Service oriented architecture solutions
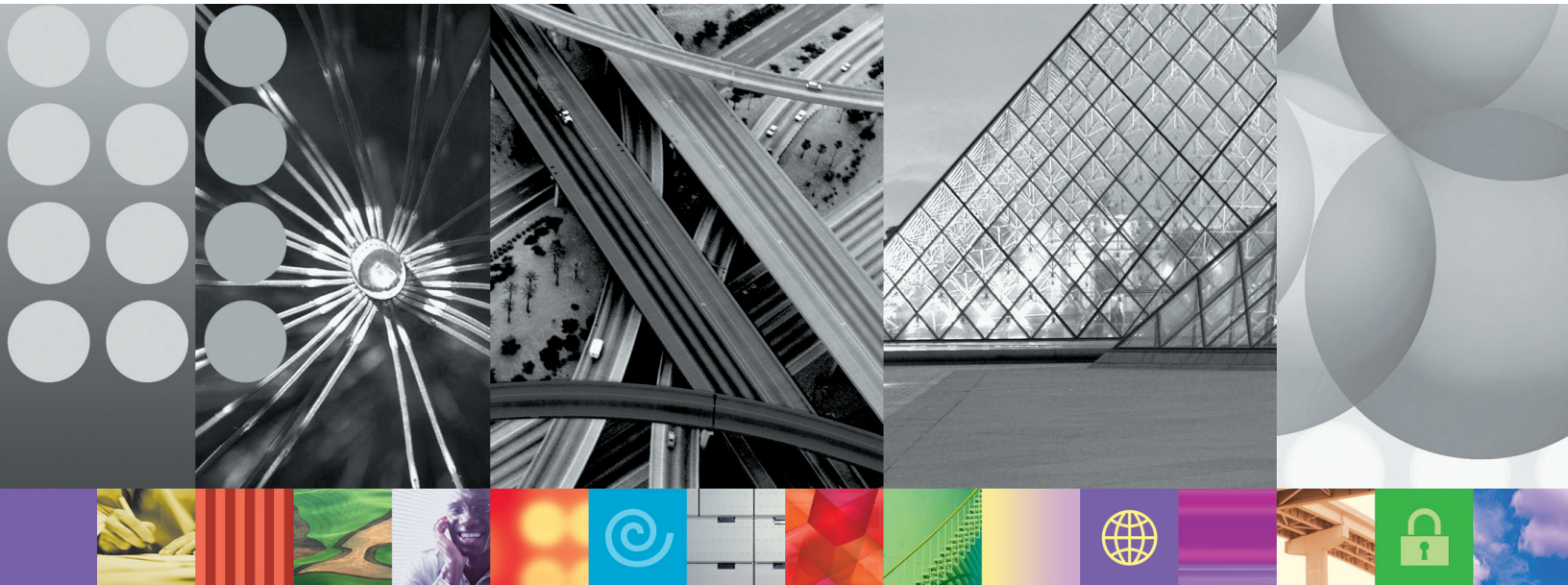**To support your business objectives**

IBM

Five SOA projects that can pay for themselves in six months.

The SOA revolution

Service oriented architecture (SOA) could revolutionize the way we think about IT. Why is that possible? Because SOA finally has the potential to make the concept of reuse real. Companies have been talking about reuse for years, but have never been able to transform that talk into full-scale reality. Now, you might be asking, "How can SOA succeed where previous approaches have failed?" Because the standards, best practices and governance models have finally matured to the point where reuse can actually work.

SOA is, by definition, an architecture. At the same time, SOA is an approach to IT that can help solve immediate business challenges. And SOA can begin paying for itself quickly. In fact, the number of opportunities for quick return on invest-ment can be surprising. For example, many businesses are unaware of the number of duplicate process that occur in separate departments and applications—and how much these duplicate processes are costing them. When you examine the costs and lost revenue attributable to redundant function and duplicated effort, you begin to see the value of centralized services over having to manage multiple competing and overlapping functions.

Business has gotten increasingly complex over the past couple of decades. Factors such as mergers, regulations, global competition, outsourcing and partnering have resulted in a massive increase in the number of applications any given company might use. Even if these applications were developed in the context of a logical master plan, the situation could be difficult. But it's often the case that these applications were built with little knowledge of the other applications with which they would be required to share information in the future. As a result, many companies are trying to maintain IT systems that coexist, but are not integrated.

SOA offers an approach that can be incrementally adopted as organizations start the necessary but daunting task of bringing order to what might have become chaos. The projects outlined in the following sections have helped many companies move further along the SOA adoption path. But just as important, these projects have produced clear and demonstrable business results in just a few months.

As companies use SOA to provide standardized services and business processes, the value of IT grows exponentially. When there is only one view of each customer, supplier and business partner, and only one business process for each specific business need, organizations can run more smoothly and put more of their energy into growing their businesses rather than taming their IT infrastructures. In the past, the prevailing belief was that the value of reuse was in eliminating duplicate development and maintenance. Now, however, it is widely accepted that the true value of reuse is in the standardization of business processes. The following projects help demonstrate how organizations have used SOA to their immediate advantage, while at the same time laying the groundwork for more flexible IT systems that can provide competitive advantage in the years ahead.

Project 1

***Delivery-date notification service: Providing a single source
of information to improve customer service***

This example shows how one IBM client increased revenue
and reduced costs by creating a centralized delivery-date
notification service. This major retailer has estimated the
revenue impact of its SOA project at US$20 million per year.
The project, which cost a small fraction of that US$20 million,
consisted of creating a centralized service used as a single
source of information for delivery-date commitments.

Many businesses can suffer in the marketplace because they
do not provide the necessary means to effectively manage
customer expectations. Consider what can happen when a
service company makes an appointment to send a service
representative to a customer's home to repair a problem.
When making the appointment, the company can commit only
to having the representative arrive at any time during an eight-
hour timeframe. Then, after the customer has waited at home
for an entire day, the company calls to inform the customer
that the service representative cannot make the appointment
and needs to reschedule. This inability to manage customer
expectations can result in alienating customers—and even
in losing some customers to competitors. This concern
transcends industries and is important to any organization
that depends on the quality of its customer service to maintain
and grow its business.

In the retail industry, successful retailers are those who can
manage, meet and exceed customer expectations. Whether
it is the quality of the merchandise, maintaining necessary
inventory levels or fulfilling orders expeditiously, the retailer
is constantly under each customer's microscope—and is
expected to meet high expectations. Delivery dates must
be met, and if a date has changed, the customer must be
informed, regardless of who is at fault. This issue prompted
this IBM retail client to implement an SOA solution.

The retailer's fulfillment chain comprised multiple
systems—each of which could update the promised delivery
date for an order. When someone changed a delivery date in
one of the many fulfillment applications, the information wasn't
consistently updated in the order-processing system. As a
result, when a delivery date was adjusted, the customer was
frequently not properly informed. Because these disparate
systems didn't reliably communicate these updates to each
other, customer representatives had to spend valuable time
checking multiple order-management systems to try to answer
customer queries or resolve a complaint when an order did
not arrive on the expected day. The long wait for information
and the inability to keep the customer informed of changes
frustrated both customers and employees.

The solution? A centralized delivery-date service. Now, when a delivery date is changed, the fulfillment system sends a delivery-change notification to this event-driven service through the retailer's enterprise service bus (ESB). As a result, the order system database—and any other system that subscribes to this service—is immediately updated. This capability allows customer service representatives to respond to and address customer issues more quickly and accurately.

The retailer's delivery-date service was a Simple Object Access Protocol (SOAP) over HTTP service managed by an ESB. The existing IBM WebSphere® MQ interface on the fulfillment system was routed to call the new service through the ESB. To support the protocol transformation of WebSphere MQ to SOAP over HTTP, a mediation was developed that sits in front of the service implementation. All requests from the fulfillment system's WebSphere MQ interface are routed to this mediation where the WebSphere MQ to SOAP transformation occurs. The SOAP request is then forwarded to the delivery-date service. The product used to develop the ESB provides all of the administrative support necessary to enforce the routing of these WebSphere MQ to SOAP over HTTP services, avoiding the need to do this routing programmatically and minimizing development time and resources. The design also enables publish-subscribe capabilities where the delivery-date service is published on the ESB. Through these publish-subscribe capabilities, other systems within the retailer can subscribe to the service so that they are also notified of the delivery date change, encouraging further component reuse. This publish-subscribe model also enables all dependent systems to be notified simultaneously.

In technical terms, this is a simple SOA project. The total effort to create two centralized services and build an ESB required four developers for four months. But the impact to the business is extensive. Customers are no longer disappointed with missed delivery dates and contradictory delivery information. The number of cancelled orders has declined, as have the number of delivery attempts that are unsuccessful due to scheduling confusion. Even more important, the retailer is profiting from the repeat business of customers who value the company's ability to meet its delivery commitments.

Project 2

Transaction dispute service: Automating processes across multiple companies and users

Financial services organizations all over the world are struggling to reduce costs and increase customer satisfaction, while at the same time dealing with an increasingly complex array of security concerns. These competing challenges often result in situations where companies must choose to spend more or risk customer dissatisfaction. The financial services organization discussed in this scenario used SOA to help lower costs and improve customer satisfaction, all without compromising security.

A common thread among all industries is the movement to automate processes that are traditionally labor- and cost-intensive. Often the most expensive processes are those that are required to handle the exceptions in everyday business. Whether it concerns reverse logistics within a supply chain or billing disputes for services rendered, handling such exceptions can be expensive because of the enormous amount of human intervention that is often required.

This project enabled a financial services organization to significantly decrease the labor-intensive processes and high costs associated with disputed transactions. Automating this process by creating a centralized service helped enable the organization to realize estimated cost savings of more than US$200 million per year. It achieved these savings by replacing a manual and archaic set of processes with an automated service that was surprisingly simple to implement.

This financial services organization functions as a support network for many large retail establishments. The process this organization had in place to handle disputed charges is an example of an extremely expensive business exception. When a retail customer disputed a transaction, the financial services organization would manually print all transactions and send them through ground mail to the customer to identify which transactions were being challenged. The customer would then have to sign these papers and send them back through ground mail to the financial services organization, which would then package them and send selected documents to the retail institution. After this paperwork was received, the retail institution would decide whether the charge should be removed. This process could take up to 20 days to complete and typically cost the organizations involved between US$400 and US$700 per transaction.

Because security was such an enormous concern, any new solution had to help ensure the integrity of the financial services organization's core transaction system. Protection of customer data was essential, so this financial services organization wanted to maintain only one access point to its transaction system. In the past, the organization had not been comfortable with integrating this system with its retail partners because it would have required a unique, point-to-point connection for each partner. Maintaining these point-to-point connections would have been cost-prohibitive and would have most likely resulted in a higher than acceptable error rate. The financial services organization would only allow partners to communicate with its transaction system if it was possible for all of them to share the same connection— reducing the financial services organization's security and maintenance responsibilities to a single connection. SOA turned this potential obstacle into an opportunity.

To create a solution, the financial services organization deployed a service in front of its core transaction system that now allows retail partners to transmit dispute claims to the financial services organization on behalf of the partners' retail customers. To register a dispute, customers now simply log into the retail institution's Web site and view a list of transactions that have posted to their account. Customers can then select the transactions they wish to dispute. The Web site sends this request to the financial services organization's transaction-dispute service. The authentication that customers provide while logging on to the Web site enables the financial services organization to stop requiring paper documentation with a handwritten signature. Today, the transaction-dispute process averages a total of three hours, reduced from 20 days, and costs only US$40 to US$70 per transaction, instead of the previous US$400 to US$700 per transaction—representing a 90 percent reduction in costs.

By creating a simple SOAP-based service, the financial services organization can maintain its strict security requirements. Its retail partners simply send their dispute requests to the financial services organization's SOAP-based service using HTTP or Java™ Message Service (JMS). The financial services organization placed a virtually impenetrable security layer between this service and their transaction system to help ensure that sensitive data is not breached. Enabling all retail partners to share a single connection, or service, allowed this financial services organization to fulfill the tightest security requirements—and saved it a significant amount of money in the development, deployment and maintenance costs associated with supporting multiple connections.

Through the use of SOA, this financial services organization was able to eliminate conventional business silos and satisfy the most intensive security requirements. Furthermore, SOA allowed the automation of a cost- and time-intensive process, resulting in millions of dollars in savings and more satisfied customers.

Project 3
Document verification service: Delivering cost savings through service reuse

Economies of scale: it's the cost savings that large organizations enjoy by virtue of being large. It's based on the premise that many services can be provided to the second hundred of users for a fraction of the cost of providing them to the first hundred. Think of a bridge. It costs basically the same amount of money to build and maintain a bridge that will serve two cars an hour as one that will serve 30 cars an hour. On a per-car basis, the second bridge cost 15 times less. You can think of software services in much the same way. If an organization has five duplicate services, it is losing out on significant economies of scale. SOA can help organizations take advantage of economies of scale by providing standard methods of connection to centralized software services.

In this example, a government agency in the Asia-Pacific region was using a combination of manual and partially automated processes to verify documents such as passports, driver's licenses and birth certificates. These verification methods were both labor-intensive and lengthy. To eliminate this manual workflow, the agency decided to serve its citizens better and help reduce costs by providing a fully automated document-verification service.

After the agency successfully deployed this service, four other agencies throughout the government requested to use this automated function for their own document-verification purposes. The four agencies' systems were based on a variety of platforms and protocols. Realizing that this project could provide the foundation for similar centralized services, the government requested that the agency host the service so that it could be used by the other four organizations.

To accomplish the centralization of this document-verification service across all five government organizations, the agency needed to maintain the service interface used by its existing system, and also develop an interface that could be used across the other four systems. The service had to interoperate across three different protocols: SOAP over HTTP, SOAP over JMS and SOAP over WebSphere MQ. To make this interoperability possible, the existing service required another layer of abstraction to support the protocol transformations and to help ensure the service's loose coupling. The agency also needed to be able to guarantee a valid response to possible exceptions. Instead of just returning an error in the event of an exception, the service had to be able to handle the error response, and, in an asynchronous manner, resubmit the request until a valid response could be processed.

To solve these issues, the agency implemented an ESB to serve as the gateway to this existing document-verification service. The ESB provides the layer of abstraction necessary to satisfy the agency's defined requirements. The product used to implement the ESB provides the SOAP over HTTP and SOAP over JMS protocol transformation capabilities and the means to correctly route messages to the different organizations without requiring any additional development. Using the packaged functionality the product provided enabled the government to help reduce its investment costs by avoiding the need to develop this functionality in-house.

To support the SOAP over WebSphere MQ transformation and assured-response requirements, the agency developed mediations deployed within the ESB that handle these needs programmatically. The SOAP over WebSphere MQ mediation supports the protocol-transformation capability used to interoperate with this protocol. The guaranteed-response mediation intercepts any error response that is returned by the service and resubmits the request at a given time interval until the document-verification service can return a valid response. Then the response is forwarded to the original requestor.

Through the use of SOA, this Asia-Pacific government was able to quickly deploy an infrastructure to host services in a centralized fashion—within only five weeks. The ESB functionality provides the government with the flexibility to integrate different technologies and protocols so that components can be reused. Also, the infrastructure that was built to support the ESB can easily scale to host other centralized services besides document verification. By adopting SOA to reuse service components, this government is helping reduce its IT costs and providing its citizens with faster, more-accurate customer service.

Project 4
E-commerce connectivity service: Selling through partner Web sites to increase sales
Developing strong trading-partner relationships has always been at the forefront of business. No organization can be expert at all things; nor can many organizations have access to all the customers who might be interested in their products. Throughout the history of business, organizations have teamed up with partners who can enhance the quality of their products and services—without competing for the same customers.

Although partnering has been standard business practice for many years, the difference today is that companies are more likely to allow partners to perform high-value customer-facing activities. These companies are increasingly using business partners in the service of their own customers. This type of partnering can be tricky: it only works when the handoffs between partners are seamless. SOA can help organizations take advantage of the services offered by business partners in a way that is transparent to their customers.

This example focuses on a company that sells upscale gift merchandise. The company has its own sales channel, but knows that to find a larger number of potential customers, it must also take advantage of third-party sales channels. In other words, the company is willing to pay a commission to partners for bringing customers to its doorstep. In this case, however, the doorstep is a partner's online retail site.

The challenge for the gift merchandiser is to seamlessly connect its catalog, inventory-management and order-fulfillment systems to its partners' front-end Web sites. The process the company had been using involved fulfilling orders that came from its partners' sites in a batch fashion. This archaic method made accurate inventory management on partner sites difficult, and delayed the order-fulfillment process—which had a negative impact on customers' shopping experiences. Adding to the complexity of the issue was the fact that the partner sites were deployed on different platforms, such as Java, COM and Microsoft® .Net, making the task of business-to-business (B2B) integration even more challenging. Hard-coding the connections would have been slow and expensive, and would have led to ongoing problems as interfaces changed and applications were updated. Even more important, hard-coding would have limited the number of partners the company could do business with, and would have delayed each partnering relationship by several months.

The company's vision was to expose the real-time capabilities of its order-fulfillment system by providing an interface that allowed partnering sites to report more-accurate inventory levels during the ordering process and to convert the overall order process from a batch to real-time type. This would improve customers' shopping experiences and strengthen the relationship with the partnering sites.

To accomplish this, the merchandiser provided its partners with a series of SOAP interfaces. These services tie into the order-processing application through an existing WebSphere MQ link. By taking this approach, the company provides a loose coupling of components that abstract the partnering sites' platforms, because Java, COM and .Net all have the capability to communicate using SOAP over HTTP. To integrate the services with the WebSphere MQ technology, the company transforms requests from SOAP to WebSphere MQ and then performs a WebSphere MQ to SOAP transformation for the response. By providing the means to perform the necessary transformations for this protocol interoperability, the company can reuse its existing informational components while still maintaining the loose coupling of the components. This loose coupling also enables future reuse of components as the company improves both its internal and external integration capabilities.

This project exemplifies one of the central themes of SOA—that reuse can be greatly increased by using standards-based interfaces to hide the complexity of existing systems. In many cases, companies do not need to redevelop what they already have, but can instead provide loosely coupled interfaces that mask the underlying complexity. These interfaces can be easily connected to and promote greater reuse throughout the organization. Through SOA, this gift merchandiser was able to take advantage of its existing assets and set the stage for quicker and more-productive application development and integration in the future.

Project 5

Criminal justice service: Building an enterprise SOA using CICS systems

Why should an organization port application functionality to a new technology when the current system hosting this application still provides the basic capability? Many firms believe that the underlying technology in their existing systems is the most efficient way to host certain applications. This situation especially holds true for applications that require high transactional volumes like IBM CICS® applications. However, you can realize measurable business value by extending the use of these applications to create new solutions and apply new technologies—particularly if you can achieve this business value quickly using an approach that minimizes your risk of change.

Thousands of organizations run CICS mainframe technology. These systems have earned their reputation as transactional work horses and are estimated to handle US$1 trillion in transactions every day. Although many organizations that have implemented CICS technology wouldn't think of replacing it, they would readily consider how they can use this valuable CICS functionality across other enterprise systems.

For this project, a government agency in North America recognized a critical need to update the technology used to access its criminal justice system. This system provides authorized officials with important legal information to track defendants, including vital details about arrests, bookings and probations. The agency realized that the underlying CICS system supported its functional needs, but that the user interface terminals used to access this information were too difficult to operate and did not support the required accessibility to packaged software commonly used by legal professionals.

The organization made a decision to retire these terminals and redesign the interaction model using a portal-based Internet interface. This solution provides easier access to a much broader audience, helping this government agency to significantly reduce its training costs and enabling it to support a growing number of users. Also, professionals can use this new portal interface in conjunction with common industry software, which allows them to work more efficiently. The agency can continue to take advantage of the existing functionality of the CICS system, and is therefore spared the time and expense of rewriting this logic on another platform.

To provide the necessary interface, this organization used the SOAP for CICS technology. SOAP for CICS enables the government agency to host SOAP interfaces directly on IBM CICS Transaction Server. Through these interfaces, the entire enterprise can gain access to the functionality hosted on these critical CICS systems. To help define these SOAP interfaces, the COBOL interface was cast into a Web Services Description Language (WSDL) definition using tools provided with CICS Transaction Server. After these WSDL definitions were established, the government agency used them to create a Java technology-based SOAP proxy that can be called within the agency's Internet portal solution.

The project delivered high mutual benefit to both the agency and its users. The agency has magnified the value derived from the existing applications by extending their usage. Legal professionals have benefited from improved access to criminal justice information on the CICS system, and from the ability to have a single, integrated view of information from both the CICS system and other sources.

One of the key ideas behind SOA is to reuse existing components and avoid developing functionality that you already have. SOA provides the ability for existing systems to quickly become accessible throughout the enterprise. By masking the underlying system complexity through standard interfaces, many of these systems now have the ability to easily communicate with the latest technologies. SOAP for CICS technology provides these interfaces, allowing the entire enterprise to gain access to important CICS functionality. This capability translates to increased flexibility and greater economies of scale, while decreasing development effort and cost.

Make your move

As these projects have shown, SOA can solve immediate business problems while simultaneously laying the ground-work for flexible IT that is capable of adapting to quickly changing business conditions. Although these solutions were implemented by specific organizations to address specific business needs, the basic concepts are universal. Concepts such as reuse of existing business functions, improvement of business processes, easier application connectivity using open standards, and separation of back- and front-end systems —all are relevant to virtually every enterprise that uses IT. As you think about these projects and the benefits they were able to deliver, you should also think about how your company can use SOA to solve similar types of business problems.

The future success of IT will be based on how easily services can be pieced together to quickly create new business solutions. SOA, backed by robust standards, technologies and best practices, will enable this success. Just as the companies discussed here used a service-oriented approach to solve nagging business problems, so too can your company take advantage of SOA to improve short-term business performance, while establishing an IT framework that allows your business to be competitive for many years to come.

For more information

To learn more about how service oriented architecture can help your business, contact your IBM representative or IBM Business Partner, or visit:

ibm.com/soa

IBM®

G224-7538-02