A Model-driven, Component-based and Service-oriented Approach for Designing an Autonomic Transport Protocol

Ernesto Expósito
ernesto.exposito@laas.fr

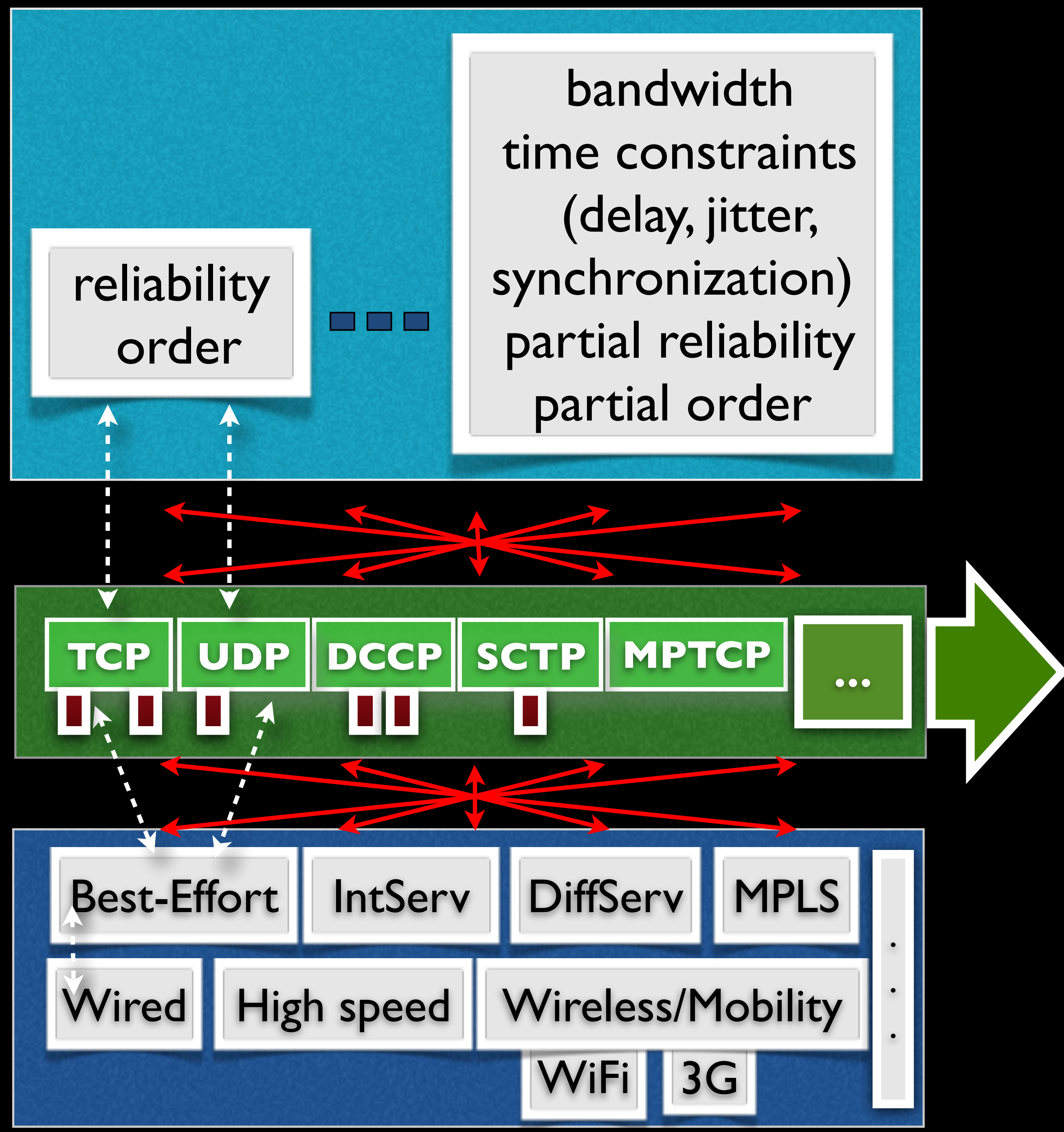
Université de Toulouse
LAAS-CNRS
INSA

Context and problem statement

Distributed applications requirements

Transport protocols

Network services and technologies



New generation transport layer

Which service?

- ***Application reqs + network const.***
- ***monolithic vs composite?***

How to adapt?

- ***Component behavior***
- ***Composite struct.***

How to be autonomous?

- ***Self-config***
- ***Self-adapt***

Approach

Methodology

requirements-oriented model-driven unified process:

1) Requirements
2) Design
3) Specification (model) and validation
4) Implementation
5) Tests and performance evaluation
6) Deployment

Models

Software architecture: UML

Knowledge: OWL (ontologies)

Decision: analytic/ learning based model

Paradigms

Component-based design (vs. monolithic)

Adaptive and autonomic management (vs. manual-human mg.)

Service-oriented design (vs. static/hard-binding)

Outline

- New generation transport layer (incremental design)

Application and network aware

Component-based and (micro)service-oriented

Adaptive

Autonomic

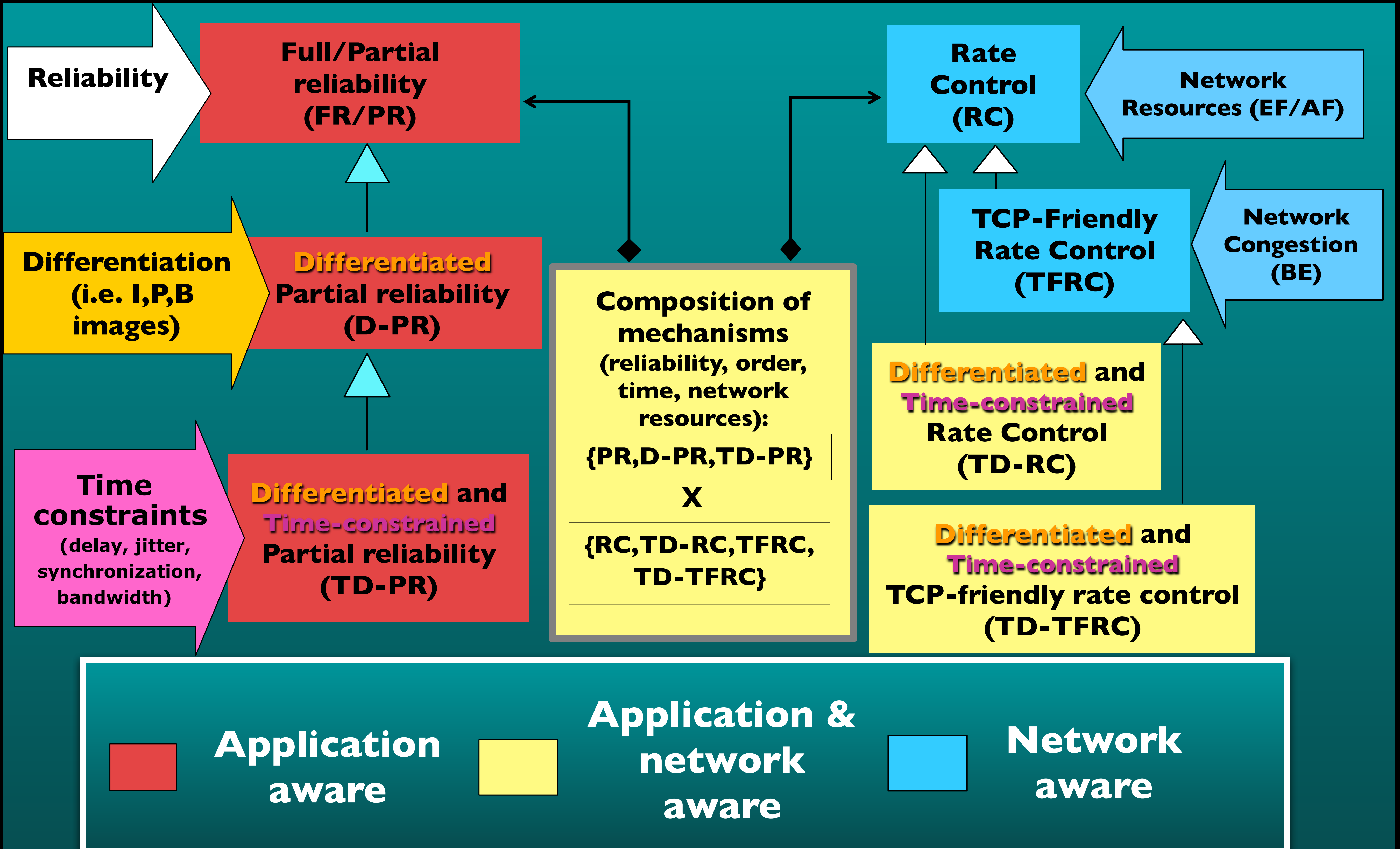
Phase 1: the basis

- Requirements
- Design of the Fully Programmable Transport Protocol (FPTP)
- Specification and validation
- Implementation and test/performance evaluation
- Deployment

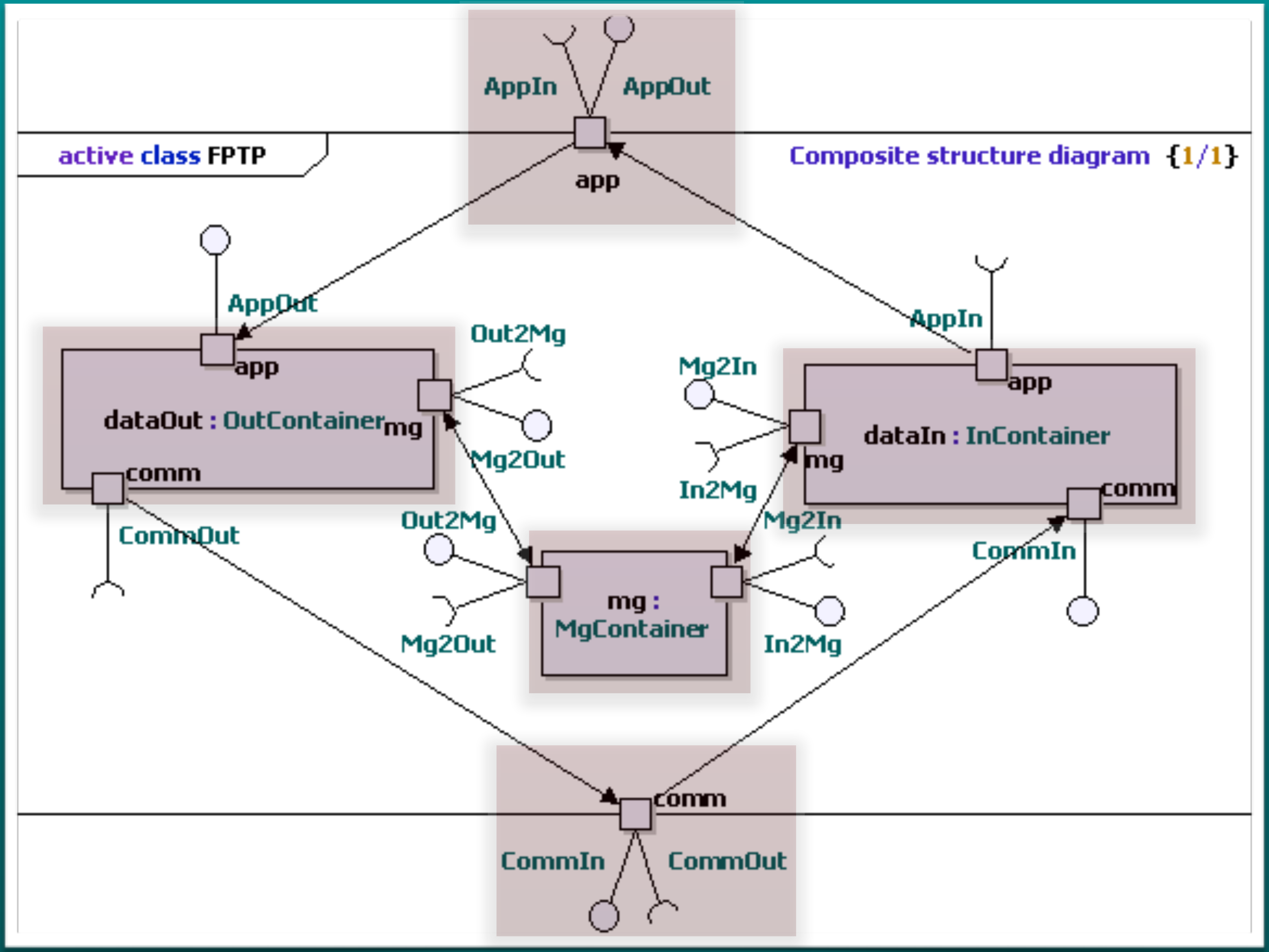
Requirements

- R1: How to provide the most adequate service taking into account application requirements and network services
- R2: How to easily integrate future components (more specialized mechanisms)

Design (R1): mechanisms

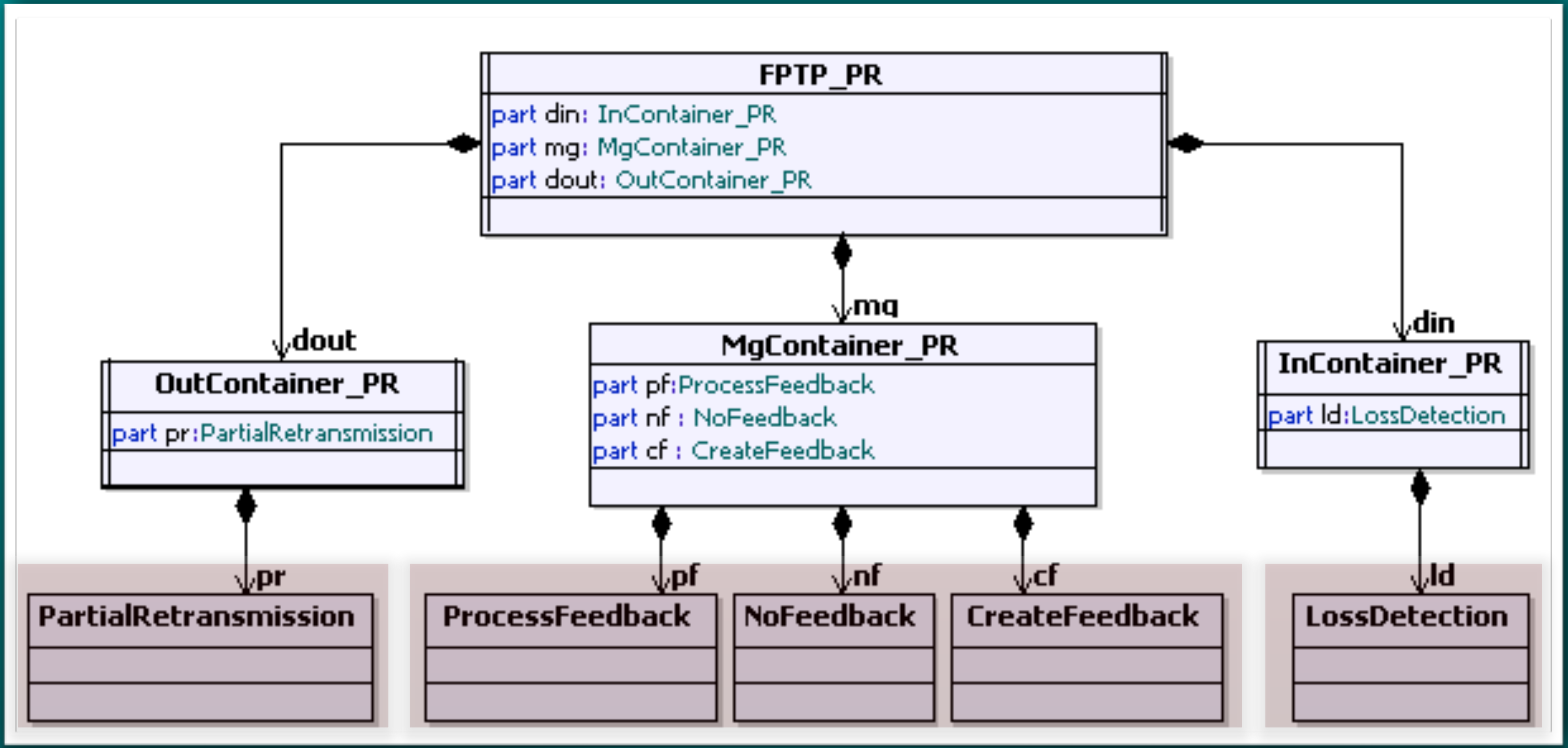


Design (R2): architecture



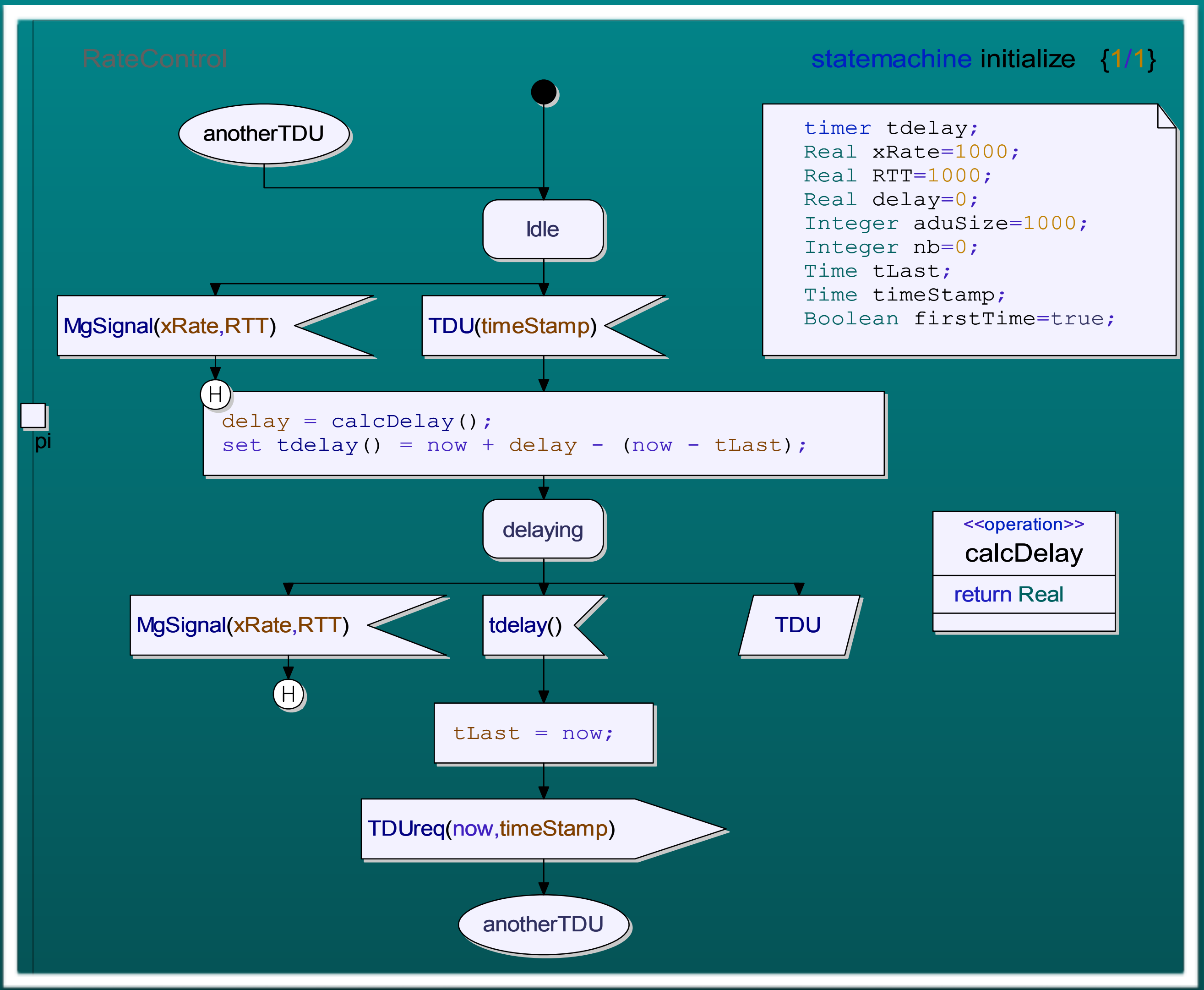
Specification: transport mechanism composition

Partially reliable function (PR)

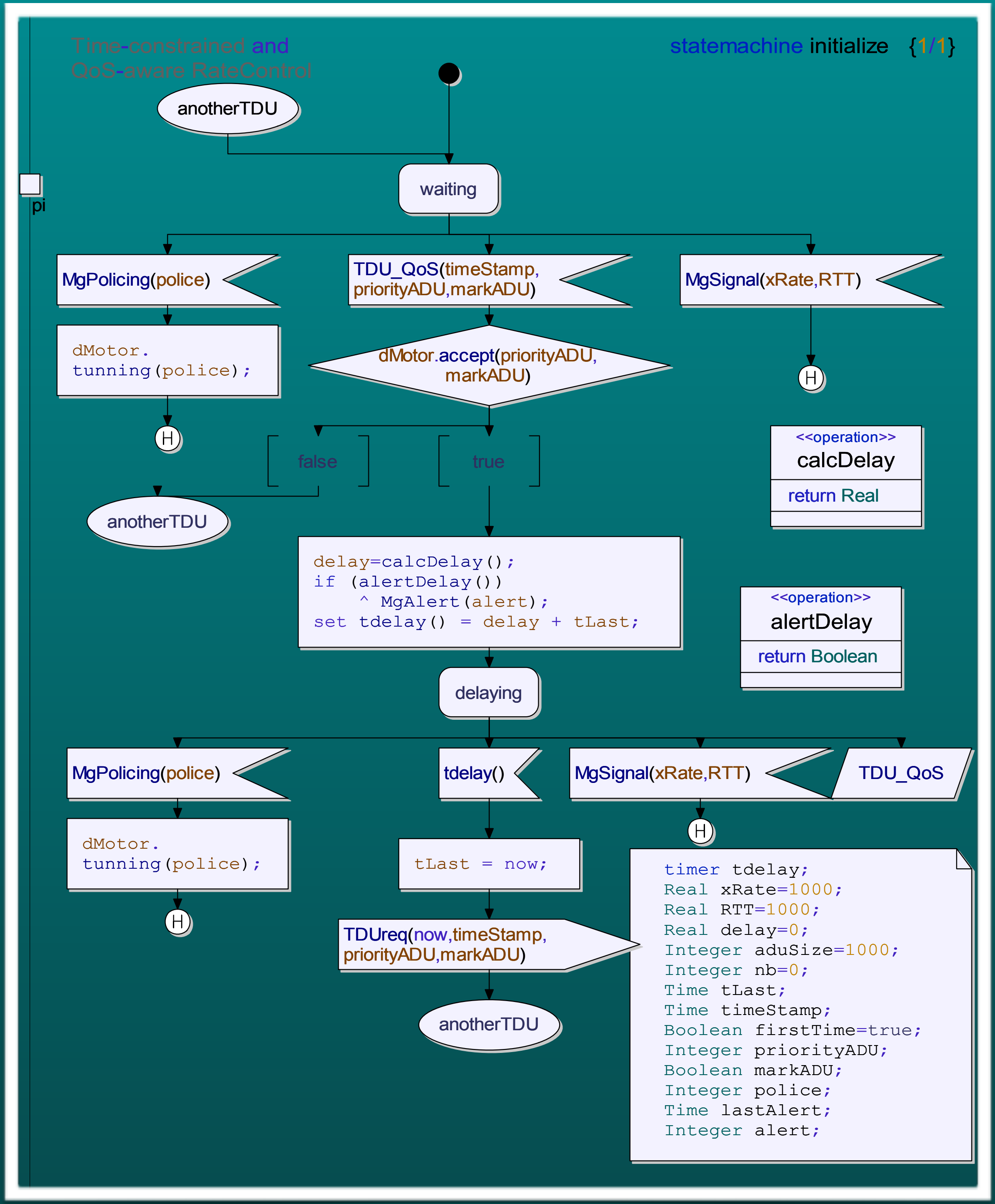


Specification: mechanism active behavior

TFRC mechanism



TD-TFRC mechanism



Validation of UML specification

- Environment: IBM-Rational TAU platform (profile: UML-verification)
  - Generation of executable model
- Approach
  - Dynamic model consistency
  - Interactive simulations for functional validation:
    - Validation per use case: instantiation, interconnection, communication, deadlocks free
  - Limitations: complexity to cover all potential protocol states

Implementation and Test/ performance evaluation

- JAVA implementation
- Evaluation:
  - Experimental network environment based on a network emulator (Dummynet) and streaming audio/video applications

Deployment

- European Project GCAP (1999-2001):
  - Active deployment of FPTP services (active networks)
- European Project EuQoS (2004-2006):
  - Deployment and evaluation of application-aware/ network-aware mechanisms over heterogeneous network services

Incremental design....

- Benefits of FPTP by offering
  - A component-based architecture
  - A large set of application and network-aware composite transport mechanisms
- New requirements: more elaborated adaptive strategies

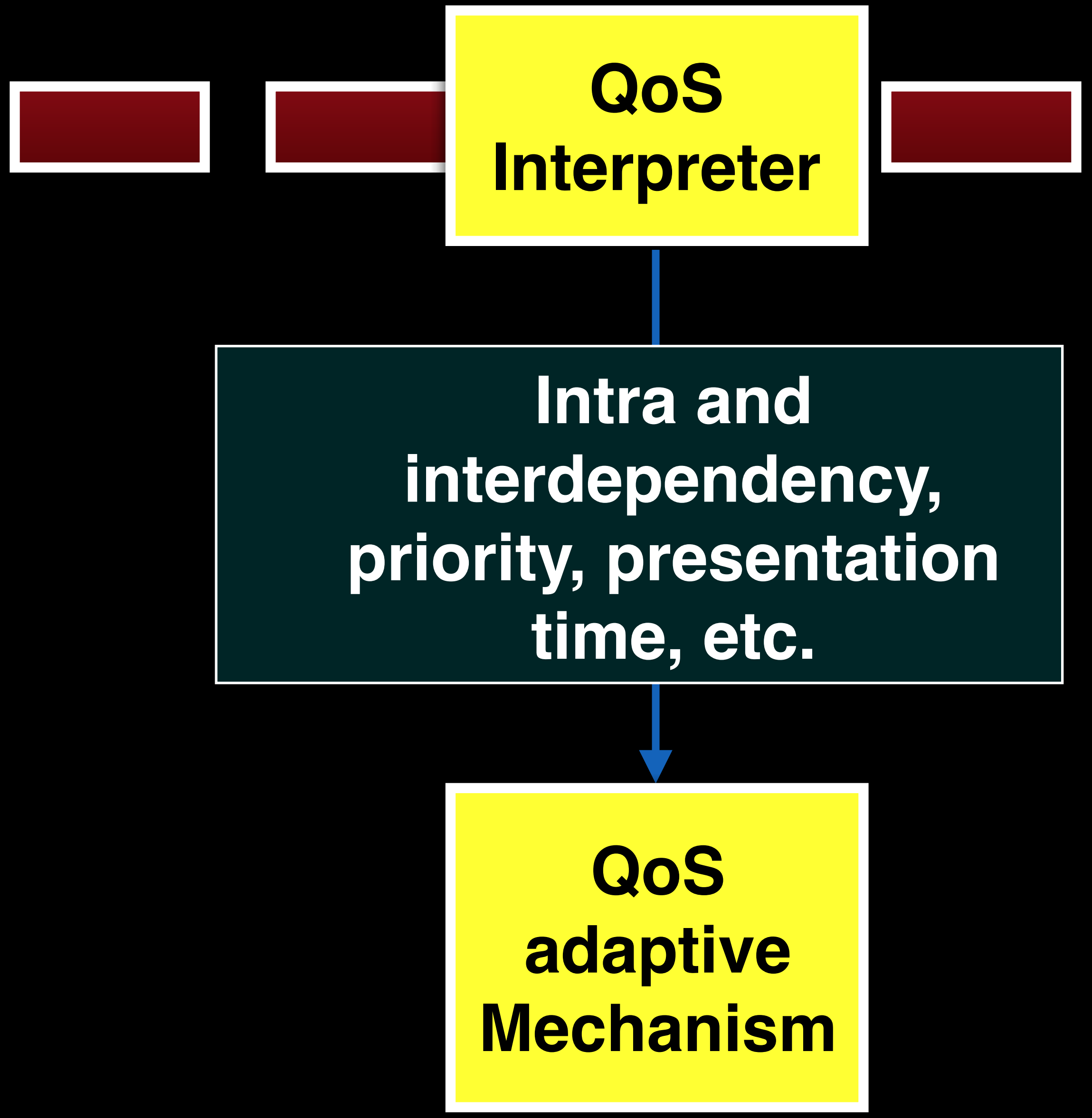
Phase II: adaptation

- Requirements
- Design of the Enhanced Transport Protocol
    - Adaptive: Behavioral and structural adaptation
- Specification and validation
- Implementation and test/performance evaluation
- Deployment

Requirements

- Req 3: Behavioral adaptation
  - Adaptive mechanisms based on generic application traffic semantic
- Req 4: Structural adaptation
  - Dynamic configuration of compositions in response to network changes
    - learning based decision model

Design of a model-driven QoS interpreter to allow generic behavioral adaptation (R3)



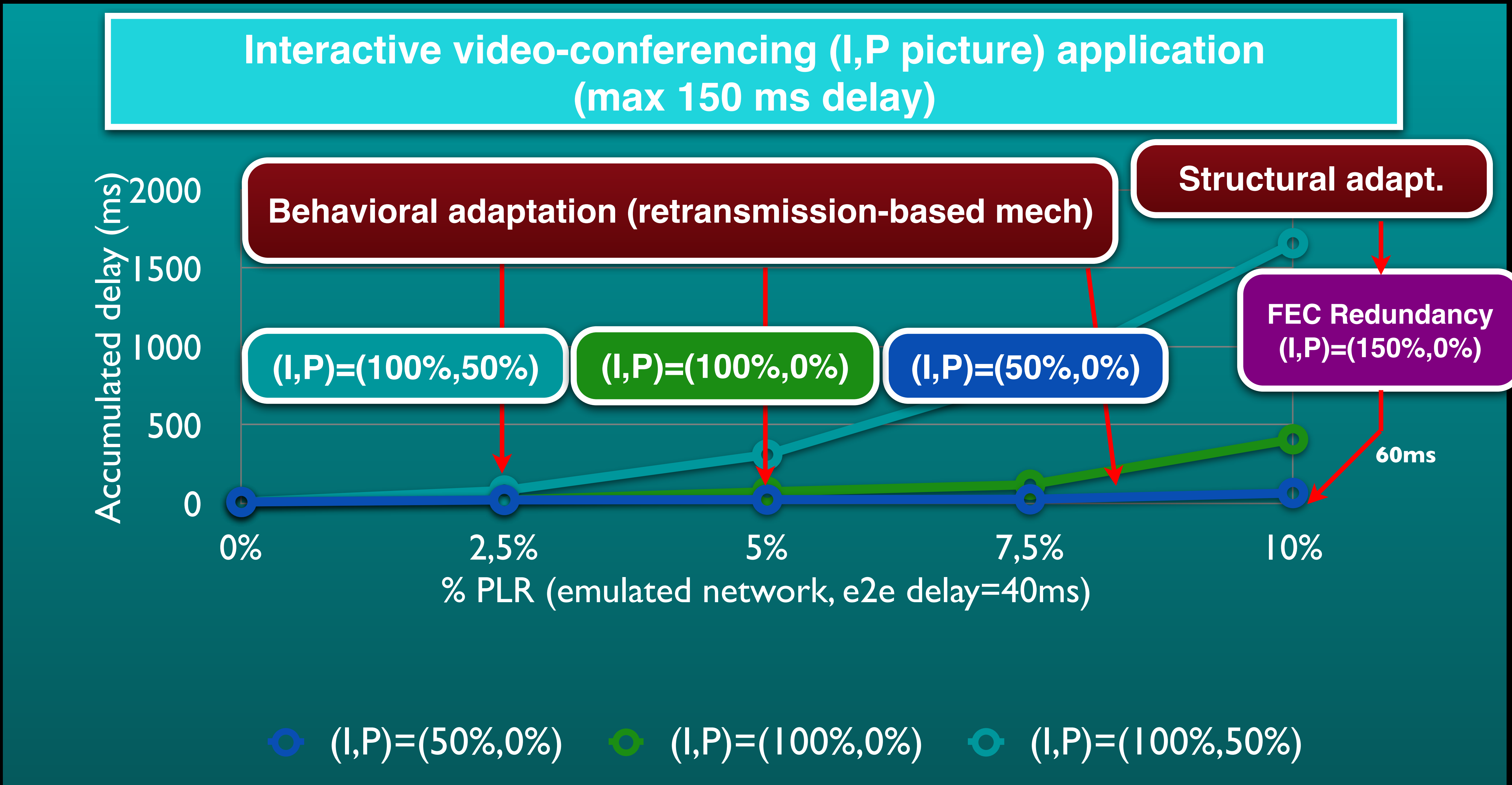
- Generic interpreter (vs. ad-hoc solutions) offering standard interface to retrieve properties/constraints of multimedia streams (i.e. H.264, MPEG2, H.263, etc.)

- Used for designing/developing QoS adaptive mechanisms (i.e. error control, rate/congestion control)

Design of model-driven structural adaptation (R4)

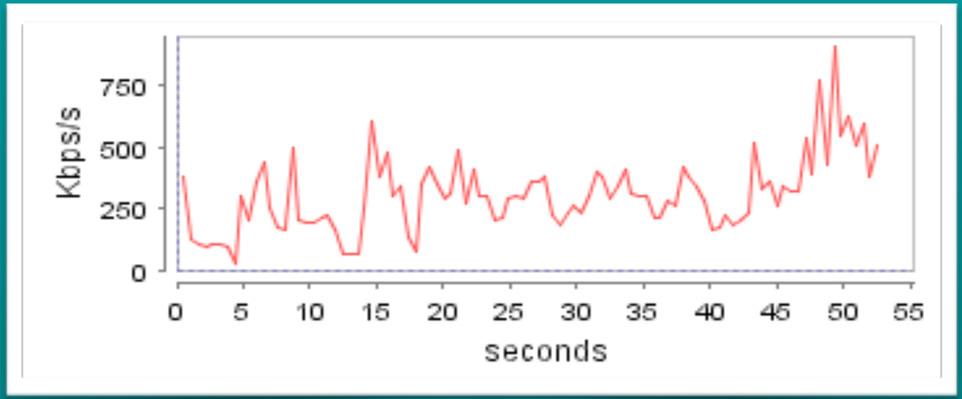
- Analytic model approach
  - Includes all the valid compositions
  - Guides the selection based on requirements and network conditions
- Learning-based model approach
  - Extension of the Markov Decision process (eMDP)
  - Obtained by reinforcement learning techniques

Behavioral and Structural adaptation (PR)



Behavioral and Structural adaptation (TFRC)

Traffic profile for a H.263 video stream

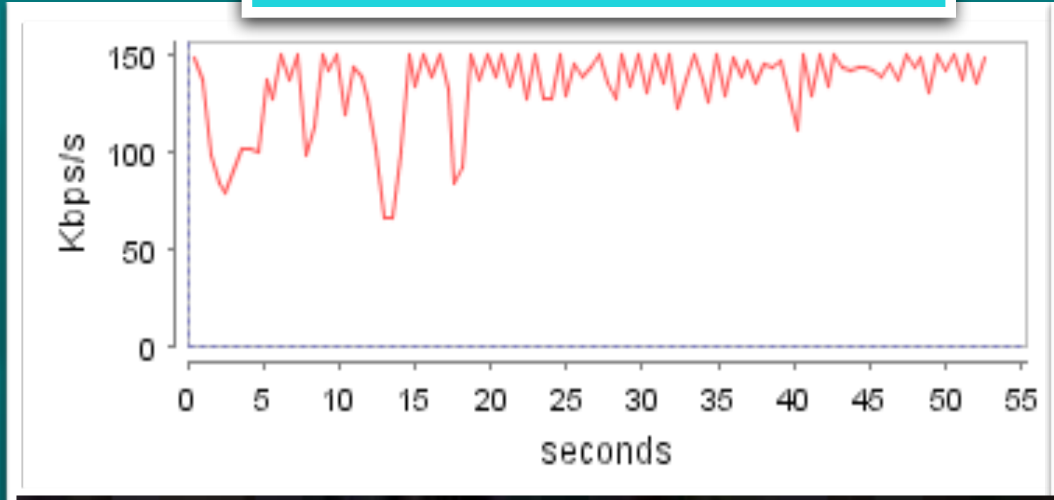


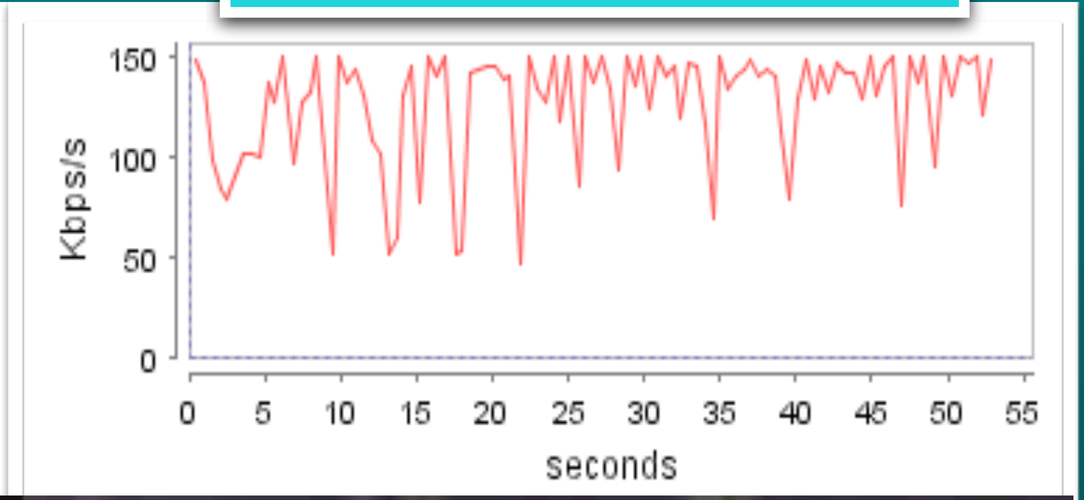
Comm. channel simulating congested, delayed and lossy network scenarios

a)Standard rate control

b)adaptive rate control

Video data received by





QoS improv. 15%-56% (PSNR)

Deployment

- European Project EuQoS (2006-2007):
  - Behavioral adaptation strategies (RC) over heterogeneous network services
- European Project NetQoS (2006-2008):
  - Integration of adaptive ETP services within the autonomic NetQoS system

Incremental design....

- Adaptive transport protocol mature for autonomic
  - Behavioral adaptation
  - Structural adaptation
- New requirements:
  - Self-configuring
  - Self-adapting

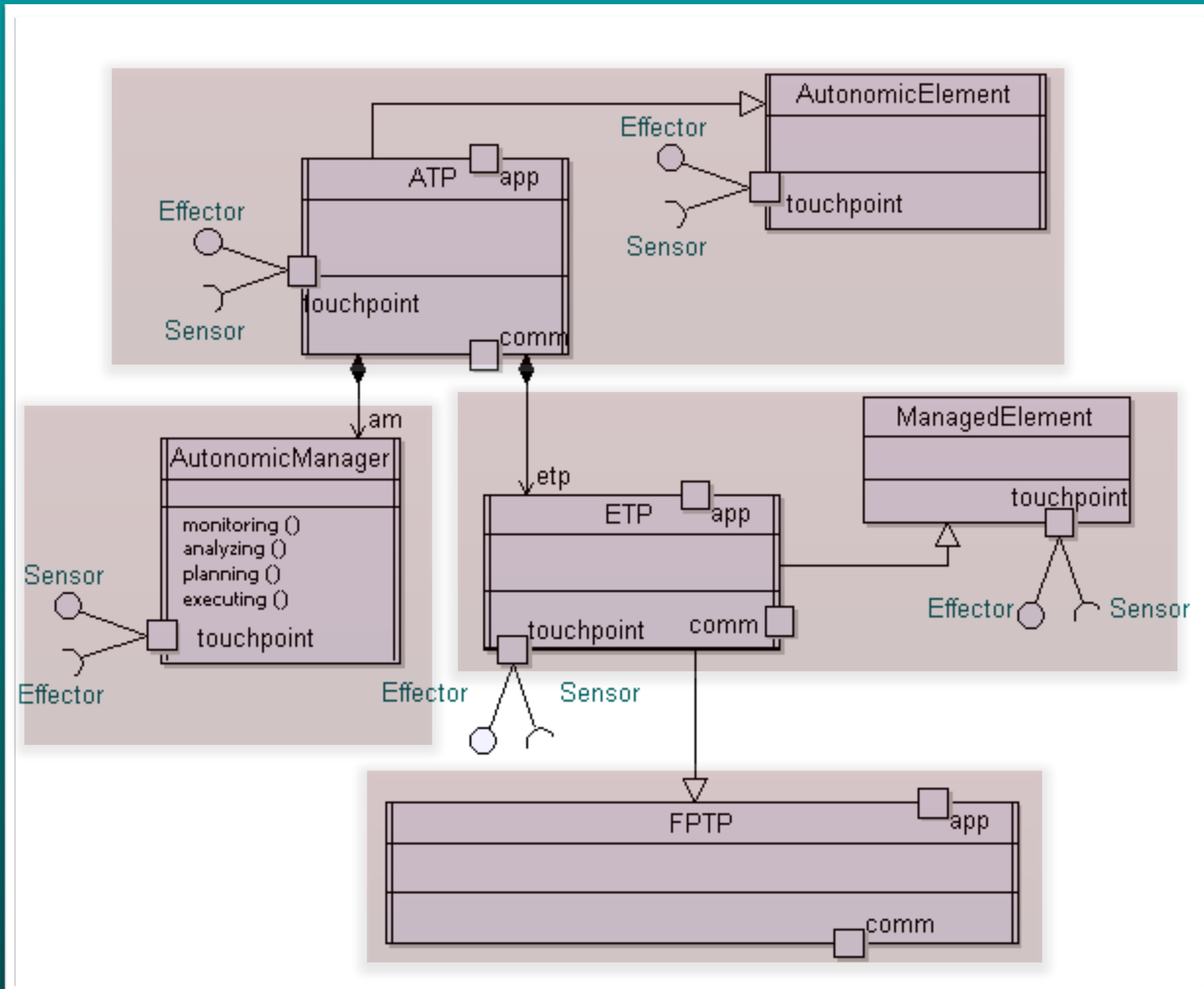
Phase III: autonomic

- Requirements
- Design of the Autonomic Transport Protocol
  - Autonomic, service-oriented and ontology-driven architecture
- Specification and validation
- Implementation and test/performance evaluation
- Deployment
- Conclusions

Requirements

- Req 5: Integration of current solution within an autonomic architecture
- Req 6 : AC knowledge base for self-configuring and self-adapting properties

Specification of the Autonomic Transport Protocol Architecture (R5)



Implementation of an Ontology-driven Autonomic-manager offering self-configuring and self-adapting functions (R6)

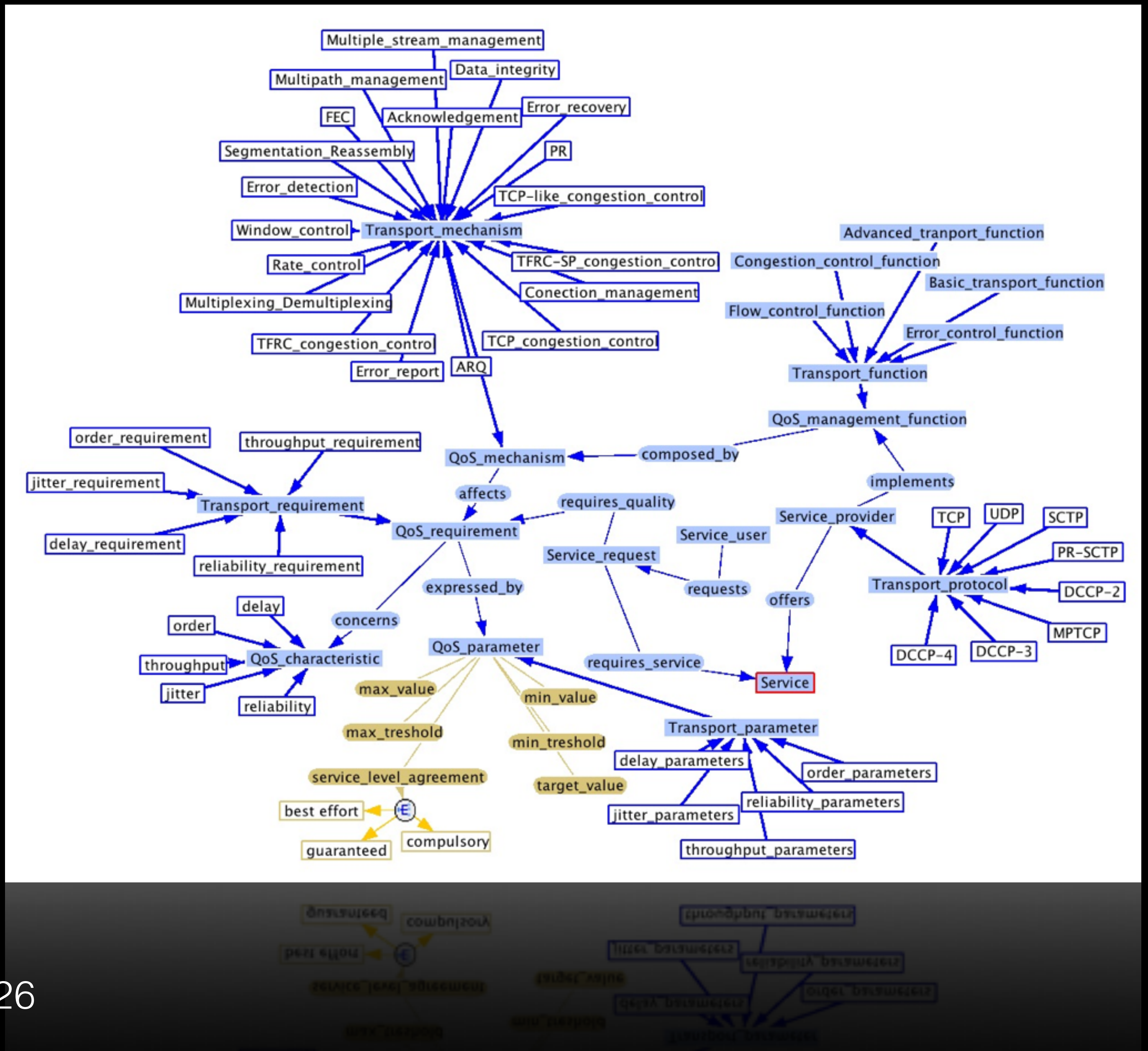
ODA
Ontologies

V1: representation and consistency:
transport mechanisms, functions, protocols and services

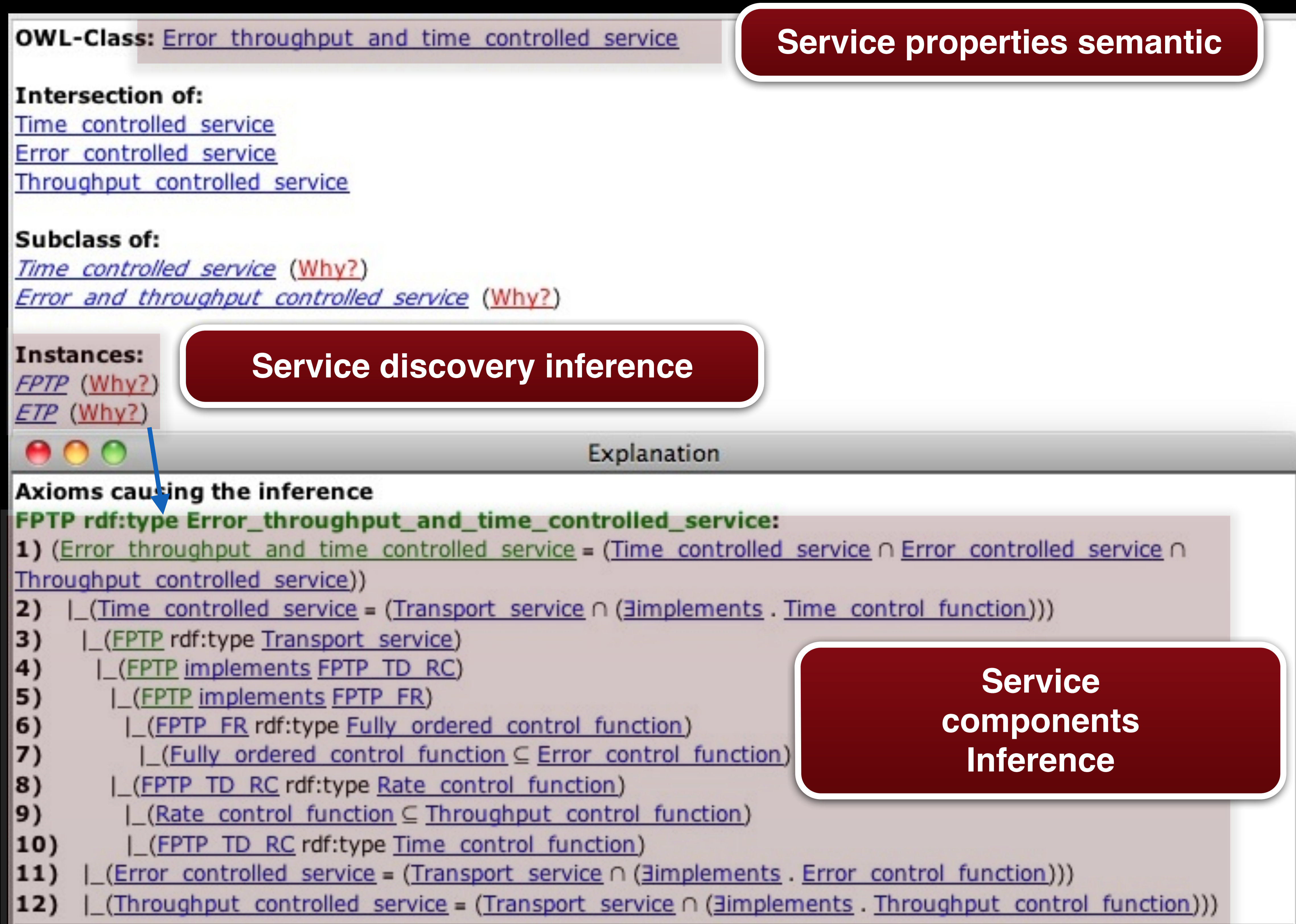
V2: inferencing capabilities for SOA/CB self-configuring:
services properties, components and composites

V3: inferencing capabilities for self-adapting
behavioral (tuning)
structural (reconfiguring)

QoS Transport Ontology OWL implementation



Example: Error/throughput/time controlled service discovery



Deployment

- European(Celtic) Project Feel@home (2008-2010):
  - Design and development of an Ontology-driven architecture for autonomic QoS management in home networks (UPnP)
- IMAGINE (starting from 2011) IP Project (Virtual Factories/Enterprises)
  - Design and development of an autonomic service bus integrating ATP services for heterogeneous information systems

Conclusions and Perspectives

Conclusions and Perspectives