Modeling Quality of Service and Continuous Media Communications
Tom Ritter, Dr. Marc Born
Workshop on UML for Enterprise Applications
October 2002
Agenda

- Telecom Domain
- CORBA Components
- CCM Metamodel
- Continuous Media
- QoS Metamodel
- Current Status
Telecom Domain Characteristics

• Distributed systems
  – Remote deployment and configuration

• Large-scale systems
  – Software components as building blocks
  – Component market

• Special demands
  – Stream interaction
  – Better quality of service to achieve higher price
Modeling Concepts

• There are a variety of modeling concepts for functional properties
  – Interface,
  – Operation, ...

• Such concepts are supported by different techniques
  – UML,
  – EDOC,
  – CORBA, ...

• Those techniques often lacks of support for non-functional properties (e.g. dependability, bandwidth, priority).

• Existing approaches support static set of non-functional properties.
General Aims of Work

• Development of a metamodel for QoS
• Application of this metamodel to CCM
• Implement these concepts

• Constraints
  – Embrace continuous media streams
    • Transmission protocol and presentation layer
  – Usage of Open Source software if possible
  – Integration with already existing systems
  – Work was aimed on CCM only
CORBA Components

• Based on CORBA (OMG)
• Platform and language independent
• Extend the CORA Object Model by introducing the notion of component
  – Multiple interfaces
  – Unit of deployment
• Container model
  – Simplifies the service interfaces
• Deployment and configuration
  – Remote installation and configuration of applications
CORBA Components

- No support for streams
- No support for QoS
CIF – Component Implementation Framework

- CORBA Base Set

- Component IDL Extensions
  - New meta classes
    - `ComponentDef`, `ProvidesDef`, etc.

- CIF Extensions
  - Component Implementation
  - New meta classes
    - `ComponentImplDef`, `SegmentDef`, etc.
Component IDL Package
Streaming Metamodel

- Defined as extension to CIF

- Stream types

- Stream interaction points
  - SourceDef
  - SinkDef
Streaming Metamodel

- Stream interaction points (SourceDef, SinkDef)
- Associate stream ports and components
- Adds the stream ports as component features
Streaming Metamodel

- Stream type
- Implicit association of stream type and interaction element

Value Def
(from BaseIDL)

- is Abstract: boolean
- is Custom: boolean
- is Truncatable: boolean

Stream Def
QoS Metamodel

• Aims
  – Support free set of QoS Categories
  – Support a multi category QoS Contracts
  – Support for multiple domains

• New Meta classes
  – Contract Type
  – Dimension
  – Binding
QoS Metamodel - Contract Type

- Contract Type is an aggregation of dimensions
- Example
  - Contract type: bandwidth
  - Dimension: bit_rate
QoS Metamodel - Dimension

- Dimension is single or composed
- To specify more complex dimensions
QoS Metamodel - Binding

- Associate a Contract Type with a Context
- The Binding can be restricted to interaction elements
- Context and Interaction element are connections to modeling techniques
Example

• Component C
• Provides interface i
• Contract Binding on interface
• Bandwidth Contract
QoS and CORBA Components

- Client has requirements
- Server offers contracts
- Client accepts one offer
- Contract will be established
Starting Position

• Open Source CCM Implementation
  – Renamed to Qedo

• Code generator
  – Repository based
  – Handcrafted repository
  – Only text-based front-end

• H.323 Streams implementation (Open Source)
• Real Streams implementation
Development Steps

- Metamodel for QoS
  - Applied to CCM

- Integration of both streams types
  - H.323, Real Streams

- QoS Contracts are managed by QoS providers

- Automatic negotiation of contracts at connection time

- Integration with Qedo deployment
Development Steps

• „UML Profile for QoS ...“ RFP
  – In contact with one of the submitters
  – Try to integrate the concepts

• Code generator
  – Generate the repository out of the metamodel (MOF)
  – Graphical front-end (e.g. Rose)

• Improvement of the concepts
  – QoSProvider Framework (CIF extension)
  – Chaining QoS contracts
References

• Fraunhofer FOKUS
  – [http://www.fokus.fhg.de/platin](http://www.fokus.fhg.de/platin)
  – Tom Ritter, ritter@fokus.fhg.de

• Qedo – QoS Enabled Distributed Objects
  – Open Source CCM Implementation
  – [http://qedo.berlios.de](http://qedo.berlios.de)