Session: EM02

RU READY TO SAVE THE DAY

### Service Enable Your New and Existing IBM

System z Applications

WHERE TEAMS ARE

Russell Bonner Consulting IT Specialist, Rational Software bonnerr@uk.ibm.com

**IBM Rational Software Development Conference 2008** 







## Agenda

Why service orient your System z Applications?

Approaches to service orientation

Wrapping

Re-engineering

Redeveloping





### Who has seen this screen?

21 Character 1 (24 - 90)	
Pusti Deckeber, Och Kennediction Alikanan Frankris (Ilfr	
DIGIGI/BO3 QPADEVOOUS LIEFERANTEN	ERWALTUNG MUEVE 1.06.05
FA 1 NL Lieferant	6177 Renderg
	<u>INF NOT</u>
Anrede <u>4</u>	
Name <u>CS-Schmalmöbel GmbH &amp; Co</u> .	Kurzname <u>SCHMAL</u>
	StatiNr
	Verband-Nr.
Strasse / Nr.,	Art P
Postfach 1164	ILN-Nummer.
Land Plz <u>66900</u> Ort Wald	dmohr
Telefon 06373-1230	Telex
Telefax	Edifact
Unsere Kdnr. <u>875330</u>	Konv.Strafe
Bestell/Zahlnr	Liefertreue 00:00
Verband L	UStKennz. V
Zahl.art/Abw /	Ueb.weisG
AB-Antwortzeit 15	DLfz/BBZt. 23 J /
Fracht frei ab 1534	Sprache 001
Mind Bestellw.	Planumsatz.
Allo Hinweis	
Restallhinweis	
E2=Ident -Nr E3=Verlassen E4=Obbruch	E5=KD-Zeiten E24=Weitere
12-1dente Min 13-rel tassen. 14-hbbruch.	TO RO Letten 124-weitere
	04/010
FLI 1992. Steven and address and address to	04/018
(3), tans - pissing wurde ettoigreich gestertet.	

### Pros

- Works today!
- Existing staff is skilled and efficient
- Minimal work to maintain

### <u>Cons</u>

- Screen space is limited
- Text only
- Not easy to expose to new audiences (e.g., online check-in)
- High training costs for inexperienced users



## **Enterprise Modernization – Solution Overview**



 Leverage value in existing assets

 Drive innovation with SOA and web technology advancements

 Leverage existing and new staff on multiplatform projects

 Improve quality and flexibility with consolidated team infrastructure

✓ Reduce maintenance costs

http://www-306.ibm.com/software/info/developer/solutions/em/



### Modernize Architectures Flexible architectures to enable business agility



"SOAs cost 20% less to implement and saves 50% more with each reuse than traditional component-based development... the level of reuse in SOA development averages 2.5 times more than non-SOA development" Jeffrey Poulin, PH.D. and Alan Himler, MBA, 2006, "The ROI of SOA – Based on Traditional Component Reuse"



### **Enterprise services**

- Services provide standardized access to assets for different software applications residing on disparate platforms
- Service definitions provide abstract interfaces which allow for loose coupling between business components – implementation can vary without affecting consumers
- You can reuse applications exposed as Services in a variety of service-oriented architecture frameworks, such as a process choreographer or an enterprise service bus.





### Modern "System z" architecture



Best practice in System z application design is to separate key elements of the application, in particular:

- Presentation logic 3270, HTML, XML
- Integration or aggregation logic Menu, router, tooling
- Business logic COBOL, PL/I, Reusable component
- Data access logic VSAM, DB2, IMS, ...

Provides a framework for reuse and facilitates separation of concerns, clear interfaces, ownership, and optimisation



## Composite Workload Application Components

Spans multiple system and middleware boundaries



Session: EM02



### Modern Multi-tier architecture





#### Web Services UDDI Registry Architecture for Service Broker Application to application publish find Communication **WSDL** WSDL Browser **Browser** Interoperation SOAP SOAP Definition: bind, invoke Service Service **Provider** Web Services are software Requester components described via SOAP Web Client **WSDL** that are capable of being Service Application accessed via standard network protocols such as SOAP over HTTP WS-I.org (Web Services Interoperablity Organization)

Ensure interoperability

The entire industry is agreeing on one set of standards !!

INTEROPERABILITY ORGANIZATION







## Agenda



### Approaches to SOA

Wrapping Re-Engineering Redeveloping



### Three ways of re-using services from existing applications



Source: Gartner



## We need interfaces to talk "XML" ....





### Agenda



## Approaches to SOA Wrapping Re-Engineering Redeveloping



## Host Access Transformation Services (HATS)

### What can you do with HATS?

Quickly and easily create Web, portal or rich client applications that provide an easy-to-use GUI for your green-screen applications

Low skills requirement

Highly customizable

Iterative development process

Transformation "on the fly"

Extend terminal application tasks as Web services

### **Benefits**

Extend host application to new users

Improve the navigation of your host application

Reuse your existing assets in a Service Oriented Architecture

Avoid rewriting or reengineering host applications







#### Host Access Transformation Services Values **Rich Client** Mobile Web Services Portal Browser Integration at the desktop Zero footprint Integration at the glass Access host applications Build self-service with other Eclipse-based from mobile devices transactions Pure HTML Cooperative portlet support applications Expose host business JSR 168 compliant Access through your favorite Client side processing processes as Web browser, including Internet Services Rich set of user interface Explorer and Firefox. widgets Provide controlled access to vital host applications and Supports Lotus Expeditor host data. deployment

Quick ROI : Host applications can be quickly deployed with a new user interface

Low cost : No need to rewrite application

Low risk : Leverage open, proven platforms and technologies

Increase productivity and reduce training costs

Improving work flow from multiple applications

Provides integration of host business processes and data with other Web, portal, and rich client applications



## The Value of HATS SOA

#### HATS is an important part of the IBM SOA strategy!

IBM SOA Foundation helps to <u>extend</u> the value of the applications and business processes that currently run your business, <u>not replace</u> them.

#### HATS does not require application reengineering or rewriting

Rewriting can consume significant amounts of time and resources

Rewriting requires access to and understanding of application source code

Rewriting inherently introduces new bugs that must be found and fixed

Simply put - rewriting is not always a realistic option!

#### HATS is unique

HATS is the only IBM product that Web service-enables 3270, including CICS and IMS, 5250, and VT terminal applications – great for customers with a mix of platforms.

#### HATS generates standard Web Service interfaces

HATS services can be integrated into an existing SOA.



### Agenda



## Approaches to SOA

Wrapping

**Re-Engineering** 

Redeveloping



### IBM Rational Developer for System z V7.1

#### Develop EGL using RBD

- Quick and easy development of modern enterprise applications for business programmers
- Simplify and speed up creation of Web applications and services without having to learn Java or J2EE

#### **JES and PD Tools**

•Debug zOS applications from the workstation as they execute live in the remote runtime

•Read/Write/Update VSAM datasets via integration with IBM File Manager

•Access IBM Fault analyzer reports for analyzing ABENDS and associating back to source code

•Interact with the Job Entry Subsystem (JES) to submit jobs, monitor jobs, and review job output

### IBM Rational Developer for System z

zOS Host **Application Enterprise Tooling Development** Service Tools Integration [COBOL, PL/I, Web Services C/C++, JCL, [JES, FA, FM, For CICS/IMS] Screens, Stored **RBD** Debug Tool] Procedures, etc]

#### Host / Distributed SCM Integration

**IBM Rational Application Developer** 

#### Traditional Development

#### Development Environment

- Connect to z/OS systems
- Work with z/OS resources like COBOL, PL/I, C, C++, JCL, assembler, etc.
- Perform dataset management actions like allocating datasets and migrating datasets
- Perform typical edit, compile, and debug tasks on remote z/OS resources from the workstation
- Create, build, and catalog DB2 stored procedures on zOS
- Compile and test programs locally to ensure correctness

#### Screen design

 Visually create, modify, build, and deploy BMS maps sets or MFS/IMS maps remotely or on the local workstation

#### Code Generation

 Generate CRUD DB2 program code from UML, which can also be easily integrated into web service applications

#### **zOS Web Service and Flow Creation**

- Implements SOA and Web Services
- SOA access to CICS V3.X and IMS V9/V10 COBOL and PL/I applications
- Bottom-up/Top-down/meet-in-the-middle COBOL or PL/I to XML mapping support
- Integrated XML converters, XML schemas, and WSDL generation
- Service Flow Modeler to build/deploy service flows out of your existing Commarea, Channel, MQ, and Terminal CICS applications.

#### **SCM Support**

- Access to host SCMs such as SCLM
- Framework for writing/deploying custom SCM integration code
- Support for storing zOS resources in distributed SCMs such as ClearCase

#### Web and JEE Development

- Create Web Pages / JSF / Struts
- JEE/Java Development
- JCA Connectors
- Distributed debugger
- Web Services and Test environment

Session: EM02



Create web services for z	OS runtimes
Build, Deploy, and Test Web services from existin applications	19 Web Services for CICS - Create New Service Interface (b
Create source code skeletons from web service definitions	The language structures have been imported from the language source. Select the inbound and outbound language structures.
Map web service definitions to existing application modules	Inbound language structure Outbound language structure Select the language structure for the inbound XML converter
Supports traditional languages	₩-INPUT-DATE W-INPUT-DATE-INT W-INPUT-DATE-INT
COBOL	U W-CURRENT-DATE-INT
PL/I	
Supports zOS specific runtimes	E FC W-OUT-DATE
CICS	
IMS	Web Service Runtime and Scenario Selection       Specify options to start a web service wizard
Batch	Runtime: Web Services for CICS   Scenario: Create New Service Interface (bottom-up)   Conversion type: Compiled XML Conversion
	Scenario description:
	Generate a Web service description and runtime specific XML message processing from a high level language data structure. You can use this option when you expose an application program as a service provider.



### XML Services for the Enterprise (XSE) Example <u>Bottom-up</u>





Session: EM02



# XML Services for the Enterprise (XSE): Example Meet in the middle

### Meet in the middle



	web services for cica	*
Scenario:	Map an Existing Service Interface (meet-in-middle)	*
Conversion type:	Compiled XML Conversion	- 0
KML, XSD or DTD file KML message proce	es. You can use this option to generate runtime specific essing based on the mappings.	۹ T



### CICS usage of the WSBind file







### IMS SOA Solutions – Service Provider and Service Consumer





### Scenario – SOAP Client Invokes IMS COBOL Application as a Web Service





## Agenda



## Approaches to SOA Wrapping Re-Engineering Redeveloping



## **Application Modernization Options**





### XML Services for the Enterprise (XSE): Example Top-down New Project

### **Top-down**





Scenario description:

**Buntime** 

Scenario:



Session: EM02



## Orchestrate CICS services and screens

Model, Deploy, and Test Service Flows using Service Flow Modeler

- Aggregates multiple CICS transactions into high-level business processes through visual modeling
- Supports CICS BMS (terminal-based) applications & CICS commarea/container/channel applications

Highly optimized CICS runtime supporting Web services and XML interfaces







## What is a Service Flow?



#### **Business Level**

- CICS Service Flow Feature provides the capability to aggregate existing CICS applications into composed business services which may be integrated into an SOA environment
  - Aggregate multiple calls to CICS applications into one business level service call
  - Automate the interaction with 3270 terminal based applications and expose as a business level service

#### **Technical Level**

- A service flow is a non-interruptible micro-flow that is constructed from a collection of nodes that represent the invocation of CICS resources
- The flow describes the navigation of the nodes and allows data mapping between the nodes
- A single request may cause the execution of many CICS resources
- Allows for the development of coarse grained services from fine grained resources



### **Traditional Access**



Multiple requests from client application Expensive

Low potential for reuse



## Aggregated Access



Single request from client Potentially reusable component More efficient

Session: EM02



## How are Service Flows related to SOA?

### SOA gives :

The flexibility to treat elements of business processes and the underlying IT infrastructure as standardized components (services) that can be reused and combined to address changing business priorities

Services are the "building blocks" for business processes

Business processes can be offered as new services Business processes can be composed from services

A Service flow is the aggregation of CICS assets into a new business service That business service can be consumed by a business process













© Copyright IBM Corporation 2008. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. IbM shall not be materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or allering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature available in any way. IBM, the IBM logo, the on-demand business logo, Rational, the Rational logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service or both.

