Which Business Integrations Provide the Greatest Cost Value?

By Mark Temple-Raston

Enterprise architects are asked daily to align IT design with business requirements. Initially, this wasn't so difficult. The business requirements were mostly functional and were mapped to functional capabilities in application design (through use cases, for example). In recent years, enterprise architects have been asked to support more sophisticated enterprise business requirements that span multiple applications, systems, and business units. They've also been asked to explain if or how contemporary technologies could be used to support corporate strategies. Those who can explain the opportunity clearly in the decision-maker's language stand to help the company most. Unfortunately, few enterprise architects today are comfortable with architecture analysis based on cost management, although I believe that's likely to change as the business and IT architecture continue to align.

There's probably a reason why enterprise architects have hesitated to learn cost management. While all proposed enterprise business integration projects should be based on solid cost analysis, traditional cost analysis is often not up to the task. In the last 20 years, we've seen a growing number of detractors to traditional cost accounting systems. Such systems were developed to produce financial information for the markets and tax authorities (e.g., Internal Revenue Service [IRS]).

Many experts argue that numbers essential to financial reporting aren't sufficient to manage costs (or savings) in a modern company. For more on this, see Robert S. Kaplan and Robin Cooper's book, *Cost & Effect: Using Integrated Cost Systems to Drive Profitability and Performance* (Harvard Business School Press, 1998).

Further evolution in cost management is needed to accompany technology advances. Activity-based costing (ABC) provides a good start. Many principles of ABC were developed for modern manufacturing. This article summarizes ABC as applied recently to the service industry, identifies basic lessons in integration learned from ABC, and maps them to capabilities in service-oriented architectures (SOAs) to clarify the cost value that SOAs provide. Although the emphasis of this article is on cost management, it must be stated that cost management can't ignore issues of quality and agility. When properly designed, SOAs can also contribute to quality and agility, as well as cost management. Quality and agility in SOAs is an interesting topic, but won't be discussed here.

We identify two areas in which ABC has proven value: shared component services and overhead costs. The first is perhaps familiar and formalizes arguments sometimes presented to support SOAs. The second reduces the setup costs for executing business processes, and has not been emphasized in the literature. A special case of horizontal integration applies to postmerger integration.

Cost Management

A traditional enterprise costing system often focuses on the direct labor required to perform a given activity, then assigns overhead to the activity in direct proportion to the activity's labor content. This approach gives the impression that labor is the fundamental variable upon which all other activity costs are derived, or equivalently, that direct labor is the only controllable expense. This is false. For example, we know that reduced labor costs through process automation usually aren't directly proportional to overhead, because automation tends to increase overhead, including engineering, programming, and maintenance support. In Figure 1, we separate the cost of a service industry activity into:

- Direct labor costs
- Overhead costs
- Transaction costs.

The transactions slice of the cost pie refers to the transaction systems that

business integration journal takeaways

BUSINESS

- Lessons learned from activity-based costing (ABC) identify two types of business integration that provide significant recurring cost savings to the business: reusable service integrations and horizontal process integrations.
- Service-oriented architectures (SOAs) are nicely aligned with the objectives of modern cost management systems.

TECHNOLOGY

- Enterprise architects will need to know more about modern cost management to align IT with the business.
- Applications that exploit service standardization or that horizontally integrate business processes provide excellent opportunities for providing cost value to the enterprise.
- SOAs can play an important role in helping improve ABC initiatives.

provide services to the users. According to Michael C. O'Guin, author of *The Complete Guide to Activity-Based Costing* (Prentice Hall, 1991), direct labor is usually a small portion of the overall cost of the activity, typically less than 5 percent. With such a small slice of the pie assigned to direct labor, business integration projects that focus on direct labor (e.g., automation) seem unlikely to provide the greatest cost value to the organization. For this reason, direct labor and automation don't play a significant role in this article. We'll focus on the remaining 95 percent of the activity cost—split between overhead and transactions.

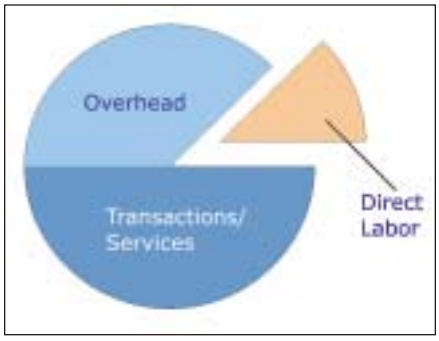

Figure 1: Distribution of Costs for a Typical Service Activity

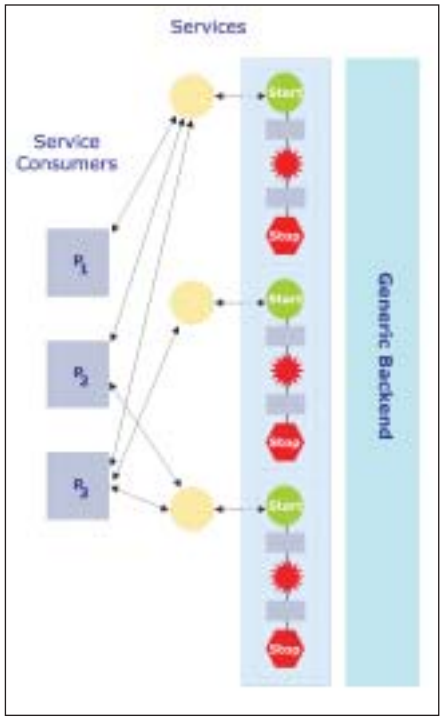

Figure 2: Generic Service-Oriented Architecture

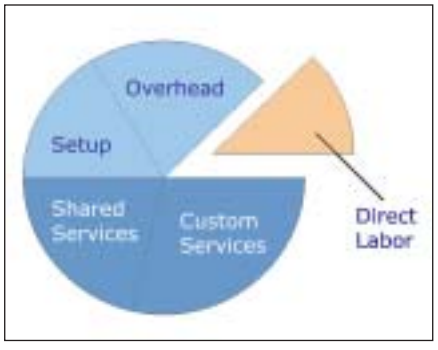

Figure 3: Detailed Costing for a Typical Service Activity

Traditional costing systems do nothing to encourage the reduction of overhead spending by process managers, because such schemes usually spread overhead across all processes (appropriately called "peanut buttering"). The whole organization benefits from the process manager's selfless act, but the process manager gains negligible cost benefit. These traditional cost system limitations have been well-documented, and enhancements to costing systems that focus on cost management are emerging; foremost among them is ABC.

ABC is a set of accounting techniques that seeks to accurately assign all costs to a given activity or to the user who invokes the activity to uncover hidden costs and to achieve greater levels of efficiency, quality, and profitability. ABC is used to:

- Capture the current costs of performing an activity
- Target high-cost activities
- Provide a context for establishing and monitoring performance measures
- Establish a link between activity modeling and economic analysis.

Activities are categorized as either value or nonvalue added, primary or secondary, or required or nonessential. A value-added activity must be directly related to customer requirements, service or product, as opposed to administrative or logistical. For example, if the output of an activity were an inventory report or update for products, the output would be nonvalue-added but necessary to the organization.

ABC was first developed in the manufacturing industry, where the goal was throughput. However, since the emphasis has always been on activities, the ABC model can be easily extended to include tasks performed by marketing, sales, logistics, purchasing, and corporate staff. Our interest is in business activities that are executed through IT and can be potentially mapped to software services in an SOA.

Reusable Service Integration

Contemporary SOAs, particularly those based on Web services, wrap application(s) with a well-defined XML interface to make the underlying language(s), platform(s) and database(s) less important, or ideally unimportant, to users of the service. The benefits are that in an SOA technological compatibility doesn't dominate what's possible for the business. Also, in an SOA, implementation of the business requirement is usually less expensive. The savings mentioned usually stem from development costs.

ABC is mostly concerned with recurring costs. So, for example, ABC would identify the number of transactions in a service activity as a major cost driver. ABC would then attempt to:

- Reduce the number of transactions
- Introduce standard-shared transactions
- Modularize transactions.

Consider a manufacturing analogy: The cost of manufacturing multiple products is significantly reduced by designing products with fewer components and standardizing the components that remain across all products. ABC for the services industry seeks to do the same thing for transactions.

In Figure 2, we see three Websites (P1, P2, and P3) that manage distinct sets of banking products and services (e.g., a retail banking, a trading, and a loyalty credit card Website). All three Websites will likely have their own customized functionality, but are expected to share some services. For example, to transfer funds from your retail banking accounts to your trading site accounts, or vice versa, we could reuse the business processes that get information about your accounts. The reuse of business processes makes it practical to optimize the design to improve reliability and consistency, and reduce maintenance costs.

So, as we see in Figure 3, ABC suggests that we divide transaction costs into shared service costs and activity-specific service costs (custom services). ABC states that the greatest benefit is achieved when the total number of services for a given activity is reduced, and the services that are shared across activities are increased. This reproduces from a cost-management perspective an ideal often presented by SOAs: shared business components. This recommendation from ABC must be balanced against increases in overhead, particularly maintenance costs. For example, a "Swiss army knife" service that does everything for everybody reduces the

total number of services, but is likely to have a complex ownership model that results in significant maintenance costs.

In the next section, the business integration we propose based on ABC is less conventional and perhaps a bit more interesting.

Horizontal Process Integration

ABC identifies setups as another major cost driver. Setups in a Web-based services environment are various navigations and authorizations that must occur before a user can execute a given business process. Consider enterprise approval processes. By definition, approvals can't be automated, and they're usually executed internally. Often, senior management is required to provide final approvals to incentive bonuses, projects, invoices, travel, expenses, etc., in separate, browser-based applications despite the fact that the high-level approval process is similar for each approval type. The set of actions required to launch an approval process comprises the setup overhead and has a cost associated with it. With a multitiered approval process, the approval setup costs accumulate as approvals move up through the levels of management approval. The setup costs can be somewhat reduced by implementing single sign-on (SSO) authentication, but the navigation through distinct applications remains.

Horizontal process integration is among the easiest ways to significantly reduce set-up overhead costs and provide business value. Horizontal process integration identifies a set of similar business processes or activities and integrates them into a unified presentation layer. Figure 4 presents a high-level example. This example shows four ERP systems that are identified to have an overlapping set of similar business processes (e.g., approvals). A horizontal integration would require that a universal workflow be defined and exposed to users through a presentation layer. Often, the workflows in each ERP back-end system aren't required to change. The same approach can be used for postmerger and acquisition integration, since selecting the correct enterprise workflow can be viewed as an extra step in the setup process.

The universal workflow presented previously can be implemented in an SOA as well by introducing both a services layer (middle tier) and service consumer layer (applications, presentation layer). Besides improved cost management, the following benefits can accrue:

- Universal workflows implemented in a middle-tier insulate users and presentation layer developers from variations in workflow on the back-end.
- The middle tier can provide management processes as shared services to the company, enabling business users through several applications to manage their own objects (self-service) subject to company policy and best practices implemented on the middle tier.
- Universal shared services make it easier to integrate new business processes into existing business applications, particularly during mergers and acquisitions.

Conclusion

Antiquated cost management systems block the business benefits of advanced technologies. However, it's IT's responsibility to demonstrate its business value with a current cost management system. Fortunately, industry experience with modern cost management systems can tell us where business integrations are likely to be most cost-effective and where IT should argue for value. We've identified two types of business integrations from our high-level understanding of ABC: reusable service integrations and horizontal process integrations. There are many more integration opportunities, but these two seem the easiest to lobby for and implement. **bij**

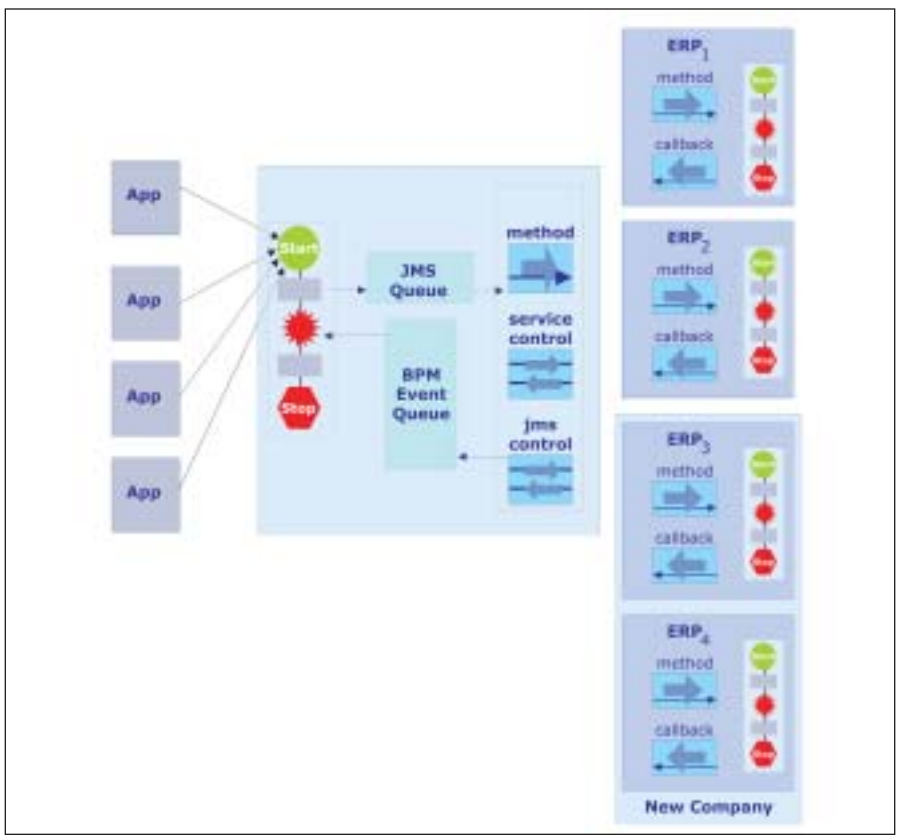

Figure 4: Horizontal Integration

About the Author

Mark Temple-Raston, Ph.D., Cantab, is a consultant and senior architect with MphasiS Corp. in New York City. He designs enterprise architectures for large tier-one financial services, pharmaceutical, and logistics companies.
e-Mail: mark.temple@mphasis.com
Website: www.mphasis.com