Jumpstarting SOA Projects

**_From Reference Architectures, Assets and Best Practices
to Rapidly Deployed Solutions_**

Olaf Zimmermann
Executive IT Architect
Emerging Technologies Customer Innovation Team (CIT)
ozimmer@de.ibm.com

Abstract

In this presentation, we describe how reference architectures and other assets can jumpstart SOA development and integration projects.

Starting from the business-level rationale for SOA, we visit reference models such as the IBM On Demand Operating Environment. We then drill down one level and feature the main architectural templates for SOA, including key patterns such as Enterprise Service Bus (ESB) and Process Choreography (PC). We then apply the template to two recent SOA and Web services application examples from the finance and the telecommunications industry.

The presented examples include the BPEL enablement of the order management application of a major telecommunications wholesaler, and the Web services externalization approach followed by a shared service provider in the banking industry. We will discuss drivers for the projects as well as the the solution architectures, and share some of the project results and lessons learned.

Copyright © IBM Corporation 2005, unless indicated otherwise. All Rights Reserved.

This presentation is intended to assist IBM practitioners and IBM business partners in understanding IBM Software directions during 2005.

This presentation can be used in sales situations except individual charts labeled VENDOR CONFIDENTIAL or IBM CONFIDENTIAL, in which case they should be considered confidential under the practices in place in your firm and under any existing agreements with IBM regarding disclosure of confidential information.

For questions or to request permission for any other use of the information or distribution of the presentation, please contact any member of the IBM software sales team.

Agenda

- **SOA value proposition and definitions revisited**
- Candidate reference architectures for SOA
  - ▶ Service-Oriented Modeling and Architecture (SOMA)
  - ▶ On Demand Operating Environment (ODOE)
  - ▶ Key architectural templates and patterns
    - Enterprise Service Bus (ESB)
    - Business process choreography
- Project examples
  - ▶ Finance industry
  - ▶ Telecommunications industry
- Roadmap for project adoption
  - ▶ SOA/Web services adoption levels and entry points, IBM assets
  - ▶ „Perspectives on Web Services" text book
  - ▶ IBM jStart program

Agenda

- SOA value proposition and definitions revisited

- Candidate reference architectures for SOA
  - ▶ Service-Oriented Modeling and Architecture (SOMA)
  - ▶ On Demand Operating Environment (ODOE)
  - ▶ Key architectural templates and patterns
    - Enterprise Service Bus (ESB)
    - Business process choreography

- Project examples
  - ▶ Finance industry
  - ▶ Telecommunications industry

- Roadmap for project adoption
  - ▶ SOA/Web services adoption levels and entry points, IBM assets
  - ▶ „Perspectives on Web Services" text book
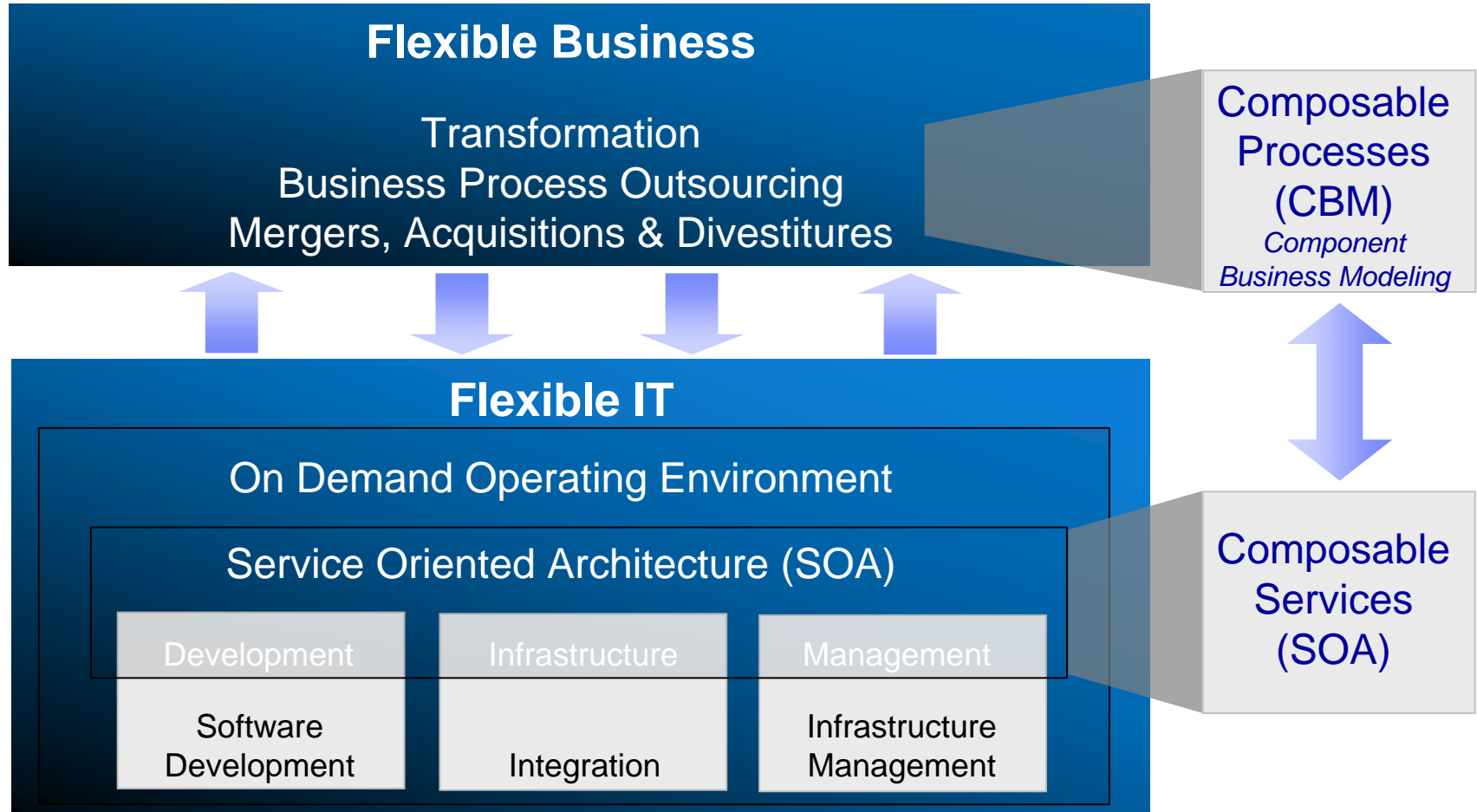 - ▶ IBM jStart program

Typical Status Quo in Many Enterprises IT Architectures

- Functional and technical **application monoliths** ubiquitous
  - ▸ Individual agendas dominate, leading to stovepipe architectures, application scope creep, redundant implementations, data management and many other agility issues
  - ▸ Architectural governance ... or at least guidance missing often

- Development and integration project **costly and long running**
  - ▸ Propretary point-to-point connections, often developed from scratch
  - ▸ File transfer is the most frequently used integration pattern (which has numerous architectural drawbacks)
  - ▸ Roll-your-own philosophy works short term, but leads to maintenance headaches

- As a result, horizontal, customer-focused initiatives are **much harder to implement** than they have to be!

Greater Flexibility Required From Business Models And The Supporting IT Architecture

Flexible Business

Transformation
Business Process Outsourcing
Mergers, Acquisitions & Divestitures

Composable
Processes
(CBM)
*Component
Business Modeling*

Flexible IT

On Demand Operating Environment

Service Oriented Architecture (SOA)

| Development	Infrastructure	Management
Software Development	Integration	Infrastructure Management

Composable
Services
(SOA)

A Service Oriented Architecture (SOA) is Key

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

Services are the building blocks

- Packaging business functions from new and existing applications in a simple and standardized way creates services that are available for use

- Services are used to help get the right information to the right people at the right time

- Services can be reused and combined to deploy composite applications to address new opportunities

- Increasing use of "Web" services based on open standards complements existing services technology

Reuse
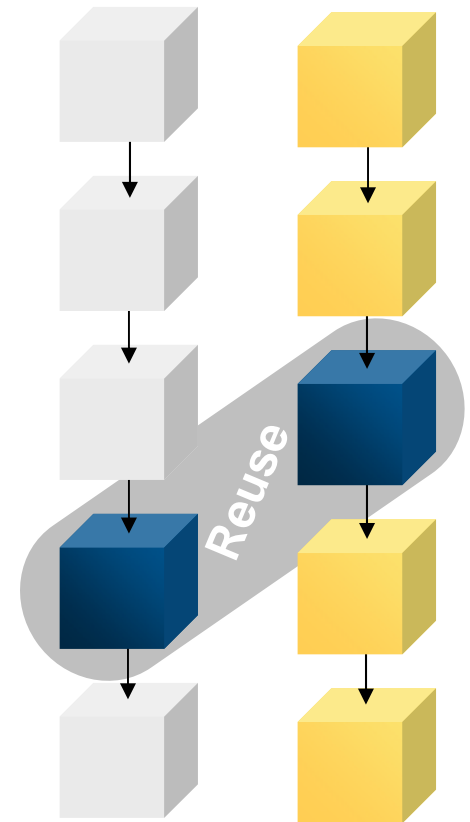
Now, what is a Service-Oriented Architecture (SOA)?

"SOA is different things to different people"

▶ a <u>set of services</u> that a business wants to expose to their customers and partners, or other portions of the organization

Business Executive, Consultant

▶ an <u>architectural style</u> which requires a service provider, requestor and a service description

▶ a <u>set of architectural principles, patterns</u> and criteria which address characteristics such as *modularity, encapsulation, loose coupling, separation of concerns, reuse, composability and single implementation*

Architect

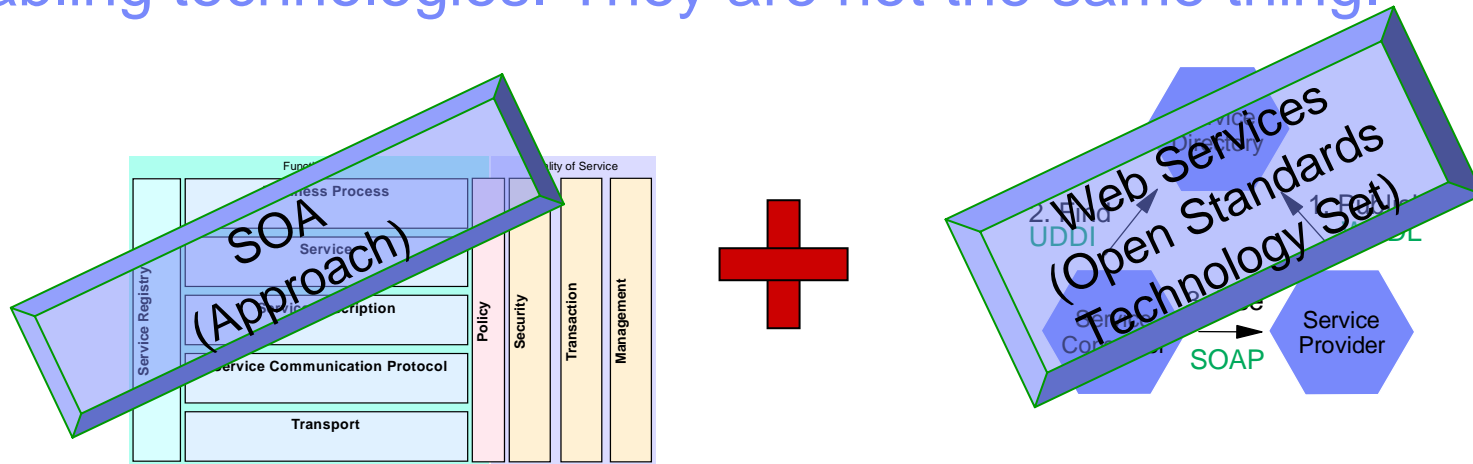▶ a <u>programming model</u> complete with standards, tools and technologies such as Web Services

Programmer

While SOA is an architectural approach, Web services are enabling technologies. They are not the same thing.

- ***Simplified definition: "SOA is an architectural style whose goal is to achieve loose coupling among interacting software agents"***

  - SOA proposes an advancement in the programming model

  - ▸ It is the next evolution in software engineering from Object Oriented Design & Component Based Development – but can be layered on top of these approaches

- Web Services and SOA are not the same thing:

  - ▸ Most of today's production SOAs don't primarily use Web Services – they are built on Message Oriented Middleware (MOM)

  - ▸ Not all deployed Web Services-based systems necessarily embrace all the guiding principles of SOA

Agenda

- SOA value proposition and definitions revisited
- **Candidate reference architectures for SOA**
  - ▶ **Service-Oriented Modeling and Architecture (SOMA)**
  - ▶ **On Demand Operating Environment (ODOE)**
  - ▶ **Key architectural templates and patterns**
    - **Enterprise Service Bus (ESB)**
    - **Business Process Choreography**
- Project examples
  - ▶ Finance industry
  - ▶ Telecommunications industry
  - ▶ Best practices for SOA and Web services
- Roadmap for project adoption
  - ▶ SOA/Web services adoption levels and entry points, IBM assets
  - ▶ „Perspectives on Web Services" text book
  - ▶ IBM jStart program

Layered SOA Reference Architecture and Service Modeling

An SOA is composed of multiple layers. At the heart of the SOA is the Service Model that defines Services and Components that realize them.
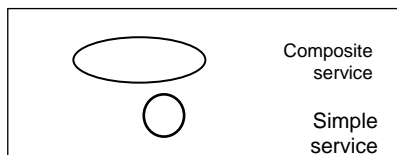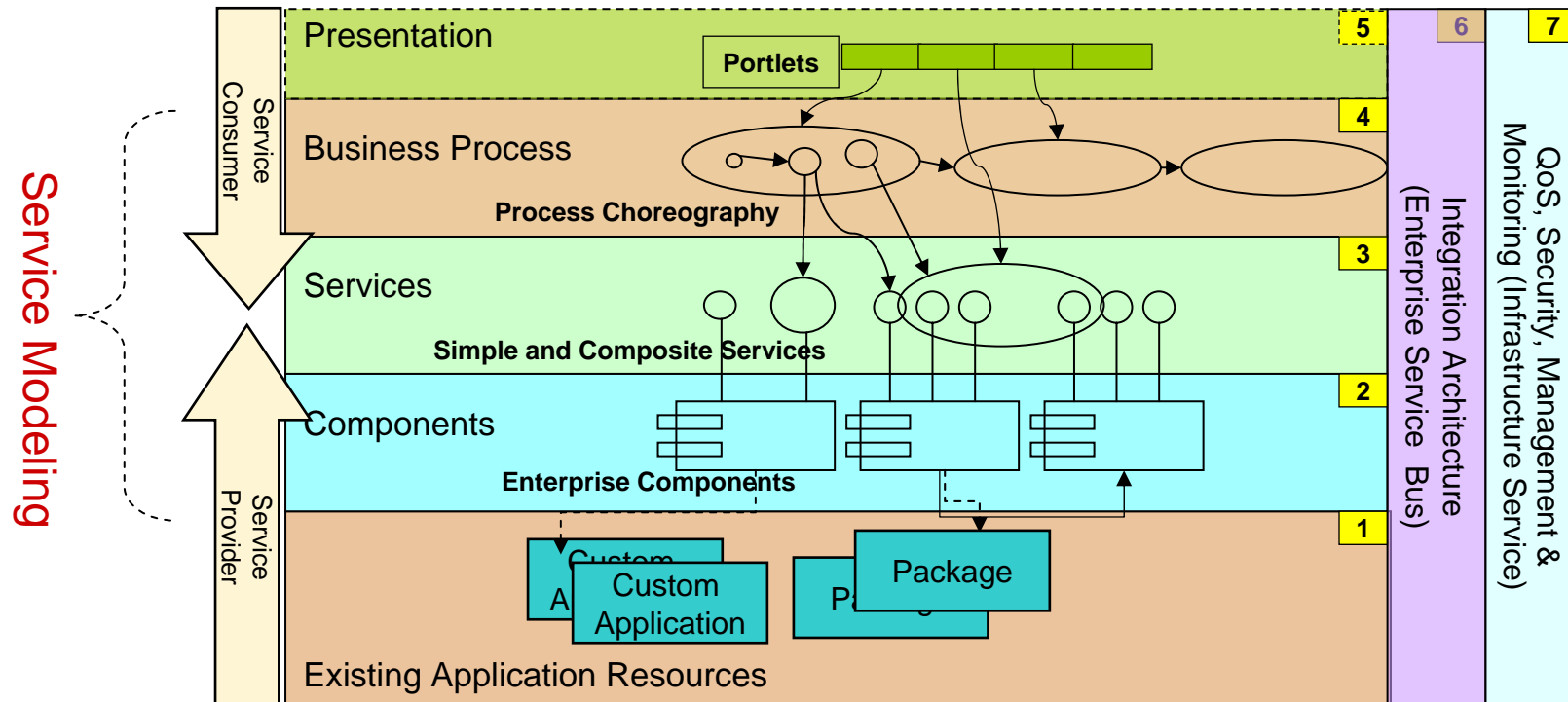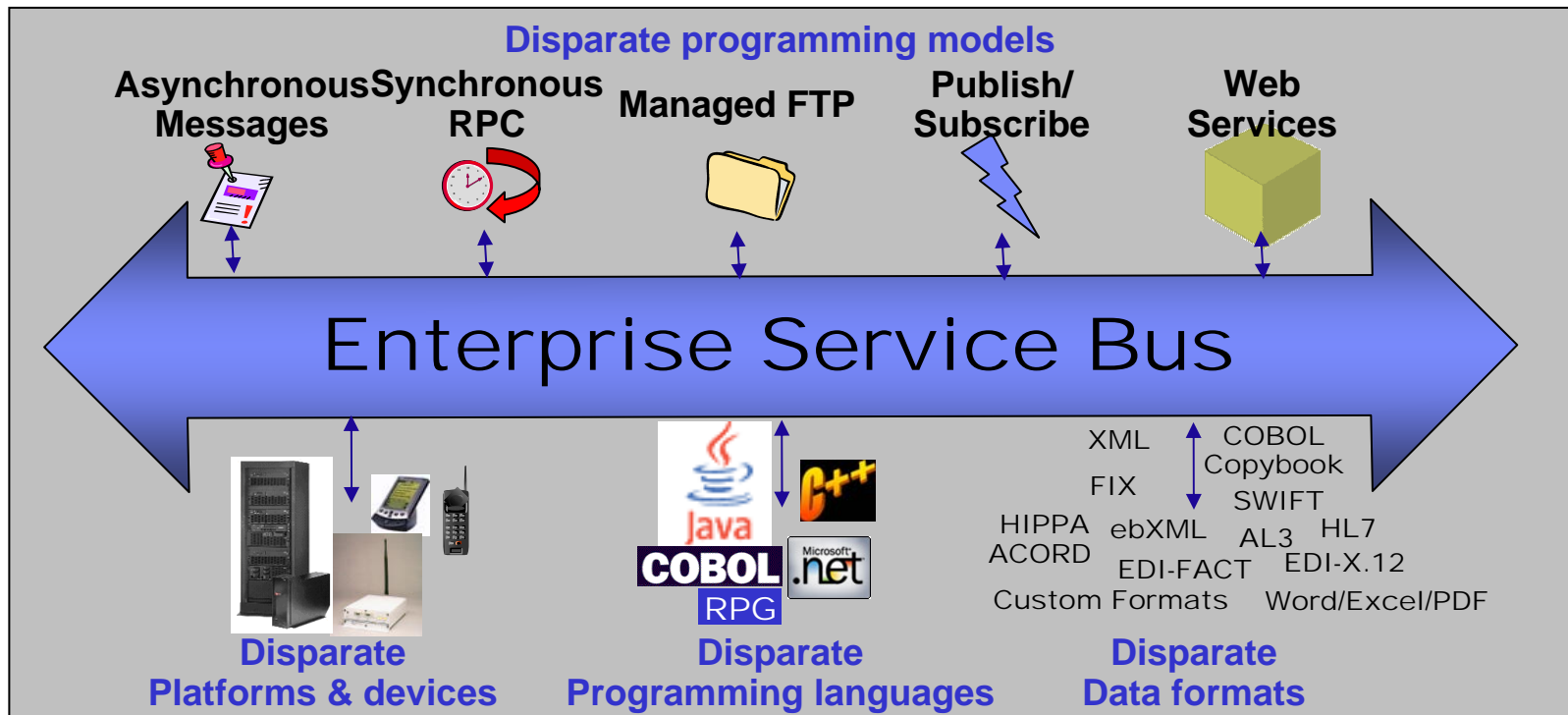
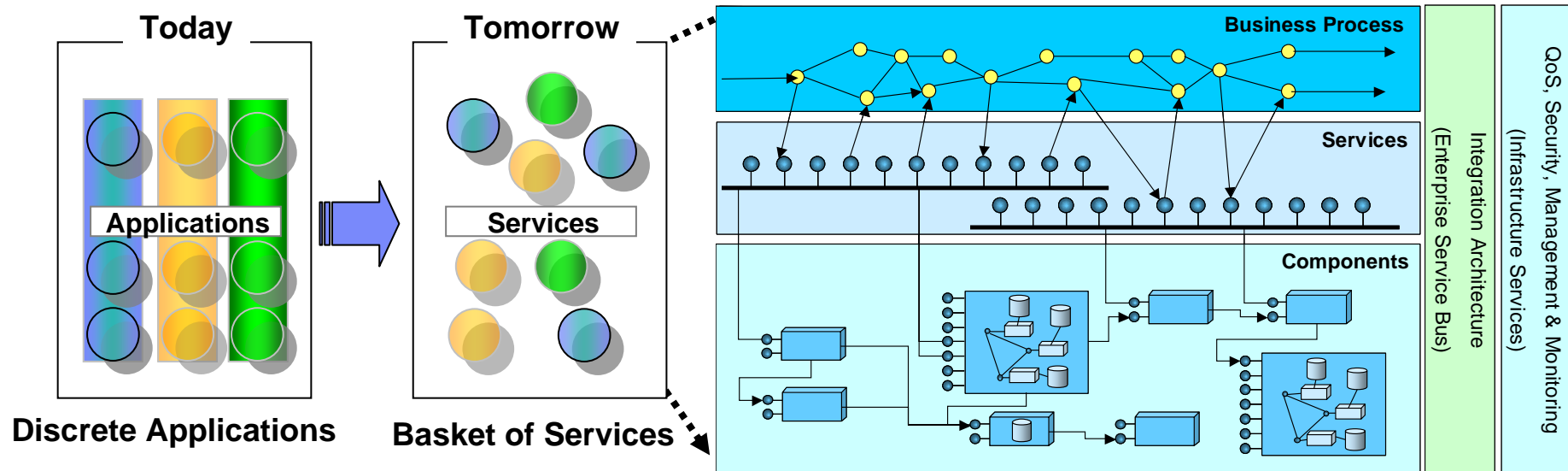# An ESB Addresses End-to-End Integration Challenges

A communications "architecture" that enables software applications that run

- on *different platforms and devices*
- written in *different programming languages*
- use *different programming models*
- require *different data representations*

to communicate *with no disruption to existing applications or interfaces.*

Disparate programming models

Asynchronous Messages	Synchronous RPC	Managed FTP	Publish/ Subscribe	Web Services

Enterprise Service Bus

XML
FIX
COBOL Copybook
SWIFT
HIPPA ebXML AL3 HL7
ACORD
EDI-FACT EDI-X.12
Custom Formats Word/Excel/PDF

COBOL
RPG
.net

Disparate Platforms & devices

Disparate Programming languages

**Disparate Data formats**

Vision of Service Oriented Architecture

Business Process Execution Language for Web Services (BPEL)

- Useful in defining both concrete and abstract processes
- Joint IBM/Microsoft proposal, being standardized through OASIS
  - ▶ There are some competing languages, including BPML from the BPMI
- Each activity is represented as a service with a WSDL interface
- Supports compensation model of transactions for long-running processes
- Already implemented by a number of vendors in their products, including IBM
  - ▶ Although most use extensions to the specification for a complete solution
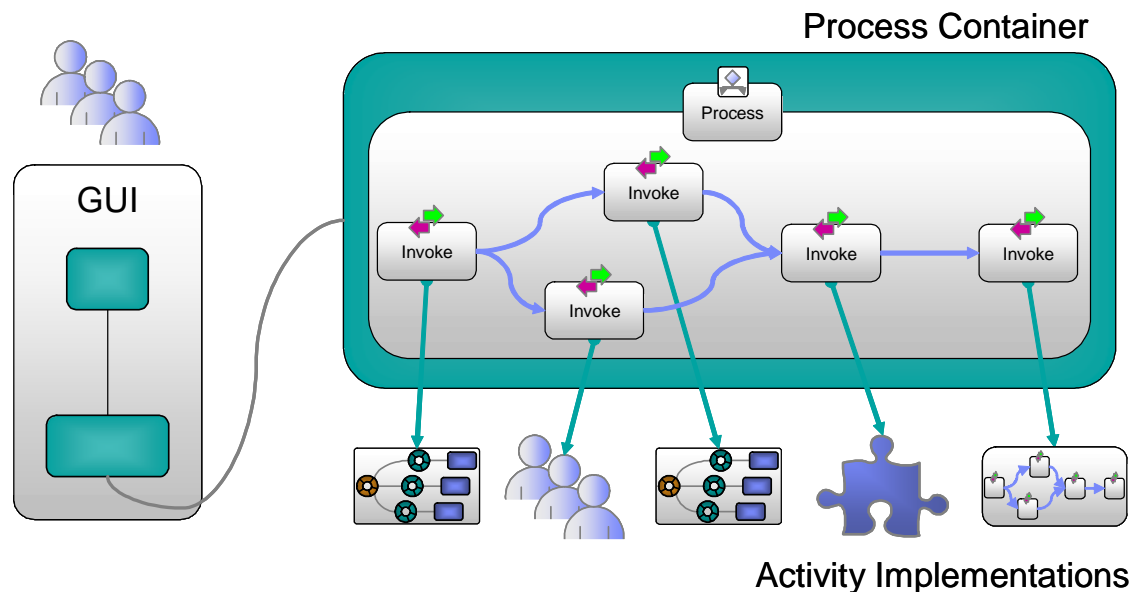
Process Container

GUI

Process

Invoke

Invoke

Invoke

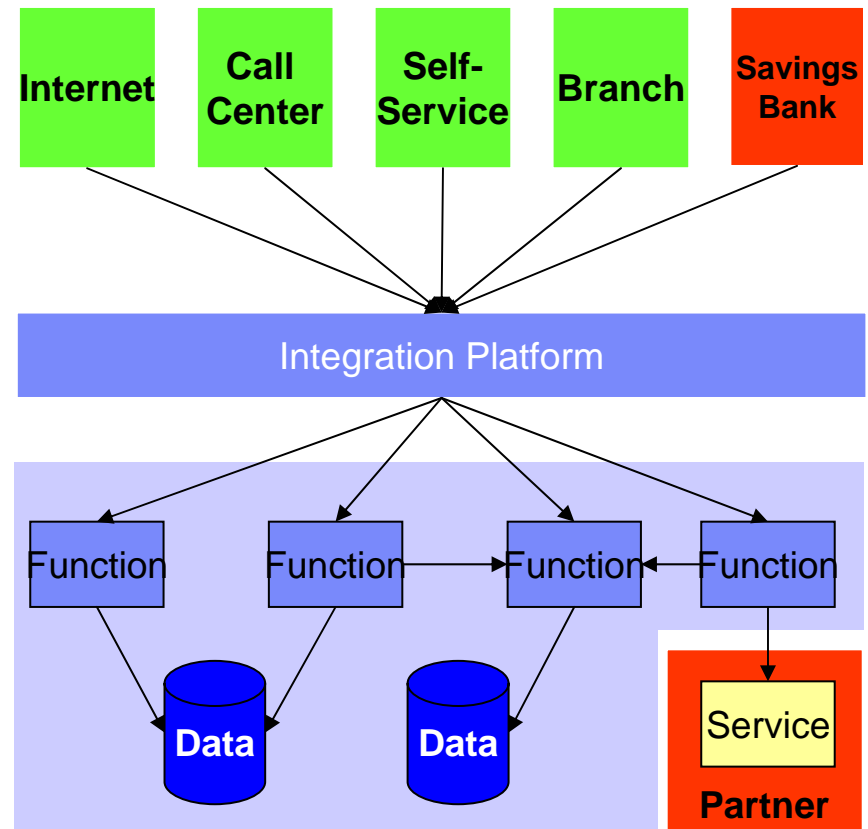Invoke

Invoke

Activity Implementations

Agenda

- SOA value proposition and definitions revisited
- Candidate reference architectures for SOA
  - ▶ Service-Oriented Modeling and Architecture (SOMA)
  - ▶ On Demand Operating Environment (ODOE)
  - ▶ Key architectural templates and patterns
    - Enterprise Service Bus (ESB)
    - Business Process Choreography
- **Project examples**
  - ▶ **Finance industry**
  - ▶ **Telecommunications industry**
- Roadmap for project adoption
  - ▶ SOA/Web services adoption levels and entry points, IBM assets
  - ▶ „Perspectives on Web Services" text book
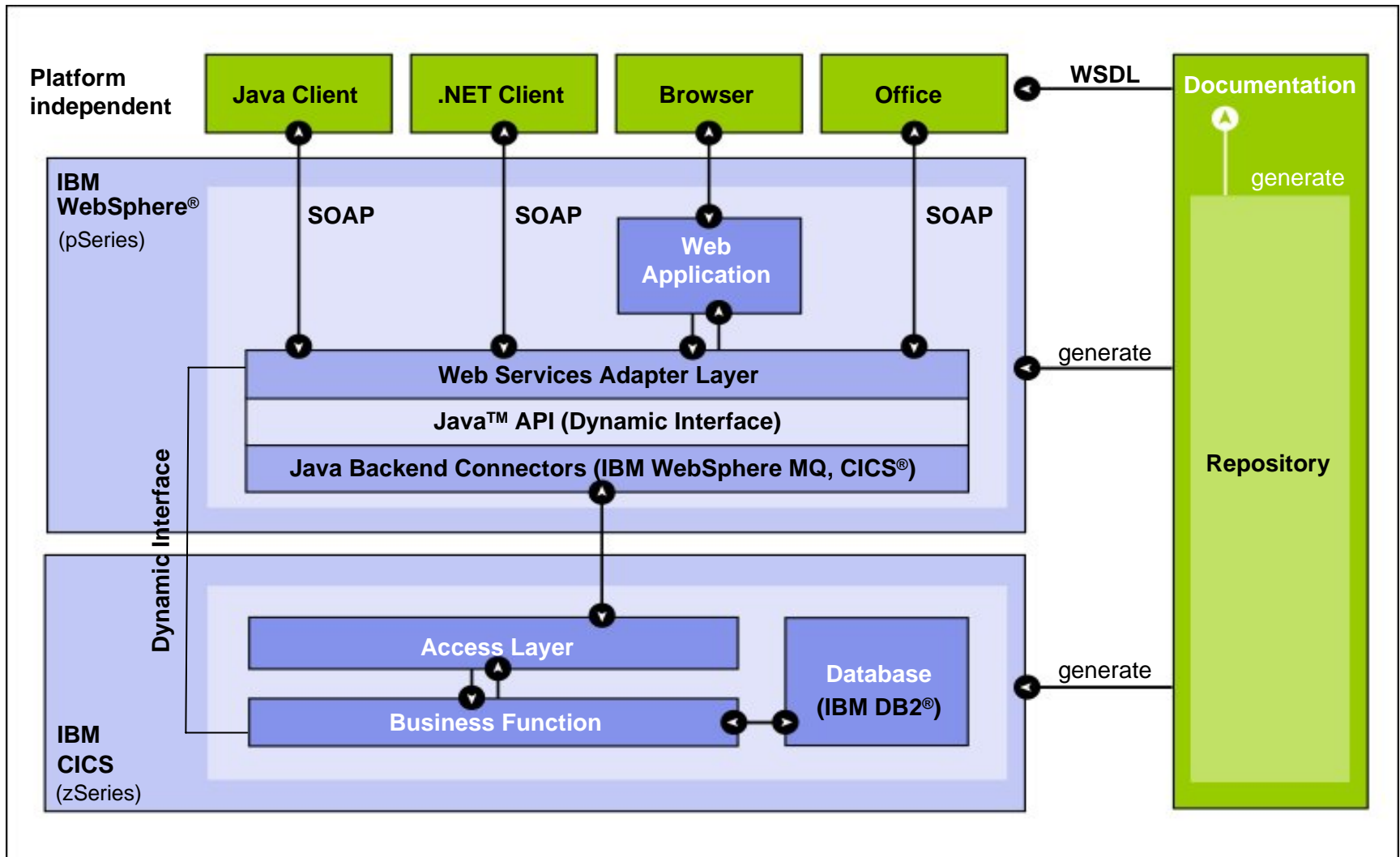 - ▶ IBM jStart program

Sparkassen Informatik's Vision in 1996:
From monolithic applications to a service-oriented architecture

- **Traditional architectures**
  - Providing redundant and often not reusable functions and data
  - High complexity and complicated integration

- **Goals**
  - Provide savings banks controlled access to core banking functionality
  - Efficient integration capabilities for best-of-breed solutions
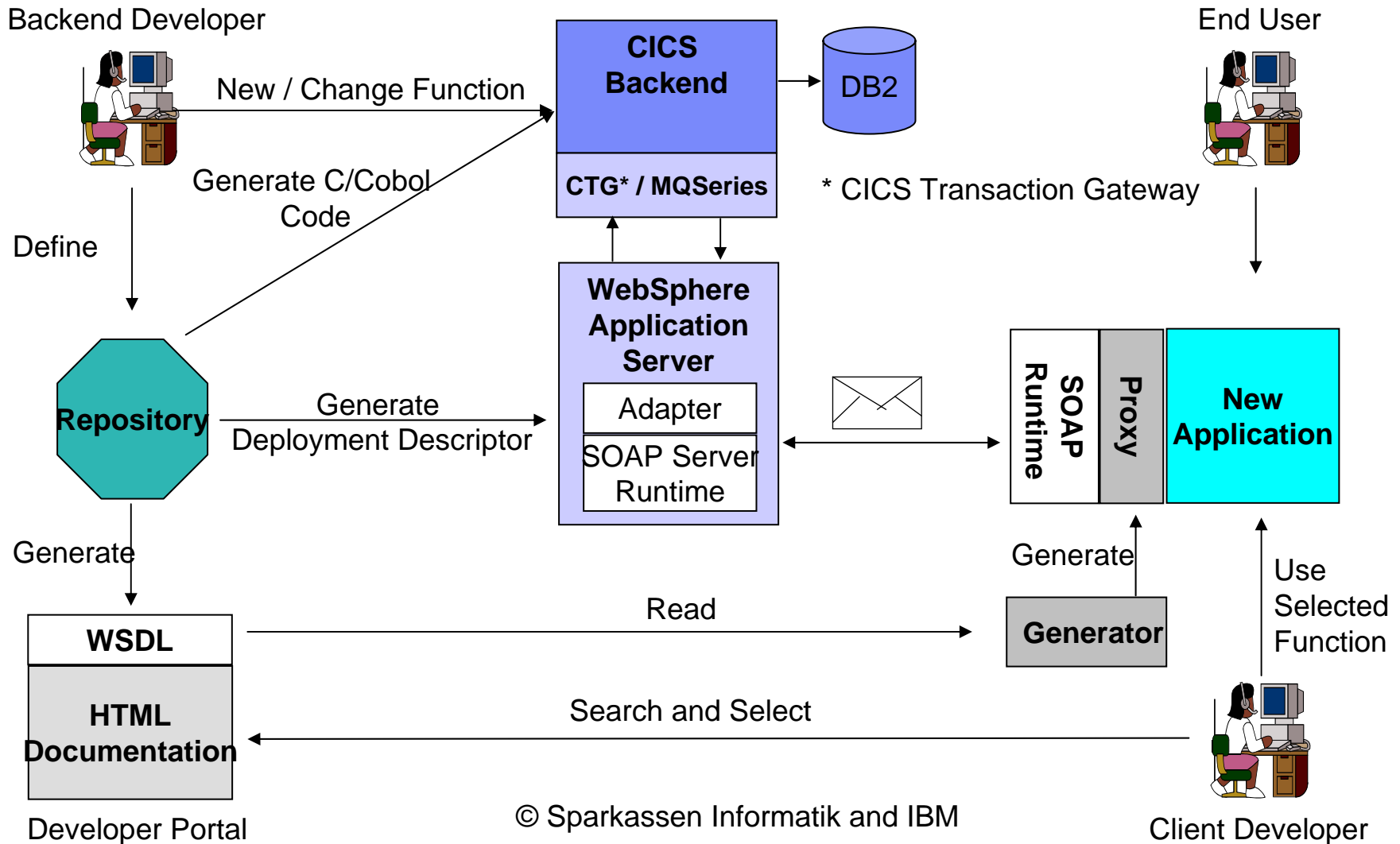 - Faster development cycles

- **Solution Approach (One System Plus)**

| Internet | Call Center | Self-Service | Branch | Savings Bank |

Integration Platform

Function | Function | Function | Function

Data   Data   Service

Partner

Architectural Overview

Platform independent

Java Client	.NET Client	Browser	Office

WSDL

Documentation

IBM WebSphere® (pSeries)

SOAP   SOAP   SOAP

Web Application

generate

Dynamic Interface

Web Services Adapter Layer

Java™ API (Dynamic Interface)

Java Backend Connectors (IBM WebSphere MQ, CICS®)

generate

Repository

Access Layer

Business Function

Database (IBM DB2®)

generate

IBM CICS (zSeries)

Integrated and Automated
Application Development Environment and Process

Backend Developer

New / Change Function

Generate C/Cobol Code

Define

Repository

Generate Deployment Descriptor

Generate

WSDL

HTML Documentation

Developer Portal

CICS Backend

DB2

CTG* / MQSeries

* CICS Transaction Gateway

WebSphere Application Server

Adapter

SOAP Server Runtime

Read

Search and Select

© Sparkassen Informatik and IBM

End User

SOAP Runtime | **Proxy** | **New Application**

Generate

Generator

Use Selected Function

Client Developer

Detailed Project Report Available!
http://www.perspectivesonwebservices.de or ACM.org

OOPSLA 2004
Practitioner Report P06
Session: Building Distributed Systems

Second Generation
Web Services-Oriented Architecture
in Production in the Finance Industry

Michael Brandner and Olaf Zimmermann,
IBM Germany

October 24-28, 2004   Vancouver, BC, Canada

Telco Wholesaler: Business and Technical Context

- **Large telco wholesaler** serving 190+ service providers (retailers)

  - Typical business events: end customers order new telephone service or indicate external relocation via service provider call center

  - Actual network provisioning and resource management performed by wholesaler

- **Order entry management application** for PSTN, DSL, value-added services

  - Browser channel (standard J2EE e-business application) in use since 1999/2000

  - System context: 20+ backend connections (MQ, ftp, ...)

  - B2B Web services channel introduced two production releases ago

  - Only simple (stateless, atomic) use cases without long-running workflow semantics supported in B2B channel so far

Multi-Channel Order Management Supporting a Wholesaler-Retailer Business Model in the Telecommunications Industry.

- **Hot component and functional domain:**
  - Order entry management
  - Two core business processes selected initially
- **Main SOA drivers**
  - Deeper automation grade (-> cost savings, flexibility)
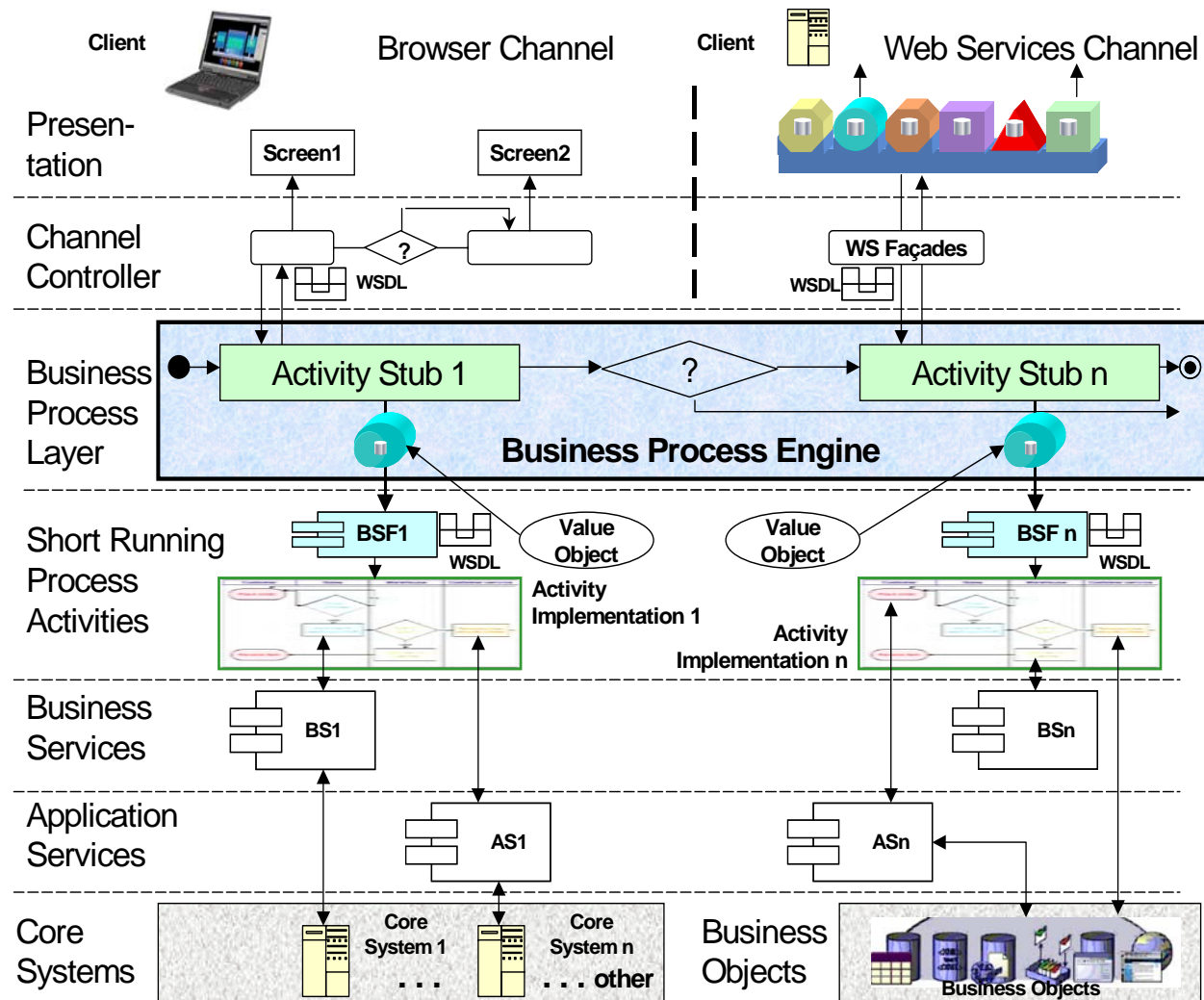 - Speed up SW development lifecycle (-> agility)
  - High-quality services shared within and between domains (-> ease of maintenance)
- **Service modeling**
  - Top-down (retailer interface)
  - Bottom-up from existing wholesaler systems
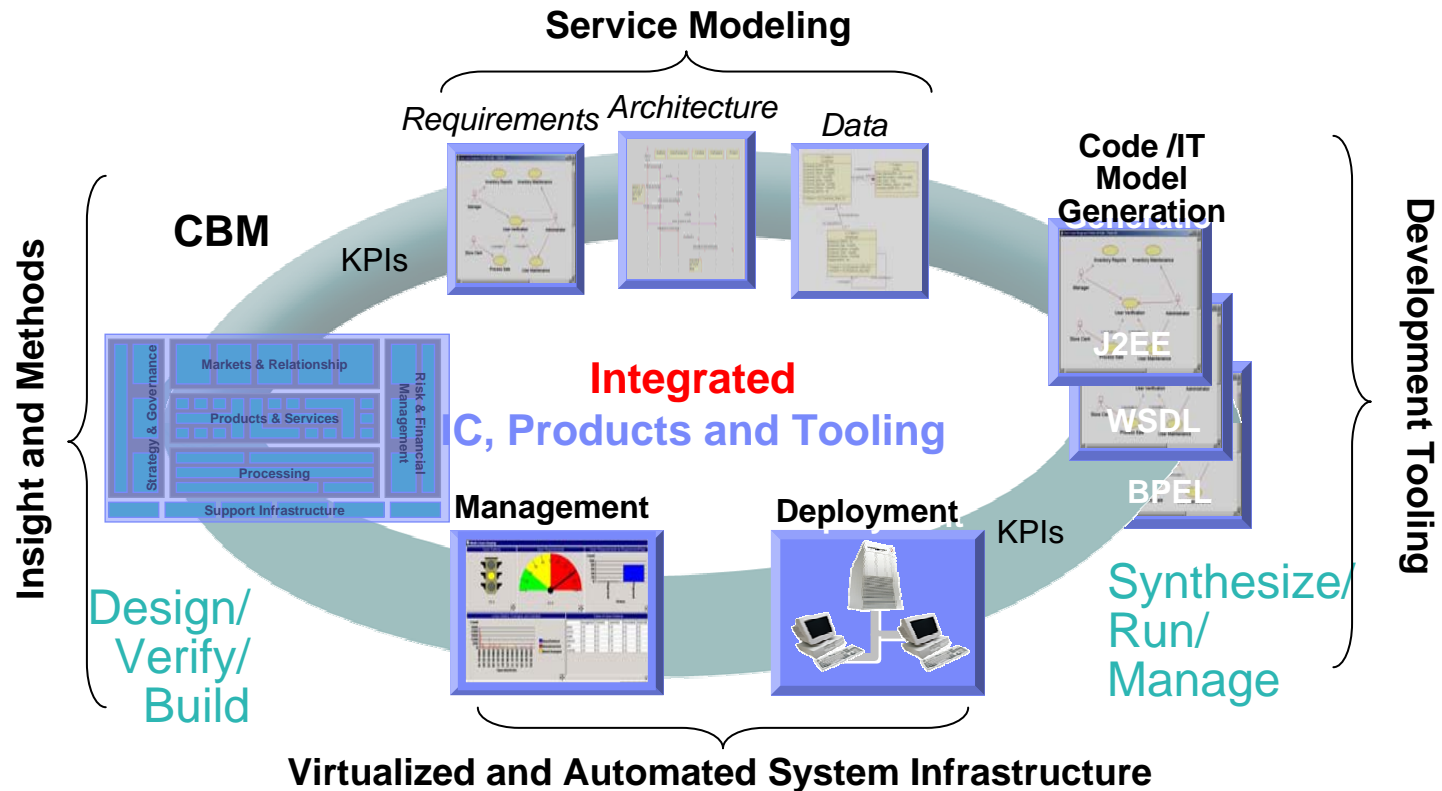 - Categorization taxonomy

Agenda

- SOA value proposition and definitions revisited

- Candidate reference architectures for SOA
  - ▶ Service-Oriented Modeling and Architecture (SOMA)
  - ▶ On Demand Operating Environment (ODOE)
  - ▶ Key architectural templates and patterns
    - Enterprise Service Bus (ESB)
    - Business process choreography

- Project examples
  - ▶ Finance industry
  - ▶ Telecommunications industry

- **Roadmap for project adoption**
  - ▶ **SOA/Web services adoption levels and entry points, IBM assets**
  - ▶ **„Perspectives on Web Services" text book**
  - ▶ **IBM jStart program**

IBMs key benefits for our clients result from a closed loop across business and IT transformations reflected in integrated product and services deployment for Business Performance Management (BPM).
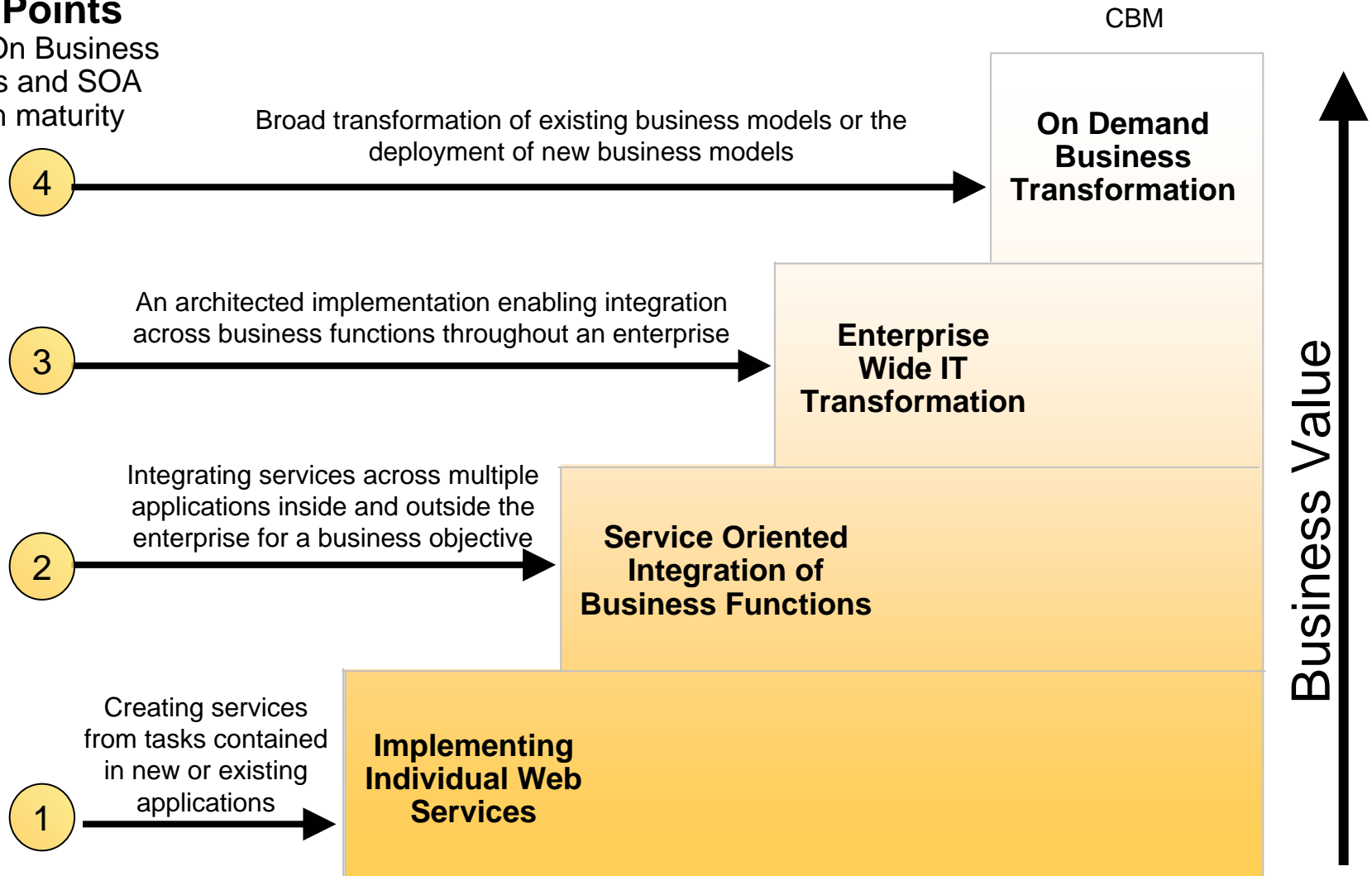
Service Modeling

Requirements   *Architecture*   *Data*

Code /IT Model Generation

Insight and Methods

CBM

KPIs

J2EE

WSDL

BPEL

Development Tooling

Integrated
IC, Products and Tooling

Markets & Relationship

Products & Services

Processing

Support Infrastructure

Strategy & Governance

Risk & Financial Management

Management   **Deployment**   KPIs

Design/
Verify/
Build

Synthesize/
Run/
Manage

Virtualized and Automated System Infrastructure

Improved Productivity for Architects, Designers, Developers, Administrators

- Lower cost of Application Development and Management – find errors early

- Enhanced Portability and Interoperability

- Models and Technologies evolve at own pace on platform(s) of choice

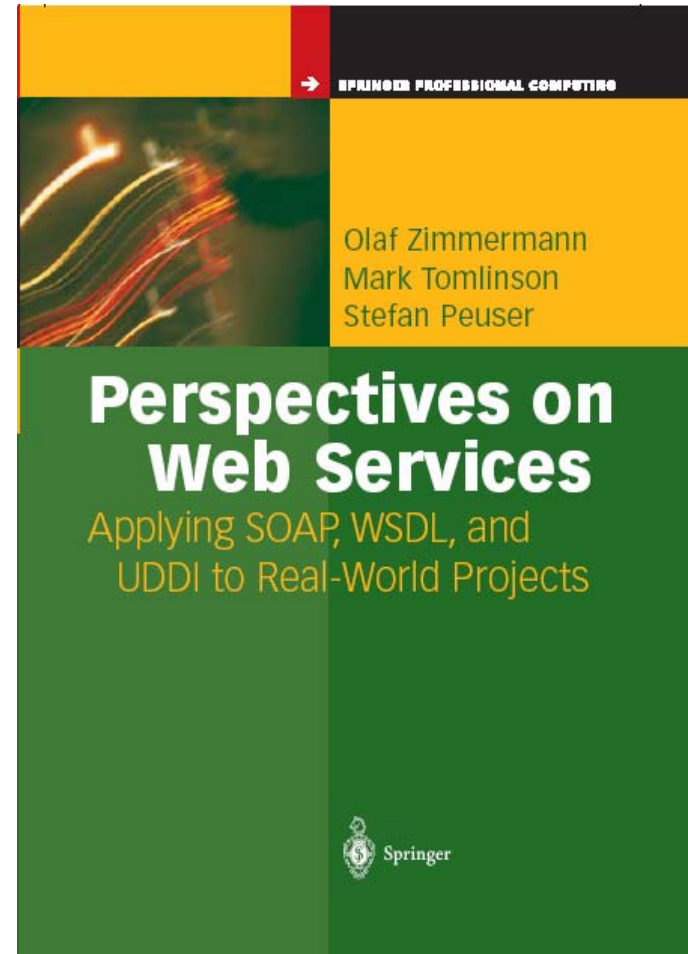# SOA and Web Services Standards Adaptation

Entry Points

Based On Business
Priorities and SOA
adoption maturity

CBM

Broad transformation of existing business models or the
deployment of new business models

4

**On Demand
Business
Transformation**

An architected implementation enabling integration
across business functions throughout an enterprise

3

**Enterprise
Wide IT
Transformation**

Integrating services across multiple
applications inside and outside the
enterprise for a business objective

2

**Service Oriented
Integration of
Business Functions**

Creating services
from tasks contained
in new or existing
applications

1

**Implementing
Individual Web
Services**

Business Value

Springer Text Book Authored by IBM Practitioners: Perspectives on Web Services (PoWS)

- The IBM architects for the Sparkassen Informatik and other projects have written a book, with an expansion on their ideas for Web services architecture design. Emphasis is on lessons learned and best practices.

- **Perspectives on Web Services**, by Olaf Zimmermann, Mark Tomlinson, Stefan Peuser, Springer 2003, ISBN 3-540-00914-0

- Foreword by Grady Booch, IBM Fellow and Chief Scientist IBM Rational

Olaf Zimmermann
Mark Tomlinson
Stefan Peuser

Perspectives on Web Services
Applying SOAP, WSDL, and UDDI to Real-World Projects

Springer

Building Blocks for Delivering a Web Services Architecture
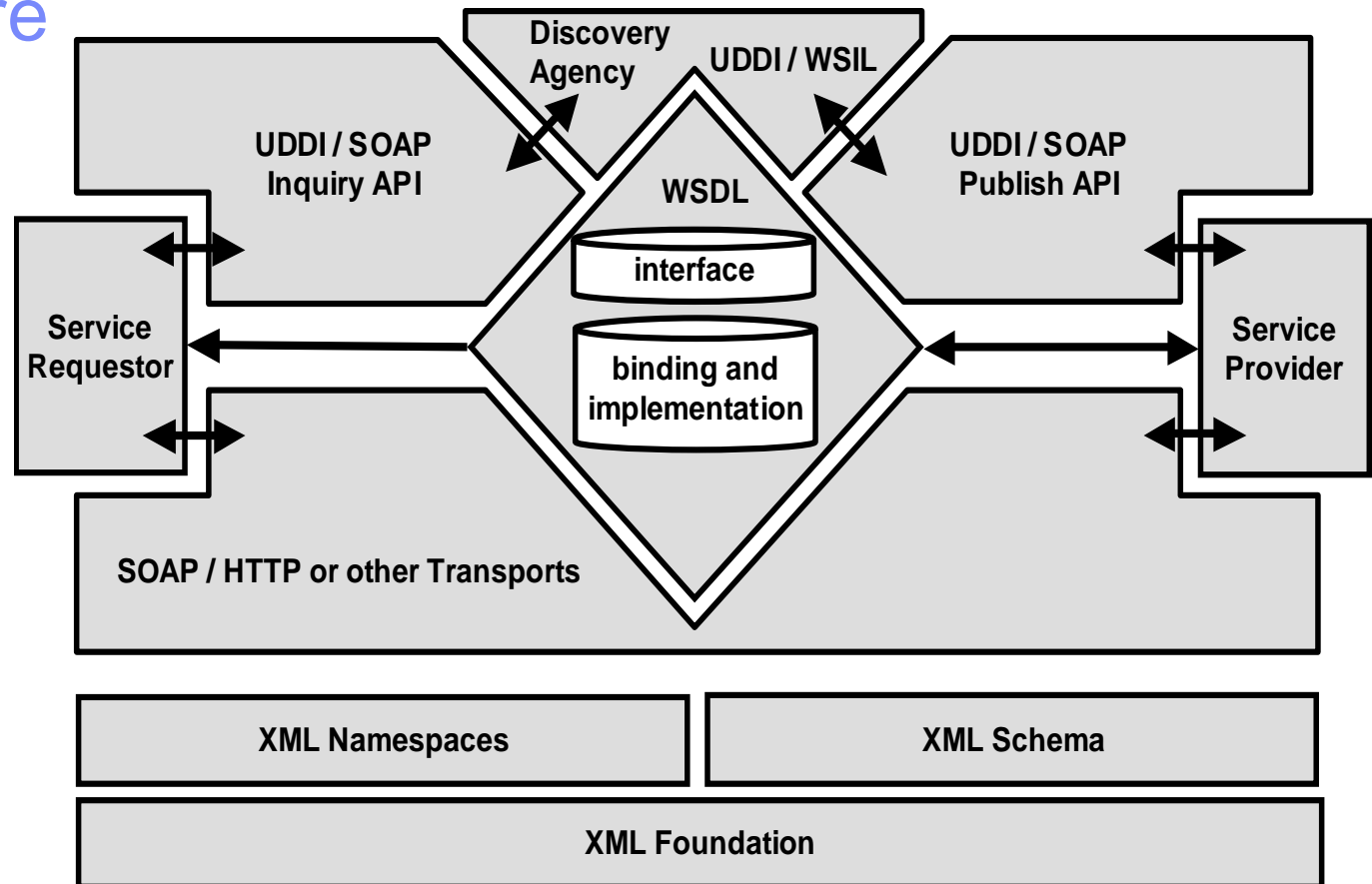
Interpretation of the core specifications and links through the WS-I basic profile 1.0

SOA and Web Services Standards Adaptation

Entry Points
Based On Business
Priorities and SOA
adoption maturity

CBM

Broad transformation of existing business models or the
deployment of new business models

4

On Demand Business Transformation

An architected implementation enabling integration
across business functions throughout an enterprise

3

Enterprise Wide IT Transformation

Integrating services across multiple
applications inside and outside the
enterprise for a business objective

2

Service Oriented Integration of Business Functions

Creating services
from tasks contained
in new or existing
applications

1

Implementing Individual Web Services

Business Value

Outstanding Redbooks on SOA / ESB

http://www.redbooks.ibm.com

IBM developerWorks SOA and Web Services Zone

SOA and Web Services Standards Adaptation

Entry Points
Based On Business
Priorities and SOA
adoption maturity

CBM

4 Broad transformation of existing business models or the deployment of new business models → **On Demand Business Transformation**

3 An architected implementation enabling integration across business functions throughout an enterprise → **Enterprise Wide IT Transformation**

2 Integrating services across multiple applications inside and outside the enterprise for a business objective → **Service Oriented Integration of Business Functions**

1 Creating services from tasks contained in new or existing applications → **Implementing Individual Web Services**

Business Value

Service Oriented Modeling and Architecture (SOMA) is a technique to build a flexible enterprise by combining CBM and SOA across scenarios.

CBM
Strategy

Break down your business into components

1. Decide what is strategically important, and what is just operations in the value chain domains
2. Analyze the different KPIs attached to these components
3. Prioritize and scope your transformation projects

SOMA
Modeling

Define a Service Model

1. Identify your services based on your business components
2. Specify the services and components accordingly
3. *Make SOA realization decisions based on architectural decisions*

SOA
Realization

Implement a Service Model

1. Develop a service-oriented architecture to support the Componentized Business
2. Implement service based scoping policy for projects
3. Implement appropriate governance mechanism

Business-Aligned IT Architecture

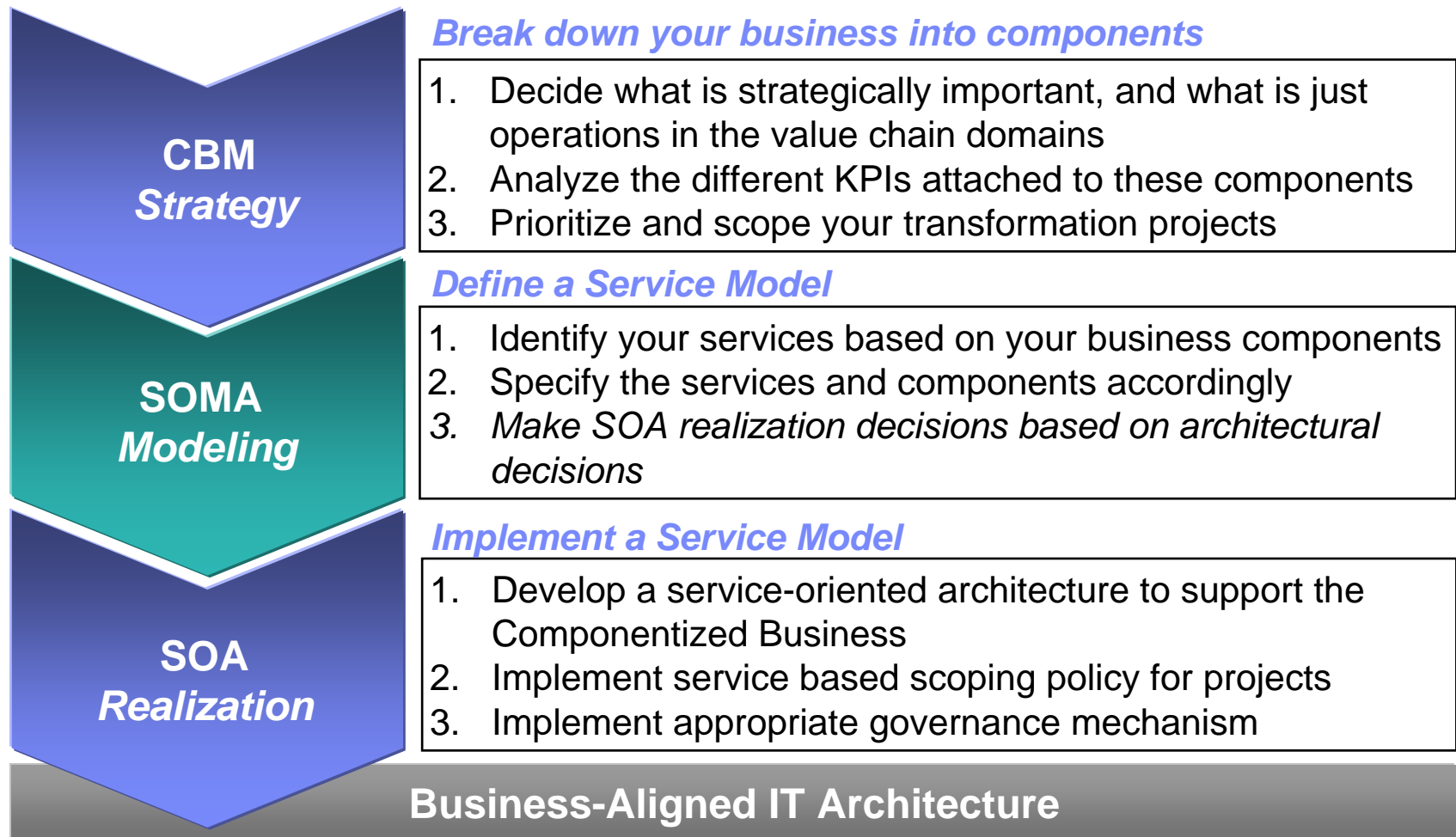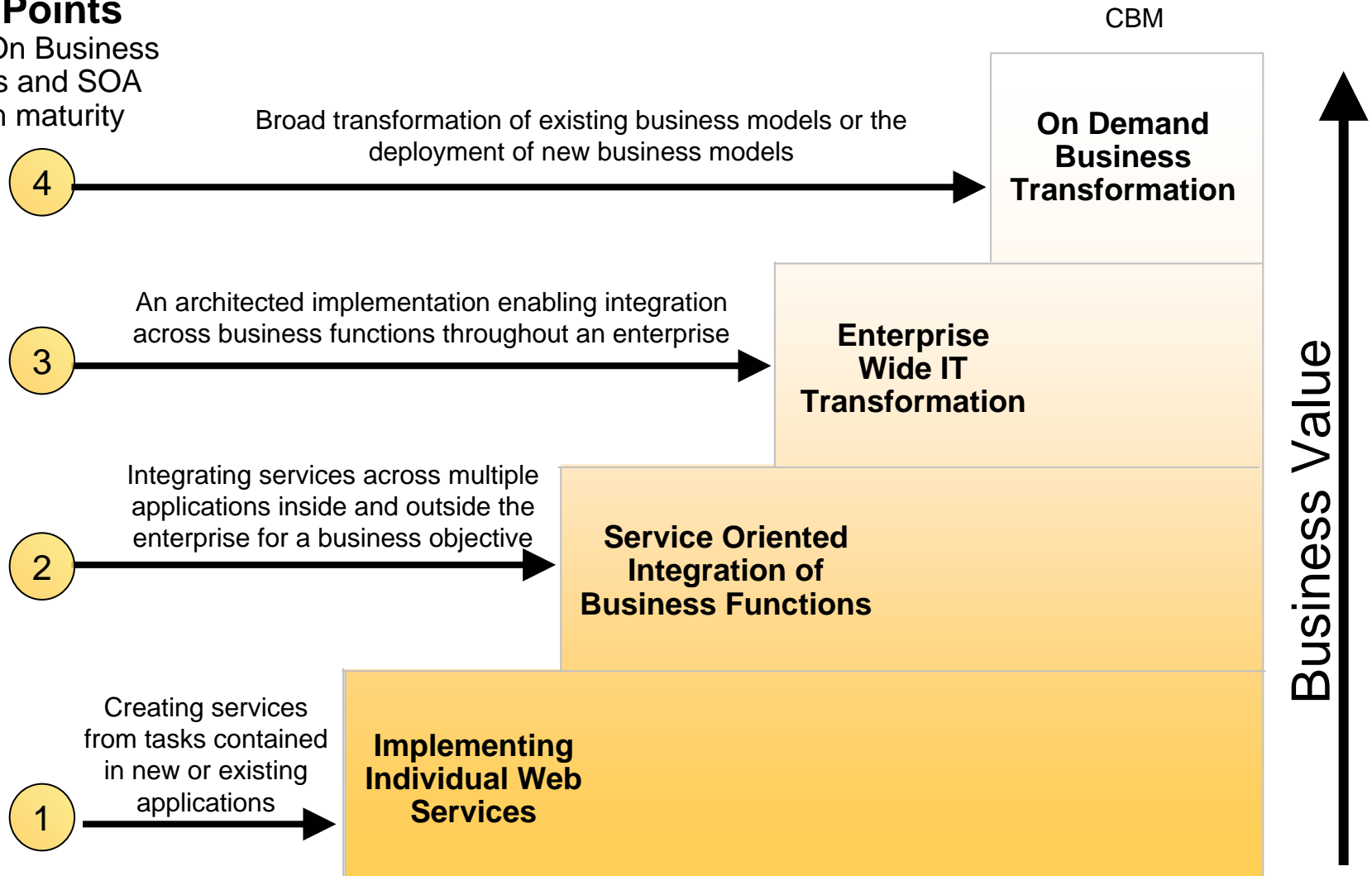# SOA and Web Services Standards Adaptation

Entry Points
Based On Business Priorities and SOA adoption maturity

CBM

Broad transformation of existing business models or the deployment of new business models

4 → **On Demand Business Transformation**

An architected implementation enabling integration across business functions throughout an enterprise

3 → **Enterprise Wide IT Transformation**

Integrating services across multiple applications inside and outside the enterprise for a business objective

2 → **Service Oriented Integration of Business Functions**

Creating services from tasks contained in new or existing applications

1 → **Implementing Individual Web Services**

Business Value

jStart and Advanced Technology Services (ATS)

Mission Statement

Develop, refine and deploy advanced technologies solutions to drive adoption of open standard technologies and IBM products that deliver business value and competitive advantage to our customers.

Achieving the mission statement requires meeting the objectives of:

- ☑ Accelerating speed to market of business solutions leveraging evolving technologies
- ☑ Reducing risk and cost of solution implementation
- ☑ Understanding technology trends, while advancing emerging technologies skills required to adapt technology to business requirements
- ☑ Enabling customers to quickly and confidently adopt advanced technologies so they can become solid, proactive IBM reference accounts

ATS and jStart Engagement Model

Business Qualification

Phase I
- Evangelism
- Business Value Proposition
- Determine Readiness
- Identify Candidate Projects
- Gain Executive Commitment

Project Definition

Phase II
- Business Objectives
- Requirements Gathering
- Prioritization and Selection
- Project Scope and Definition

Project Readiness

Phase III
- Project Design
- Use Case Analysis
- Produce Project Plan / Sizing
- Develop Business Proposal

Customer Commitment

Phase IV
- Proposal Presentation
- Contractual Agreement

Solution Building

Phase V
- Implementation and Deployment
- IBM Reference Activity

Engage ⟶ Deploy ⟶ Promote

Contact Information

- ## When to call on jStart/ATS?

  - ▸ Embracing emerging technologies and "jump-Start" would be advantageous

  - ▸ Emerging technology skills or product skills

  - ▸ Services pricing flexibility

  - ▸ Speed to market track record, short duration projects

- ## jStart/ATS contacts in EMEA

  - ▸ Anton (Tony) Fricko, anton_fricko@at.ibm.com

  - ▸ Stefano Pogliani, stefano.pogliani@fr.ibm.com

  - ▸ Olaf Zimmermann, ozimmer@de.ibm.com