THE POSSIBILITIES ARE INFINITE FUJITSU

WHITE PAPER

Service-oriented Architecture with BS2000/OSD

Issue April 2009

Pages 6

Introduction

BS2000 applications are deployed to handle core processes in industrial and commercial organizations and public authorities wherever maximum efficiency and reliability are paramount.
These applications run so smoothly it can easily be forgotten what assets are contained in them in terms of information and services.
The application scenarios in which these services are actually used often make up only a fraction of the scenarios in which they could be used extremely productively even outside of their previous application context.
Often, too, these services are isolated IT islands. In many cases the efficiency of working and business processes can be increased merely by using IT techniques to link together the functionality of such IT islands.

With the aid of a service-oriented architecture (SOA), the functionality of proven BS2000 applications can be reused in a flexible, quick and cost-effective manner to optimize online collaboration between employees and business partners, to automate business processes to a significantly deeper degree, and to ensure that business processes and IT services interact seamlessly.

This white paper describes the opportunities presented by SOA projects and how such projects can be realized successfully and without risk.

Contents

Problems facing today's IT

In the past flexibility came at a high price

The desire for greater flexibility in IT comes to the fore whenever the current requirements facing IT on the one hand and the historically evolved application environment on the other hand no longer fit together in an optimal manner.
And this inevitably happens time and time again, because working processes, the market situation or the corporate structure change, because new statutory regulations have to be taken into account, because technologies and cost structures evolve, or because automating processes requires previously independent applications to be integrated. The list of such causes can be extended more or less at will.

While it is true that in most cases this change in the challenges facing IT takes place only in small increments, the effort necessary in order to adapt the applications to the changed requirements is often considerable. Thus, although the number of lines of code to be changed is usually small, the risk involved in such change interventions can be very high and the new production and testing of the modified applications are therefore disproportionately costly and time-consuming.

The high costs of adapting the existing IT infrastructure to new requirements all too easily led in the past to a way of dealing with this problem that either produced damaging side-effects or postponed necessary changes for too long.

Flexibility without a suitable IT architecture makes the IT infrastructure unnecessarily complex and inefficient

To keep the change overhead to a minimum, attempts were often made in the past to leave the existing applications untouched as far as possible, and to implement the necessary adaptations by means of add-ons in pre- and post-handling stages. This approach comes at a price, however. Such pre- and post-handling stages are props which cause the IT infrastructure to become more and more complex over time. This leads to inefficiency, excessive consumption of resources and long response times, and makes the IT infrastructure increasingly difficult to manage.

Doing without a flexible IT infrastructure means sacrificing business opportunities

But it is also no solution simply to defer making adaptations to meet new requirements or using new capabilities. IT and processes will then not fit optimally together.

An IT infrastructure inadequately tuned to the needs of the working and business processes necessarily requires an additional investment of effort, because adapting the IT to the processes must then be carried out by additional working steps outside of the IT. This lengthens process times, pushes up costs, increases susceptibility to error and compromises process efficiency, thereby also jeopardizing competitiveness and customer satisfaction.
Furthermore, failing to innovate working and business processes in order to save the expense of the IT adaptations necessary for this impedes the exploitation of all business opportunities that rely on the modernization of the processes.

A sustainable solution to these problems is therefore only possible if flexibility can be successfully achieved at reasonable cost. Precisely this is what is promised by a service-oriented architecture.

Solution concepts for a flexible IT infrastructure

A service-oriented architecture essentially helps solve two problems:

Firstly, the aim is to make adapting the IT infrastructure to new requirements and technologies much easier, quicker, more cost-effective and less risky in the future than in the past.

Secondly, the IT infrastructure should optimally match the business and professional processes, and moreover in such a way that the IT processes are aligned to the business processes rather than the business processes having to be geared to the capabilities of the IT.

Good prospects of organizing efficient flexibility with a service-oriented architecture

What is special about SOA is not this objective, but that is offers a series of **new concepts** for arriving at a solution. What is also new is that many basic technologies necessary for its implementation are already available today on all platforms. The **Web Services** make technologies available which establish interoperability between the clients and services running on different hardware and system platforms. Building a service-oriented architecture today therefore requires considerably **fewer technical preparations** than was previously the case, when similar goals were pursued with the aid of interoperability technologies which were less powerful, such as DCE, or too complex and not so widespread, such as CORBA.

Orchestration of services instead of applications

A big disadvantage of today's IT environments is that any change to working and business processes involves both sides – end user departments and IT experts – because adapting the IT processes to new working and business processes necessitates changes to the applications. The aim is to avoid this disadvantage by means of one of the central new SOA concepts – the concept of the SOA services.
An SOA circumvents this problem by separating the actual business logic from the applications. SOA-compliant applications – called services – provide their specialized functionality in the form of services via standardized interfaces.

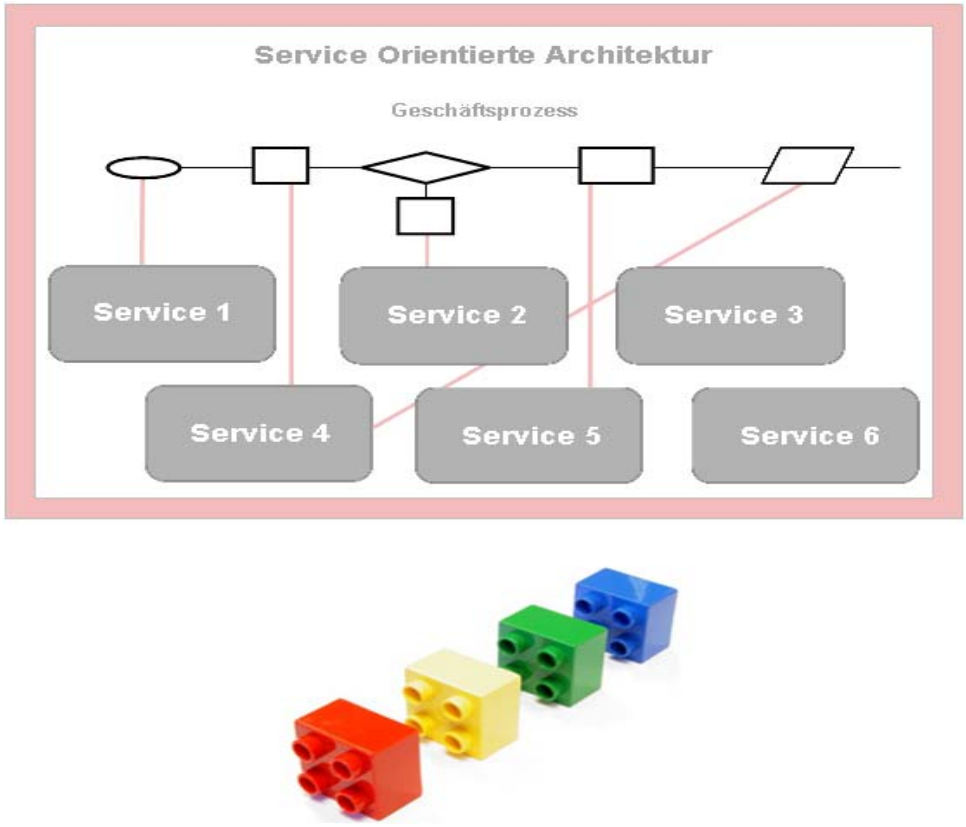

Figure 1

The actual business logic is then implemented only on the basis of a composition of such services outside of the applications. This composition is called an orchestration of the services and should be so simple that it can be accomplished in the organization by specialists who basically have a precise knowledge of the business logic, but no longer need to have any detailed knowledge about the applications.

Formal specification of the working and business processes

In information technology, formal specifications are the prerequisite for automated entities to be able to process the information. The most notable feature of a service-oriented architecture is that the specialized descriptions of the working and business processes are already clearly and formally specified. This requirement is a direct consequence of the two initially presented problems that an SOA is intended to solve:

- Simple, quick, cost-effective and risk-free flexibility is only achievable if, after changes have been made to processes, it can be established automatically which other processes are affected by the changes and whether their consistency has been preserved. This implies an automatic check to verify the correctness of the process specifications, and this is possible only if the specifications have been formalized and are therefore present in a checkable syntax.
- Deriving the information technology processes from the specialized descriptions of the processes is also only possible if the specialized descriptions are complete and clear and available in a formalized form.

Automatic mapping of working and business processes to the services

In the ideal case that would mean that the orchestration of the services could be derived automatically by tools from descriptions of the business logic. For example, the business processes could be formally described in the Business Process Execution Language (BPEL) and executed by means of a Business Rule Engine (BRE) via an Enterprise Service Bus (ESB).

Support for an SOA by BS2000/OSD

BS2000/OSD as a runtime platform for services

Services as part of a service-oriented architecture are the building blocks from which different working and business processes can be constructed in a flexible manner. This only works, however, if the systems on which these services run are able to provide the relevant resources flexibly and quickly in accordance with demand.

This imposes the highest demands on the robustness and scalability of the runtime platforms. BS2000/OSD is renowned for fulfilling these requirements in an exceptional manner.

- **Requirements for reuse of services**
  - APIs for all reusable interfaces
    BS2000 strength: API, not GUI, is the standard for mainframes
  - Easy encapsulation of existing interfaces using web technology
    BS2000 strength: client/server architecture based on web technology in BS2000
- **Consequences of service reuse**
  - Maintenance time windows are lost
    BS2000 strength: ideally suited to 24/7 operation
  - Load is less predictable / peak load becomes more frequent
    BS2000 strength: high scalability / highly available even when bottlenecks occur
- **Consequences of the new flexibility of the application environment**
  - Physical resources are no longer dedicated to a specific purpose
    BS2000 strength: virtualization of all resources
  - Resource management must extend across all servers
    BS2000 strength: cross-server resource sharing
  - Resource management must be automated
    BS2000 strength: this has always been a requirement for multiprogramming and batch processing

BS2000 applications provide a particularly promising starting point for establishing a service-oriented architecture.
One reason for this is that the applications running on BS2000/OSD are often the proven and stable core applications of business organizations and administrative agencies, i.e. the centralized IT functionality for handling working and business processes.
Another reason is that it is precisely these core applications that generally already possess many of the features that are demanded of SOA-compliant services:

- They are geared to ongoing, long-term evolutionary development
- Many of these applications use the interface concepts and formats standardized in the BS2000 environment by the SDF command language and the UTM transaction monitor. This not only guarantees high quality based on long-established, tried-and-tested interface technology; it also simplifies the encapsulation via standardized internet interfaces, since reusable encapsulation tools and features of the openSEAS middleware suite and other BS2000 products can be used.
- Because BS2000 applications address all server and peripheral resources only via system interfaces that are strongly abstracted from the physical characteristics of these resources, they are effectively independent of their physical properties. This high degree of virtualization affords BS2000 applications a very cost-effective flexibility.

For this reason all that is additionally required in many cases in order to transform such applications into services in the sense of a service-oriented architecture is to separate out the high-level business or process logic from the applications and to provide the external services of these applications in an SOA-compatible interface format.

Providing BS2000 application functionality as SOA services

Providing IT functions that are independent of the platform on which they run, independent of the programming language in which they are written, and independent of implementation details requires their interfaces to be mapped to standardized interface formats. In most cases internet standards will represent the interface technology of choice.

BS2000 already supports the provision of BS2000 application functions as web services today by means of an array of features and tools. The most important are briefly outlined below:

Providing subfunctions of existing UTM applications as web services

- WebServices for UTM (WS4UTM V1.0)
  WS4UTM can run on all platforms on which Java Tomcat and AXIS run.
  It simplifies access to UTM services for web service clients using SOAP.
- WebTransactions V7.0 (BizTA Studio)
  Logical objects, commonly known as business objects, are created from host applications; these objects can then be used e.g. as a web service in any form of business process integration.
- UTM-XML
  UTM-XML enables UTM applications to provide their data in XML format or to process XML input.
- Enterprise Java Beans (EJBs) with BeanConnect V2.0
  BeanConnect™ 2.0 connects applications based on a J-EE application server to applications of the openUTM (Fujitsu) and CICS (IBM) transaction systems.
- BeanConnect as web service encapsulated in Oracle BPEL PM
  BeanConnect enables UTM and CISC applications to be integrated into Oracle-BPEL-based SOA solutions.

Adapting the input and output data of COBOL programs to XML and vice versa

- Cobol2XML
  For SOA services implemented in COBOL, COBOL2XML facilitates the conversion of input data from XML format to COBOL

format, and output data from COBOL format to XML format. At design time, the tool automatically generates the conversion functions from COBOL Copy elements. At runtime, these functions tied to the application then handle the format conversions.

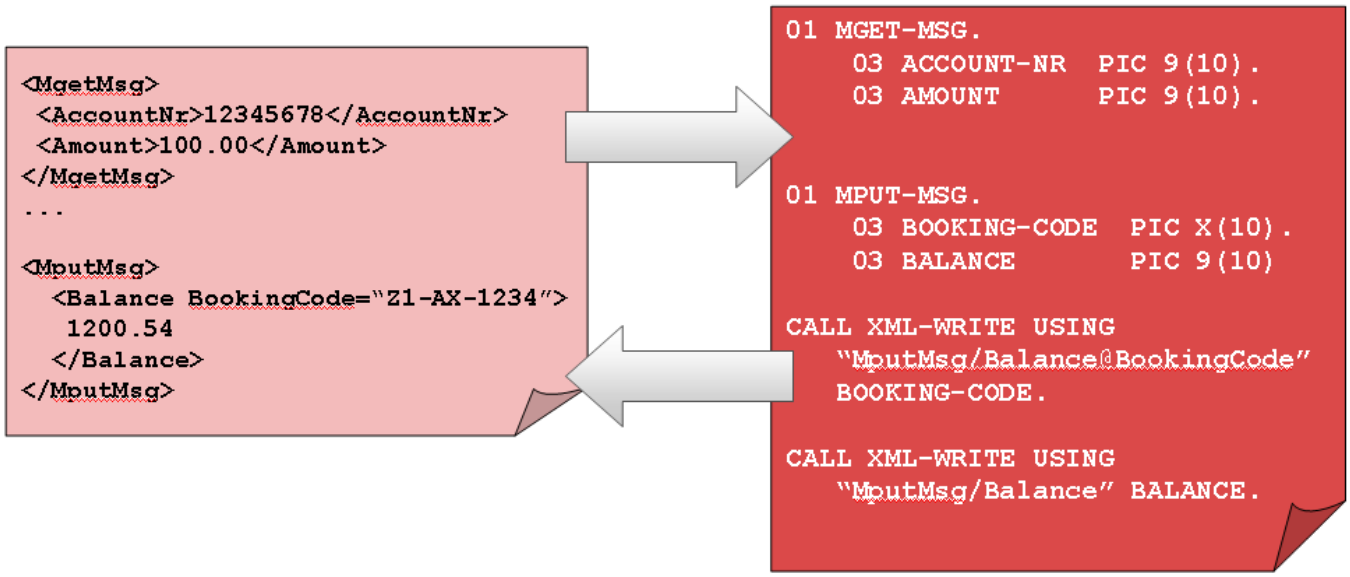

Figure 2

How can an SOA project be planned?

SOA starts with the organization's architecture

SOA is no out-and-out IT technical issue, but concerns end-user departments, IT and management in equal measure. The objective of optimizing working and business processes is actually only attainable if the organization – whether business enterprise, public authority or not-for-profit organization – is analyzed specifically with regard to the services and service areas organized within it.

The core processes of the service areas must be identified. The staff involved and affected must be made familiar with the paradigms of the service-oriented architecture. Potential areas for improvement based on role relocation, role centralization and role reallocation must also be identified.

SOA demands a clear and full description of the services

As soon as the processes and their interaction in the individual areas have been clarified, the specialized end-user requirements to be met by the individual services can be recorded and documented. SOA services should be redundancy-free, decoupled

Figure 2

from the implementation components, and capable of being orchestrated into business processes flexibly and on a demand-driven basis.

A formal specification of the services using modeling tools, e.g. in the Business Process Execution Language (BPEL), makes sense regardless of whether the processes are to be orchestrated automatically from this formal process description or not. One of the most important aims of an SOA is to ensure that the specialized description of a service is clear and complete and leaves no room for interpretation.

The specifically defined services must be mapped to IT services

Only once the services have been precisely specified in terms of end-user requirements are they mapped to IT technical services, assuming they are to be provided by the IT function. Usually this will mean adapting the existing IT application environment to the target architecture in accordance with the SOA paradigm.
In most cases the existing functions will, in essence, not be very far removed from the required service specifications, so a pragmatic adaptation of the existing IT to the requirements of an SOA service is possible.

The IT infrastructure for the IT services must be specified

After the service functionality and its implementation, the matching runtime environment for the SOA services must be specified, along with the matching server and server architectures, the matching storage concepts, and the matching IT infrastructure for fulfilling all the demands on availability, scalability and security.

The migration steps must be specified

Finally, the existing organization and IT architecture is modified in accordance with the designed target architecture. It is one of the main advantages of a service-oriented architecture that this **migration can be completed in small steps**.

This is an important prerequisite for successfully combining **investment protection and flexibility** on the basis of an SOA. The step-by-step approach enables a quick return in terms of benefits to be achieved from the innovations and makes it easier to correct decisions which prove to be less successful in real-world scenarios.

Service offerings for SOA

A pragmatic approach, at the outset of a migration to an SOA, is to investigate existing applications with regard to their potential to be made dynamic and at the same time to enable a suitable medium- and long-term architecture to be established. We conduct strategic dialogs with partners providing consulting services and know-how, as well as offering technical collaboration on projects, in order to optimize this phase. We also offer service packages for openSEAS which can be used when building a service-oriented architecture.

Summary

The concepts of service-oriented architecture have the potential to leverage the benefits of BS2000 applications by converting existing BS2000 applications into SOA services which provide their specialized end-user services via standardized web service interfaces.
Providing the services as web services guarantees interoperability in an arbitrarily heterogeneous IT infrastructure. At the same time the standardization of the interface technology facilitates the free and automatic composition of such services into complex working and business processes. The formal specification of the processes or the business logic permits the IT infrastructure to be quickly adapted to satisfy new requirements without the need for complicated and time-consuming interventions in the applications. The end result is that SOA establishes the basis for affordable flexibility.

BS2000 applications are particularly well-suited for integration into a service-oriented architecture – and not just because they are often the core applications that implement the vital core processes of a commercial enterprise or administrative agency. As applications running on BS2000/OSD, they also boast an array of technological features which make it particularly easy to integrate them into an SOA. All system resources (CPU, memory, disks, and tape and printer peripherals) that a BS2000 application draws on are virtualized to a high degree. This optimally decouples these applications from the physical characteristics of these resources and makes them independent when it comes to replacing and modernizing these resources. BS2000 resource management also masters extreme load peaks, supports huge numbers of concurrent users, and copes with the total failure of resources, thus guaranteeing optimum availability and scalability of these applications without the customer having to get involved.

The excellent customer proximity customary for BS2000/OSD ensures a quick response to customer requests during ongoing SOA projects and in conjunction with the SOA service portfolio makes the project risk easily manageable.

All this combined creates the best foundation for a quality leap in your procedures and business processes.

Published by department:

BS2000 Product marketing
Phone: ++49 89 3222 2612
Fax: ++49 89 3222 329 2612
BS2Marketing@ts.fujitsu.com
ts.fujitsu.com/bs2000

Partner login
partners.ts.fujitsu.com