What Is Quality of Service (QoS)?

- **Functionality/service** = “WHAT operations does the system execute?”
  - E.g.: Returns current price for a stock symbol
- **Quality of service (QoS) = “HOW WELL” the system performs its operations?”
  - E.g.: Average response time is 2 seconds, availability in the last 24 hours is 99%,...
  - Price and security info sometimes included
  - Synonyms: non-functional properties, ’ilities’
  - QoS exists even when not specified or measured

The Need for QoS Specification for WSes

1. Publish It differentiates from competitors
   Service Provider (Supplier, Server)

   Service Registry (Broker, Directory)

2. Find It determines the best match
   Many WSes with similar functionality

3. Bind It is used for management (monitoring and control)
   Service Consumer (Requester, Client)
Definition of Management – Monitoring

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

Definition of Management – Control

- **Control** tries to ensure that the 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: e.g., penalty for not meeting guaranteed response time is AU$1.00, ...
  - Modification of requirements/guarantees
  - Notification of human administrators

Benefits of QoS Management

- **QoS (performance) management** helps to:
  - ensure correct operation,
  - attain or surpass guaranteed QoS,
  - discover and fix problems,
  - accommodate change,
  - balance price/performance ratio,
  - maximize profits, ...
- **QoS specification** is essential for QoS mgmt.
  - You cannot control what you do not monitor
  - You cannot monitor what you do not describe

Presentation Progress

- About NICTA
- Need for QoS specification for Web services
  - **Main concepts and languages for QoS specification for Web services**
  - Some issues in QoS analysis for Web services and their compositions
- Conclusion, resources for further study, Q&A

The Main QoS Specification Concepts

- **QoS specification** – description of what, where, when, how to monitor and control
- **QoS information** – QoS specifications + QoS analyses + monitored QoS metric values
- **Contracts** – formal agreements
  - Service Level Agreements (SLAs) – QoS contracts
  - Classes of service – predefined SLAs
- **Policies** – high-level operation and management goals and/or rules

Contract

- **Binding and enforceable formal agreement** between 2 or more parties
  - Defines requirements and guarantees of parties
  - Can be used for monitoring and control
- Not only QoS information
  - A WSDL file is a contract
  - Can contain information about prices, penalties, ...
- **QoS description and QoS differentiation**
  - Different QoS contracts for different (types of) consumers
WS-Agreement

• **General framework** for XML specification of agreements and agreement templates
  – plus a simple agreement negotiation protocol and run-time agreement monitoring interface
  – Standardized by the Global Grid Forum (GGF)
  – Intended for multiple domains, not only WSes
• **Strengths**: widely used
• **Weaknesses**: no built-in constructs for QoS specification – any language can be used
  – This flexibility can produce incompatibility

WS-Agreement: Agreement Structure

• **Name**
• Context: involved parties (initiator & provider); expiration; template name; related agreements
• ExactlyOne or OneOrMore or All compositors
  – service description terms: service descriptions, service references, service properties
  – guarantees: service scope, qualifying condition, service level objective (SLO), business value list
• **Constraints**

Service Level Agreement (SLA)

• A special type of contract for QoS (and often price/penalty) requirements and guarantees
• Many different formats, one of which is:
  • **Parties** (incl. 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: what happens if SLOs met / 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 (up to the limit of 100 concurrent invocations), RT-OP1-C will be less than or equal to 2 seconds
**Action guarantees**: (AG1) If SLO1 was met, C pays P price of AU$0.20 per invocation;
(AG2) If SLO1 was not met, P pays C penalty of AU$0.10 per invocation

QoS Specification Must Be Precise

• Which QoS metric, how measured, when, where, by which party, circumstances, …
• It is a common mistake to specify SLOs without limiting the number of requests
  – E.g., response time of operation X of Web Service A is max 1 second
  – What if there are 1000 (or million) concurrent requests competing for the same resources?
• Response time (availability) depends on the number of requests!

Strengths and Weaknesses of SLAs

• **Strengths**:
  – Formal contract specification of QoS and related management aspects
  – Widely used
• **Weaknesses**:
  – Negotiation of custom-made SLAs can require complex analysis of offers and generation of counter-offers (using templates can alleviate this)
  – Management of many concurrent custom-made SLAs can be complex and with high overhead
  – Cannot be used for QoS-enabled WS selection
Web Service Level Agreement (WSLA)

- **QoS language & management infrastructure**
  - From IBM Research (H. Ludwig, A. Keller, et al.)
  - Compatible with, but not restricted to WSes
- Custom-made SLAs (the example format, plus many additional details)
- **Strengths**: detailed and precise specification of monitoring and control; several tools exist; used in practice; widely referenced
- **Weaknesses**: the weaknesses of custom-made SLAs; QoS metrics defined within SLAs

Class of Service

- A special type of SLA that is not custom-made, but predefined, anonymous & reusable
  - 1 provider offers many classes of service with same functionality, but different QoS
  - 1 class of service can be used by many consumers
- **Strengths**: usable for QoS-enabled WS selection, no complex negotiation (only simple selection), simpler management, lower run-time overhead, faster adaptation
- **Weaknesses**: limited choice

Policy

- **High-level** (possibly business-level) operation and management goals and/or rules
- A classification of policy types:
  - **Goal policies**: Describe desired state (e.g., “Response time of operation A is less than 2 sec”)
  - **Action policies**: Describe what should happen (e.g., “If response time of operation A is greater than 2 sec, provider pays penalty of US$0.10”)
  - **Utility policies**: Quantify “goodness” of a state (e.g., “Add to the goodness measure [2 sec - response time of operation A] * 10 units”)

WS-Policy

- Web Services Policy Framework (WS-Policy)
  - **General**, flexible, and extensible framework for specification of policies for WSes
  - **Strengths**: policies can be in or out of WSDL files, some reusability constructs, …
  - **Weaknesses**: no constructs for actual QoS specification – this is left for extensions, …
  - Several QoS extensions of WS-Policy
    - E.g., WS-Policy4MASC: detailed, precise, with unique support for business value metrics

Discussion of QoS Specification Options

- **Contracts vs. policies**
  - Analogies: SLOs can be viewed as goal policies, action guarantees as action policies
  - Different management infrastructures
  - Internal policies and external multi-party contracts
- **Which type of contract to use** depends on circumstances
  - For comprehensiveness: general contracts
  - For flexibility of QoS specification: custom SLAs
  - For low overhead: classes of service

Presentation Progress

- About NICTA
  - Need for QoS specification for Web services
  - Main concepts and languages for QoS specification for Web services
  - Some issues in QoS analysis for Web services and their compositions
  - Conclusion, resources for further study, Q&A
QoS (Performance) Analysis

- How to determine appropriate numbers for SLOs or goal policies?
- This is very difficult and a combination of approaches should be used:
  - Analytical methods (queueing networks, layered queueing networks, software performance engineering, …)
  - Simulation (e.g., discrete-event simulation)
  - Monitoring (the system, its parts, related systems)
- Some amount of “guestimates” necessary

Using Historical QoS for WS Selection

- Collect past QoS measurements and publish in a directory for QoS-enabled WS selection
- Some potential problems:
  - From all consumers: consumers have different characteristics (e.g., geographic location), other consumers’ reports can be fake
  - From probes: easy for providers to give good QoS to probes, while bad QoS to real consumers
  - From the same consumer: what if it did not previously invoke this operation of the provider, circumstances of different invocations are different

QoS Depends on Circumstances

- It is a common mistake to rely on historical QoS without considering circumstances
  - E.g.: When the number of provider’s concurrent consumers grows, it is likely that QoS perceived by individual consumers will drop
  - Past QoS measurements with different circumstances (contexts) can be misleading!
  - Historical QoS info (even with same context) can be useful indication, but it provides no guarantees and cannot guide control activities

Determining QoS in WS Compositions

- How to select QoS of an individual WS to satisfy an overall QoS requirement from a given composition?
- Given a set of WSes with known QoS, what is the QoS of their composition?
  - E.g., if max response time of all WSes WS1-WS5 is 1 sec, is the max overall response time 4 sec?

Analyzing QoS in Compositions Is Complex

- It is a common mistake to think that response time of a sequence of services is the sum of response times of composed services
  - Under some circumstances, it is appropriate
  - But what about # of requests & context?
  - What if dependencies (e.g., WS4–WS3)?
  - What is the request probability distribution?
- Advanced analytical methods often needed
  - Math is only an abstraction of reality – clarify assumptions & validity limits of all analyses!

Business Value Metrics

- Business value metric = any measure of business worth
  - Financial: income, cost, profit, margin, ROI, …
  - Non-financial: # of customers, market share, customer satisfaction, … (argument for capturing them: balanced scorecard – BSC)
- Is it QoS? Yes, but not in the traditional sense
- Business value metrics are subjective
  - It is not the same for consumer, provider, …
- Related term: key performance indicator (KPI)
Technical QoS Management Is Not Enough

• Technical QoS is important, but it is subordinate to business value
  – E.g., do customers 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
  – E.g., will 1% higher availability increase profits? Not always … (even if yes: amounts differ)
  – Depend on domain, context, business strategy, …
• Business-driven IT management (BDIM) researches maximizing business value metrics

Presentation Progress

• About NICTA
  • Need for QoS specification for Web services
  • Main concepts and languages for QoS specification for Web services
  • Some issues in QoS analysis for Web services and their compositions
  • Conclusion, resources for further study, Q&A

Many Results on QoS for Web Services

• Many results on QoS specification and analysis for Web services
• Key QoS specification concepts: contract, SLA, class of service, policy
• Popular languages: WS-Agreement, WSLA, WS-Policy (but they are not enough)
• Which one to use depends on circumstances!
• There is no “silver bullet”, you have to know strengths/weaknesses of various approaches

QoS Specification & Analysis Is Complex

• QoS specification and analysis for Web services is more difficult than it may seem
• Three common mistakes:
  1. Specifying QoS guarantees without limiting the number of requests
  2. Using past QoS to predict future QoS without considering request context
  3. Calculating QoS of a WS composition from QoS of individual services without discussing assumptions and validity limits of analyses

Some Resources for Further Study

• List of many relevant resources (in PDF) is available through my Web page: http://nicta.com.au/people/tosicv/tutorials
• My specialized conference tutorials contain many additional details and can be repeated
• NICTA short course and seminars by my colleagues Paul Brebner & Liam O’Brien
  • http://www.businessdrivenITmanagement.org
• Contact our NICTA research group – we seek industry collaboration!


vladat (server: computer.org)