ICT 2003, April 8-10, 2003

On Adaptable Networking

**Finn A Aagesen, Bjarne E Helvik,
Chutiporn Anutariya and Mazen M Shiaa**,
Department of Telematics,
Norwegian University of Science and Technology

"Adaptable networking" is the handling of dynamic changes in *time* and *position* related to *Nodes*, *Capabilities*, *Users* and *Services*.

Capabilities are Node inherent or Node related *data*, *resources* or *functions*, which act as an application *basis* for the building of dynamic functionality.

Contents

Telematics

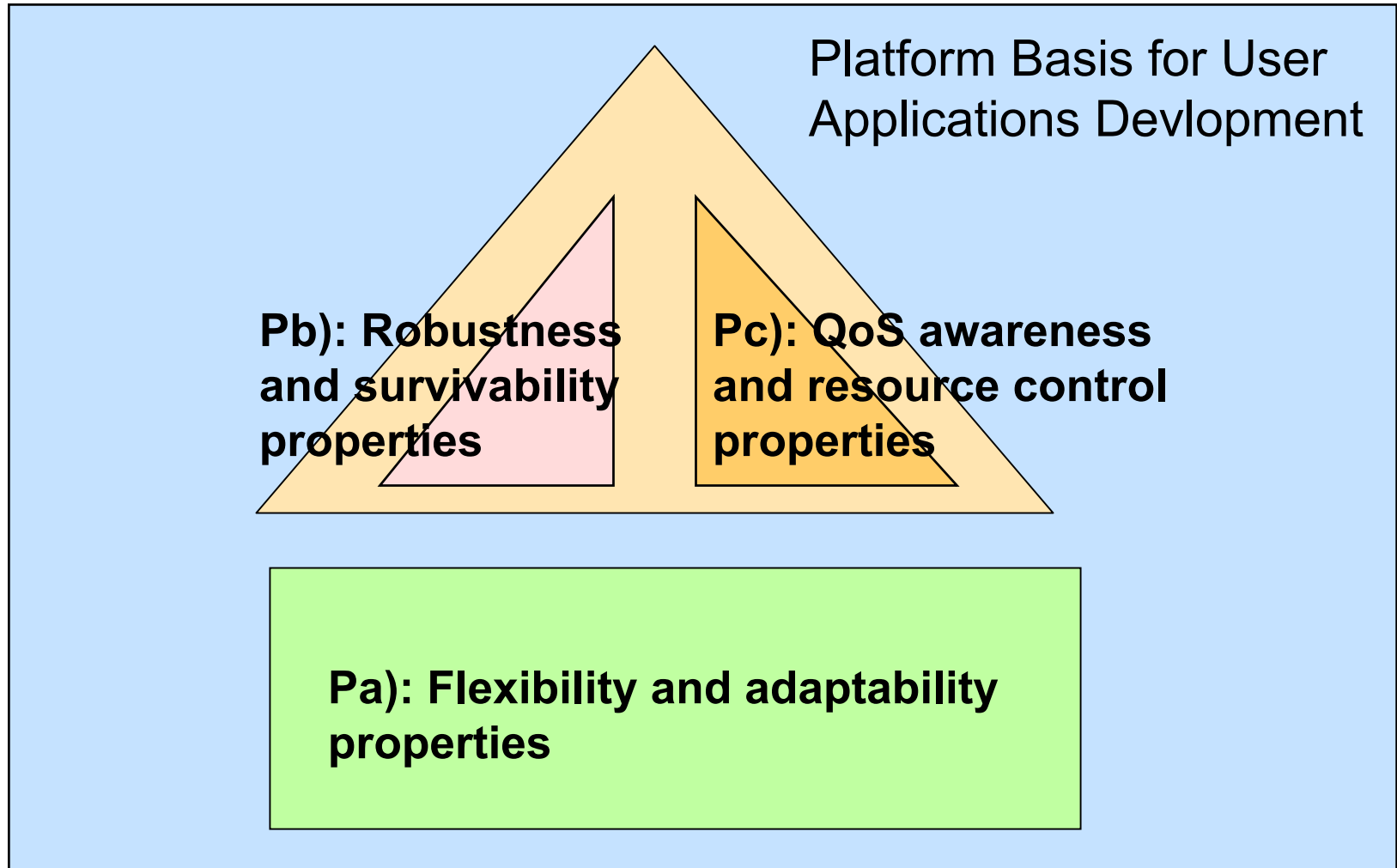**A**rchitecture

Plug-

And-play

Systems

Motivation: To be able to handle

- the increasing heterogeneity, complexity and diversity *for technical telecommunication solutions*
- that qualified staff *is the critical factor for development, installation, deployment, operation, maintenance and evolution* of *Telecom Service software*

2. Property Requirements

Platform Basis for User Applications Devlopment

Pb): Robustness and survivability properties

Pc): QoS awareness and resource control properties

Pa): Flexibility and adaptability properties

Flexibility and adaptability:

- **System structure and functionality is not fixed** *(adding, moving, removing components and changing component functionality according to needs)*

- **New components and their capabilities are found automatically when introduced** *(awareness of new components and capabilities, propagation of needed information about changes)*

- **Continuous adaption to changed environments and operation strategies/policies** *(new component functionality, new service and network management functionality, new policy functionality)*

- **Appropriate functionality building priciples**

Robustness and survivability:

- **Dependable distributed architecture.**
  (Replicated resources and functionality, inhibition of malicious and unauthorized components.)

- **Reconfigure itself in the presence of failures.**
  (Detection of hardware component failures and reconfiguration, detection of software failures and reinitialization, prevention of error import and propagation)

- **Provide continuous operation.**
  (Continous operation during fault handling, PaP of components and changes in the functionality repertoir.)

QoS awareness and resource control:

- **Negotiatiation about QoS and optimum resource allocation.**
  (Transmission capacity, storage capacity, computation power, functional handling capacity, functional handling capability, QoS contracts, negotiation functionality, optimisation functionality)

- **Monitoring of resource utilization and actions and reallocation of resources**.
  (Detection, rearranging of workload by PaP, relocation of functionality by PaP, information and guidance to network and service operators about needed manual actions)

3. TAPAS – Basic Architecture and Platform

- **The theatre metaphor**
- **The basic architecture (Object model)**
- **Capabilities**
- **The platform**

The theatre metaphor

Theatre: A metaphor for concepts and functionality definition.

Repertoire: The set of **Plays** that may be performed at the theatre.

Play: Defines a set of logically related functionality.

Director role figure: The manager of plays, and supervisor for application role figures, constituted by an **actor** .

Application role figures : The performers of plays. Constituted by **actors** playing **roles**.
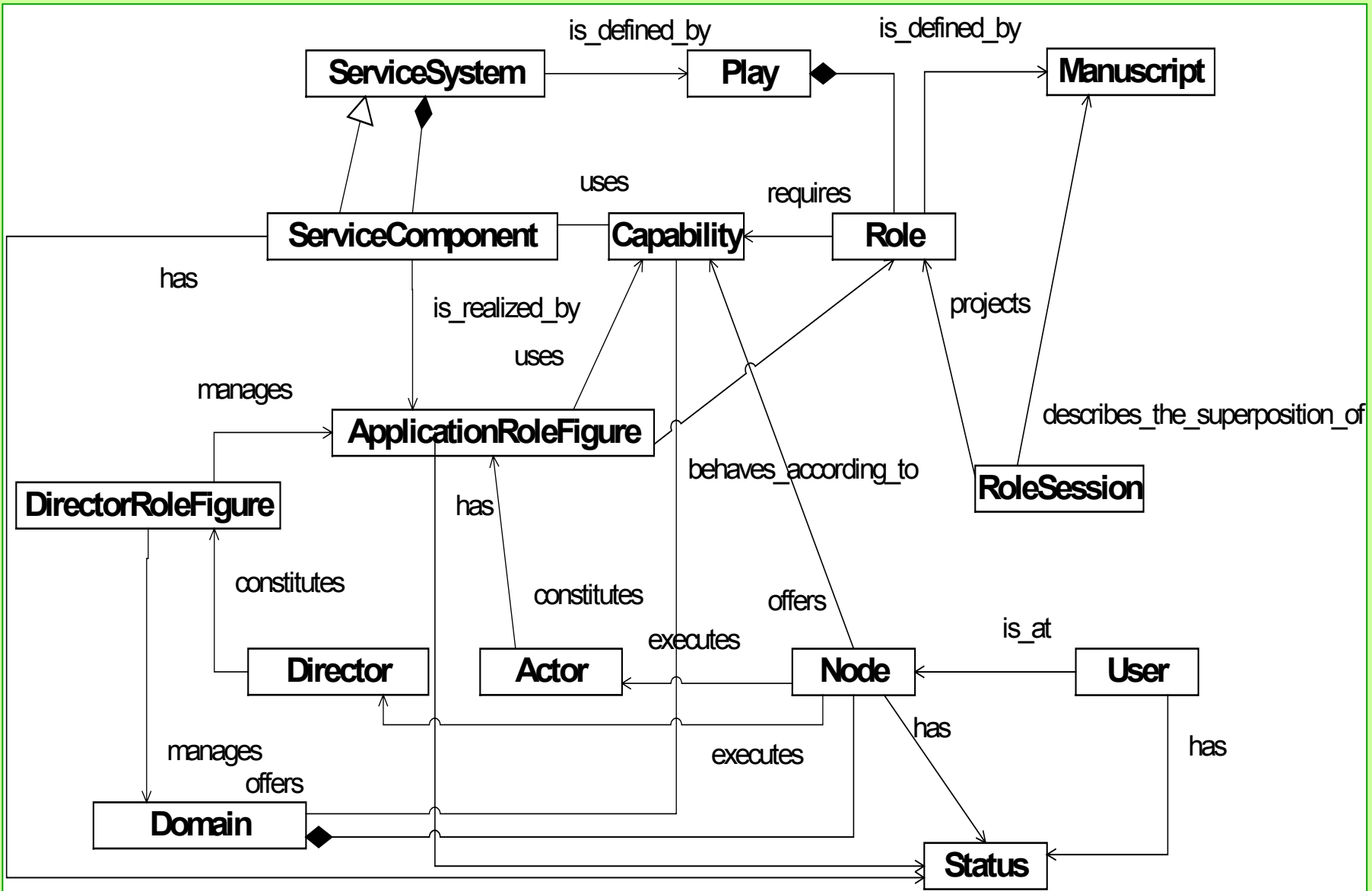
Capability: A unique set of properties of an **actor** at the stage where he is playing.

Actors

Role session: A dialogue between two **role figures**.

Manuscript: The assigned behavior, i.e. the defined **role** of a **role figure**, constituted by an an **actor**.

# The basic architecture (Object model)

Capabilities

Primitives:

- **Functions:** Pure software or combined software/ hardware components used for performing particular tasks,

- **Resources:** Hardware components with finite capacity, such as processing, storage and communication units,

- **Data:** Just data, the interpretation, validity and life span of which depend on the context.
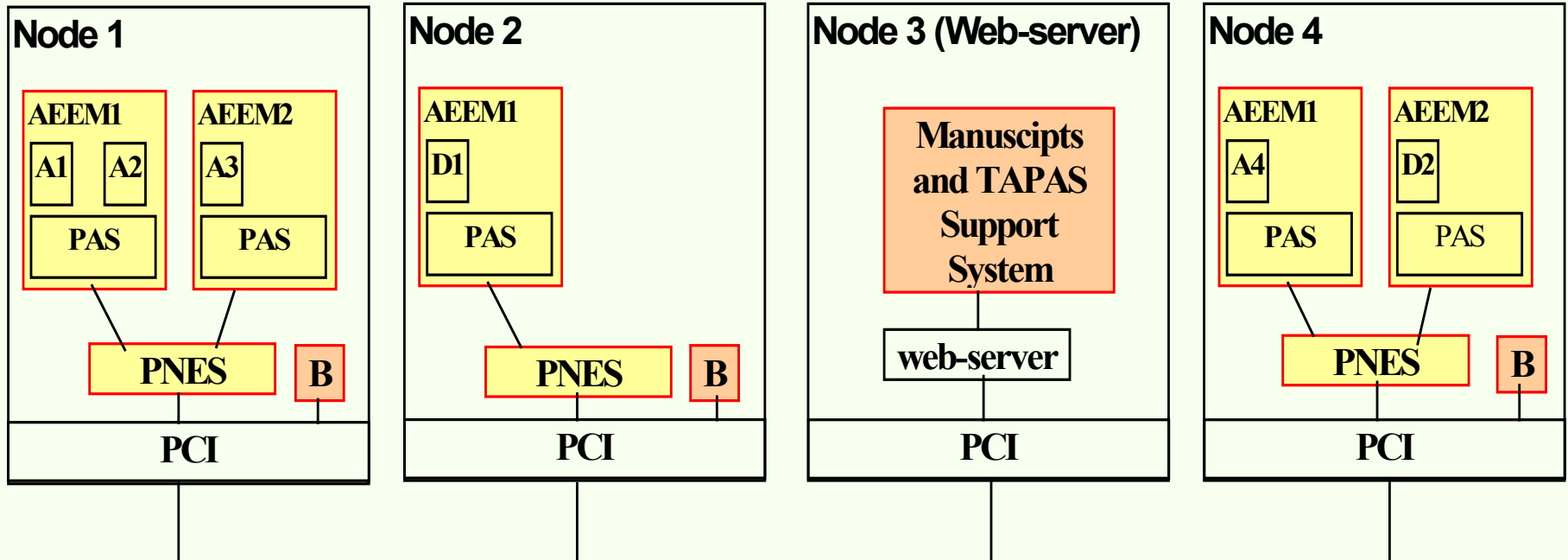
Classification:
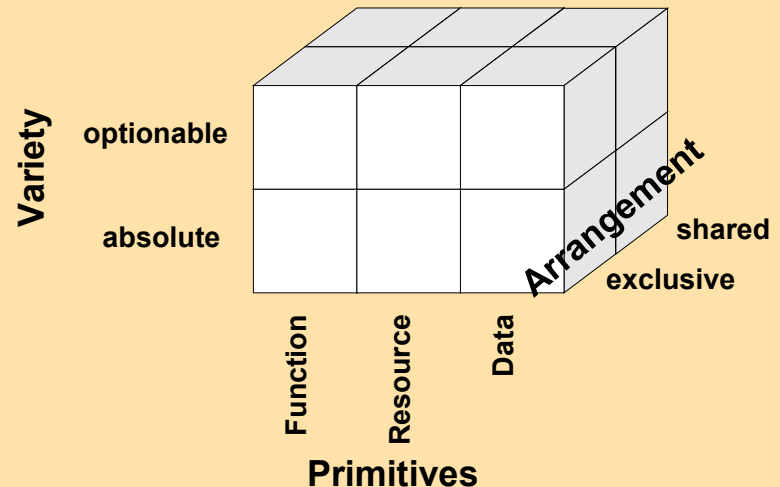
Status

Status reflects the resulting state of the system, which indirectly can be changed.

- **Status is, at a certain time instant, the situation in a PaP Service System system with respect to the actual number of nodes, playing plays, traffic situation, etc.**

- **It can both comprise observable counting measures, measures for QoS or calculated predicates related to these counts and calculated measures.**

Example view of TAPAS platform for software execution

Node 1

AEEM1
- A1
- A2

PAS

AEEM2
- A3

PAS

PNES B

PCI

Node 2

AEEM1
- D1

PAS

PNES B

PCI

Node 3 (Web-server)

Manuscipts and TAPAS Support System

web-server

PCI

Node 4

AEEM1
- A4

PAS

AEEM2
- D2

PAS

PNES B

PCI

Ai :   Actor no i ,   Di: Director no i,   B: Plug-and-Play Boot
AEEMi: Actor-environement-execution-module no i
PAS: Plug-and-Play Actor Support,
PNES   Plug-and-Play Node Execution Support
PCI: Plug-and-Play Communication Infrastructure

Static available

Dynamic available

NTNU

4. TAPAS – Mobility Handling

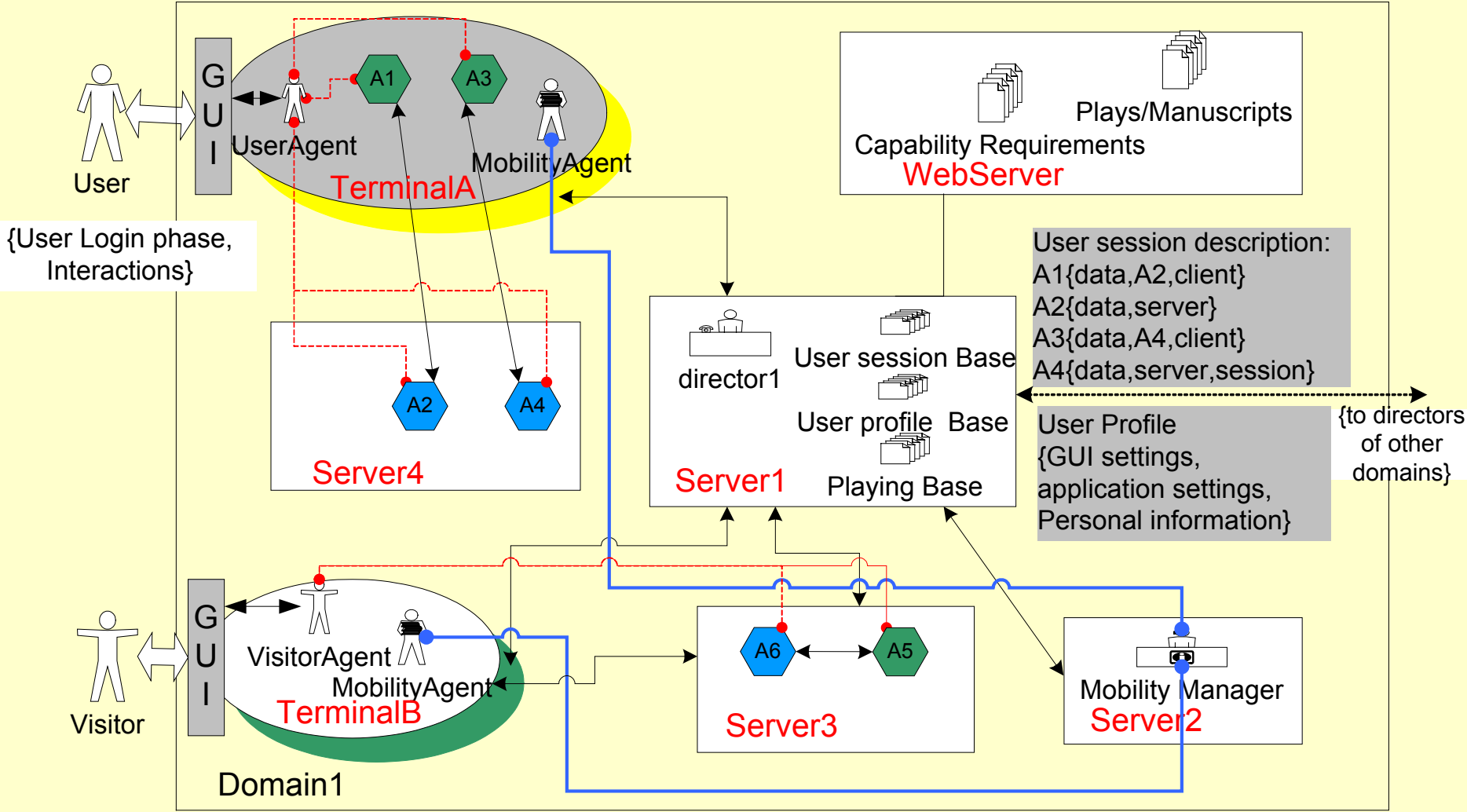**Mobility handling is a rather comprehensive issue:**

- **User mobility**: Changing access point
- **Actor mobility**: Moving Role figures to a new Node
- **Role session mobility:** Moving ongoing Role sessions
- **User session mobility:** Moving Applications, Actors and Role sessions to make the User able to continue his User sessions from a new access point
- **Node mobility:** Changing physical positions of Nodes (and terminals)

***Personal Mobility** is the provisioning of services that are personalized with end user's preferences independent of both physical location and equipment type*
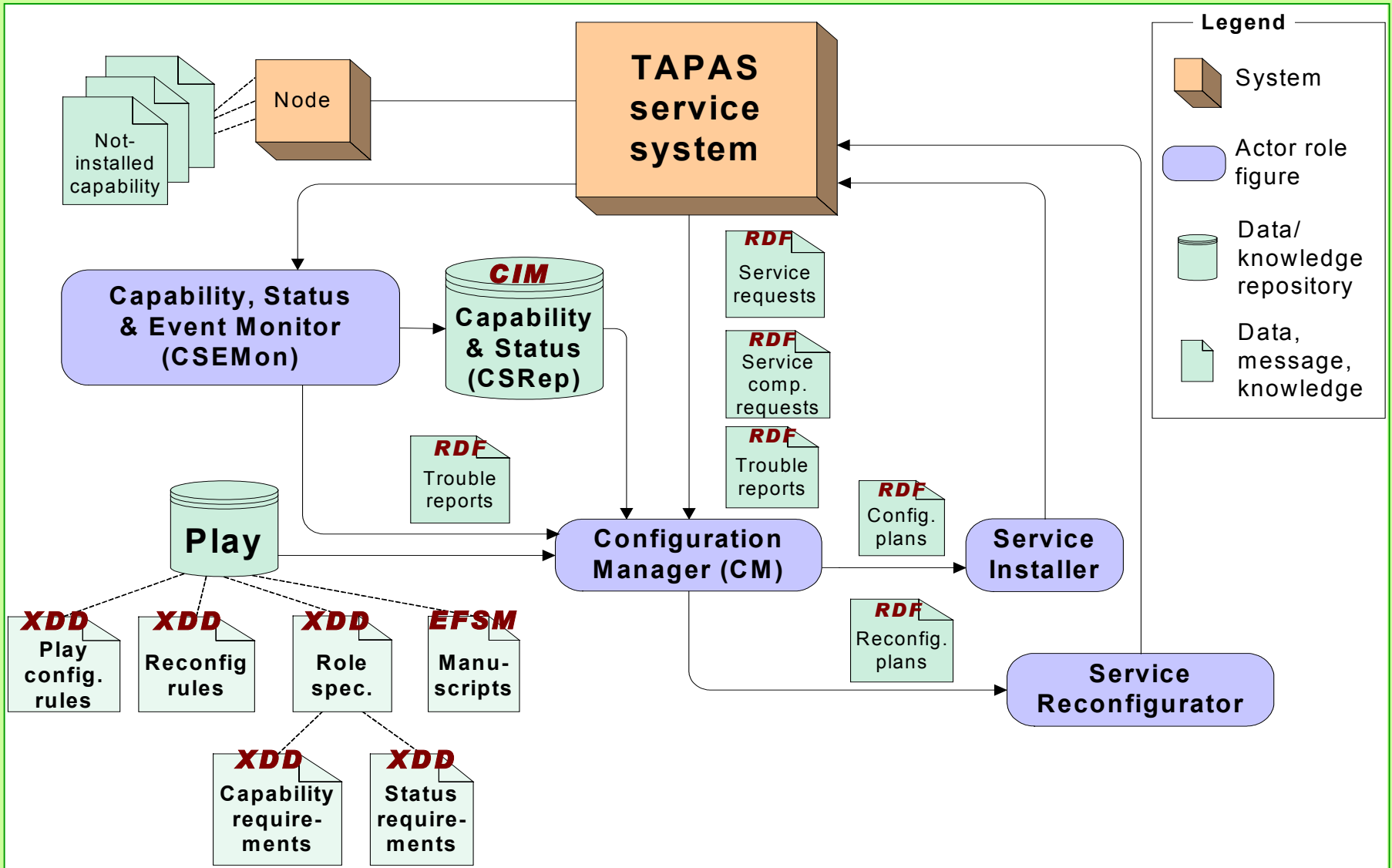
TAPAS Mobility Architecture (Object model)
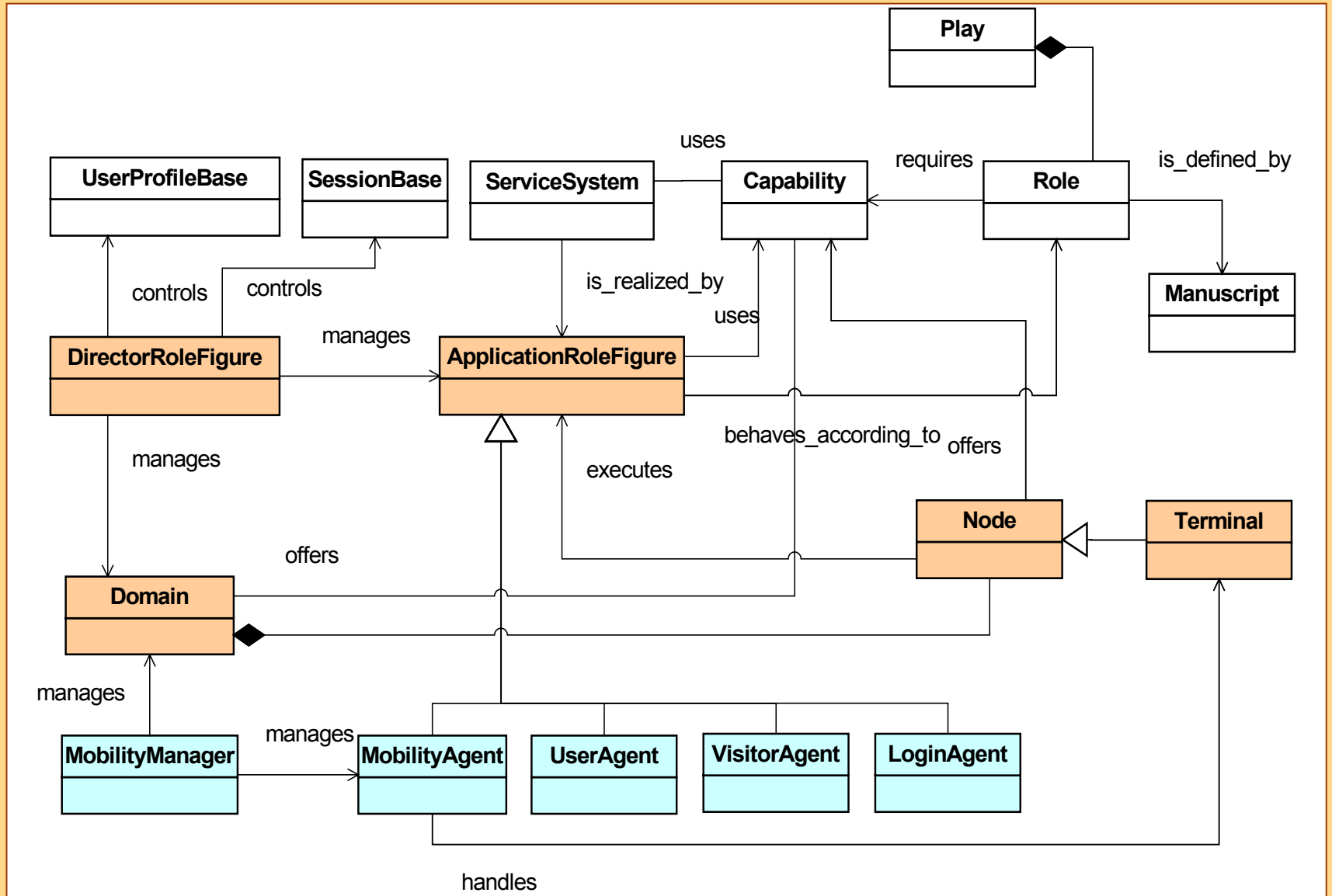
An illustration of TAPAS Mobility Architecture

5. TAPAS - Configuration Management

Modeling Configuration Management

Ontology

Configuration Management Ontology

-- CIM model

XML-based ontology language

Play Configuration

Constraints on Roles

Relationships among Roles

Instance of

Instance of

Uses

XML clauses

Node

Capability$_1$

Capabilityj

Service System

Status$_1$

Status i

Role

Requirement$_1$

Requirement$_n$

Configuration Manager Engine

XML-based ontology language

XML-based ontology language

XML clauses

Service System Configuration

6. Adaptive Service Infrastructure

Service Composition Definition
A Service as a composition of multiple sub-services

Service Implementation Definition
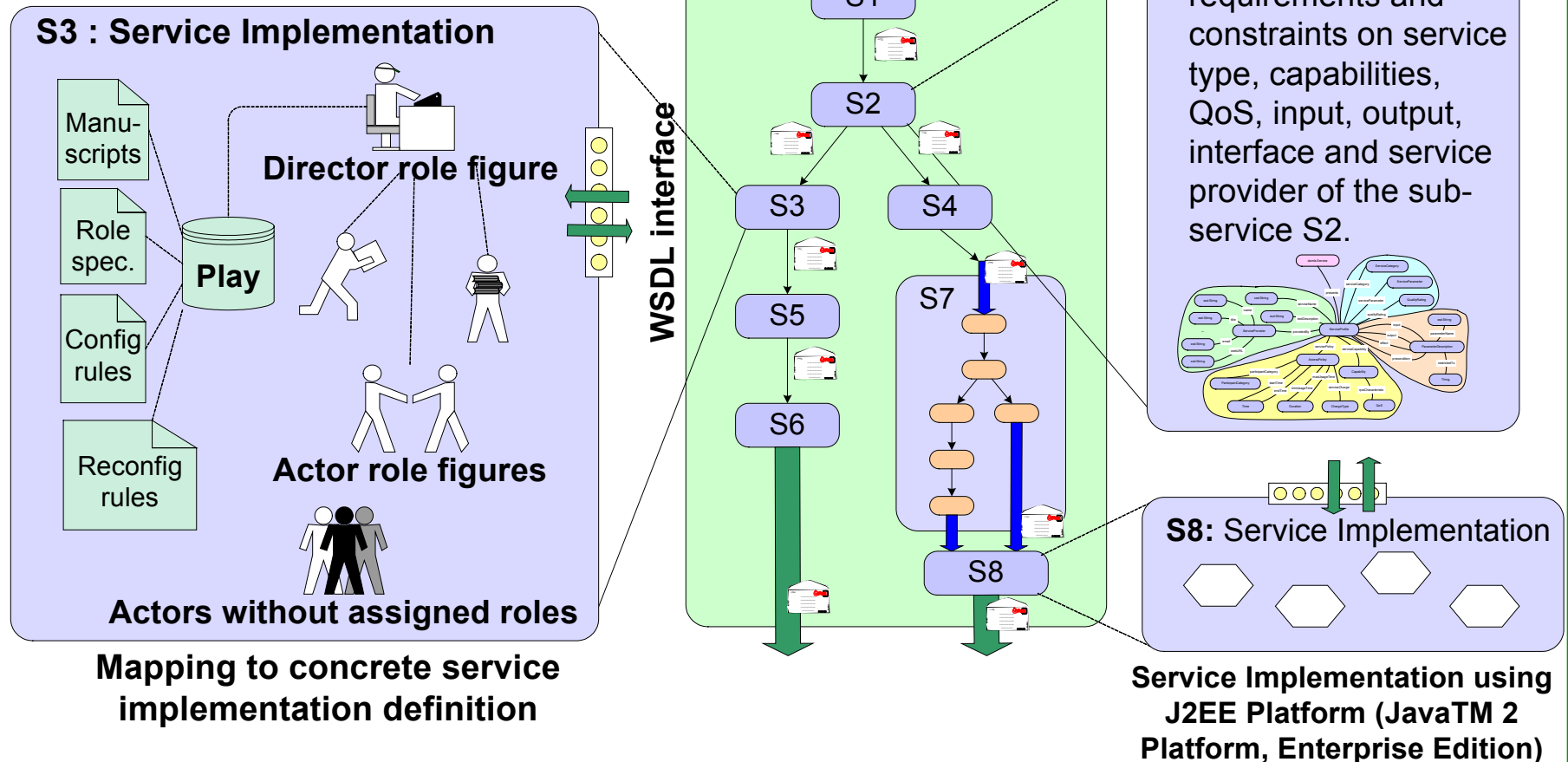**using PaP Execution Platform**

S3 : Service Implementation

Manu-scripts

Role spec.

Play

Config rules

Reconfig rules

Director role figure

Actor role figures

Actors without assigned roles

Mapping to concrete service implementation definition

WSDL interface

Service X

S1

S2

S3

S4

S5

S7

S6

S8

S2 Specification of requirements and constraints on service type, capabilities, QoS, input, output, interface and service provider of the sub-service S2.

S8: Service Implementation

Service Implementation using J2EE Platform (JavaTM 2 Platform, Enterprise Edition)

Summary and Conclusions

- **Adaptive Networking - Properties**
  - Flexibility and adaptability
  - Robustness and survivability
  - QoS awareness and resource control

- **TAPAS basic architecture with extensions**

- **Ongoing research**
  - Fault tolerance and intrusion prevention
  - Personal mobility
  - Configuration Management
  - Adaptive Service Infrastructure

- **TAPAS platform and demonstrator**. Software and documentation available: *http://www.item.ntnu.no/~plugandplay*