Management of Service-Oriented Implementations
of Business Processes: From Quality of Service (QoS) to Business Value

BPM 2008 Tutorial

Tutorial Goals
- Explain that QoS and business value specification and management are crucial for business process efficiency and agility
- Summarize and analyze the main concepts
- Inform that there have been:
  - many academic and industrial works on QoS, but
  - only a few on business value management
- List and discuss open issues
- Provide a foundation for future research and/or decision-making by the participants

Presentation Outline
I. Importance of QoS and business value
II. Specification of QoS
III. Management (monitoring & control) of QoS
IV. Business-driven IT management (BDIM)
V. From QoS to business value
VI. Summary, challenges and discussion

I. Definition of Terminology and Importance of QoS and Business Value
- What is a service? (2 different definitions)
- Services and business processes
- What is quality of service (QoS)?
- Management = monitoring + control
- Benefits of QoS management
- What is business value?
- Why QoS management is not enough?

What Is a Service? (Service-Oriented Computing View)
- In SOC, a service is:
  - a distributed software component ...
  - with a unique ID (e.g., URI) and ...
  - accessible over a network (e.g., Internet) ...
  - in a loosely coupled manner (run-time connection)
- Example: book buying Web service (WS)
- Web services: using XML-based standards (SOAP, WSDL, WSBPEL, ...) and Internet
**What Is a Service? (IT Service Management View)**

- In ITSM, a service is:
  - a means of delivering value to customers...
  - by facilitating outcomes customers want to achieve...
  - without the ownership of specific costs and risks
- Examples:
  - e-mail service provided by a specialized department or company
  - laptop repair service

**Services and Business Processes**

- Multi-faceted relationship
  - Depends on the definition of ‘service’
- Business process: coordinated set of activities producing an outcome that creates value
- SOC services and ITSM services implement activities in business processes
- ITSM services support business processes
- Business processes can be exposed through SOC services and/or as ITSM services

**What Is Quality of Service (QoS)?**

- Functionality/service = “WHAT operations does the system execute?”
  - Example: Returns current price for a stock symbol
- Quality of service (QoS) = “HOW WELL does the system perform its operations?”
  - Examples: Average response time is 2 seconds, availability in the last 24 hours is 99%, ...
  - Cf.: quality of product/object (e.g., book bought)
  - Synonyms: non-functional, extra-functional, ‘ilities’
  - QoS exists even when not specified or measured

**Definition of Management: Monitoring**

- Monitoring = monitoring and control
  - Run-time (and some deployment-time) activities
- Monitoring determines state of the system:
  - Measurement or calculation of QoS metrics (measures of QoS): response time, availability, ...
  - Evaluation of conditions (requirements or guarantees): response time < 2 seconds, ...
  - Accounting of invoked operations, consumed resources, measured/calculated QoS metrics, evaluated conditions, taken control actions, billed prices/penalties, ...

**Definition of Management: Control**

- Control tries to ensure that the managed system is always in its desired state:
  - Starting/stopping the system or its components
  - (Re-)Configuration of the system: setting thread priorities, re-composition of Web services, ...
  - (Re-)Allocation of resources: assigning processing time to requests from different consumers, ...
  - Billing of prices or penalties: penalty for not meeting guaranteed response time is US$1.00, ...
  - Modification of requirements or guarantees
  - Notification of human administrators: alert threshold notification, ...

**Monitor-Control Loop**

- Norm
- Control
- Compare
- Monitor
- Activity
- Output
**Complex Monitor-Control Loops**

1. Control
2. Compare
3. Monitor

**Benefits of QoS Management**
- 5 functional areas of system/network management (FCAPS): Fault, Configuration, Accounting, Performance, and Security
- QoS (performance) management helps to:
  - ensure correct operation,
  - attain or surpass guaranteed QoS,
  - discover and fix problems,
  - accommodate change,
  - balance price/performance ratios,
  - maximize profits, ...

**A Motivating Example for WS QoS Specification and Management**
- Cannot be done with basic WS technologies
- One of the consequences of management: differentiation in the market of Web services

**What Is Business Value?**
- Business value (metric) = any measure of business worth
  - Financial: income, cost, profit, margin, ...
  - Non-financial: number of customers, customer satisfaction, market share, ... (argument for capturing them: balanced scorecard – BSC)
- Is it QoS? Yes, but not in the traditional sense
- Business value is subjective!
  - It is not the same for consumer, provider, ...
- Related (almost synonymous) terms:
  - KPI (Key Performance Indicator),
  - business (performance) metric, ...

**Why QoS Management Is Not Enough?**
- Technical QoS is important, but business value is more important
  - Do customers/users really care whether availability is 98% or 99%? Not really...
  - They care about its impact on their business value
- Mappings between the two are complex
  - Will 1% higher availability result in higher business value? Not always... (even if yes: amounts differ)
  - Depend on domain, context, business strategy, ...

**What Has to Be Developed?**
1. Well-defined (hopefully, standardized) formats for specification of QoS / business value
   - You cannot control what you cannot monitor
   - You cannot monitor what you cannot define
   - Basic Web service standards do not address this
2. Diverse algorithms & protocols
   - Selection of WSes using QoS / business value info, negotiation of QoS and prices, control to maximize business value, adaptation to changes in QoS, ...
3. Management infrastructures/tools
**Terminology and Importance: Summary & Discussion**

- **Service** has (at least) 2 different definitions
- **QoS** = “How well does the system perform?”
- **Business value** = any measure of business worth (not only financial)
- **Management** = monitoring + control
- **Specification** impacts management capability
- **QoS** and business value specification and management are crucial for business process
  - efficiency (e.g., attain desired technical/business performance), agility (e.g., adapt to changes), ...

**Presentation Progress**

I. Importance of QoS and business value
II. Specification of QoS
III. Management (monitoring & control) of QoS
IV. Business-driven IT management (BDIM)
V. From QoS to business value
VI. Summary, challenges and discussion

**II. Overview of Approaches to and Tools for WS QoS Specification**

- Classification of QoS specification approaches
- Contracts (example language: WS-Agreement)
- Service level agreements – SLAs (example language: WSLA)
- Classes of service
- Policies (example languages: WS-Policy)
- Summary of issues related to WS QoS specification

**Classification of QoS Specification Approaches**

- **QoS specification** = description of what/where/when/how to monitor & control
- **QoS info** = descriptions & monitored values
- Classification of QoS specification approaches:
  1. Implicit – built into the implementation (not flexible)
  2. Contracts – formal agreements (for QoS, billing, ...)
     - Service Level Agreements (SLAs)
     - Classes of service – a special type of SLAs
  3. Policies – high-level operation & management goals and/or rules (for security, QoS, billing, ...)

**QoS Specification Topics Present in All Approaches**

- Where are QoS metrics defined?
  - There are no standard QoS metrics - use, names, and definitions vary! Example: ‘response time’ can have at least 2 different meanings! 4 approaches:
    - Nowhere (implicit meaning) - not precise
    - In the QoS language grammar - not flexible
    - In QoS specification files (e.g., SLAs) - not reusable
    - In external reusable ontologies (definition files)
      - Other ontologies can define measurement units
  - For practical use, QoS specification languages must be accompanied by appropriate tools!

**Contract**

- **Contract** = binding and enforceable formal agreement between two or more parties
- Defines requirements & guarantees of parties
  - Can be used in monitoring and control
- Contracts enable not only QoS description, but also QoS differentiation
  - Different consumers can have different contracts
- Apart from QoS info, a contract can contain other information (e.g., prices/penalties)
  - A WSDL file is a contract
**Specification of QoS in Extended WSDL, UDDI, or WSBPEL Files**

- **Strengths:**
  - The extensions can be relatively simple
  - QoS discovery related to Web service discovery

- **Weaknesses:**
  - QoS specification language tied to WSDL (UDDI, WS-BPEL) in terms of tools, evolution, ...
  - Extension mechanisms are limited
  - Run-time change of QoS information requires updates of all affected copies of WSDL (UDDI, WS-BPEL) files, which is complicated

**WS-Agreement**

- from Global Grid Forum (GGF); industry support (IBM, ...)
- General framework for XML specification of agreements and agreement templates
- plus a simple agreement negotiation protocol and run-time agreement monitoring interface
- Intended for multiple domains, not only WSes
- WS-Agreement allows use of any language for the actual contained specifications (including QoS expressions, QoS metrics, ...)
- This flexibility can produce incompatibility

**WS-Agreement: Agreement Template Structure**

- **Name**
  - Context: Involved parties (initiator & provider); Expiration time; Template name, Related agreements
  - Terms: Term compositors: ExactlyOne/OneOrMore/All
    - Service description terms: Service descriptions, Service references, Service properties
    - Guarantee terms: Service scope, Qualifying condition, Service level objective (SLO), Business value list (Importance, Penalty, Reward, Preference, ...)
  - Creation constraints: item requirements and/or constraints (in some language)

**Service Level Agreement (SLA)**

- A special type of contract for QoS (and often price/penalty) requirements & guarantees
- Many different formats, one of them is:
  - Parties (including supporting management parties)
  - Service description
    - Service operations – describe available operations
    - SLA parameters – define monitoring of QoS metrics
  - Obligations
    - Service Level Objectives (SLOs) - QoS guarantees
    - Action guarantees - specify what happens if SLOs are met or not met

**A Simple Example of an SLA**

Parties: consumer C and provider P
Service operations: P has one operation (OP1) float getStockPrice(String stockName)
SLA parameters: (RT-OP1-C) Response time of operation OP1 measured at consumer C by consumer C SLOs: (SLO1) For every OP1 invocation by C, RT-OP1-C will be less than or equal to 2 seconds Action guarantees: (AG1) If SLO1 was met, C pays P price of US$0.20 per invocation; (AG2) If SLO1 was not met, P pays C penalty of US$0.10 per invocation
Service Level Agreement (SLA): Strength and Weaknesses

**Strengths:**
- Formal contract specification of QoS and related management aspects
- Widely used in computing and communications systems (also for WSes)

**Weaknesses:**
- Negotiation of custom-made SLAs can require complex analysis of offers and generation of counter-offers (can be alleviated by using templates)
- Management of many concurrent custom-made SLAs can be complex & with high run-time overhead
- Cannot be used for QoS-enabled WS selection

Web Service Level Agreement (WSLA): Overview

from IBM Research: H. Ludwig, A. Keller, A. Dan, ...
- QoS language & management infrastructure
- Compatible with, but not restricted to WSes
- Custom-made SLAs (the example SLA format)

**Strengths:** detailed and precise specification of monitoring and control;
- several tools for SLA creation, deployment, and compliance monitoring (were distributed by IBM);
- widely referenced; was used in practice
- Weaknesses: those of custom-made SLAs;
- QoS metrics defined within SLAs

Web Service Level Agreement (WSLA): Language Details

- SLA parameter - monitored property; contains 1 QoS metric & extra info for exchange of values
- QoS metric – defines where & how to measure or calculate; can be reused across SLA parameters
- An SLO contains: evaluated Boolean expression (limits values of SLA parameters), obliged party, validity periods, evaluation event or schedule
- An action guarantee contains: precondition expression, evaluation event or schedule, action to be taken, obliged party, execution modality
- Reusability: SLA templates, metric macros, ...

Class of Service

- A special type of SLA that is not custom-made, but predefined & reusable (anonymous)
  - 1 provider can offer many classes of service that refer to the same functionality, but differ in QoS
  - 1 class of service can be used by many consumers
  - Simple selection instead of complex negotiation
  - Classes of service already checked for consistency

**Strengths:** Usable for QoS-enabled WS selection, no complex negotiation, simpler management, lower run-time overhead, faster adaptation
- Weakness: Discrete differentiation - limited choice

Policies – High-Level Goals and/or Rules

- A classification of policy types [Kephart&Walsh2004]:
  - Action: Describe what should happen - "If-Then" rules ("If response time of operation A is greater than 2 sec, provider pays penalty of US$0.10")
  - Goal: Describe desired state ("Response time of operation A is less than or equal to 2 sec")
  - Utility: Quantify "goodness" of a particular state ("Add to the goodness measure [2 sec - response time of operation A] * 10 units") - rarely used

**SLAs vs. policies:** SLOs can be viewed as goal policies, action guarantees as action policies
- **Policies vs. business rules:** mostly synonyms?

Web Services Policy Framework (WS-Policy)

from BEA/IBM/Microsoft/SAP – industry support!
- General, flexible and extensible, framework for specification of (security) policies for WSes
- Many good features (e.g., policies can be in or out of WSDL files, some reusability constructs)
- QoS extensions (several exist) require:
  - Precise and detailed QoS specification
  - Contracts/SLAs/classes of service and their static and dynamic relationships
- Standardized expression mechanism
**Approaches to (WS) QoS Specification: Summary & Discussion**

- Contract = binding and enforceable formal agreement between two or more parties
- SLA is a special type of contract; many SLA formats
- Class of service is a type of a light, predefined SLA
- Which one to use depends on circumstances
  - For comprehensiveness: general contracts
  - For flexibility of QoS specification: custom SLAs
  - For low overhead: classes of service
- Contracts vs. policies: similar information, different management architectures
- One view: external contracts and internal policies

**Languages for WS QoS Specification: Summary & Discussion**

- There are many different languages
  - Most are based on contracts, particularly SLAs
- WS-Agreement & WS-Policy: general frameworks that can be extended for WS QoS specification
  - Have industry support, but the "meat" is missing
- WSLA: example precise and detailed contract-based QoS specification language
  - Its solutions could be re-used (along with some ideas from other languages) for the "meat"

**Presentation Progress**

I. Importance of QoS and business value
II. Specification of QoS
III. Management (monitoring & control) of QoS
IV. Business-driven IT management (BDIM)
V. From QoS to business value
VI. Summary, challenges and discussion

**III. Overview of Approaches to and Tools for WS QoS Management**

- Approaches to QoS monitoring (example tools: WSLA Framework, Cremona)
- Some approaches to QoS control (example tool: Smartware)
- On approaches to QoS-based Web service selection (only briefly)
- Industrial products for WS QoS management
- On QoS of Web Service compositions
- Summary of issues related to WS QoS management

**Classification of Approaches to QoS Monitoring**

- Instrumentation
  - Internal instrumentation
  - External instrumentation
- Intermediaries
- Probes
- Sniffers
- All approaches have strengths & weaknesses – which one to use depends on circumstances

**Using Provider-Side Monitoring Instrumentation**

- Invasive vs. non-invasive instrumentation
  - Invasive: within business logic code of a WS vs. within WS hosting tools (e.g., within SOAP engine)
- All monitoring on the provider side
- Strengths: Realistic & consumer-specific measures; independent from network location of measurement
- Weaknesses: Provider must have capabilities and willingness; consumers must trust the provider (no way to check results given by the provider)
**Some Instrumentation Technologies**

- Industry standards:
  - Simple Network Management Protocol (SNMP)
  - Application Response Measurement (ARM)
  - Java Management Extensions (JMX)
  - Windows Management Instrumentation (WMI)

- Other instrumentation approaches:
  - Addition of composable SOAP message processing filters to WS tools (e.g., SOAP/WSBPEL engines): handlers in Apache Axis (1 and 2), Web Service Offerings Infrastructure (WSOI), ...
  - Aspect-oriented code weaving: AO4BPEL, ...

- Most can be used not only on providers, but also on consumers and third parties

---

**Using SOAP Message Intermediaries**

- Exchange of monitored values: a) in SOAP headers; b) using special push or pull operations
- Strengths: Realistic & consumer-specific measures
- Weaknesses: High run-time overhead (can be reduced with periodic/occasional monitoring); results depend on network location of measurement

---

**Using Probes (Probing)**

- Strengths: Run-time overhead can be lower
- Weaknesses: Results not consumer-specific, provider can treat probes in a special way; not possible to re-use SOAP headers to send monitored values; results depend on network location of probes

---

**Using Sniffers (Sniffing)**

- Strengths: Very low run-time overhead; measures can be realistic & consumer-specific
- Weaknesses: Unknown SOAP message’s Internet route; WS security technologies can be a problem; not possible to use SOAP headers to send monitored values; results depend on network location of sniffers

---

**Web Service Level Agreement (WSLA) Framework**

from IBM Research; uses the WSLA language

- Prototype: SLA Compliance Monitor – module 1 is simple, 2 is implemented, 3 & 4 are general purpose, 5 & 6 missing
- Special management port types (e.g., for value exchange)

- Strengths: Comprehensive approach to QoS management; support for management third parties; was used in practice; well-known

- Weaknesses: Run-time overhead

---

**WSLA Framework: Run-Time Use**
**CREation and MONitoring of Agreements (Cremona)**

- **Architecture** for WS-Agreement middleware
  - Agreement initiator & agreement provider roles
  - Agreement Management layers
- **Java library** that:
  1. implements WS-Agreement interfaces;
  2. provides management functionality for agreement templates and instances;
  3. defines abstractions to be implemented in domain-specific environments
- **Strengths**: Relates agreements with underlying resources; reusable for various domains
- **Weaknesses**: Needs additions to be used for WSes

---

**Several Control Approaches That Try to Meet QoS Guarantees**

1. **Manipulate which request is processed first**
   - Provider has several different request queues, e.g., one for each class of service
   - Scheduler within the provider decides from which queue to process a request, depending on QoS guarantees, current load, queue lengths, ...
2. **Manipulate thread priorities** for different requests and/or OS scheduling discipline
3. **General approach**: Manipulate allocation of resources for various requests
4. **Load balancing** between replicas

---

**Smartware**

- from Infosys: A. Sharma, H. Adarkar, S. Sengupta
- **QoS control**: Differentiated scheduling of requests based on context priorities
  - Context = info about provider application, user, and client device; sent by consumer in request SOAP header
- Based on *Apache Axis* SOAP engine, adds:
  - Interceptor – reads context info and determines priority
  - Scheduler – puts request into a queue for its priority; based on scheduling policy fetches a request from a queue
  - Dispatcher – forwards request to the provider
- **Strengths**: Rare work that performs QoS control
- **Weaknesses**: Scheduling uses limited information

---

**Re-composition of Web Services vs. Re-negotiation of Contracts**

- **Run-time adaptation** of WS compositions
  - a) Re-composition of Web services – more powerful
    - Special case: Switching between WSES (only 1 change)
  - b) Re-negotiation of contracts – faster, simpler, lighter
    - Special case: Switching between classes of service (also de/re-activation, deletion, creation, (dis-)allowing use)
- **Legend**: C – consumer; P – provider; CS – class of service

---

**Using Historical QoS Information for WS Selection: Possible Approaches**

- **From the same consumer**
  - Problem: When consumer did not previously invoke this operation of the provider Web service
- **From probes**
  - Problem: Easy for providers to give excellent QoS to probes, while bad QoS to real consumers
- **From all consumers**
  - Problem: Consumers have different characteristics (e.g., could be located on different continents)
  - Problem: Other consumers’ reports can be fake
**Using Historical QoS Information for WS Selection: Discussion**

- General problem: Circumstances of different invocations are different!
  - Example: When the number of provider’s concurrent consumers grows, it is likely that QoS perceived by individual consumers will drop
- General problem: Absence of targets/goals to guide control activities (including billing)
- Conclusion: Historical QoS information can be useful, but it provides no guarantees (and can even be misleading) => contracts are needed

**Approaches to QoS-Driven WS Discovery and Selection**

1. Provider as only source of QoS specifications
2. UDDI extensions with QoS information
3. Additional QoS information registry
   - The main issues:
     - QoS publication in a registry enables QoS-based selection (selection is difficult with option 1)
     - QoS changes much more often than WSDL, so QoS updates have to be propagated to consumers (difficult with option 2)
     - Complexity (highest with option 3)

**Some Observations on Industrial Products for WS QoS Management**

- They address many practical problems
  - Academic researchers should be aware of these works and their accomplishments
  - Some works contain advanced solutions that show how SLAs and/or policies can be used in practice
- Many products have significant limitations:
  - Crucial role of human administrators (i.e., not completely automated)
  - Limited/predefined choice of used QoS metrics
  - Lack of flexible formal machine-understandable QoS specification (instead, forms are used)

**Large System Management Suites**

- Contain many different management products
  - some related to WSes (or “business services”)
  - some related to performance (QoS) management of applications, computing systems, networks
- HP: Business Technology Optimization – BTO (formerly OpenView; includes SOA Manager)
- IBM: Tivoli (includes SOA Management Suite)
- Computer Associates (CA): Unicenter (includes Wily Web Service Manager)
- BMC Software: Patrol (includes Business Service Management - BSM)
- Microsoft (includes Application Center)

**Some Products for WS QoS Management from Smaller Companies**

- Often products (1 or more) addressing several management areas, incl. performance (QoS)
  - Actional SOA Management (includes SOAPStation Web Services Broker) – policies
  - AmberPoint SOA Management System (incl. Service Level Management) – custom-made SLAs
  - SOA Software (incl. Service Manager) – policies
  - Software AG (incl. webMethods Business Process Management Suite and SOA Governance) – policies
  - WestGlobal mScape (including Performance Management Module - PMM) – custom-made SLAs

**On QoS of Web Service Compositions**

- Given a set of WSes with known QoS, what is the QoS of their composition?
  - Very difficult question, without a general answer!
  - It is not a set of simple mathematical operations, because distribution of probability varies
  - Queueing methods might help (to some extent)
- How to select QoS of an individual WS to satisfy overall QoS of a known composition?
Approaches to (WS) QoS Management: Summary & Discussion

- Several different approaches to (general and Web service) QoS monitoring and control
- No one is best for all circumstances – knowing their advantages/disadvantages will help you choose
- In B2B scenarios, QoS monitoring with SOAP message intermediaries seems most flexible
- QoS control by (re-)allocating resources to meet QoS guarantees is necessary, but hard
- For QoS-aware WS selection, contracts have significant advantages over historical QoS data

Tools for WS QoS Management: Summary & Discussion

- There are many tools; very different in power
- Most are based on SLAs, some are based on policies
- No current tool or a set of tools addresses all WS QoS monitoring and control needs!
- Industrial products address many basic issues, but have limitations (e.g., in QoS control)
- Powerful, but expensive: system management suites
- Cheaper, but limited: smaller companies’ products
- Research tools tackle advanced problems, but commercial use requires additional features

IV. Introduction to Business-Driven IT Management (BDIM)

- Drivers for BDIM
- Precise definition of BDIM
- The BDIM framework
- Relationships between BDIM and other areas
- Some questions BDIM should help answer
- 3 examples of the BDIM approach
- Summary of BDIM and related issues

Drivers for BDIM

- Fast changing world (both business context/requirements and technologies)
- Business demands
  - agility, alignment and availability
  - business intelligence
  - mergers and acquisitions
  - regulatory compliance
- Technology demands
  - Modern, flexible architecture
  - service migrations and upgrades
  - IT automation and consolidation
  - security

IT Reality

A growing gap between the demands placed on IT and IT’s ability to deliver

Expectations on IT

Business demands
Technology demands

Versus

- IT’s capability to deliver
- Flat to modest increase in budget
- Manual processes
- Disparate tools/point solutions
- Silo’d organizations

Time
Business-Driven IT Management (BDIM): Precise Definition

BDIM is the application of a set of models, practices, techniques and tools ... to map and to quantitatively evaluate ... dependencies between IT solutions and business performance and ... using the quantified evaluation to improve the IT solutions’ quality of service and related business results.

Relationships between BDIM and Other Areas

BDIM addresses IT (technical) decisions, not business decisions. It is different from (but compatible with) the related areas:

- business management
- IT governance
- business process management (BPM)
- value-based software engineering (VBSE)
- autonomic computing

Some Questions BDIM Should Help Answer

- Which of the hundreds of incidents should I take care of now?
- What SLO values does the business need?
- Which changes to handle now considering impact, risk and our corporate risk attitude?
- Which services should be part of my portfolio?
- Which tests should be run on a release to lower risk to a level acceptable to my business?

Example 1: BDIM Approach to Incident Prioritization

- How do you prioritize incidents?
- Compute the likelihood of violation of an SLO in function of the time taken to close a jeopardy incident
- From this, compute the alignment with the business objectives
  - Alignment = probability of meeting objectives
- Prioritize the incidents to maximize alignment with the business objectives

Example 2: BDIM Approach to Defining SLAs

- How do you design service infrastructure?
  - You minimize cost to yield a certain QoS expressed in the SLA
- But how do you choose the SLOs?
  - 99%?, 99.5%?, 99.9%?, 99.99%?
  - 1 s, 2 s, ½ s?
  - Finger in the air ...
- BDIM approach: Calculate business loss due to unavailability and high response time and minimize cost+loss
Example 3: BDIM Approach to Service Testing

- Testing is a risk elimination activity
  - Risk is uncertainty
- Answer testing questions using risk models
  - Should a particular test be performed or not?
  - Which set of tests should be run?
- This is still at the research stage
  - Some related research in value-based software engineering (VBSE)

BDIM: Summary & Discussion

- BDIM is
  - A research agenda for decision support and information management that brings business objectives at the core of the IT Management decision making process.
- Achieved by
  - Modeling and reasoning over the objectives, the decisional criteria and the dependencies that link IT management and IT operations

Presentation Progress

I. Importance of QoS and business value
II. Specification of QoS
III. Management (monitoring & control) of QoS
IV. Business-driven IT management (BDIM)
V. From QoS to business value
VI. Summary, challenges and discussion

V. Overview of Possibilities for Enriching QoS Management with BDIM

- Current frameworks for improving business-IT alignment
  - IT Infrastructure Library (ITIL) v3
  - IT governance and COBIT
- Some industrial products for BDIM
- Example research projects in BDIM:
  - Management by Business Objectives (MBO) and Aline
  - WS-Policy4MASC and MASC middleware

Running IT Department like a Business

- Need to break down functional IT silos to deliver positive business outcomes

The Racing Car Example

- Better performance management (and configuration management) can improve performance of a racing car
- However, there are additional issues:
  - What if the car was not designed, built or tested correctly?
  - What is the goal (finish, win race, win championship)?
  - What strategy are we following?
Information Technology Infrastructure Library (ITIL)

- A set of guidelines and best practices for IT Service Management
- Provides generic, non-prescriptive guidance
- Alignment with and guidance on:
  - Industry best practices and international standards
  - Compliance to legislative requirements
- Important evolution towards BDIM (see ITIL v3), but not enough

Evolution of ITIL

- Centered on business value
- Creating a way to integrate IT processes, people, tools with business strategy and desired outcomes
- Structured according to the ITSM service lifecycle (instead of processes)
- Functionality and manageability are two sides of the same coin

ITIL v3 Books (1 of 2)

- Service Strategy
  - Strategy Generation & Demand Mgmt; IT Financial & Service Portfolio Mgmt
- Service Design
  - Requirements Mgmt for Availability, SLM, Capacity, Service Continuity, Information Security Mgmt; Supplier Mgmt & Service Catalog Mgmt
**ITIL v3 Books (2 of 2)**

- **Service Transition**
  - Planning & Support, Change, Service Asset & Configuration and Knowledge Mgmt; Release & Deployment Mgmt; Service Validation & Testing; Evaluation
- **Service Operation**
  - Access Mgmt; Operations and Event Mgmt; Incident, Problem, Request Fulfillment Mgmt
- **Continual Service Improvement**
  - Service Improvement, Service Measurement, Service Reporting, SLM and Service ROI

---

**The Continual Service Improvement (CSI) Model**

- **What is the vision?**
- **Business Value reason, goals and objectives**
- **Where are we now?**
- **Baseline Assessments**
- **How do we keep the momentum going?**
- **Measurable Assessments**
- **Where do we want to be?**
- **Service and Process Improvement**
- **How do we get there?**
- **Measurements and Metrics**
- **Did we get there?**

Reference/source: The Office of Government Commerce (OGC) ITIL v3

---

**The 7-Step Improvement Process**

1. Define what you should measure
2. Define what you should measure
5. Analyze the data: Relational? Trended? According to plan? Targets met? Corrective action?
6. Present and use the information: Summary action plans, etc.
7. Implement corrective action

Goals

Reference/source: The Office of Government Commerce (OGC) ITIL v3

---

**IT Governance**

- A set of organization’s policies, plans, and processes that govern how the organization uses its IT resources
  - Difference from management: longer-term vs. shorter-term; strategy vs. tactics
  - Aligning IT investments with overall business goals and strategies
  - Determine who makes IT-related decisions and who is responsible for the outcomes

---

**Control Objectives for Information and related Technology (COBIT)**

- An IT governance framework that helps:
  - Align IT with the business, so that IT enables the business and maximizes business benefits (value)
  - IT resources are used responsibly
  - IT risks are managed appropriately
  - Based on industry best practices
  - Links IT activities with business requirements
  - Organizes IT activities into a generally accepted process model
  - Identifies major IT resources to be leveraged
  - Defines control objectives to be considered

---

**COBIT Framework**

- Reference/source: IT Governance Assurance (IGA) - Control Objectives for Information and related Technology (COBIT)
**Specification of Goals for Business Processes**

- Services & processes need to have clearly defined goals/objectives and measurements to guide both design-time and run-time decision making.
- Several academic works enable specification & processing of goals for business processes e.g., using the OMG Business Motivation Model (BMM), Tropos, i*, ...
- Much related work in requirements engineering
- Few works relate goals to run-time management

**ITIL v3 and COBIT in Relation to BDIM**

- Business-IT alignment has been identified as a problem several decades ago
- ITIL v3 does not provide concrete “how-to” models/techniques to use in automatic tools
- COBIT does not provide ways of propagating objectives throughout the IT organization (cmp. IT business scorecard)
- Both ITIL v3 and COBIT are important in the evolution towards BDIM, but are not enough

**Some Current Industrial Products for BDIM**

- HP Business Technology Optimization (BTO) Datawarehouse and CIO scorecard
- IBM Cognos Business Intelligence
- BMC Dashboard for Business Service Management
- CA Cleverpath AION BPM, BRE (Business rules expert)

**Management by Business Objectives (MBO)**

- A methodology for quantitative evaluation of alignment of business objectives of alternative IT options
- Aimed at supporting decisions making at the IT level
- Information model derived by COBIT objectives and balanced scorecard
- Quantitative definition of alignment with business objectives as the likelihood that the objectives will be met

**MBO Aline: Calculating Alignment**

- The alignment engine of the MBO framework

**Management of Non-Financial Business Value Metrics**

- Financial business value metrics modeled and monitored in the past relatively successfully e.g., prices and penalties in SLAs
- Accounting sub-systems
- Non-financial business metrics are difficult to model (incl. monetize), monitor, and control
- Modeling challenge: diversity of characteristics
- Monitoring: business intelligence systems?
- Control is still a challenge for both
- Business strategy determines which business value metrics to maximize (it is not always profit)
**Motivation for Business Process Adaptation Maximizing Business Value**

- **Alternatives:**
  - A: 99% availability, mid price
  - B: 95% availability, cheap
  - C: 99.99% availability, expensive

**WS-Policy4MASC Overview**

- Extends WS-Policy with policy assertions and details necessary for run-time management
- Support for management of Web service compositions (e.g., built-in actions, events)

**WS-Policy4MASC: UtilityPolicyAssertion (1/2)**

- Main distinctive features of WS-Policy4MASC:
  - Specification of both financial and (monetized) non-financial business values
  - Policy conflict resolution: maximize business value
- A utility policy assertion:
  - situations to which it applies (When construct)
  - management party and beneficiary party, ...
  - 1 or more business values
- A business value has a monetary amount and a business value type

**WS-Policy4MASC: UtilityPolicyAssertion (2/2)**

- 8 business value types: combination of characteristics along 3 dimensions
  - Tangible (financial) vs. intangible (non-financial)
  - Agreed vs. possible
  - Benefits vs. costs
- Example: AU$10 intangible possible benefits (models aspects of customer satisfaction)
- Business values can be not only absolute, but also relative (e.g., PriceB=0.75*PriceA)

**WS-Policy4MASC: MetaPolicyAssertion (1/2)**

- Policy conflict: several action policy assertions can be applied, but only 1 should
  - E.g., "skip activity X" vs. "replace activity X with Y"
- Policy conflict resolution with meta-policies
- A meta-policy assertion:
  - list of 2 or more conflicting (alternative) action policy assertions
  - specification of business strategies maximizing specified business value types
- Strategies classified along dimensions based on business value types

**WS-Policy4MASC: MetaPolicyAssertion (2/2)**

- E.g., ‘tangible-only’ vs. ‘intangible-only’ vs. ‘tangible+intangible’
- Tiebreaking in case of close alternatives
  - E.g., ‘tangible+intangible’ instead of ‘tangible-only’
- Time limit and cost limit
- Example strategy: ‘intangible-only agreed+possible benefits+costs with tiebreaking tangible+intangible’ (models maximization of customer satisfaction)
- Policy conflict resolution algorithm

---

**Main distinctive features of WS-Policy4MASC:**

- Specification of both financial and (monetized) non-financial business values
- Policy conflict resolution: maximize business value

**WS-Policy4MASC: UtilityPolicyAssertion (1/2)**

- Main distinctive features of WS-Policy4MASC:
  - Specification of both financial and (monetized) non-financial business values
  - Policy conflict resolution: maximize business value

**WS-Policy4MASC: UtilityPolicyAssertion (2/2)**

- 8 business value types: combination of characteristics along 3 dimensions
  - Tangible (financial) vs. intangible (non-financial)
  - Agreed vs. possible
  - Benefits vs. costs
- Example: AU$10 intangible possible benefits (models aspects of customer satisfaction)
- Business values can be not only absolute, but also relative (e.g., PriceB=0.75*PriceA)
**Manageable & Adaptable Service Compositions (MASC) Middleware**

- Platform independent components
- Platform specific components

**Unified Modeling Language (UML) Profiles for WS-Policy4MASC**

- Design time (modeling & tooling)
- Runtime (configuration & execution)

**From QoS to Business Value: Summary & Discussion**

- The transition from management of QoS to management of business value is not simple
- Could/should be done in several directions
- ITIL v3 and COBIT available now, but do not provide concrete “how-to” for use in automatic tools => only one step towards BDIM
- Several industrial products with some BDIM capabilities, but many improvements possible
- Compared to QoS management, not as many research projects, but the number is increasing

**Presentation Progress**

I. Importance of QoS and business value
II. Specification of QoS
III. Management (monitoring & control) of QoS
IV. Business-driven IT management (BDIM)
V. From QoS to business value
VI. Summary, challenges and discussion

**VI. Summary, Challenges for WS QoS Management and BDIM, Discussion**

- The main “take home” points
- Summary of past results on WS QoS management
- Some research topics for WS QoS management
- Summary of past results on BDIM of WSES and business processes
- Some challenges & research topics for BDIM
- Resources

**The Main “Take Home” Points**

- Web service QoS and business value specification & management is crucial
  - market differentiation, efficiency, agility, ...
- Technical QoS is important, but business value is even more important for customers
- Many things you can do today on WS QoS
- Some things you can do today towards BDIM of WSES and business processes
  - but a growing research area ...
**Summary of Past Results on WS QoS Management**

- Several QoS specification approaches
  - Contracts, SLAs, classes of service, policies
  - Know strengths and weaknesses, circumstances
- Many languages for WS QoS specification
  - General standardized frameworks such as WS-Agreement and WS-Policy are not enough
- Several approaches to QoS monitoring/control
  - None is “perfect”; control is difficult
- There are many industrial and research tools for WS QoS management, but with limitations

**Summary of Past Results on BDIM for WSes and Business Processes**

- Recognition that business value matters!
- Industry acceptance of ITIL v3 and COBIT
- Definition of BDIM as a research area
  - leveraging several other areas; multi-disciplinary
- Some approaches to and information models (languages) for specification of business value
- Some research projects and (to some extent) industrial products
- Not all BDIM work has the “BDIM” label!

**Some Research Challenges for BDIM**

- How do we cross the business-IT chasm?
  - Modeling of business values, strategies, (long-term) goals/objectives
  - Modeling their links with technical (IT) metrics
  - Predicting consequences of actions on IT/business, having in mind (long-term) uncertainties and risks
  - Making decisions maximizing business value
- The right level of abstraction for models
  - Diversity of business-IT links impacts model reusability
  - Appropriate/accurate values for model parameters

**Some Research Topics on Modeling of Business Values**

- Explicit description of various characteristics of business values
  - e.g., financial or not, agreed or not, ...
- Monetization of non-financial business values
  - e.g., 80% customer satisfaction => $100 per day
- Uncertainty of business values
  - Related to risk, trust, ...
- Time value of money (for the long term)

**Some Research Topics on Modeling of Business Strategies**

- Explicit description of various characteristics of business strategies
- Predicting/processing a chain of future events
  - The temporal scope must be limited
- Calculating the overall business value
  - Not 1 number, but a set of values
- Comparing overall business values
  - Constraints (e.g., cost limit)
  - Priorities of business value types and tiebreaking
Non-technical Challenges for BDIM

- The technical mindset of the community
  - Business? What business?
- Selling the vision to business
  - Sell on business terms, not on technical terms
  - Do we know the real business value of our approach?
- Validation is not simple
  - Complex socio-technical systems

Resources

- Publications are scattered between many different conferences, journals, and books
  - The best resource on BDIM are BDIM workshops (2006-2009) at NOMS/IM, published by IEEE
- Past specialized tutorials by the presenters
  - ICWS/SCC 2005 and CEC/IEEE 2006 on WS QoS
  - NOMS 2008 on ITSM & BDIM
  - Detailed lists of resources
- http://www.businessdrivenITmanagement.org
- http://groups.google.com/group/BDIM
- Ask the tutorial presenters (e-mail on Slide 2)