# Unlocking the Business Value in Mainframes

BY JOSHUA FOX

Mainframe systems hold the IT power that drives the largest enterprises, yet their tremendous potential often remains locked up in siloed systems. CIOs and IT executives realize the need for agility and reuse in today's environment. They must satisfy new regulatory and business requirements for reporting, security, privacy, straight-through processing, and Business-to-Business (B2B) integration. Yet there's neither time nor need to redevelop core functionality now supported by mainframes. Instead, executives have been undertaking Service-Oriented Architecture (SOA) projects to expose their mainframe functionality as services: simple, loosely coupled interfaces that reflect business needs.

To achieve this, architects and analysts must align IT data with the business by mapping technically oriented legacy systems to the organizational requirements they serve. Automated systems significantly accelerate the work of the analysts and architects. We'll describe approaches to automating the business and technical analysis of mainframe systems, reducing costs and time-to-market. These include search engines and automated classification systems specialized on mainframe metadata, as well as manual mapping tools and tagging systems.

With the mappings in place, a company can rapidly move ahead to wrapping its mainframes in services, modernizing business processes, and making the mainframe an even greater asset for driving down costs, increasing agility, and capitalizing on new business opportunities.

### Legacy Transformation

CIOs and IT executives are working to uncover the tremendous value locked in their mainframe systems: strategic "Big Iron" applications on IMS, CICS, and DB2 running on operating systems such as z/OS and Linux on System z.

Mainframes have provided the infrastructure for IT-driven business successes in large enterprises; they power the management of supply chains, customer relationships, human capital, and business processes.

The time has come to open up the mainframe to new opportunities for cost reduction and revenue enhancement, once the functionality can be leveraged for reuse outside the silo of the application for which it was developed. So, for example, if a mainframe provides inventory information in a siloed Customer Relationship Management (CRM) application, it should also flexibly and quickly provide the same information for a modern CRM system, or even for a B2B integration.

But you can't reuse something you don't understand.

When mainframe systems were first developed, their focus was strictly in technical processing, with little concern for business-oriented rapid reuse. In the early days, software had to execute close to the metal for maximum performance, with a minimum of abstraction layers to hide low-level detail. Code had to fit into limited memory and disk space, which meant identifiers were short and cryptic. Code was tightly coupled and fine-grained; each unit of code addressed only a narrow area of technical functionality. Comments were lacking, and manuals were often in hard copy. Often, there was no clear record of the connection between these mainframe IT systems and the business requirements they addressed.

Over the years, vendors have added extensive support for business-oriented interfaces, clear identifiers, and machine-readable documentation. Nonetheless, many of the original limitations still cloud the meaning of older systems.

The full power of the mainframe emerges when it is leveraged for new business goals in SOA.

An SOA is built of loosely coupled, coarse-grained, business-oriented services that add flexibility and accelerate reuse of the existing code. A single CICS transaction might, for example, calculate the income tax on monthly salaries, and use hard-to-understand technical identifiers such as SALBNDX4. Such code is difficult for architects to find and reuse. On the other hand, an SOA would expose a comprehensible, easily reusable employee compensation service. To achieve this, architects bundle up code from multiple mainframe transactions and data structures into larger units of business functionality, exposed in modern, standard protocols with fully specified, natural-language identifiers such as salaryBand, compensationPackage, etc.

The CICS income-tax calculation would be aggregated

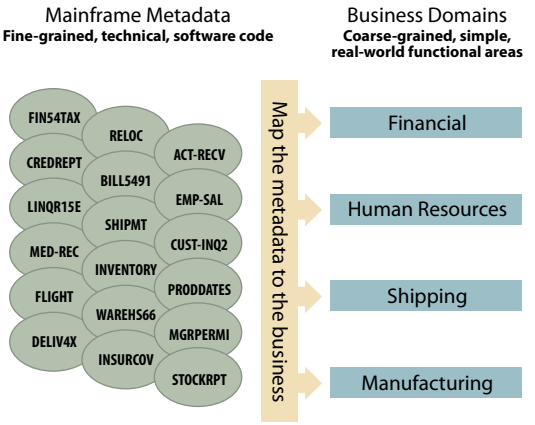


**Figure 1: Mapping Mainframe Metadata to Business Domains**

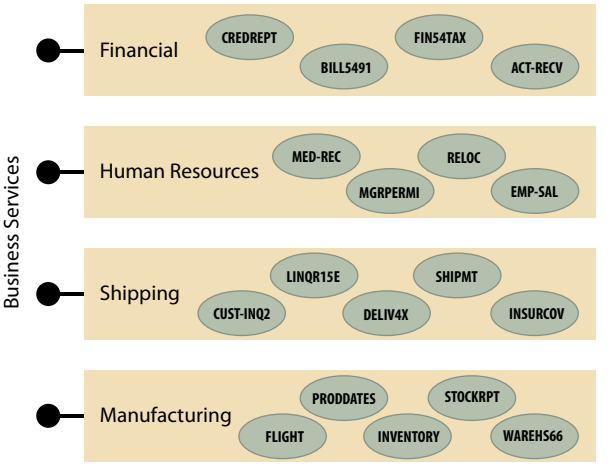


**Figure 2: Wrapping Mainframe Systems With Business Services**

with dozens of other transactions in the area of employee compensation; the legacy code would be wrapped in XML Web services running on application servers such as WebSphere. Java or .NET Web applications then readily provide a front-end, reusing the power of the mainframe for new applications. With agile application development, executives can rapidly get the information they need for business decisions from new, Web-based integrated applications.

### Understanding Your Mainframe Systems

SOA projects begin by understanding the business. For example, new labor regulations imposed on a *Fortune* 100 company meant that the CIO needed to deliver an updated human capital management system for rapid compliance.

The existing human resources functionality worked well, but only in its original environment, distributed among a variety of isolated CICS transactions supported by COBOL code and DB2 databases on z/OS. This functionality wasn't suited to the new legal environment. There was no way to change the fragile legacy codebase, which processed tens of thousands of mission-critical transactions a day, especially since it was unclear which systems might be impacted by any change.

It also was clearly not cost-effective to recode these applications from scratch in scalable, robust, and secure Java or .NET applications, while meeting both old and new requirements.

The most efficient path forward was to wrap SOAP services over the relevant legacy behavior and data in the human resources domain, selected from more than 60,000 units of code in their enterprise.

The methodology for such legacy transformation projects starts with a technical investigation phase, when architects gather metadata: IT data that characterizes information systems. This may come in many forms, including COBOL copybooks, DB2 schemas, CICS transaction specifications, or simply human-readable documentation. Architects load the metadata into a repository for convenient access and searching.

Simultaneously, business analysts thoroughly investigate the requirements and capture the business concepts in a semantic model, in the form of an ontology, taxonomy, or business glossary. The model includes broad functional domains such as human resources, manufacturing, and sales, as well as narrower domains such as new employee training and regional sales reporting. The model also includes key business entities, such as employee and shipment. Business analysts generally start with industry standard models, often related to standard interchange formats, including Insurance Application Architecture (IAA) and Association for Cooperative Operations Research and Development (ACORD) in insurance, HR-XML in human resources, or Health Level Seven (HL7) in healthcare. They then refine the model to the needs of the organization based on an in-depth understanding of their own business.

The next step is to tie the technical metadata and semantic model together, determining the business concepts served by each transaction or data structure (see Figure 1).

The mapping process is expensive; only experts who understand both IT and the business can do it. Moreover, the experts must understand, in detail, a variety of highly specialized mainframe-based technologies, as well as semantic modeling methodologies such as taxonomies, ontologies, or glossaries.

Although senior programmers understand their mainframe systems, few have the breadth and depth of knowledge to analyze metadata from multiple underlying technologies. Business analysts who know the ins and outs of the organization are rare, too. People who combine these

two skill sets are hard to find and expensive when found.

Moreover, even the most highly paid consultant can't accurately and reliably understand the technical meaning and business context of tens of thousands of IT artifacts.

Next, at the development phase, analysts and architects identify the legacy code in their functional area, and then wrap it into coarse-grained, business-oriented services (see Figure 2). Their work is enabled by the clear understanding captured by the technical analysis, the business analysis, and the mapping between them. In the case previously described, they found COBOL code from the human resources business domain, then drilled down to the copybooks that specify the needed entities (employee, role, compensation, etc.).

Finally, the development organization layers XML Web services on top of the mainframe systems. With easy-to-use, business-oriented interfaces, it is straightforward for the development team to tie together their previously isolated functionality into integrated systems that comply with changing regulations and business requirements.

### Mining for Meaning

This methodology is highly effective in leveraging mainframe code for flexible reuse. However, as mentioned, only experts can create the IT-business mappings by hand. Automated mapping tools can accelerate their work and drive down labor costs by helping analysts and architects find what they need and capture the results of their work in a reusable manner.

Search engines are the most basic type of automated metadata analysis system. Whether running against a centralized IT repository or connecting to multiple sources, a search engine helps analysts and architects find the IT artifacts relevant to their business requirements. A basic search engine lets analysts search for relevant strings in the metadata, finding, for example, EMPLOYEE when looking for data sources that hold human resources data. More advanced tools can search for similar strings, finding EMPL and EMPLOYEES on a search for EMPLOYEE. The most sophisticated search engines can recognize semantic similarity, finding, for example, EXECUTIVE when the search is for CIO.

Specialized functionality for abbreviated, cryptic, technical software code is generally required; most Commercial Off-The-Shelf (COTS) search tools are optimized for human-readable texts, which is suitable for manuals and other documentation.

Simply searching for IT data with a given business purpose is not enough; it is essential to persist the associations between metadata and a standard model of business domains and entities. Semantic mapping tools can accomplish this. They let analysts capture mappings from artifacts such as database schemas or COBOL copybooks to business domains; and

from the data structures within them, such as relational tables or COBOL element groups, to business entities. With the semantic mappings in place, it is easy to identify the bundles of low-level technical functionality that together constitute a business service.

Web 2.0 has introduced tagging, in which large numbers of users freely assign labels to data. "The wisdom of crowds" often captures knowledge unavailable to more formal methods. Mainframe metadata doesn't generally have a large user community. Still, tagging can be useful in uncovering knowledge from large numbers of departmental developers, data managers, and administrators who lack the skills for IT-business mapping, but who know their systems. Skilled experts can then use the tags as guides in assigning the metadata to business concepts. Automated mapping technology also can give a boost to the experts.

The simplest such technology uses a dictionary, finding mainframe metadata with keywords known to be relevant to a business domain or entity. For example, PRODUCTION might be a keyword for the manufacturing domain, and DELIVERY for the shipment entity.

However, mainframe metadata is often so abbreviated and non-standard that known keywords are unavailable. An automated classification system can learn the actual terms associated with each business domain or entity. Just as a spam filter identifies the words most characteristic of spam and of good email, an automated metadata classifier learns, for example, that the word SALARY, EMPL, EMPLOYEE, EXECUTIVE, and MANAGER often appear in the human resources domain, but less commonly in CRM. In fact, if a cryptic code such as QXR758 was an internal standard meaning "customer," the machine-learning engine will identify that, too.

A skilled expert always reviews the automated mappings. The best analysts have deep domain knowledge not found in any model and not accessible by any tool, and only they can interview domain experts and technical specialists to understand the business concepts supported by mainframe systems. Likewise, top architects understand their mainframes as no software can. But with various tools to help in searching and mapping, experts can more efficiently sort out the relevant concepts in their IT.

Even partial automation has value. An Internet search engine doesn't guarantee that the page you're looking for will appear at the top of the results, yet you couldn't navigate the Web without one. A spam filter doesn't catch every bad message, yet email would be unusable if all the spam reached your inbox, and anomaly detection software doesn't detect every fraudulent credit transaction, yet such software has greatly increased the integrity of our financial networks.

These automated tools can cut expenses and accelerate time-to-market in the analysis phase of SOA transformation projects. First, they easily identify the bulk of mainframe metadata, which is too technical and low level to be exposed in an SOA business service. Next, they capture the meaning of metadata with standard identifiers and comments, saving the expense of repetitive human effort. Since older standards are often built of abbreviated, cryptic, and repetitive terms, automated tools excel at learning and detecting them; a human, on the other hand, would need to constantly refer to documentation to understand their meaning. Last, automated tools tease out the hard-to-detect patterns in non-standard, heterogeneous, and technically specified mainframe metadata, providing a basis for humans to continue their analysis.

Machines are tireless. They can quickly review tens of thousands of mainframe IT artifacts, leaving only the harder cases requiring deep domain-specific knowledge to the highly paid experts. As the project grows to cover mainframe systems across the organization, the machines are always available to repeat and expand the analysis.

Machines make mistakes, but so do humans. The most accurate, most efficient results come from leveraging the best of both. Machines are good at churning through massive amounts of data. Humans are good at learning and understanding the real world.

Once the analysts have captured the business meaning of the mainframe metadata, it's easy to identify the mainframe systems that should be bundled into comprehensible, reusable, business-oriented Web services.

## Conclusion

Mainframe systems provide the engines behind key business processes in the largest organizations. With new regulatory and business requirements demanding agility and flexibility, CIOs and IT executives have a mandate to open up the tremendous potential now locked up in isolated mainframe applications.

To do this, IT organizations must capture the business context of their mainframe functionality. Architects and analysts gather and analyze their mainframe metadata, model the business, and then map the two together.

The skills for this process are expensive since both technical and business knowledge are needed. However, recent improvements in data mining, search, and pattern recognition aid the experts in mapping between IT and the business, significantly improving the efficiency of their work.

Architects and analysts, supported by automated tooling, capture the business value in technical metadata, opening up the power of the mainframe for rapid integration inside the organization and beyond. By helping an enterprise modernize its mainframe-based business processes, SOA transformation makes the mainframe an even greater asset for driving down costs, increasing agility, and extracting maximum value from new business opportunities. ME

**Joshua Fox** is chief technologist for Metadata Analytics at IBM. Previously, he was principal architect and director at Mercury Interactive and chief architect of Unicorn Solutions. He is a frequent author and lecturer in the fields of enterprise architecture and software engineering.
Email: joshuaf@il.ibm.com
Website: http://joshuafox.com