. Today, Sun has released all of its core products under open-source licenses and is involved in over 750 open-source communities. Sun is also the largest contributor of code to the open-source movement.

Sun’s philosophy of free and open-source licensing is not to necessarily favor any one license or category of license but to make decisions that match the project or community with the license that best supports it.<sup>3</sup> Sun believes open source is about creating a thriving, active, and innovative market. The ecosystem includes many different actors, including those providing development, support, documentation, consumers, and more. Each of these market segments has its own needs and capabilities.

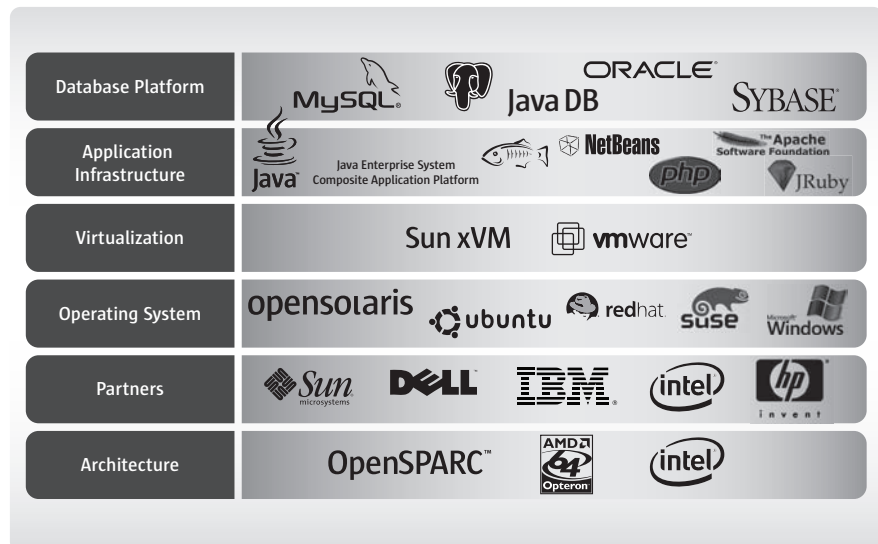


Figure 2. Sun offers a robust and flexible set of open source products that is supported by leading solution providers.

3. <http://news.zdnet.co.uk/software/0,1000000121,39407332-1,00.htm>

4. [se.sun.com/sunnews/events/sunexpo/pdf/Sun\\_Software.pdf](http://se.sun.com/sunnews/events/sunexpo/pdf/Sun_Software.pdf)

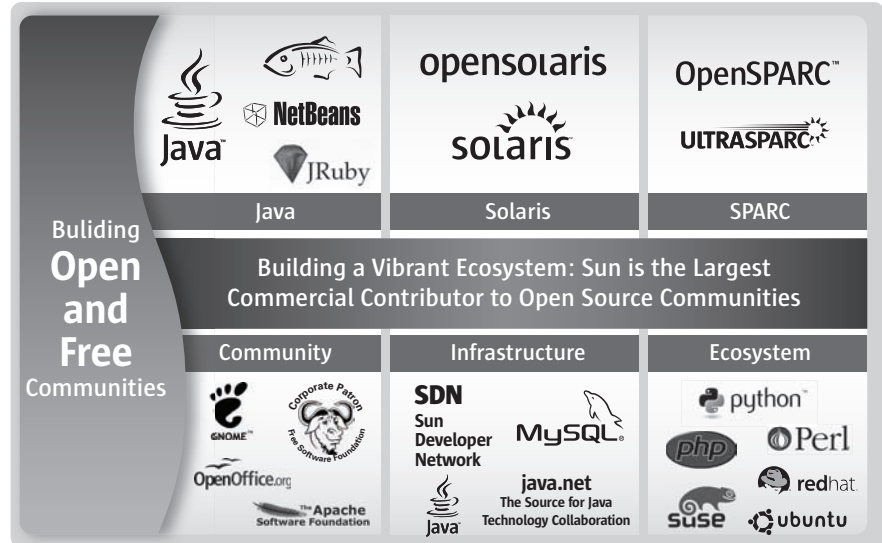


Figure 3. Sun is the largest commercial contributor to open source communities.

Overall, Sun's goal is to make our technology source code as open as possible to the largest developer audience, to lower the barriers to entry, and to seed the market. We're continuing to evolve our licensing models to meet this goal. Sun believes that the variations in different licensing models are based upon stewardship: who will take ownership and responsibility for evolving the technology in the appropriate manner.

Open source development provides a collaborative way for all organizations, both large and small, to contribute and invest in open source development. By delivering on top of open source software, government agencies are helping to encourage genuinely open innovation; open source is available to everyone.

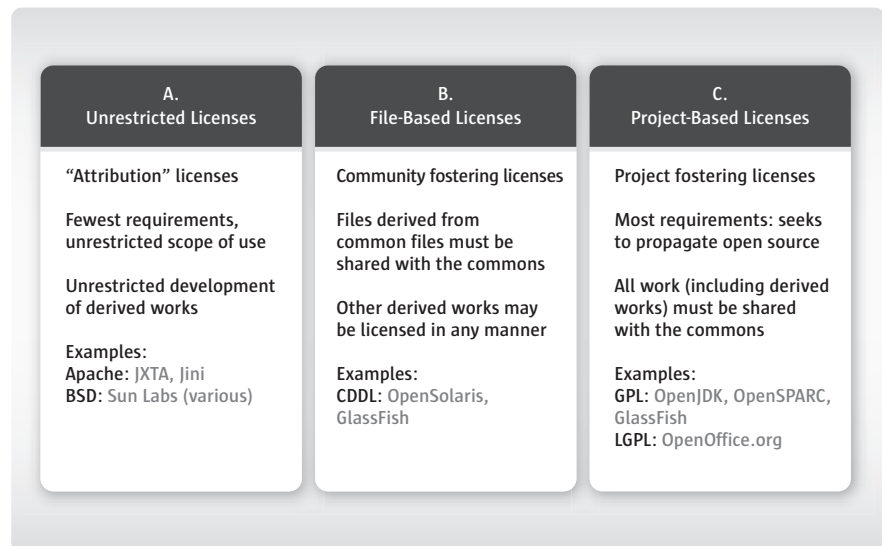


Figure 4. Open source licensing can be divided into three categories.

Sun has come up with three overall categories to contain these many different licenses. An understanding of these licensing categories provides a foundation for understanding Sun's approach to making decisions about open source licensing for its products.

- Unrestrictive licenses generally only require that you give attribution to the source, and places no requirements on what you do with the code. Apache licenses are an example of unrestricted licenses.
- File-based licenses are designed to enrich the source code commons by requiring that any source modifications must be returned to the commons, but other files may be licensed according to the author's wishes. Source code under this type of license require that any such source file that contains code from the commons must be licensed with the same license as the commons. They do not, however, impose this requirement on files that do not contain code from the original commons — those files can be licensed in any way the developer wishes. CDDL is an example of file-based licenses.
- Project-based licenses seek to preserve freedom and increase open source software by requiring that your product, should it include this software, must also be licensed under the same terms. This type of license is sometimes known as "viral." Project-based licenses require that any file, regardless of code origin, which is combined under certain circumstances with the commons file must be licensed under the same license as the commons. The effect of this is to make as much free software as possible available publicly, and in a practical way, through the commons. GPL is an example of a project-based license.

File-based licenses are considered *copyleft*, and project-based licenses are considered strong copyleft licenses. Unrestrictive licenses are not considered copyleft. The principle of copyleft requires derivative works to be licensed under the GPL. Linux is probably the best-known GPL-licensed software. (In general, copyright law allows an author to prohibit others from reproducing, adapting, or distributing copies of the author's work. Copyleft may be characterized as a copyright licensing scheme in which an author surrenders some but not all rights under copyright law.<sup>5</sup>)

For more information, please refer to Sun's white paper on open source licenses, at [http://www.sun.com/software/opensource/whitepapers/Sun\\_Microsystems\\_OpenSource\\_Licensing.pdf](http://www.sun.com/software/opensource/whitepapers/Sun_Microsystems_OpenSource_Licensing.pdf)

5. <http://en.wikipedia.org/wiki/Copyleft>

### Community Perspectives

“I think Sun...has contributed more than any other company to the free software community in the form of software. It shows leadership. It's an example.I hope others will follow.”

#### Richard Stallman

Free Software Foundation<sup>6</sup>

### Federal Health Architecture (FHA) on open source:

“The NHIN-CONNECT software solution, developed by ONC and FHA, enables federal agencies to tie into the Nationwide Health Information Network and was intentionally built on open-source technologies. The open-source, unified software provided to the CONNECT team satisfied a need for functionality and scalability and helped deliver a required solution for the federal agencies to advance exchange of interoperable health information.”

It is important to note that open source does not necessarily mean free. Vendors may opt for an open-source process in product development but still offer products under a commercial license. This process may be suitable for those customers who don't want to adhere to the open-source license, or are interested in the benefits a vendor can provide. For example, a commercial license can include indemnification, packaging, testing, support, patches, updates, integration with other value added functionality, and other capabilities that are seen as valuable by the community.

Sun is one of the only major SOA vendors to develop production software in open-source communities. Sun has contributed billions of dollars, as well as more code, to free software than any other organization in the public or private sector.<sup>7</sup> Sun remains committed to open-source communities to develop quality software and leverage the technological innovation happening in the quickly evolving open-source realm. Sun's commitment to open-source software helps bring greater value to developers, consumers, and the open-source ecosystem to help drive IT innovation and value.

An excellent example of the success of the 'open source' model is the Nationwide Health Information Network (NHIN) being developed by Health and Human Services (HHS)/Office of the National Coordinator (ONC). The ONC decided to use 'open source' technology to develop a reference implementation for the nations Health Information Exchange (HIE) infrastructure. While establishing standards, policies and governance is key to the success of NHIN, the open source technology use, is also key.

The ONC decided to use Sun's open source technology for the following reasons:

- **Cross Platform Compatibility.** Compatibility in a heterogeneous environment is critical to the success of any project in today's environment. Sun is committed to continue its cross-platform support of its open source software.
- **Open Source and Open Standards.** The NHIN infrastructure was designed to withstand the test of time and with 'future proofing' in mind. It was built to endure, irrespective of corporate or economic changes.
- **Commercially Supported Open Source Code.** Sun contributes extensively to the open-source community, but has a complete engineering staff to provide commercially deployable, tested and certified products to its customers.
- **Major Health Information Technology Provider.** Sun understands the challenges our customers are facing and their business processes. Sun has been applying its technology to address these challenges for many years.
- **Technology Partner, not just another Vendor.** Sun is committed to its customers and will provide the technical support required to ensure their success.

6. <http://channelsun.sun.com/video/open-source/java/1631259654/richard+stallman+3/1817711041>

7. Press release December 5, 2007 <http://www.bizjournals.com/eastbay/stories/2007/12/03/daily25.html>

## Chapter 3

# Business Drivers for Open-Source SOA

Over the past decade many organizations have integrated their enterprise applications into accidental architectures, laboriously connecting different application components, data repositories, and creating ad hoc business logic. As the number of point-to-point instances increase, the incompatibility and inconsistency between the technologies also increase, resulting in reduced responsiveness to on-going business requests and lower quality due to the increased complexity.

Many organizations and agencies are turning to an SOA approach to automate, optimize, and integrate business processes, and use their existing IT assets in new ways. An SOA provides a framework for taking information already contained in the organization's IT resources to provide a more insightful, comprehensive view of information. In keeping with the Federal Healthcare example, one of the agencies responsible for promoting healthcare may require information from multiple sources to determine medical claims payment or disability eligibility.

Most of the leading SOA providers use proprietary code. Sun is the only major technology provider currently developing production SOA software in open-source communities. Only Sun provides the combination of open-source leadership combined with the assurance of worldwide, enterprise-class support.

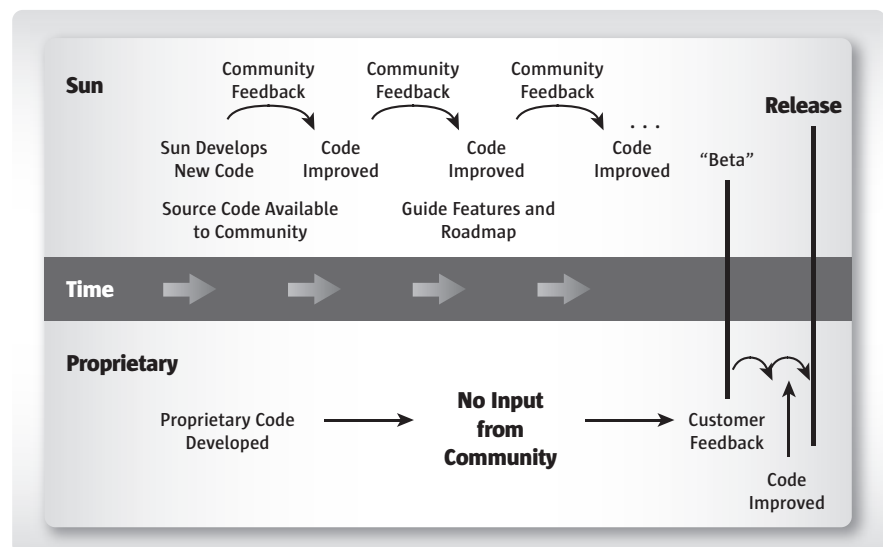


Figure 5. Transparency and continuous community feedback help lower risk.

Sun SOA solutions can help companies, organization, institutions, and government agencies reduce costs, lower risks, and increase flexibility. Sun's modular, open-source SOA platform:

- **Reduces costs by —**
  - Reducing the complexity when developing new services and leveraging existing functionality in legacy applications for new SOA-based applications. Sun's SOA offering is a complete and integrated suite that provides a common development, management, and administration environment.
  - Offering a subscription pricing models that eliminates the burden of traditional pricing schemes. Open-source software companies typically earn revenue through support subscriptions renewed on an annual basis, giving the vendor an incentive to provide high-quality support each year and ensure that customers continue to be successful with their software. Because access to new versions is included in the subscription fee, costs are predictable and consistent compared with the up-front purchases that characterize the traditional proprietary software model.
  - Freeing users from high software license fees. With free downloads available, open-source software eliminates costly software licenses and the cumbersome processes associated with a software purchase. The result is higher return on investment in application development.
- **Lowers risk by —**
  - Providing transparency into the solution architecture, enhancing stability, increasing security, and providing insight into future functionality.
  - Enabling a comprehensive trial before purchasing. Open-source software binaries are freely available for download, which allows users to try it, provide feedback, and suggest changes before making the decision to purchase the software. This can significantly reduce the risk and buyer's remorse.
  - Ensuring production-level quality and stability. A robust testing process reduces validation time for IT organizations, and minimizes downtime and troubleshooting.
  - Indemnifying organizations and agencies against potential intellectual property infringement related to product code.
  - Providing the technology and integration expertise required to build an application environment that can be counted on. Sun continues to extend its leadership by building on more than 18 years of expertise with SOA, enterprise service bus (ESB), and business integration platforms and technologies.
  - Building an open-source based SOA solution prevents vendor lock-in and provides a flexible, standards-based software foundation for the future.

- **Increases flexibility by —**
  - Using a “pluggable” architecture, based on open-source and modular standards. This offers IT managers more choice as other open-source or standards-based functionality can be easily added.
  - Sun’s open-source SOA environment reduces time-to-market, enabling IT departments to deliver new applications and features faster than some other vendor solutions.

Strong, vital communities of open-source projects can offer more of the benefits of an SOA when compared those closed-source SOA vendors. Companies and large organizations can implement features and capabilities at their own pace, paying at the point of value, with reduced risk and increased security.

Sun is a strong, committed proponent of open-source development and software, contributing more source code and support to open-source communities than any other organization. Key open-source projects and communities include OpenSolaris™<sup>8</sup>, OpenOffice.org™<sup>9</sup>, OpenJDK™<sup>10</sup>, NetBeans™<sup>11</sup>, GlassFish™<sup>12</sup>, OpenSSO<sup>13</sup>, OpenDS™<sup>14</sup>, OpenESB<sup>15</sup>, and Mural<sup>16</sup>. Several of these are core or foundational to Sun’s SOA products.

Sun adds value to its open-source SOA offering in several ways to deliver a solution that meets enterprise and business-critical requirements, such as installation, use, and management experience. Only after the products meets well-defined and rigorous criteria is it released, and Sun will support it as a standard commercial product with support offerings including 24x7 support, patches, updates, indemnification, consulting, and more.

### Sun’s Unique Business Model

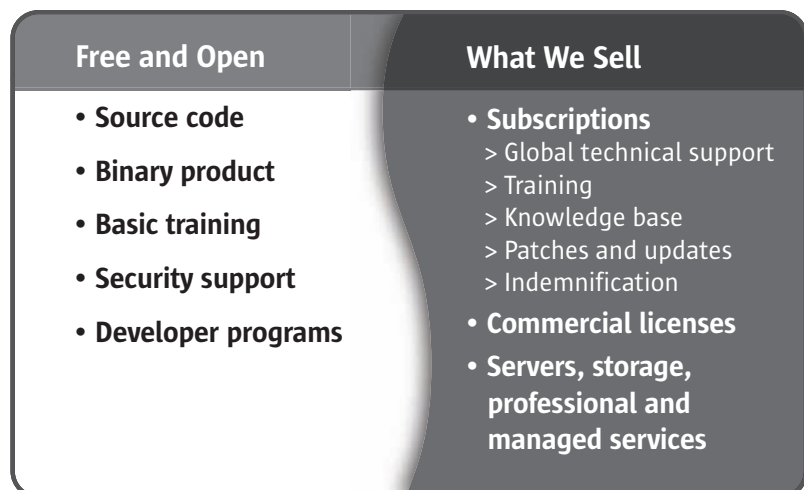


Figure 6. Sun provides enterprise value for open-source software.

8. <http://opensolaris.org>  
 9. <http://openoffice.org>  
 10. <http://openjdk.dev.java.net>

11. <http://netbeans.org>  
 12. <http://glassfish.dev.java.net>  
 13. <http://opensso.dev.java.net>

14. <http://opends.dev.java.net>  
 15. <http://openesb.dev.java.net>  
 16. <http://mural.dev.java.net>

## Chapter 4

# Modular, Open-Source SOA

Sun continues to extend leadership by building on more than 18 years of Sun expertise within SOA, ESB, and integration platforms. With over 2,000 customers, Sun is a recognized leader in providing a comprehensive SOA solution with open-source core components. The cornerstone product offering is the Sun Java™ Composite Application Platform Suite (Java CAPS). Java CAPS is the most unified business integration platform based on open-source and modular standards-based framework. A pluggable architecture provides IT managers with choice and flexibility. Sun's SOA solutions are built with open source and based on standard APIs, so there is no vendor lock-in. As a leader in open source, Sun offers developers the benefit of transparency, early access, and ability to influence and contribute to future Java CAPS enhancements through the OpenESB, Project Mural, and other open-source communities.

Key advantages of Java CAPS include:

- Open-source, pluggable architecture helps lower risk and eliminate vendor lock-in.
- Unified design and implementation — the product was built from the ground-up as a unified suite so the product integration is already done. The result is shorter development time, faster application deployment and lower TCO.
- Unified and comprehensive
- Standards leadership
- Proven business integration specialists

Java CAPS is based on a component architecture that allows rapid integration of standards-compliant components into the core OpenESB that enables rapid plug-in of partner components. Sun works with the community to develop Java CAPS innovations in OpenESB, an open-source community. As the functionality matures, it is incorporated into the final Java CAPS product. Customers gain access to the innovation of the OpenESB community — along with the assurance of proven quality standards and world-class support from Sun. Open-source software also gives Java CAPS customers transparency into the underlying source code, virtually eliminating vendor lock-in.

One of the available options for utilizing external components within Java CAPS is Java Business Integration (JBI), one of the only open and pluggable standards for building an SOA available today. Many SOA solutions rely solely on traditional adapter technology to enable application integration. Java CAPS supports JBI and a wide variety of standards to offer more choice and flexibility by allowing developers to more easily interact with standards- and non-standards-based applications and data stores.

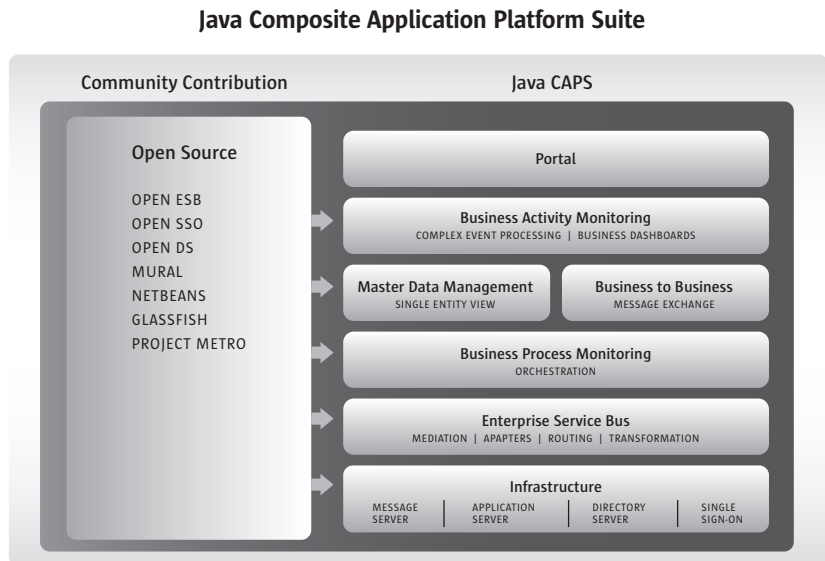


Figure 7. Java CAPS is a comprehensive modular, open-source SOA implementation.

This open, modular approach can lower TCO by implementing only the necessary components to meet specific project needs. Additional standards-compliant components can also be added. Java CAPS also offers the ability to orchestrate and interact with Web and non-Web services seamlessly across the enterprise by leveraging enhanced interoperability with Java and Microsoft .NET platforms.

Open-source components in Sun's SOA include the following:

- **OpenESB:** A Java technology-based open-source enterprise service bus, which can be used as a platform for both enterprise application integration and SOA. OpenESB hosts a set of pluggable components, which integrate various types of IT assets. The runtime can be collocated with the GlassFish server for support of J2EE components. There is also support for other application servers.
- **GlassFish ESB:** A Sun-supported platform based on Project OpenESB. This light-weight and agile ESB platform packages the innovation comprising Project OpenESB, the GlassFish application server, and the NetBeans IDE into a commercially supported, enterprise-class platform. GlassFish ESB is ideally suited for projects and departmental level implementation, but has the ability to scale as the business evolve over a period of time.

- **OpenSSO:** Provides core identity services, including access management and federation services, that can help simplify the implementation of transparent single sign-on (SSO) as a security component in a network infrastructure.
- **OpenDS:** A comprehensive next generation directory service based on LDAP and DSML standards. OpenDS is designed to address large deployments, provide high performance and extensibility, and to be easy to deploy, manage and monitor.
- **Project Mural:** Provides tools for management facilities that enable organizations to create and deploy master data management solutions, resulting in high levels of master data quality and reliability.
- **NetBeans:** A free, open-source Integrated Development Environment for software developers.
- **Project GlassFish:** GlassFish is the name for the open-source development project for building a reference J2EE application server. It is the foundation for the Sun GlassFish Portfolio, the most complete open-source Web application platform, offering enterprise-class features and support, and best-in-class performance.
- **Project Metro:** Metro is a high-performance, extensible, easy-to-use web service stack. Metro provides comprehensive, reliable, secure, and transacted Web services, including .NET services. The Metro Web Service stack is a part of the GlassFish community, but it can be also used outside GlassFish.

## Chapter 5

# Conclusion

Sun is a strong leader in the open-source model. As a leading provider of service-oriented architectures, Sun provides an affordable, secure, low-risk, and flexible platform. Unlike other business integration technology providers, Sun develops software in the open-source community, enabling developers, integrators, and end-users to participate. Extensive integration capabilities, including Web Services interfaces, OpenESB plug-ins, and connectors, means government organizations and agencies can get and reuse the functionality they need from a diverse ecosystem of vendors. For mission-critical product deployments, Sun adds value such as enterprise quality assurance, support, and indemnification. Sun's open-source SOA products are transparent and accessible throughout the life cycle, helping to lower costs, improve security, reduce risk, and enhance agility.

Sun continues to extend leadership by building on more than 18 years of Sun expertise within SOA, ESB, and integration platforms. Sun has over 2,000 customers from a variety of industries utilizing Java CAPS. Java CAPS is the enterprise integration standard for some of the largest, most complex organizations in the world. With Sun's extensive partner and services network, customers benefit from extensive SOA implementation experience, broad global resources and industry leading SOA delivery methodologies, to reduce the cost and complexity of SOA.

## Chapter 6

# More Information

Additional information on Sun's open-source SOA is available:

[sun.com/javacaps/](http://sun.com/javacaps/) and [sun.com/soa/](http://sun.com/soa/)

Sun GlassFish ESB: [sun.com/glassfishesb](http://sun.com/glassfishesb)

Sun Open Source Resources: [sun.com/opensource/](http://sun.com/opensource/)

**Sun Microsystems, Inc.** 4150 Network Circle, Santa Clara, CA 95054 USA **Phone** 1-650-960-1300 or 1-800-555-9SUN (9786) **Web** [sun.com](http://sun.com)



© 2009 Sun Microsystems, Inc. All rights reserved. Sun, Sun Microsystems, the Sun logo, OpenSolaris, OpenOffice.org, OpenJDK, Netbeans, GlassFish, OpenDS, and Java are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the United States and other countries. Information subject to change without notice. SunWIN #564002 05/09