

# Modern Software Engineering Methods for IP-QoS Resource Pool Management

Gerald Eichler

T-Nova Deutsche Telekom Innovationsgesellschaft m.b.H.  
Technologiezentrum Darmstadt



Anne Thomas, Falk Fünfstück

Dresden Technical University, Dept. of Computer Science, Institute SMT



Fabio Ricciato

Consorzio di Ricerca sulle Telecomunicazioni - CoRiTeL, Roma



Charilaos Tsetsekas

National Technical University of Athens  
Dept. of Electrical and Computer Engineering



Martin Winter

Siemens AG, Munich, ICN WN CC EK A19



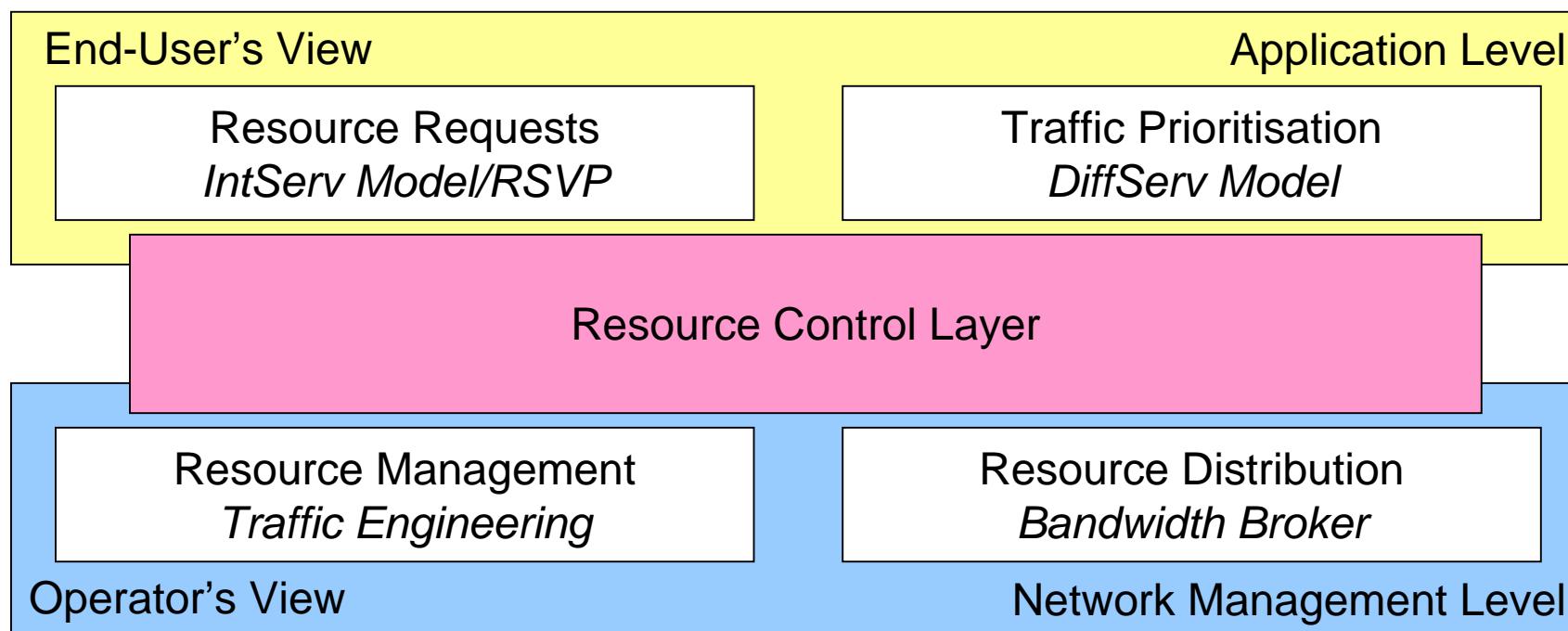
# Outline

- **Introduction: Gap between Applications and IP-Networks**
- **Resource Control Layer Model**
  - Architecture and Components
  - Development Methodology
- **Resource Offering**
  - Network Services and Application Profiles
  - Accountability Knowledge Level & Item-Item Description Analysis Pattern
- **Resource Management**
  - Resource Distribution Algorithms and Resource Pool
  - Composite Structure Design Pattern
- **Conclusion: Extension of the Basic Model**

# State of the Art

## ■ Gap between Application Level and Network Management Level

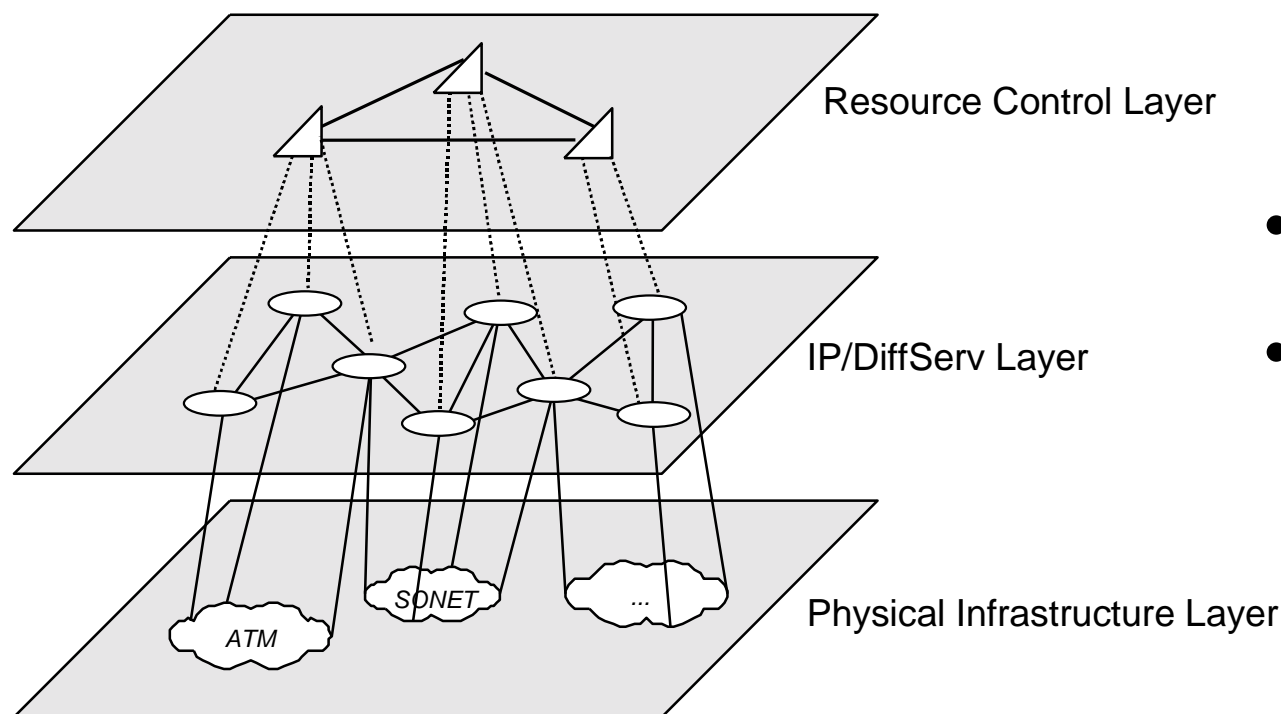
- existing and upcoming QoS supporting models vs.
- classical traffic engineering methods



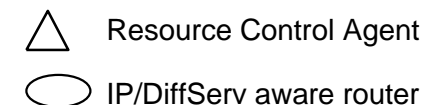
# AQUILA's Resource Control Layer

## ■ AQUILA ... Adaptive Resource Control for QoS Using an IP-based Layered Architecture

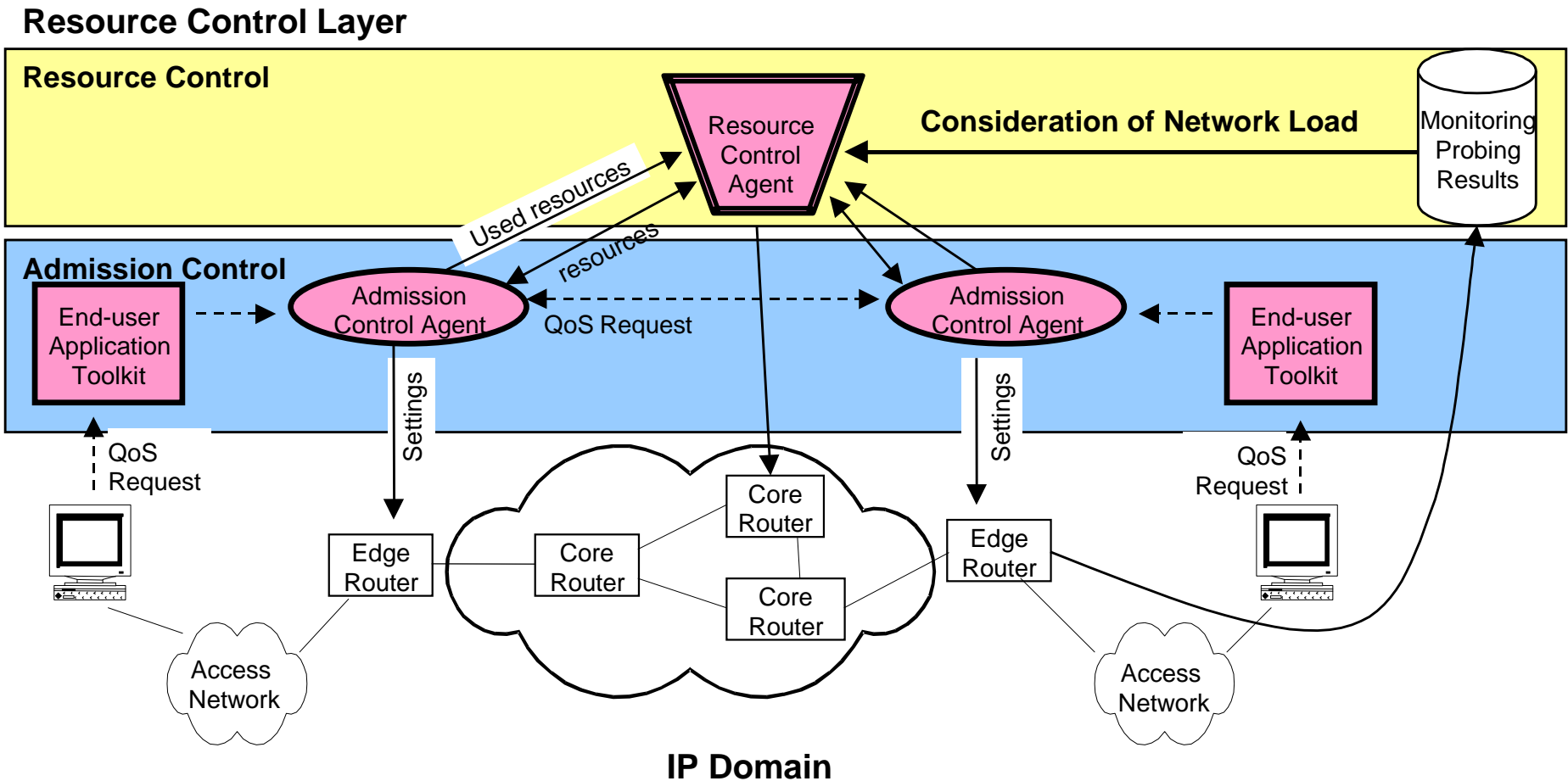
- IST project of the Next Generation Networks Cluster (IST-1999-10077)
- URL: <http://www-st.inf.tu-dresden.de/aquila/>



- introduction of additional Resource Control Layer
- agent based solution on top of a DiffServ aware IP-network



# Resource Control Layer - Architecture



# Resource Control Layer - Main Components

## ■ Resource Control Agent (RCA)

- keeps track of overall resource usage within an IP domain
- distributes resources to Admission Control Agents
- one instance per IP domain

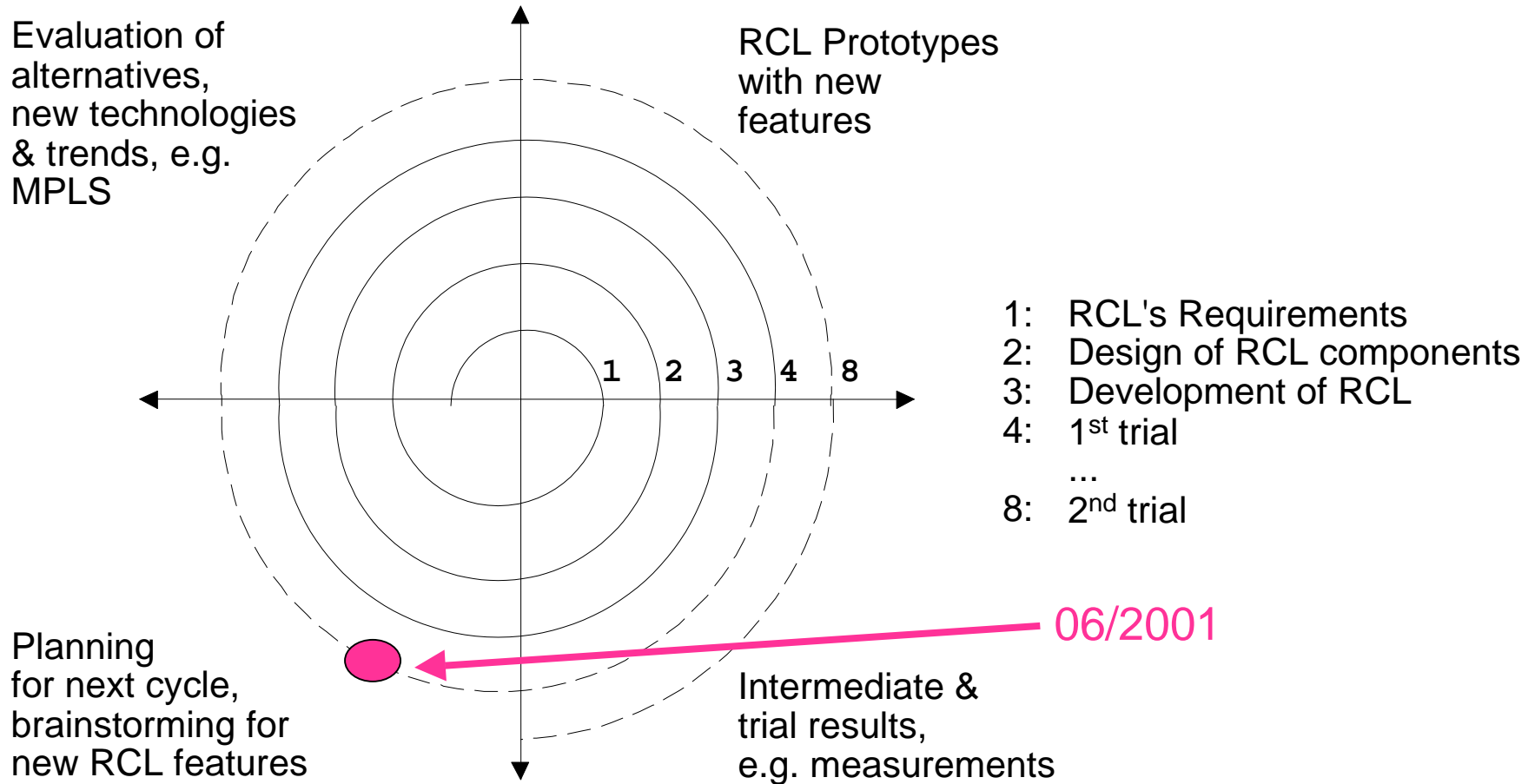
## ■ Admission Control Agent (ACA)

- performs authentication, authorisation and admission control
- acts autonomously within assigned limits
- one instance per edge router

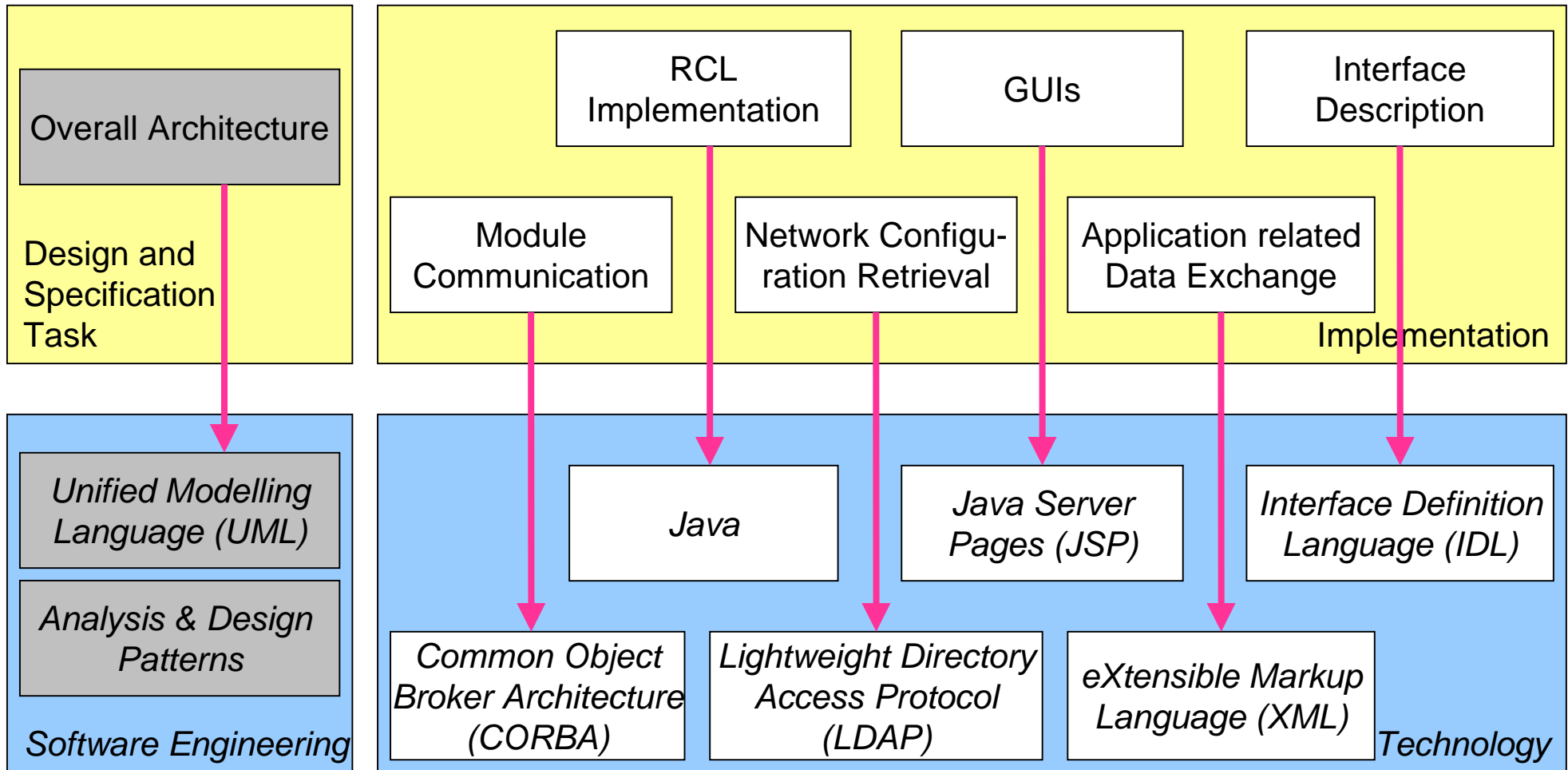
## ■ End-user Application Toolkit (EAT)

- offers Network Services to the customer
- translates and forwards QoS requests from the application to the ACA
- provides proxies for selected applications

# Software Engineering Process



# AQUILA's Software Development Support



# Resource Offering - Network Services

## ■ Objective

- dynamical assignment of network resources to requesting applications

## ■ Solution: Network Services

- suited to data type (streaming multimedia, VoIP, ...)
  - Premium Constant Bit Rate, Premium Variable Bit Rate
  - Premium Multi-Media, Premium Mission Critical
  - Standard
- defined by the network operator
- composed of parameter sets
  - combination of QoS performance parameters ranges (delay, loss, ...)
  - set of traffic parameters ranges (peak rate, token bucket size, ...)
- requested by application / end-user
  - high complexity of the request
  - mapping mechanism needed between end-user and network level

# Resource Offering - Mapping

## ■ Application QoS Request Message at Network Level

- list of low level network specific parameters
- include a *Network Service*
- *Technical Characteristics*

## ■ Application QoS request at Application Level

- correlated to the application type (VoIP, video conference, ...)
- verbal description of the wished quality (picture size, sound quality, ...)
- *Session Characteristics*



## ■ Two way Mapping between both Levels done by the EAT

- using the *Application Profile* information

# Resource Offering - Application Profile

## ■ Application Profile (Definition)

- all relevant session and technical characteristics of an application
- described as data structure & analysis pattern

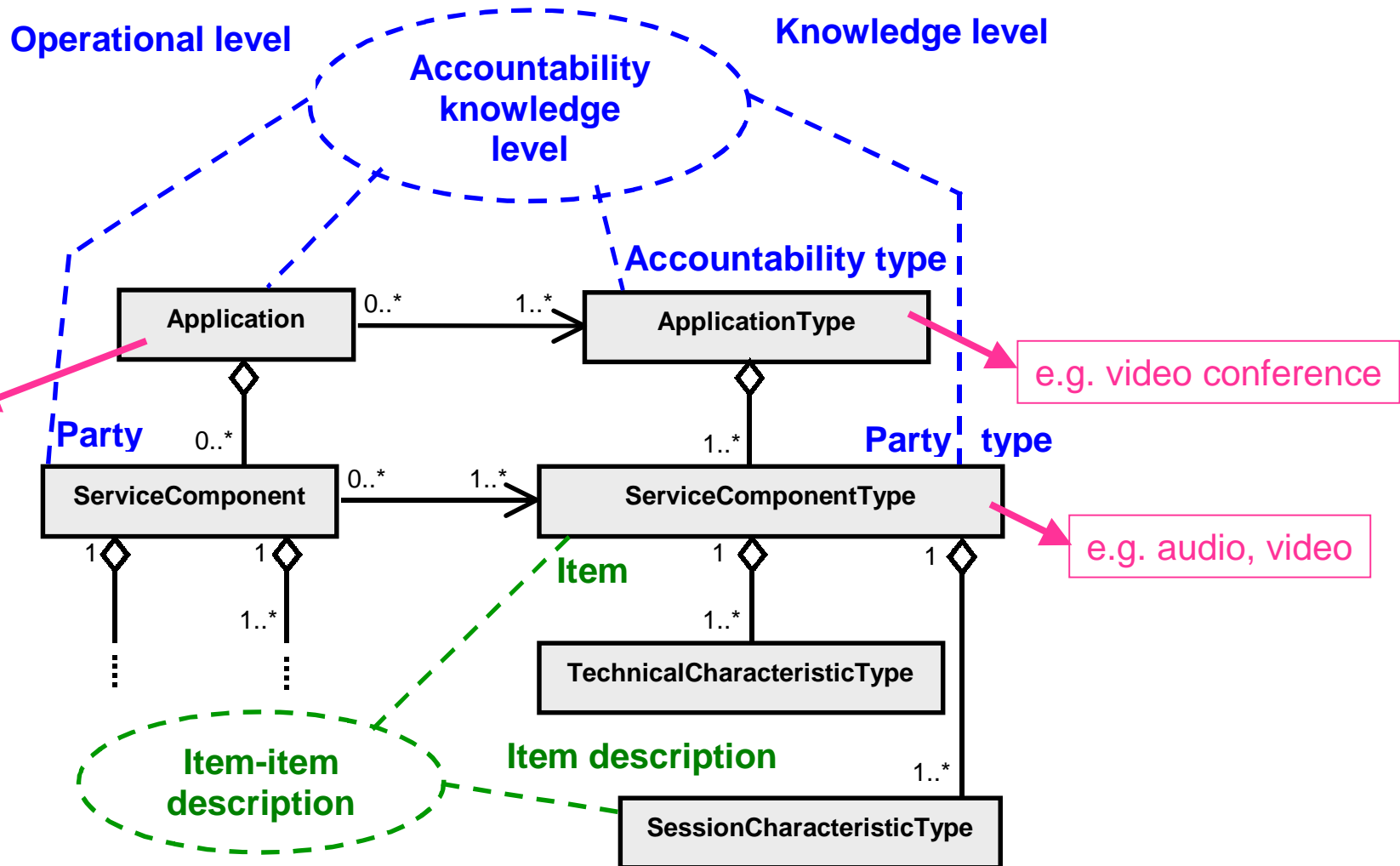
## ■ *Accountability Knowledge Level Analysis Pattern*

- operational level
  - corresponds to the concrete application
- knowledge level
  - meta-level description (application type)
  - rules governing the structure

## ■ *Item-Item Description Analysis Pattern*

- relation between concrete objects and a shared description
  - each item object is described by a description item

# Analysis Patterns



# Resource Management - Distribution

## ■ Resource Distribution Algorithms

- based on the “Hose Model”

- Request Rule
  - Whenever the amount of used resources raises above the upper watermark, ask the root for a block of more resources per Network Service.
- Release Rule
  - Whenever the amount of used resources falls above the lower watermark, return a block of free resources to the root per Network Service.

- Dynamic Distribution
  - Dynamically shift resources within the group
- use of the **Resource Pool** data structure

# Resource Management - Resource Pool

## ■ Hierarchical Structure

- “groups of groups”
- hierarchically organised tree

## ■ Resource Limits

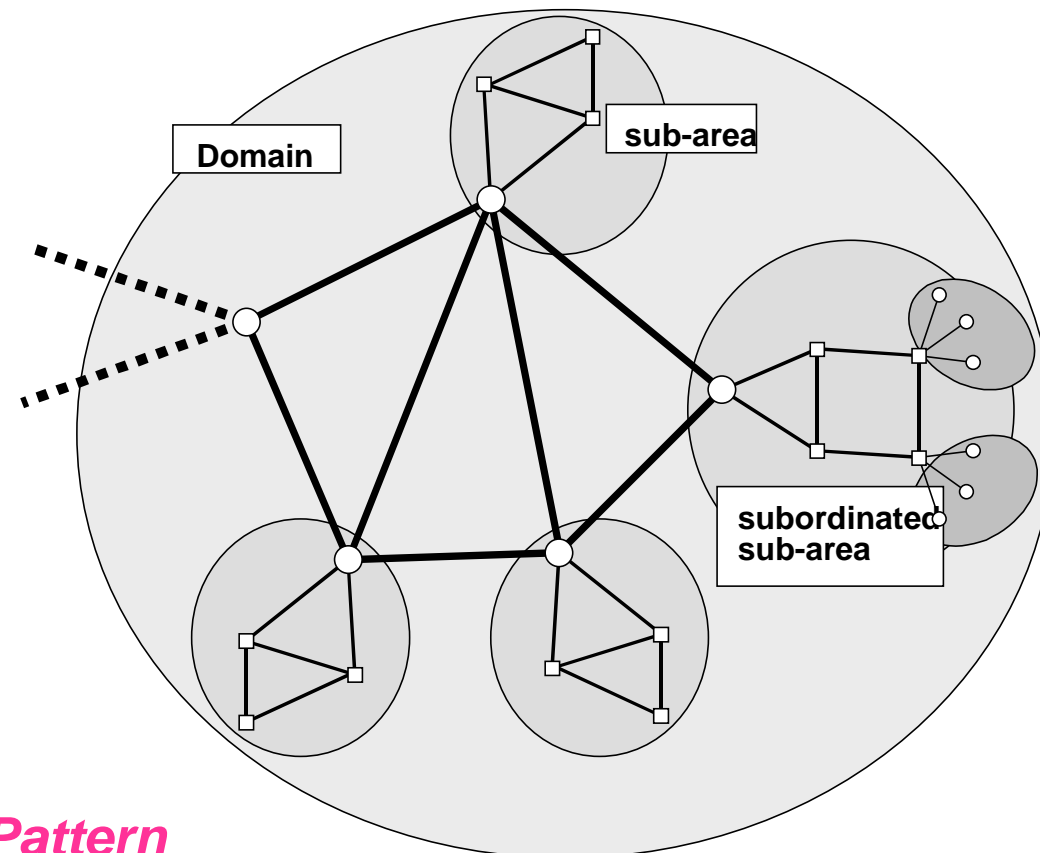
- limit amount of QoS traffic from each edge router

## ■ Group neighboured Routers

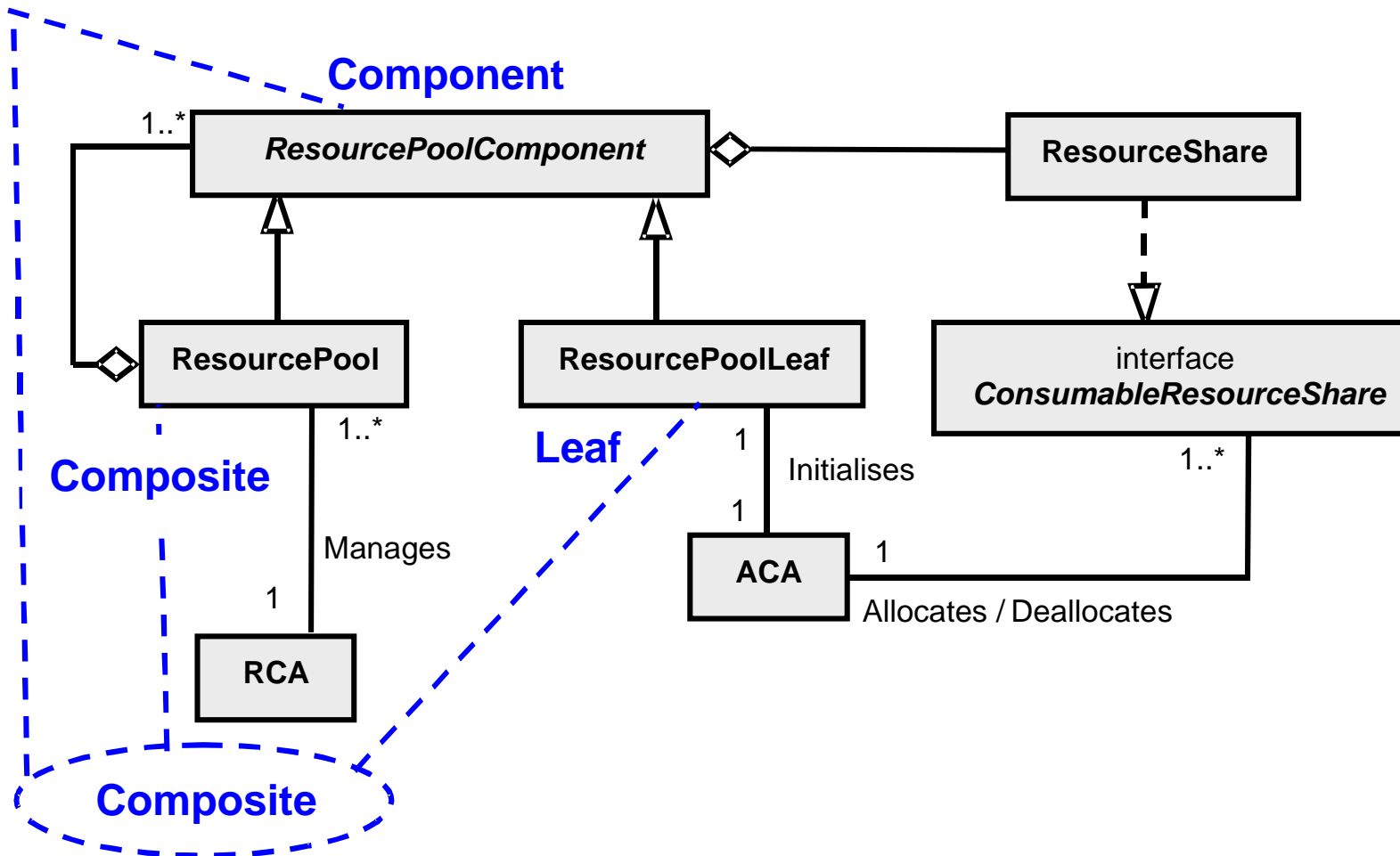
- limit amount of QoS traffic from each adjacent group

## ■ Composite Structure Design Pattern

- data structure underlying the resource distribution algorithm
- representation in a tree structure (part-whole hierarchies)



# Composite Structure Design Pattern



# AQUILA Project Goals for the 2<sup>nd</sup> Phase

## ■ Scalability

- single domain solution  $\Rightarrow$  multiple domain solution

## ■ Flexible “Vertical” Resource Shift

- dynamical resource sharing between Network Services

## ■ Quality Adjustment

- measurement driven control loops
  - to optimise admission control
  - to adapt router parameters (queuing, scheduling)
- optimisation of traffic parameters for each Network Service

## ■ Contribution to Standardisation

- IETF drafts