

CLOUD COMPUTING AND PUBLIC SAFETY SERVICES

This paper examines the emergence of Service Oriented Architectures, their implementation in Cloud computing services and the likely impact on Public Safety systems and applications.

Dr. Toufic Boubez, SOA Craftworks



Introduction

Information Technology is in the process of a fundamental architectural change that is having a profound effect on systems design and planning. Over the past decade, a new model of architecting systems called Service Oriented Architecture (SOA) has arisen to provide an advanced alternative to traditional Client/Server systems. SOA is a critical element in a series of new software technologies that are enabling a move to so-called Cloud computing, in which premise-based computing systems are being supplanted by services that are remotely located in large, aggregated data centers. This trend is accelerating because of the flexibility of these services, their ease of implementation, their high performance, high reliability and dramatic economies of scale. This paper examines SOA and Cloud computing as it relates to the unique requirements of Public Safety applications and outlines critical success factors that public agencies need to consider in evaluating Cloud services.

Public Safety IT Challenges and Requirements

Traditionally, public safety has been focused on the rapid and accurate delivery of Law Enforcement, Fire and Emergency Medical responses to local incidents via emergency 9-1-1 systems. These local Public Safety Answering Points (PSAPs) have been the backbone of a national system that has grown, in the last four decades, into a system that manages the incoming phone calls, determines the nature of the emergencies, directs the response to the emergencies, correctly deploys the first responders to the emergency site and records the resolution of the incident for later analysis and adjudication.

In any comprehensive public safety system, these five separate functions need to be integrated and coordinated to successfully resolve incidents. The systems supporting these functions are very complex and comprehensive in their own right, and Public Safety applications demand systems that can meet the rigorous performance, reliability, usability and flexibility standards that incident management requires.

For the past 20 years the dominant computing and communications architectures have been based on a model called Client/Server, which was developed to take advantage of LAN-based data communications technologies in the Enterprise. As it evolved, Client/Server also grew to greatly expand the role of the "middleware" systems connecting the Client to the Server. Middleware, in the Client/Server model is a layer that synchronizes operations between Client applications and Server computers and databases. The result is a robust, well-proven architecture for successfully managing complex applications in a widely distributed environment with the assurance of maximum availability and reliability.

The drawback of Client/Server architectures, though, is that implementing and scaling these systems can be difficult and expensive. Middleware systems operate in a meticulously synchronized environment, which becomes increasingly complex and inflexible as the system grows. Each of the Client systems is polled by the Server databases in an effort to maintain a consistent response time and traffic balance in the network. As these systems grow larger, response time and performance become larger issues in systems engineering.

Applied to public safety, Client/Server applications in large metropolitan centers require continuous, careful attention to design and systems engineering support to function optimally. In these applications, integration and implementation are the biggest cost factors in a successful installation. Moreover, there is a practical limit to how large these systems can grow and still meet the design goals of the project. This is because of the inherent conflict between the complexity of public safety incident management solutions and the high performance response needed to resolve emergencies successfully. As a result, the middleware systems used to implement public safety Client/Server systems have become too complex and unwieldy to adapt to changing needs. Over time, this has resulted in smaller scale installations that can't be easily grown to serve larger groups of users and larger scale systems that can't be economically downscaled to smaller installations.

Today, the public safety sector faces a number of challenges that are affecting the traditional application model and causing agencies to look closely at alternatives. The growing costs of Client/Server solutions, new mobile data

technologies, greater demand for consolidation into larger, more economical PSAP centers, funding pressures and new technologies are causing public safety professionals to take a fresh look at their systems and consider more cost-effective alternatives.

The critical factor for public safety agencies in saving lives is system performance. The elapsed time between the initial call and appearance on site is the biggest factor in enabling first responders to successfully resolve emergency incidents. Seconds literally do count. To meet these demands, public safety systems require architectures and supporting infrastructures that fully enable rapid, accurate incident analysis and deployment of responders. The key to this operational flexibility and agility is systems that provide on-demand scalability and throughput. These are essential requirements in building successful, real-time public safety systems. Fortunately, two major trends in recent years have converged to provide for these requirements; namely Service Oriented Architecture (SOA) and Cloud computing.

SOA Overview

Beginning in the late 1990's, as Internet usage grew exponentially, it became apparent that synchronizing communications between client applications and server databases couldn't scale to support the information delivery demands of the emerging World Wide Web. The rapid growth in Client/Server systems had produced an explosion in demand for enterprise application integration (EAI) and interoperability between software systems. But the arrival of the Internet increased the requirements for organizational agility and exposed the inadequacies of proprietary vendor "middleware" stacks and technologies.

At the same time, software implementations had been steadily evolving from procedural methods and processes to Object Orientation and Component Orientation. The next logical step in this evolution was Service Orientation. The Service Oriented Architecture (SOA) model arose to promote developing software as a set of reusable, standards-based services that can be composed into various applications, depending on business requirements. It emphasizes the concepts of standards-based interoperability, vendor neutrality and loose-coupling between a service's contract (or interface) and its implementation.

In the SOA model, a service is a core software component that implements a unit of business logic; can be developed over a number of different technology platforms; can be invoked over the network through a standards based interface, and can be reused in a variety of contexts and applications. Service Orientation encompasses both the methodology for developing these services-based software systems, and the technologies that enable them, i.e., Web Service technologies and standards (SOAP-based) and RESTful services (REST-based).

Today, SOA-based enterprises such as Google, Salesforce.com and Amazon have successfully proven the scalability and vast economies of scale inherent in the model and have deployed massive Cloud computing platforms. The key reasons for their ability to deliver this capability lie in the message-based switching functions of SOA, a new generation of modular programming languages and the flexibility of the model to adapt to very high performance applications. SOA and Cloud-based services offer similar potential benefits to public safety agencies, but how should they approach these technologies in their planning?

Enabling Cloud services through SOA and ESB

Transitioning from a well-proven, though cumbersome, Client/Server model to a Cloud model requires careful planning and analysis. The benefits are compelling, but what should public safety agencies look for in Cloud Computing and how do they avoid the pitfalls of inadequate planning and implementation?

Probably the most fundamental hurdle in migrating a traditional Client/Server architecture to a Cloud-based architecture is the tightly-coupled, monolithic aspect of the Client/Server application. Moving monolithic applications to the Cloud does not realize some of the most important benefits of Cloud computing. In addition to lacking the

flexibility of Cloud services, the user experience with Client/Server applications is more rigid and rules-bound and difficult to adapt to the Cloud. The user may have a much greater freedom of action in Cloud services and a different interaction with the application.

SOA-based applications composed of reusable, interacting, loosely-coupled, distributed and self-contained services provide an architecture that is perfectly suited for the Cloud. In a SOA application, loosely-coupled services can be deployed on various servers and scaled up or down as needed without affecting the rest of the application. Fundamental to maintaining this loose-coupling is the communications backbone of the SOA infrastructure, the Enterprise Service Bus (ESB).

In SOA, an ESB (named by analogy to the data bus concept in computer hardware and operating system design) is a middleware pattern that provides a set of common functionalities required to facilitate the integration of disparate and distributed services. These functionalities are fundamental to maintaining the characteristics of a smooth functioning SOA implementation through loose-coupling and standardized interfaces. By industry consensus, the four basic functionalities of an ESB are:

Message Delivery

The main function of a data bus is to manage the delivery of messages. Services in a SOA implementation communicate through messages and, therefore, the main function of an ESB is to manage and monitor the exchange of messages between distributed services. Having an independent message delivery layer facilitates loose-coupling of services by allowing services to be implemented and deployed independently and by delegating their communication to the messaging backbone.

Message delivery in a SOA can be broken down into several aspects:

Message Routing: The ESB will determine the service that is the intended recipient of a message, and will guarantee its delivery regardless of the active state of the service, or the messaging protocols used by the service or its consumers. Message routing can be based on a variety of parameters such as message content, consumer identity or Service Level Agreement (SLA).

Message Transformation: The ESB might need to transform the message (either its content or its format) to fit the expectations of the intended recipient. This functionality greatly facilitates the integration of disparate systems that were developed independently, and possibly using different message and data standards.

Redundant Services and Versioning

In addition to managing multiple, redundant services for scaling and high availability, the ESB allows the deployment and management of multiple versions of the same service in order to facilitate managed migrations and prevent service outages.

Service Orchestration

The ESB provides a process execution environment that typically allows users to graphically define long-lived processes with their corresponding exception handling mechanisms and specify how different services interact to fulfill these processes.

Common Utility Services

The ESB provides a number of common utility services such as exception handling, message and transport security handling, queuing and persistence, a service coordination framework, a transaction framework, event notification and audit logging.

Cloud Computing Overview

As SOA has moved into the mainstream of software development methodologies, the concept of Cloud computing has begun to revolutionize the way IT services (whether software or infrastructure) are delivered and consumed. While there is a lot of confusion around what constitutes Cloud computing, the National Institute of Standards and Technology (NIST) has published a paper that provides some fundamental characteristics of what constitutes Cloud computing. NIST considers the following characteristics to be fundamental to Cloud computing:

On-Demand Self-Service:

Users can provision computing resources automatically, as needed.

Broad Network Access:

These resources are accessible over the network, using standard networking technologies.

Resource Pooling:

These resources are pooled in a multi-tenant model, and assigned to consumers on demand as needed.

Rapid Elasticity:

These resources can be rapidly provisioned or decommissioned in any quantity at any time.

Measured Service:

These resources are metered and monitored to allow a pay-as-you-go model.

At this point it is important to remember that the flexibility, agility and adaptability of the SOA model to user requirements will produce several different interpretations of service that need to be evaluated in public safety applications. There are several Cloud models that are already in Enterprise deployment, which may result in services tailored to public safety implementations. Each of the following models is based on specialized SOA middleware adaptations and more will follow as the services market becomes more segmented. Among the various Cloud services being marketed today are:

Private Cloud

This cloud infrastructure is operated exclusively for an organization. It may be managed by the organization or a third party and may exist on premise or off premise. The Private Cloud represents a level of security and restricted access that agencies may require if the nature of the information is highly sensitive, confidential or secret.

Public Cloud

This cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services. The Public Cloud gives access to the public of any and all published information.

Community Cloud

This cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy and compliance considerations). The organizations or a third party may manage a Community Cloud and may exist on premise or off premise. The Community Cloud allows certain agencies or organizations to confidentially share, exchange and view information.

Hybrid Cloud

This cloud infrastructure is a composition of two or more clouds (Private, Community or Public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting or elastic computing for load balancing between clouds). The Hybrid Cloud provides scalable services delivery but has inherent risks, given its need for strict security protocols and management. Access and postings to this cloud level needs to be both monitored and tracked to ensure accurate alignment of information with appropriate recipient.

Conclusion

The goal of this paper has been to examine the factors driving the emergence of Cloud computing services and evaluate them in the context of public safety. Cloud services are already gaining wide acceptance in Enterprise applications and offer many potential benefits to public and private safety agencies as well.

The importance of Cloud computing to the future of public safety can't be overestimated. Only Cloud-based services offer the performance, scalability and adaptability to enable the next generation of incident management and mobile applications that are now beginning to be deployed. Without SOA-based systems, the necessary economies of scale and operating cost reductions of the next generation of public safety systems simply can't be achieved.

Added to that, the ability to scale as needed and to provide consumers with a widely distributed set of services without their having to physically provision their own data centers is becoming a very attractive proposition. Cloud-based solutions offer greater flexibility, easier adoption of new technologies and potentially very large cost savings.

As Public Safety agencies evolve towards Cloud Computing services, achieving operational improvements while greatly reducing costs will require careful planning and implementation. Many observers have recommended a measured evaluation of Cloud services; develop a clear vision of what needs to be accomplished, but start small with projects that have a high probability of success. Then, use the experience gained to implement Cloud services with the greatest payback and benefits.

Public Safety agencies implementing Cloud-based services will see a whole new world of opportunities to expand the scope of their capabilities, while simultaneously reducing the recurring costs of hardware and software systems. Larger county, state and national applications will be possible from the services available through a variety of Cloud services providers. Agencies that carefully consider the required elements of SOA architectures and available Cloud services will reap the benefits of this technology for years to come.

How Cloud Computing and the SOA architecture evolve is still in the formative stages. According to the U.S. Department of Commerce and the National Institute of Standards and Technology, "Cloud computing is still an evolving paradigm. Its definition, use cases, underlying technologies, issues, risks, and benefits will be refined and better understood with a spirited debate by the public and private sectors."

References

1. The NIST Definition of Cloud Computing (Draft), Peter Mell & Timothy Grance, NIST Special Publication 800-145 (Draft), January 2011, http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf

About the Author

Dr. Toufic Boubez is a well known and respected SOA and Web services pioneer and co-author of the SOA Manifesto. He is a Certified SOA Architect and Security Specialist, as well as a consultant and Certified SOA Trainer for SOA Systems Inc. He is the founder of Metafor Software, SOA Craftworks and the founder and CTO of Layer 7 Technologies, one of the most successful vendors in SOA Governance and Security.

Prior to Layer 7, he was the Chief Architect for Web Services at IBM's Software Group, and the Chief Architect for the IBM Web Services tools. At IBM, he founded the first SOA team and drove IBM's early XML and Web Services strategies. As part of his early SOA activities, he co-authored the original UDDI specification, and co-authored a service description language that was a precursor to WSDL.

His current activities span SOA Security, SOA Governance and the impact of Cloud Computing. Toufic is a sought-after presenter and has chaired many XML and Web services conferences, including XML-One and WebServices-One. He has also been actively involved with various standards organizations such as OASIS, W3C and WS-I. He was the co-editor of the W3C WS-Policy specification, and the co-author of the OASIS WS-Trust, WS-SecureConversation, and WS-Federation specifications. He has also participated on the OASIS WS-Security, SAML and UDDI Technical Committees.

He is the author of many publications and several books, including "Building Web Services with Java", "SOA Security: Practices, Patterns, and Technologies for Securing Services" and the upcoming title "Service-Oriented Infrastructure: On-Premise and in the Cloud". InfoWorld named him to its "Ones to Watch" list in 2002, and CRN named him a Technology Innovator for 2004. Dr. Boubez holds a Master of Electrical Engineering degree from McGill University and a Ph.D. in Biomedical Engineering from Rutgers University.