

ARCHITECTING FOR SOFTWARE AS A SERVICE

Colleen Smith
Director, Software as a Service

CONTENTS

Introduction	3
Architecting for SaaS	4
Service-Oriented Architecture (SOA)	5

DISCLAIMER

Certain portions of this document contain information about Progress Software Corporation's plans for future product development and overall business strategies. Such information is proprietary and confidential to Progress Software Corporation and may be used by you solely in accordance with the terms and conditions specified in the PSDN Online (<http://www.psdn.com>) Terms of Use (<http://psdn.progress.com/terms/index.ssp>). Progress Software Corporation reserves the right, in its sole discretion, to modify or abandon without notice any of the plans described herein pertaining to future development and/or business development strategies. Any reference to third party software and/or features is intended for illustration purposes only. Progress Software Corporation does not endorse or sponsor such third parties or software.

INTRODUCTION

Just because it is now possible to deliver any application using any user interface, into a browser over the internet, does not mean that in doing so a Service Provider will be able to profitably deliver that application to a chosen class of users – at massive scale. Profitable delivery of Software as a Service (SaaS) requires a transformation of the architecture, as well as an understanding of the nature of the application to be delivered and a determination of how appropriate that application will be with the SaaS delivery model.

The Business Application

Buyers of technology are primarily seeking solutions that allow them to grow their core business, but with lower IT costs and less management complexity. Market indicators point to the fact that the majority of applications ideal for service delivery will be ones that have a specific function or vertical orientation – as organizations are primarily seeking solutions that allow them to grow their core business, but with lower IT costs.

Any one or a combination of, the application types below represent significant opportunity in the SaaS Provider marketplace:

- 1) Verticalized/specialized applications – Examples include specialized ERP applications, such as lumber, plastics, or distribution, also education administration systems, medical claims processing, and legal/property management- etc.
- 2) Applications not requiring customization - those requiring ease of use for self-service and personalization because the application touches many different types of users. Such examples include human resources, materials/warehouse management, facilities management, etc.
- 3) Domain specific business-to-business applications - where it is almost a necessity to be available via a hosted service to other third parties. These applications usually offer services in conjunction with the business application process.
- 4) Applications that are departmentally critical - but not core to the entire business and where the support of the application required from corporate would be out of proportion to the use of the application at the departmental level. This would make the additional services component of the offering become critical to the line of business/department owning the business process. For example, consumer affairs division applications at a large CPG organization, the parts recall applications at a large automotive organization, the transportation/freight departmental systems at a large manufacturer, etc.
- 5) Customer/Supplier facing applications -those applications where the majority of users are outside of the four walls of the organization and ubiquitous web access is critical to the success of the use of the application. Examples include sales force automation, customer service, call center, supplier portals, and e-commerce applications.

ARCHITECTING FOR SAAS

Moving to a SaaS delivery model requires a careful examination of both the application architecture, and operational aspects of day to day service provisioning. One of the greatest 'levers' that can be pulled is that of an appropriate application architecture for the centralized deployment of SaaS.

Application Architecture Options

A large majority of legacy applications have been architected as a two-tier (client/server) architecture, usually with a Windows front end (the client or user interface portion) that communicates to back end database server processes. While applications that have this as their architecture can be delivered via SaaS, there are issues that may arise, based on performance and cost associated with this architectural design.

The "remoting" of the client portion of an application utilizing two-tier architectures entails a large capital and management expense at the central data center site. Applications of this type will require implementation and licensing of technologies such as Microsoft® Windows® Terminal Services, or Terminal Services plus additional licensed technologies such as Citrix®. This results in a potentially large embedded cost that must be passed along to the service consumer.

Stateless application architecture will deliver the best performance, at the lowest incremental cost, for the widest array of users. Utilizing technologies such as Progress Software's WebSpeed® coupled with appropriate internal architectures as described by the OpenEdge® Reference Architecture, the SaaS provider will be able to take advantage of rapidly declining hardware and managed hosting costs.

The major elements of the recommended SaaS architecture are:

- Presentation Layer - Service Interface Proxy
- Business Layer - Service Interface, Business Entity Layer, Data Access Layer, Managers and Utilities
- Data Layer - RDBMS

SaaS applications should be architected vertically in specific tasks, and the tasks aggregated in business processes. Progress Software offers consulting services and although it is not required to move to the SaaS delivery model, recommends looking at the Application Transformation process. Capabilities exist such as Web-centric Application Enablement, for assisting our application partners in determining the quickest way, to migrate their current character or client server application to the Web.

Methodologies have been built to assist application partners transform their applications. Allowing them to exploit new market opportunities, while developing an application that is less costly to support and maintain, lowering overhead and increasing profit margins.

SERVICE-ORIENTED ARCHITECTURE (SOA)

To facilitate the need of SaaS Providers to deliver integrated sets of applications in a service model, Progress has created a developmental framework and set of technologies around the OpenEdge platform. Progress focuses on technologies that will facilitate the development of a new generation of highly specialized applications by creating a framework that has been specifically designed to expose defined business processes for integration within and between applications/organizations.

SaaS business applications based on a Service-Oriented Architecture (SOA) are designed to make it easy for applications to share information and processes within a business or among global partners. Applications designed with a SOA can more efficiently meet SaaS technology challenges.

SOA is not as much of a revolution as an evolution in the model for developing software. SOA provides a framework that allows software components, residing on any network, to be published, discovered, and invoked by each other. This "Publish, Find, and Bind" paradigm allows a Service Provider to publish the name of its service, its location, and its interface in a directory. A service requestor looks up the services and then dynamically binds to the service using the information that is provided by the directory.

SaaS Development/Deployment Success Factors

One critical SaaS requirement is a low total cost of ownership – and that is what Progress Software strives to deliver with our development, deployment, and management tools. Another key requirement for SaaS is scalability –whether you need to provide access to ten or 10,000 users, you need to handle sudden growth and varying loads with ease as a SaaS provider.

The use of application servers enable software components to be reused and recycled almost infinitely – keeping your investment low as the needs of your customers expand and change. From an application management standpoint, the Progress database provides capabilities to ensure optimal performance and cost efficiencies. It provides a complete set of tools to manage and monitor the Progress environment remotely, and it detects and repairs conditions that could result in application outages – which is even more critical as applications are provided via the SaaS model.

To fulfill the vision of SaaS, the Progress OpenEdge platform provides three main technology components:

- Integration infrastructure, to address the challenge of assembling, deploying, and managing Service-Oriented Architectures in a SaaS model.
- Technologies to enable the management of data and monitoring of business service levels - ensuring service level management.
- Products that enable the development of brand-new SaaS applications.

Progress continues to extend its tool set for building open, highly integratable applications with new releases of Progress OpenEdge. Its family of deployment products provides the ability to interface business logic with any application client, any data source, and any other application, be it .NET, Java, or a legacy system.

The Progress OpenEdge platform includes various avenues for integrating applications. In addition to providing objects designed to handle XML data exchange and allowing Java and ActiveX clients to access business logic on the Progress application servers, it supports Sonic Software's enterprise-class services infrastructure. When they are integrated with Sonic, Progress-based applications can share information with other applications and services.

The next step is to build on these products and features and to provide the framework for building complex business systems end-to-end. New SaaS-enabled applications will share some key characteristics:

- Central management of services – The SaaS applications are hosted, managed, and maintained in a centralized data center enabling a reduction in overall support and maintenance costs.
- Application accessibility – SaaS providers provide access to software-automated processes as a service offering. Accessibility, policy-driven configuration and personalized customization, give the users of the service the ability to have control over how the application is used, rather than how it is implemented.
- A one-to-many service- In order to accomplish the “economies of scale” associated with SaaS – the providers develop a multi-tenant application that supports multiple end-users accessing a single source of the application business logic and database.
- Guaranteed performance and reliability via service level agreements.

While most agree that these are the characteristics of SaaS – research indicates that the weight of these characteristics is not evenly distributed, when it comes to the ultimate success that a SaaS provider will have.

True next generation SaaS offerings that have been correctly architected can deliver much higher value than installed applications software. SaaS providers that design and operate their software using this service-oriented approach will result in providing higher value, but with a far lower cost structure.

Corporate Headquarters

Progress Software Corporation, 14 Oak Park Drive, Bedford, MA 01730 USA
Tel: 781 280 4000 Fax: 781 280 4095

Europe/Middle East/Africa Headquarters

Progress Software Europe B.V. Schorpioenstraat 67 3067 GG Rotterdam,
The Netherlands
Tel: 31 10 286 5700 Fax: 31 10 286 5777

Latin American Headquarters

Progress Software Corporation, 2255 Glades Road, One Boca Place, Suite 300 E,
Boca Raton, FL 33431 USA
Tel: 561 998 2244 Fax: 561 998 1573

Asia/Pacific Headquarters

Progress Software Pty. Ltd., 1911 Malvern Road, Malvern East, 3145, Australia
Tel: 61 39 885 0544 Fax: 61 39 885 9473
Progress is a registered trademark of Progress Software Corporation. All other trademarks, marked and not marked, are the property of their respective owners.

PROGRESS
S O F T W A R E

www.progress.com

Specifications subject to change without notice.
© 2006 Progress Software Corporation.
All rights reserved.