



## TAMING THE ENTERPRISE SERVICE BEAST



By Paul Ferris

Technical Architect  
Brulant, Inc.

Service Oriented Architecture (SOA) presents a solution to the problem of integrating many different systems into a coherent solution that addresses the needs of the modern-day enterprise. Unfortunately, SOA breeds complexity on an expanding order and *inter-dependency*, as increasing numbers of sub-systems depend upon each other.

### Enter ITSM

IT Service Management (ITSM) presents a discipline for managing change and ensuring quality of service. In a small shop, ITSM isn't a mandatory item, but in a medium to large enterprise - especially one with an SOA implementation - the business needs demand coordinated changes to services and optimal uptime.

Products like IBM WebSphere Application Server and WebSphere MQ present a company with pieces of the *technical* solution, whereas ITSM present an organization with process rules to govern the way that the technology is *managed*. While the technical pieces are the essential building blocks, ITSM represents the playground rule-set; they establish well understood terminology, organization bodies and boundaries for dealing with the complexity that technological change presents.

All of this complexity has business value despite the sizable cost to the enterprise. No company implements an expensive application framework like SOA, in other words, without the cost of the components being weighed against the payoff in added value. Similarly, the ITSM choice is almost a no-brainer. Without ITSM, service in an SOA environment can be chaotic.

[www.brulant.com](http://www.brulant.com)



### WHAT'S MISSING?

Inevitably, the deployment of both SOA and ITSM lead to complexity on a multitude of axis. Systems cannot be changed without extensive planning and "paperwork" by engineering and administrative staff. New software cannot be installed without compliance and auditing by architectural bodies that evaluate the software against the needs of the enterprise. Code cannot be deployed and data leveraged that does not fit the modeled backbone of the Enterprise Service Bus and repeatable, documented and automated processes.

Complicating matters is the inevitable chaos wrought attempting to manage the infrastructure needs of the modern-day enterprise; endless patch cycles, deployments, migration projects and multiple-tier dependency nightmares. All of this is complicated by the various skill levels of the staff charged with keeping the Enterprise Beast in check.

The missing link is some kind of standard methodology that clarifies the difference between engineering and administrative duties - if such boundaries even exist. Where does engineering end and deployment by administrators begin? What is a unit of work in the task list of a deployment? Even harder to define - what truly constitutes a change?

### PACKAGE MANAGEMENT FRAMEWORK

Package Management Framework (PMF) standardizes on a fundamental unit of work; a package. Systems are deployed utilizing fundamentally defined, carefully constructed, discreet packages. These packages utilize their environmental surroundings in an intelligent fashion to deploy automatically, without human (administrative)

intervention. Packages that are deployed register in such a way that all configuration is stored in the OS package repository.

System configuration is documented, soup to nuts. All OS components, patch levels, software components (such as MQ and WebSphere) - even the last ounce of added software configuration, are listed in one place. These packages are then centrally maintained with the same tools that developers have leveraged for years.

The end result is a system that is built with a well-defined list of packages. Change management is then focused upon changing the automation - not on the servers themselves. This yields obvious benefits: a common change database, reduced administrative skill levels, common language and clear engineering / administrative turn-over points.

### TAMING THE BEAST

All of these benefits translate to systems that are deployed faster, with higher quality, lower cost and greatly streamlined communication across the enterprise. The business benefits due to the greater consistency and decreased need for expensive subject matter expertise. Higher ITSM benefits are realized thanks to the common framework of package deployment and clearly delineated version boundaries during change requests.

PMF represents yet another culture shift. The framework is to systems management what ITSM is to change management - it represents a standardized way of deploying and managing technology across the ever-changing ultra-connected enterprise. The costs, similar to ITSM and SOA, are trivial compared to the expanding costs of doing business without.

