THE CLIPPER GROUP
Navigator™
Navigating Information Technology Horizons℠
Published Since 1993 Report #TCG2008027 May 28, 2008

Build a *Super-SOA* with SOLA

Analyst: Anne MacFarland

Management Summary

Superman seemed immune to the inefficiencies that plague the rest of us with less-empowered lifestyles. It is easy to think about *the plagues of inefficiencies* when thinking about IT's *service-oriented architecture (SOA)*. SOA is the logical way to both build and evolve the software that supports business operations. These business operations are typified by constant shifts and extensions needed to meet both the nature of the competition and the changing needs of customers. Their nature is well matched by a software architecture characterized by loosely-coupled components and arbitrated run-times run on hardware that is similarly flexible – either by its design or made so by virtualization. However, these architectural characteristics bring, with their flexibility, some inherent performance and security challenges. In some situations, these challenges are mitigated adequately by today's processing capacities and density and by adding vulnerability and penetration testing disciplines to the traditional IT security arsenal of authentication, access control, identity management, firewalls, and encryption.

In situations with high volumes of transactions involving sensitive information, such as the financial services industry, the nature of operations requires more. The high rate of transactions, and the extent to which they propagate changes in related systems, means that performance needs are greater than the norm. It is people's money that is being manipulated, not some inventory of pet accessories, so security mandates and regulatory requirements are more stringent. There is vast proliferation of complex financial service offerings, most involving modeling of alternatives. Such a situation demands, in effect, *a SOA with some super powers*.

Many enterprises in financial service and other transaction-intensive industries still use Mainframes for their back-end systems. By leveraging the capabilities of a Mainframe, and particularly those of the newer *IBM System z9* or *z10*, SOA Software's *SOLA* offering supports a SOA that is inherently secure and high performing because of where it runs – and how it runs. SOLA becomes a point of policy enforcement that is itself inherently secure. The multi-tenant architecture and high degree of resource and process control, built into System z, deals with many of the subtle challenges of managing an environment designed for flexibility.

SOA Software, based in Los Angeles, CA, has focused on Web Services and SOA since it bought Flamenco Networks in 2004. It bought the basics of what would become its Mainframe SOA product, SOLA, from Merrill Lynch in 2005, and focused on building the software to support a SOA for large volumes of transactions that run completely on the IBM System z Mainframe.

Like Superman, SOLA relies on some special capabilities: the capabilities of the Mainframe, the efficiency of SOLA operations, and the completeness of the solution (e.g., Superman without his *x-ray vision* would not be as effective.) Like any SOA, it can be addressed as an element in a larger environment and administered through a browser interface. For more details, please read on.

IN THIS ISSUE

SOA Operations Elements

SOAP/XML and Other Industry Standards

SOA Software[1] has done all the mapping to incorporate necessary standards[2] into its Mainframe SOA operations without requiring the re-writing of any of the participating services or applications that support them. *SOLA* accomplishes this translation using a heuristic dictionary that can tokenize *COBOL* data structures and map them into *WSDL*[3]. The dictionary can also translate Mainframe asset names into names other systems will recognize. SOLA also can parse a COMMAREA interface to ascertain the structure of a message. SOLA then produces, in WSDL, the runtime metadata needed – the load-only module, the schema of the SOAP request, and the rules for transformation. An administrator can test artifacts without running code, and then migrate them to production using SOLA's change management capabilities.

The Registry

A Registry and/or a broker (such as an *Enterprise Service Bus*) are what differentiate a services-oriented architecture from the litany of "modern" but static architectures that preceded it. SOLA supports either an ESB or a Mainframe-optimized UDDI registry – or both. What a business will need depends on the nature of its service components and the complexity of connections it must support to get business done.

The Mainframe, by its nature, supports both safe arbitration of multi-tenant environments and a variety of containers, including partitions and virtual machines.[4] Decades of closely-aligned development of *z/VM* and the *IBM System z* (Mainframe) processor means that z/VM images do not carry the overhead of newer VMs. Resource control is in hardware, not software. Therefore, the resource arbitrations of SOA do not have to be an explicit part of SOA choreography. The logging and operational rules that are inherent in z/OS both assist in the BPEL-based orchestration of services and assure that SOA operations are fully controlled and documented. Without this granularity of documentation, SOA forensics are hampered by many unknowns and require a lot of work that usually translates into people costs.

Other Operational Elements

Security is, of course, key. The folks at SOA Software have developed the bridges to bring Mainframe operations into WS-Security space. SOLA also leverages Mainframe capabilities for encryption, digital signature administration, and PKI, including synchronous and asynchronous encryption keys. SOLA security runs of System z's cryptographic co-processor, keeping operations at in-line speeds. SOLA includes identity mapping – a must for SOA operations. It addresses both authentication and authorization, via *SAML (Security Assertion Markup Language)* and *XACML (eXtensible Access Control Markup Language)*, which adds another layer of civilization.

SOLA on the Mainframe

Unlike some other "Mainframe SOA" approaches[5], SOLA does not leverage the lower-cost environments of System z's *ZIIP*[6] and *ZAAP*[7] specialty engines – because to do so would increase costs. This is because SOLA does not use Java for its software. It uses assembler language. Assembler language, unlike JAVA and many programming languages, is optimized for minimal use of cycle time.[8] The offload process to specialty engines incurs an overhead of about 5%. Assembler language is so efficient that the overhead of offload would add more latency and cost than running the code itself produces (even in a z/OS environment). (See Exhibit 1, on the next page.)

SOLA has the management and metrics to process *CICS* services at high volumes. The SOAP support of the native CICS pipeline introduced in CICS 3.1 is fine for prototyping, says Jim

[1] SOA Software also has a complete range of software products for non-Mainframe SOAs. They are not described in this bulletin for reasons of space and distraction. They, too, spring from the same philosophy of completeness as SOLA.

[2] SOLA leverages SOAP, XML, WSDL, BPEL, WS-Policy, WS-Security and uses a W3C-compliant parser.

[3] Web Services Description Language.

[4] For more information, see **The Clipper Group Navigator** entitled *Oh The Things You Can Do…With z/VM 5.3!*, dated February 28, 2007, and available at http://www.clipper.com/research/TCG2007030.pdf.

[5] For more information, see **The Clipper Group Navigator** entitled *For a More Secure SOA at Less Cost, Use System z9 and Data Direct*, dated December 3, 2007, and available at http://www.clipper.com/research/TCG2007012.pdf.

[6] For more information, see **The Clipper Group Navigator** entitled *z9 Adds zIIP to Ally with DB2 on z/OS to Better Serve the Onslaught of Business Data*, dated January 24, 2006, and available at http://www.clipper.com/research/TCG2006006.pdf.

[7] For more information, see **The Clipper Group Navigator** entitled *zSeries Sips through Java with zAAP*, dated April 7, 2004, and available at http://www.clipper.com/reserach/TCG2004030.pdf.

[8] By the estimate of Jim Crew of SOA Software, assembler code runs 20 times more efficiently than Java.

Exhibit 1 —
SOLA Parser Performance

- The SOLA assembler parser takes 0.0001 CPU seconds to parse 1000 bytes (a typical XML message)
- At a typical chargeback rate of $0.20/CPU second; this would be a cost of $20 to parse a million transactions.
- Parsing, as a problem, is similar to compiling in that it involves a lot of string leader comparisons. By writing a parser in assembler language, optimizing for speed and low memory usage, SOA Software can reduce greatly the cost of parsing and support a high rate of transactions.

Source: SOA Software

Crew of SOA Software, but not for high-volume production environments with security and audit requirements.

Super SOA Operational Requirements

Closed-Loop Governance

SOA Software's approach to SOA operations supports governance integrated across design and runtime both in their open platforms products as well as in SOLA. This closed loop eliminates sources of introduced inconsistency-induced risk.

Scalability and Adaptability

It is easy to implement a small project, but hard to scale to support high volumes of use and still harder to support a high rate of change at the same time. SOLA and System z's flexible containers offer the benefits of safe and rapid deployment of services in a number of scenarios.

Infrastructure Rationalization

With SOLA, it is possible to have an implementation *with no middle tier of servers*. This expensive hardware tier often includes security and XML appliances, as well as application servers and software.

Human Assets and Their Development Environment

The best SOLA service developer is a Mainframe developer who knows the programs from which services are to be derived. In SOA Software's experience, Mainframe developers have appreciated a development environment purpose-built for Mainframe-based service creation and publishing. The testing tools do not require *Java* or *.Net* skills.

SOLA in Your Data Center

The typical SOLA bundle includes the SOLA Development Studio and four SOLA instances – one for Development, one for QA and two for production. All bundles include:

- SOLA UDDI
- SOLA parser
- SOLA dictionary
- BPEL engine
- WS-Security engine
- WS-Policy engine
- SAML engine
- XACML engine
- Identity mapping engine
- Batch support

SOLA's ROI comes in both cost avoidance and direct savings. The cost avoidance comes in the infrastructure that does not have to be built to support a SOA at scale, and in the speed of deployment of scalable SOA services. The direct savings come from a lower cost per transaction.

SOLA services can be run under the control of *RACF*, *ACF2*, or *Top-Secret*. They can be managed by the customer's release management system. SOLA's *Identity Management* offers customers a way to extend single sign-on across their entire environment, including the Mainframe.

SOLA can also integrate into a SOA ecosystem, by integration into a corporate UDDI, and by virtue of its support for WS-Policy, WS-Security, and its BPEL engine. SOLA offers top-down, bottom-up and meet-in-the-middle development approaches to match a customer's preference. SOA Software is working on integration with the Tivoli Systems Management toolset.

SOLA, itself, can be presented as services. *The Development Studio* is shipped as a set of services. Functions, such as User Registration and Service Promotion, are published as services and can be invoked from other platforms.

Conclusion

There are few commercial enterprises whose heartbeat is not measured, somehow, in financial transactions. For business processes where time is money and money is the focus, SOA Software's SOLA offers a combination of performance, security and flexibility that is hard to beat.

SOA Software's Mainframe-based SOLA offers a SOA built lean from the inside. It does not indulge in excesses of middleware or inefficient coding – quite the opposite! For those with a Mainframe, and a need for a superbly-performing SOA for high volumes of transactions, SOLA should be of interest.

About The Clipper Group, Inc.

The Clipper Group, Inc., is an independent consulting firm specializing in acquisition decisions and strategic advice regarding complex, enterprise-class information technologies. Our team of industry professionals averages more than 25 years of real-world experience. A team of staff consultants augments our capabilities, with significant experience across a broad spectrum of applications and environments.

- ***The Clipper Group can be reached at 781-235-0085 and found on the web at www.clipper.com.***

About the Author

Anne MacFarland **is Director of Data Strategies and Information Solutions for The Clipper Group.** Ms. MacFarland specializes in strategic business solutions offered by enterprise systems, software, and storage vendors, in trends in enterprise systems and networks, and in explaining these trends and the underlying technologies in simple business terms. She joined The Clipper Group after a long career in library systems, business archives, consulting, research, and freelance writing. Ms. MacFarland earned a Bachelor of Arts degree from Cornell University, where she was a College Scholar, and a Masters of Library Science from Southern Connecticut State University.

- ***Reach Anne MacFarland via e-mail at Anne.MacFarland@clipper.com or at 781-235-0085 Ext. 128. (Please dial "128" when you hear the automated attendant.)***

Regarding Trademarks and Service Marks

The Clipper Group Navigator, **The Clipper Group Explorer**, **The Clipper Group Observer**, **The Clipper Group** *Captain's Log*, Clipper Notes, and *"clipper.com"* are trademarks of The Clipper Group, Inc., and the clipper ship drawings, *"Navigating Information Technology Horizons"*, and *"teraproductivity"* are service marks of The Clipper Group, Inc. The Clipper Group, Inc., reserves all rights regarding its trademarks and service marks. All other trademarks, etc., belong to their respective owners.

Disclosure

Officers and/or employees of The Clipper Group may own as individuals, directly or indirectly, shares in one or more companies discussed in this bulletin. Company policy prohibits any officer or employee from holding more than one percent of the outstanding shares of any company covered by The Clipper Group. The Clipper Group, Inc., has no such equity holdings.

Regarding the Information in this Issue

The Clipper Group believes the information included in this report to be accurate. Data has been received from a variety of sources, which we believe to be reliable, including manufacturers, distributors, or users of the products discussed herein. The Clipper Group, Inc., cannot be held responsible for any consequential damages resulting from the application of information or opinions contained in this report.