Questions Answered.
Solutions Provided.

VISIT WWW.QAT.COM FOR MORE INFORMATION ON QAT SOLUTIONS!

- IT Strategy & Architecture
- Service Oriented Architecture
- Application Development and Support
- Legacy Modernization
- Staff Augmentation
- CA Gen Specialty Solutions
- CA Gen Training
- QAT Exchange
- QAT Flow
- QAT Function Manager
- QAT Publisher
- QAT ReFactor
- QAT Security
- QAT WebDaptive
- QAT Wizard

Subscribe to the QAT Insights e-newsletter to have articles like this one sent directly to your inbox every month. Visit qatinsights.com to subscribe!

The Path to SOA
by Rich Barndt

Service Oriented Architectures (SOA) provide clear business benefits by making software components freely interchangeable, reusable, and loosely coupled. Because SOA isn't something you buy off the shelf, the implementation of a SOA can be daunting.

This does not have to be the case. There is a clear, step-by-step path for any company to move to an SOA. In designing and implementing Service Oriented Architectures for more than 5 years, QAT Consultants have developed the skills and experience to put you on the right path. If you're considering the advantages of SOA, but aren't sure what steps to take, consider a QAT Architecture Review.

In a general sense, the movement from a traditional application environment to a SOA involves five fundamental levels of SOA maturity. The first level of maturity in SOA implementation is Service Enablement. In this phase, you need to break your application infrastructure into modular components, which will form the building blocks of your SOA. Many of your components may already be modular so this may simply be a matter of formalizing the interface definitions. Other applications may require several method calls or CICS transactions be assembled to form a single Service module. Other application services will require complete building. In addition to being modular, Services in a SOA should be "accept and return" self-defining data structures (i.e. SOAP) and use standards based protocols (i.e. HTTP) wherever practical. This may mean adding wrappers around existing application components to give them standards based interfaces or building new standard based services.

Perhaps the most difficult aspect of the first level of SOA is determining exactly which protocols and data structures are "Standard". If you allow too few standards, then you will need to write wrappers for virtually every application, slowing your SOA development and reducing its flexibility. On the other hand, if you allow too many standards, you can create a maintenance nightmare since each protocol

requires administrative maintenance and there is little or no coherence in your data structures. Thus, it is quite important to define a clear set of enterprise protocol and data structure standards as well as the governance procedures to add standards to and remove standards from this set.

As your organization develops and defines more services, the threshold is usually 40-50 services, disparate requirements for data structures and protocols of the services make it difficult to have the services all communicating directly with each other and the effort to wrap every service with matching protocols and some form of generic superset data structure becomes unmanageable. To decouple the data structure and protocol for the service interfaces, it is normal practice to develop an Enterprise Service Bus (ESB) consisting of a set of Mediation Services that perform translation between different data structures (data transformation) and protocols (protocol conversion), keeping specific interface dependency on other services out of your core business services.

Completion of an ESB provides your organization with an extremely flexible set of services that can be used freely by any client. Since an ESB intentionally does not supply any form of control over which services are called by which other services and in which order, most organizations go on to add a set of Orchestration Services to control this flow. This allows business analysts to develop abstract process models with languages like Business Process Execution Language (BPEL). Such Process Modeling is typically the

Questions Answered.
Solutions Provided.

VISIT WWW.QAT.COM FOR MORE INFORMATION ON QAT SOLUTIONS!

- IT Strategy & Architecture
- Service Oriented Architecture
- Application Development and Support
- Legacy Modernization
- Staff Augmentation
- CA Gen Specialty Solutions
- CA Gen Training
- QAT Exchange
- QAT Flow
- QAT Function Manager
- QAT Publisher
- QAT ReFactor
- QAT Security
- QAT WebDaptive
- QAT Wizard

Subscribe to the QAT Insights e-newsletter to have articles like this one sent directly to your inbox every month. Visit qatinsights.com to subscribe!

first step in implementing this third level of an SOA. The underlying ESB allows the actual processing details of the target service to be isolated from the Orchestration Services. The technical implementation of the Orchestration Service then consists of providing an Orchestration runtime that can interpret the process model and then binding individual activities in the process to the Services that will execute them.

The real business benefit of an SOA comes not from the potential to reuse services and perform internal service updates without affecting others, but from actually doing these things with your SOA. In order to do this, you need to gather objective information about the performance and availability of your actual services to identify bottlenecks, select targets for improvement, and judge the business effects of changes to underlying services. Such Service Management is critical to getting a clear return on your SOA investment. Further, to encourage reuse, you must document dependencies both of and between services and store such dependency metadata along with the Service Level metadata discussed earlier into a commonly accessible and searchable repository and define procedures to update that data. It is this formal SOA Governance that marks the fourth level of SOA maturity.

Finally, the development of an SOA is not a one-way journey, but the establishment of a sustainable lifecycle that leads to continuous business process improvement. To get the most out of an SOA, you must put governance procedures into place not only to measure the performance and availability of your services, but you must have defined and clear procedures for making business decisions based on the information that is gathered. Thus, process modeling allows process measurement, which allows process improvement planning directly in the modeling tool allowing the cycle to repeat.

Learn more about Service-Oriented Architecture (SOA) at QAT's SOA Resource Center. You will find articles, resources, products, and services that will help you successfully implement SOA in your organization.

Other Articles of Interest:

- **AJAX - Cleaning up in IT**
- **Is SOA the Answer?**
- **SOA Business Value**
- **Ensuring SOA Success**
- **White Paper: Simple Benefits of Service-Oriented Architecture**

**Visit
QAT's SOA Resource Center
soaresourcecenter.com**