



Context-Aware Management Domains

Authors: Ricardo Neisse, Maarten Wegdam,
Patrícia Dockhorn Costa, and Marten van Sinderen



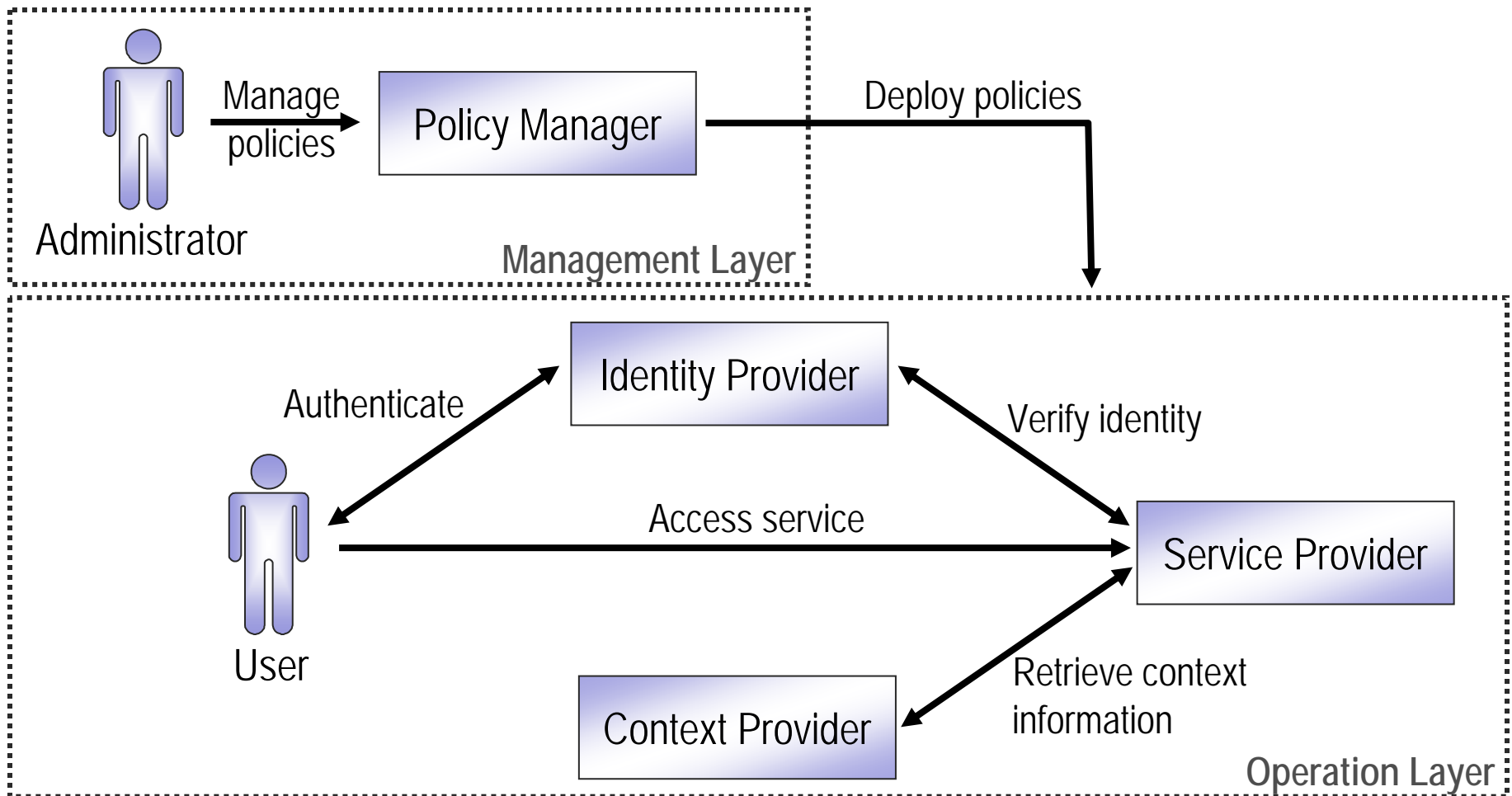
Conselho Nacional de Desenvolvimento
Científico e Tecnológico



Outline

- Policy management in context-aware service platforms
- Limitations of existing approaches
- Our proposal: context-aware management domains
- Conclusions and future work

Policy Management Scenario



Introduction

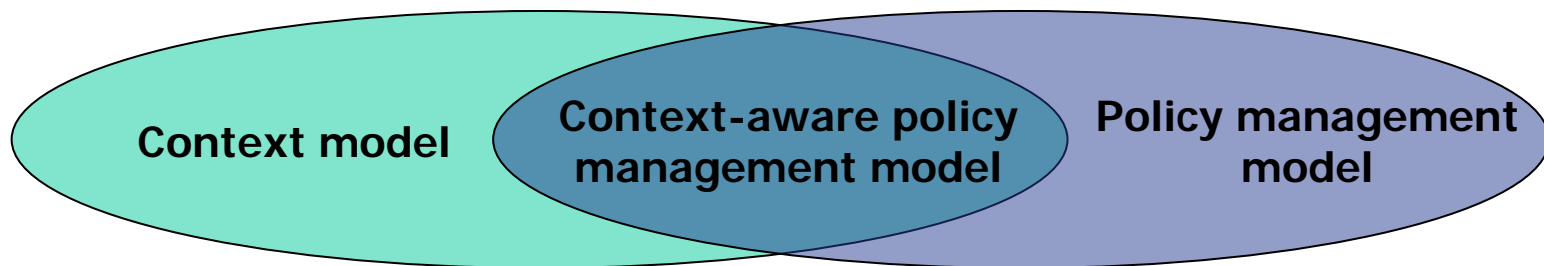
- Large number of entities for which different types of policies have to be managed
- Policies should be less intrusive for users, based on context information:
 - Allow access for working colleagues
 - Bob's identity should not be anonymized for nearby persons
 - Family members should delete my location information after one day
 - Increase trust of person every time he/she enters my home

Limitations of Existing Approaches

- Static management of policies
 - Entities are unknown at policy specification time
- Context-aware policy solutions focus explicitly on at most one policy management area
 - eXtensible Role Based Access Control (X-RBAC)
 - Context-aware trust management
- Context-aware policy management solutions express context as an attribute in the policy conditions
 - Location = X, Activity = Y

Our proposal

- New concept called: Context-Aware Management Domains (CAMDs)
- Combines a situation based context model with a policy management model
- Enables flexible context-aware policy management



Context Model

- Provides modeling of:
 - Entities, Context, Situations, Situation events
 - Examples: Person, Body temperature, Fever, EnterTrue(Fever)
- Use Event-Condition-Action (ECA) rules to trigger context-aware application adaptation

Upon EnterTrue(**Fever**)

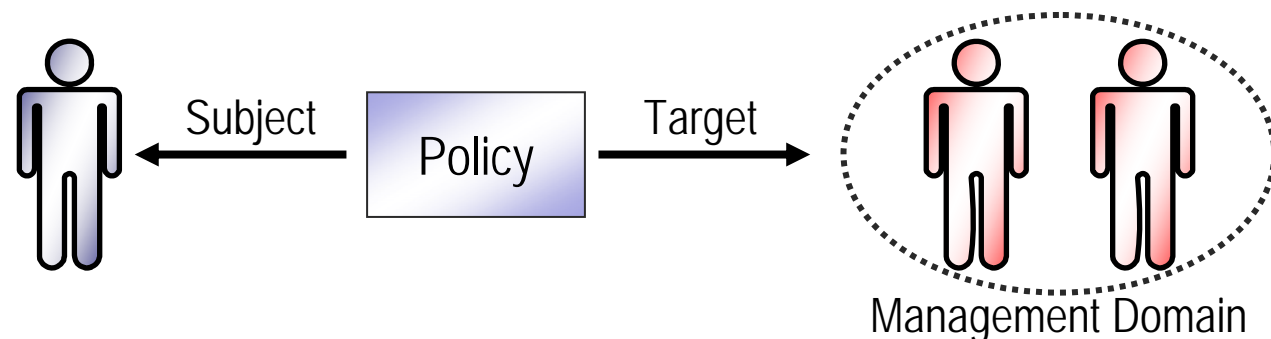
When **time.inBetween(8:00, 22:00)**

Do SendSMS(Fever.Person.FamilyDoctor)



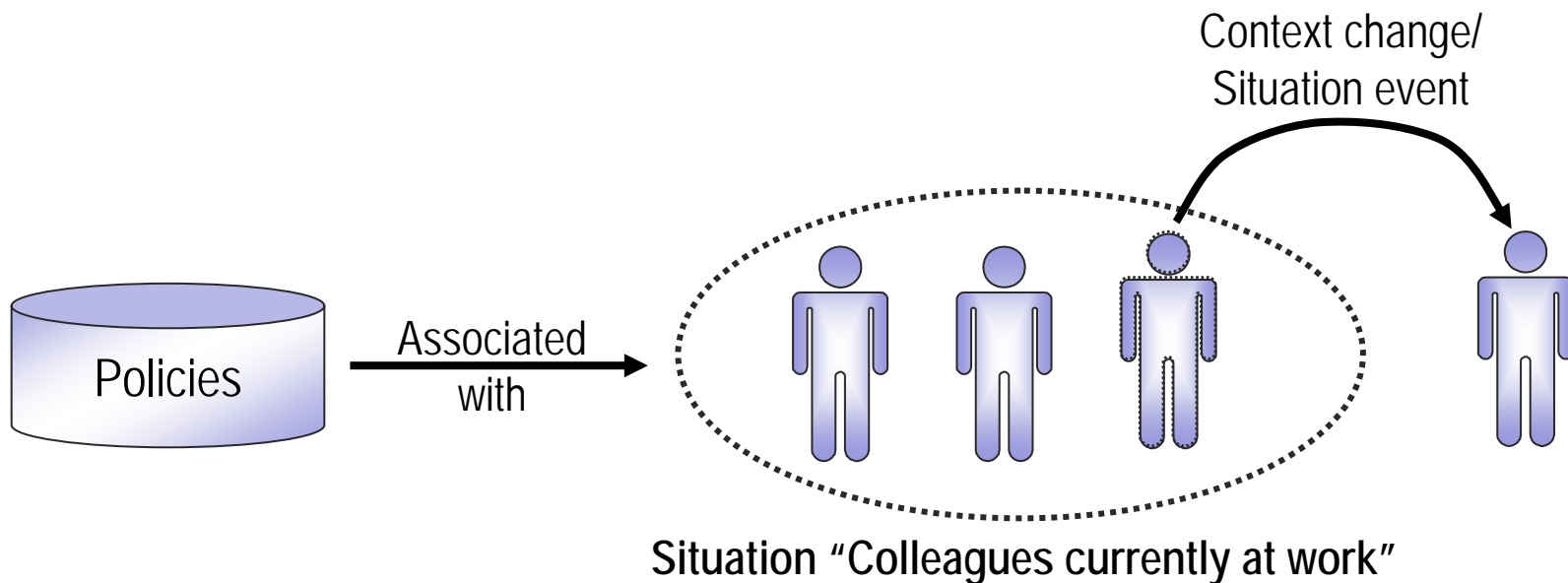
Policy Management Model [Ponder]

- Types of policies: authorization, obligation, refrain
- Policy subjects and targets can be expressed as management domains
- A management domains is a set of entities for which a common management policy applies
 - Managers have to manually include/remove entities from the domains



Context-Aware Management Domains

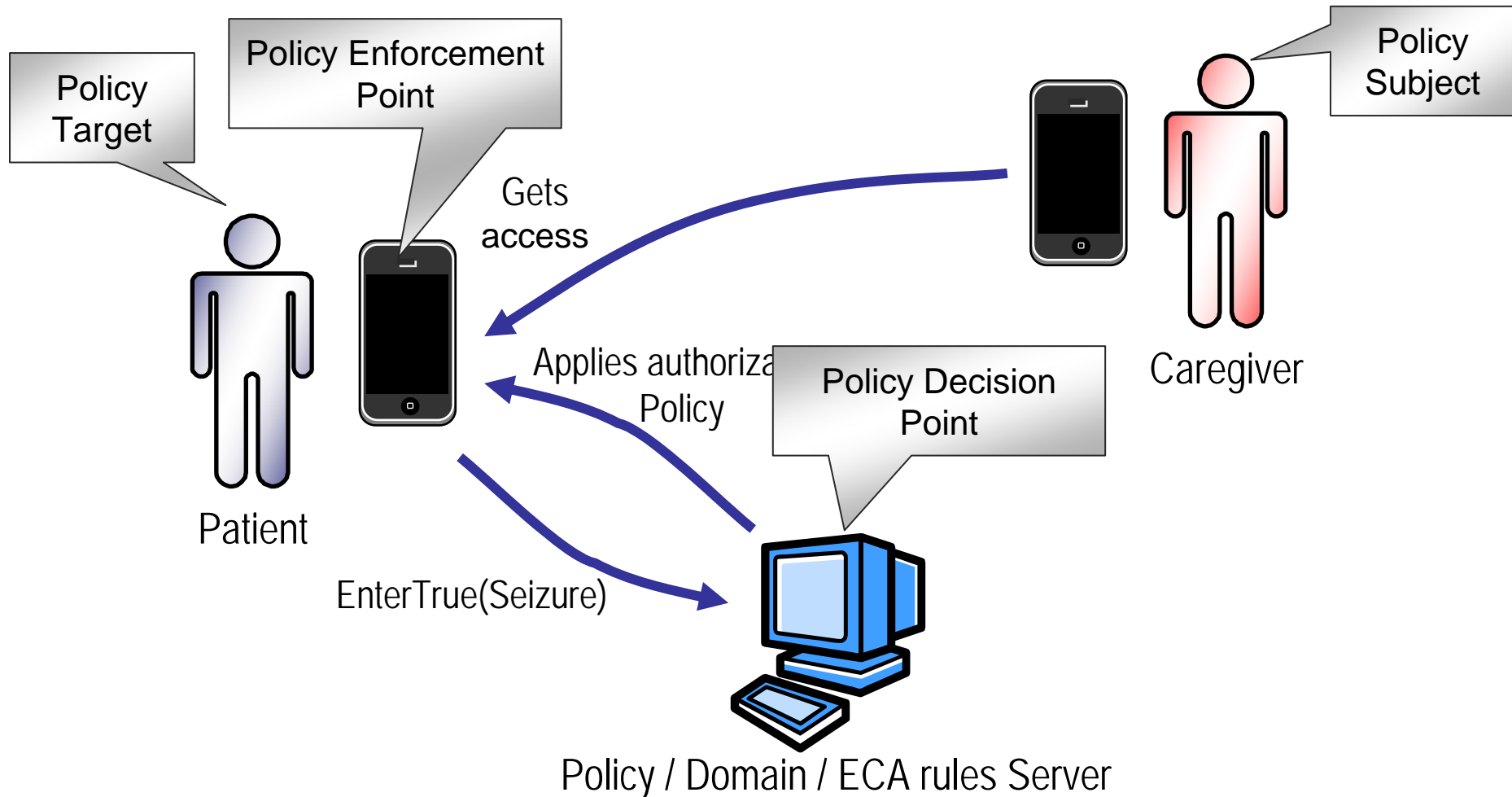
- Combine association of policies with dynamic set of entities based on context situations



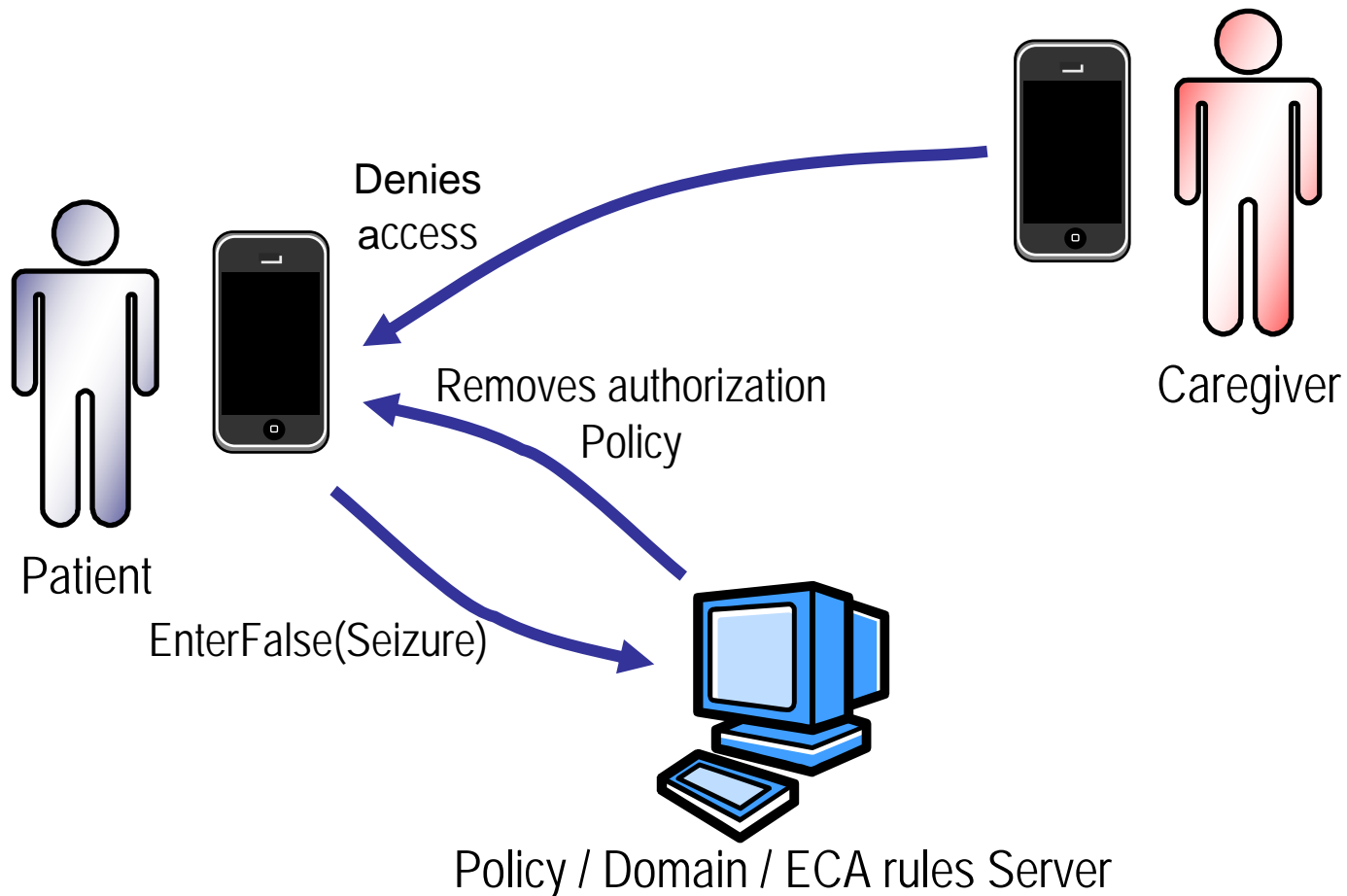
Policy example using CAMDs

- When a patient is having a seizure, nearby caregivers should be authorized to access his/her location and health data
- All the patient data accessed by the caregiver should be deleted afterwards
- Details:
 - Context situation of interest: Seizure
 - Events: EnterTrue(Seizure), EnterFalse(Seizure)
 - Entities: Patient, Nearby Caregivers
 - Policies: authorization and privacy obligation

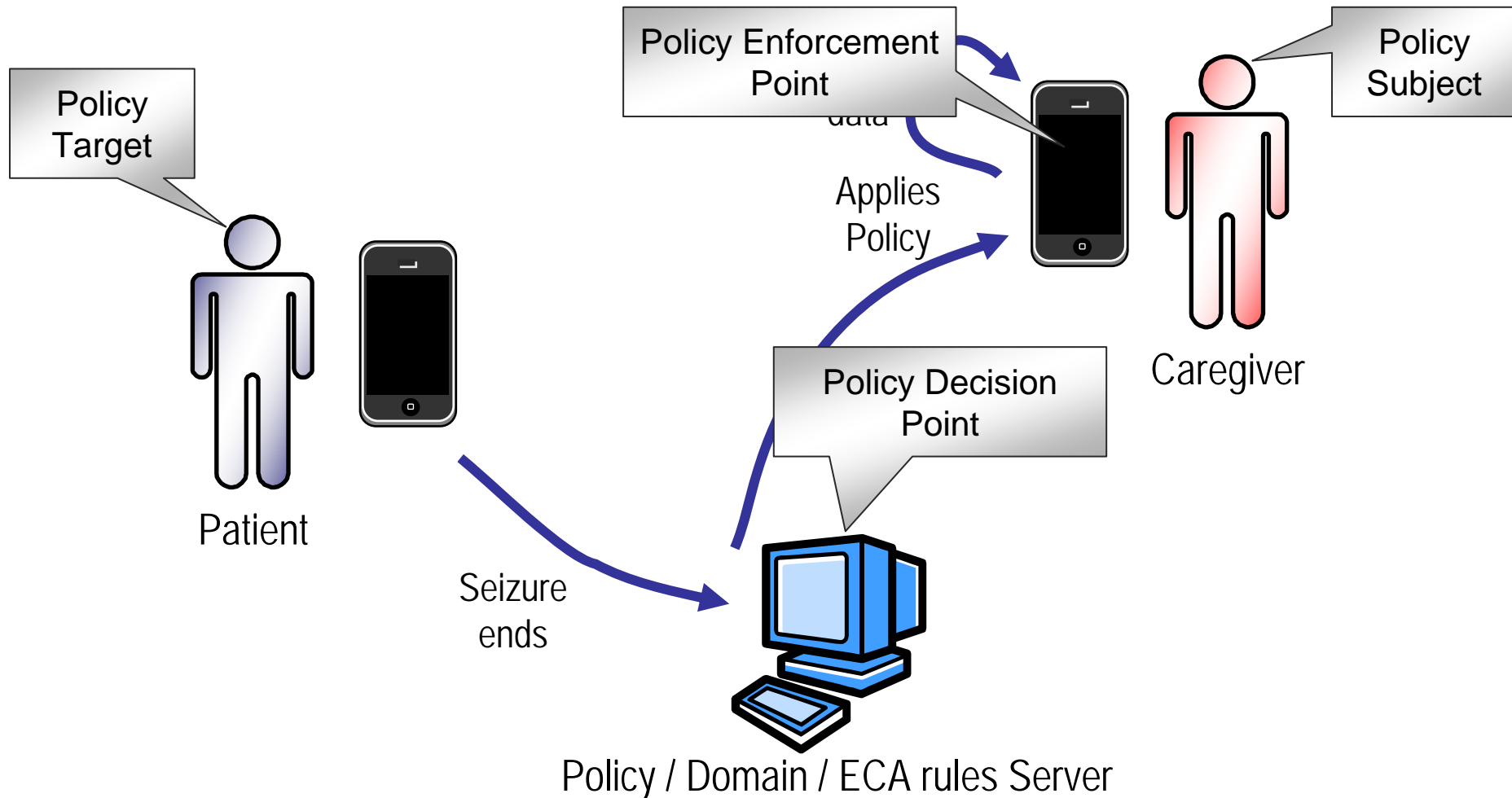
Health Scenario – Authorization



Health Scenario – Authorization



Health Scenario – Privacy Obligation



Details of Management Life Cycle

- Context situations of interest are defined during design time
- Management domains are specified based on the events of context situations and policies that should be deployed in each case
 - Administrator
 - User defined (simplified interface)
- Association of context-situations and policies can be visualized in an user interface

Ongoing Implementation

- ECA rules for context situations using JESS rule engine
- Policies using Ponder2
- PDP and context-aware management domains tool using Java

Conclusions and Future Work

- CAMDs combine existing context modeling and policy management frameworks
- Allows dynamic management of different types of policies based on context situations
- Less intrusive and flexible using context situations and events
- Future work: performance evaluation (response time, scalability, and bandwidth consumption); Usability and usefulness studies

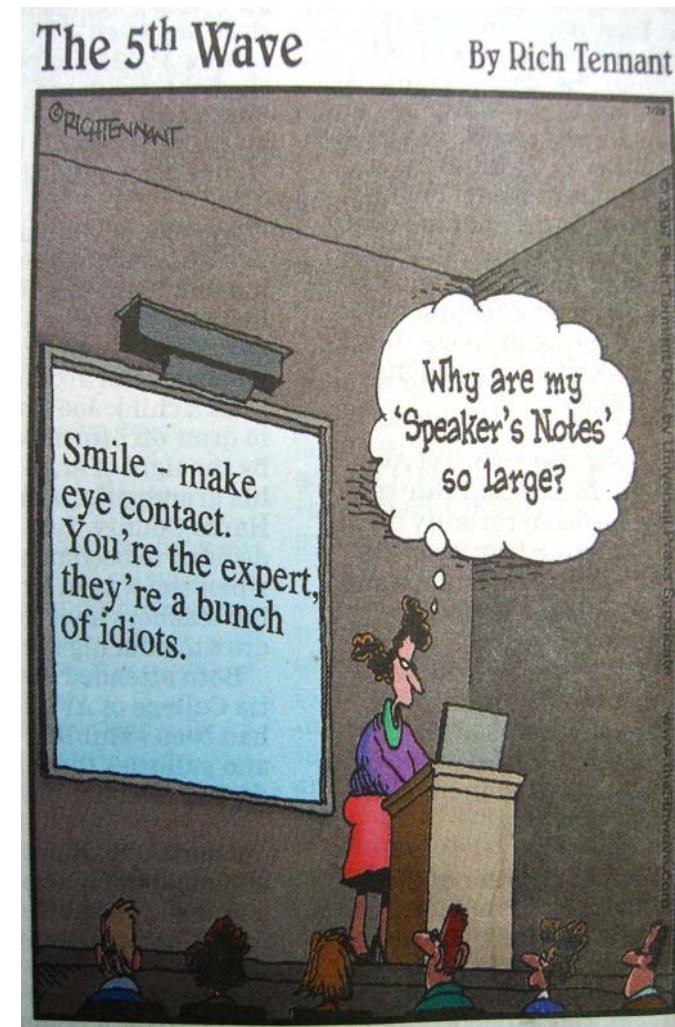
Thank you!

Contact information:

Ricardo Neisse

R.Neisse@ewi.utwente.nl

<http://www.ewi.utwente.nl/~neisser>



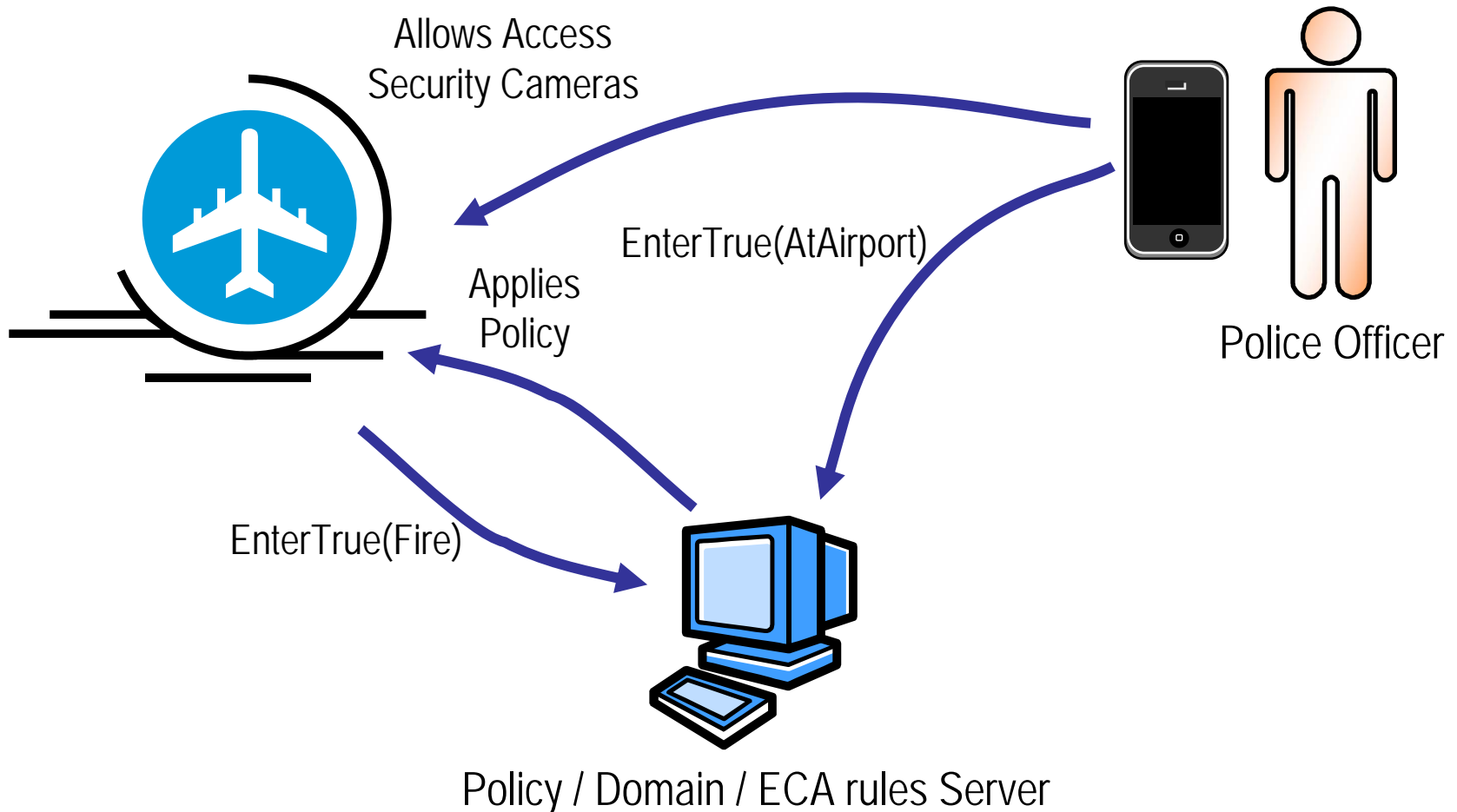


Universiteit Twente

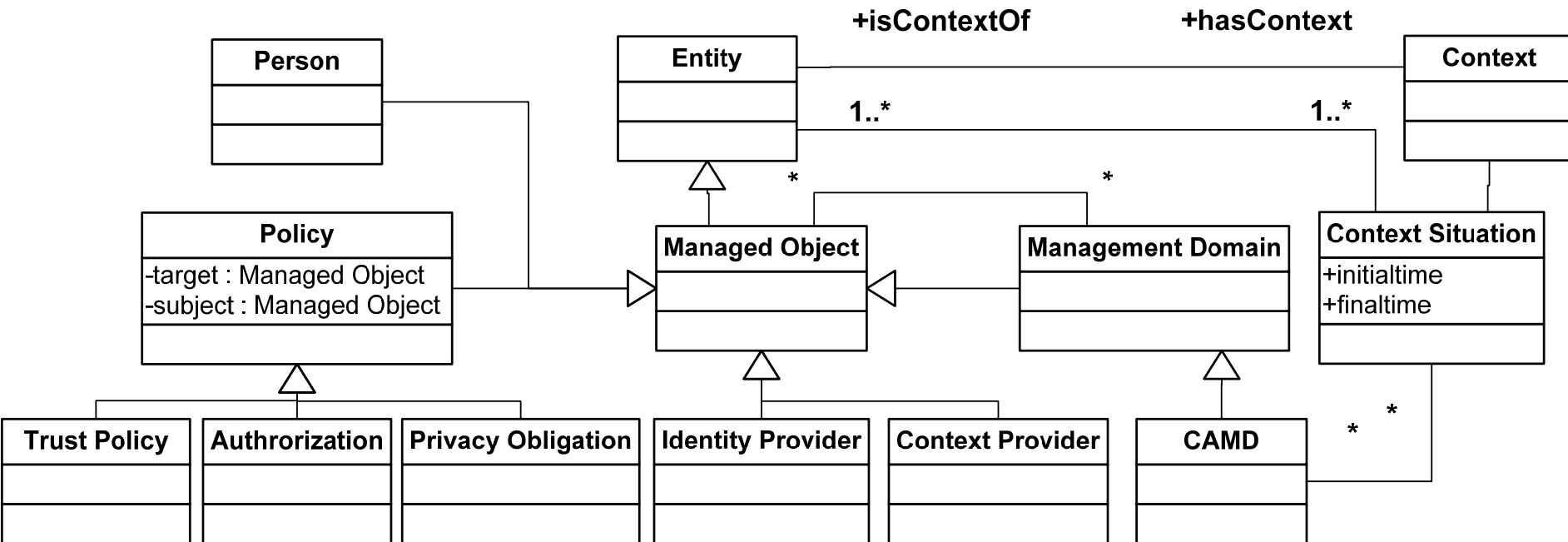


Universiteit Twente
de ondernemende universiteit

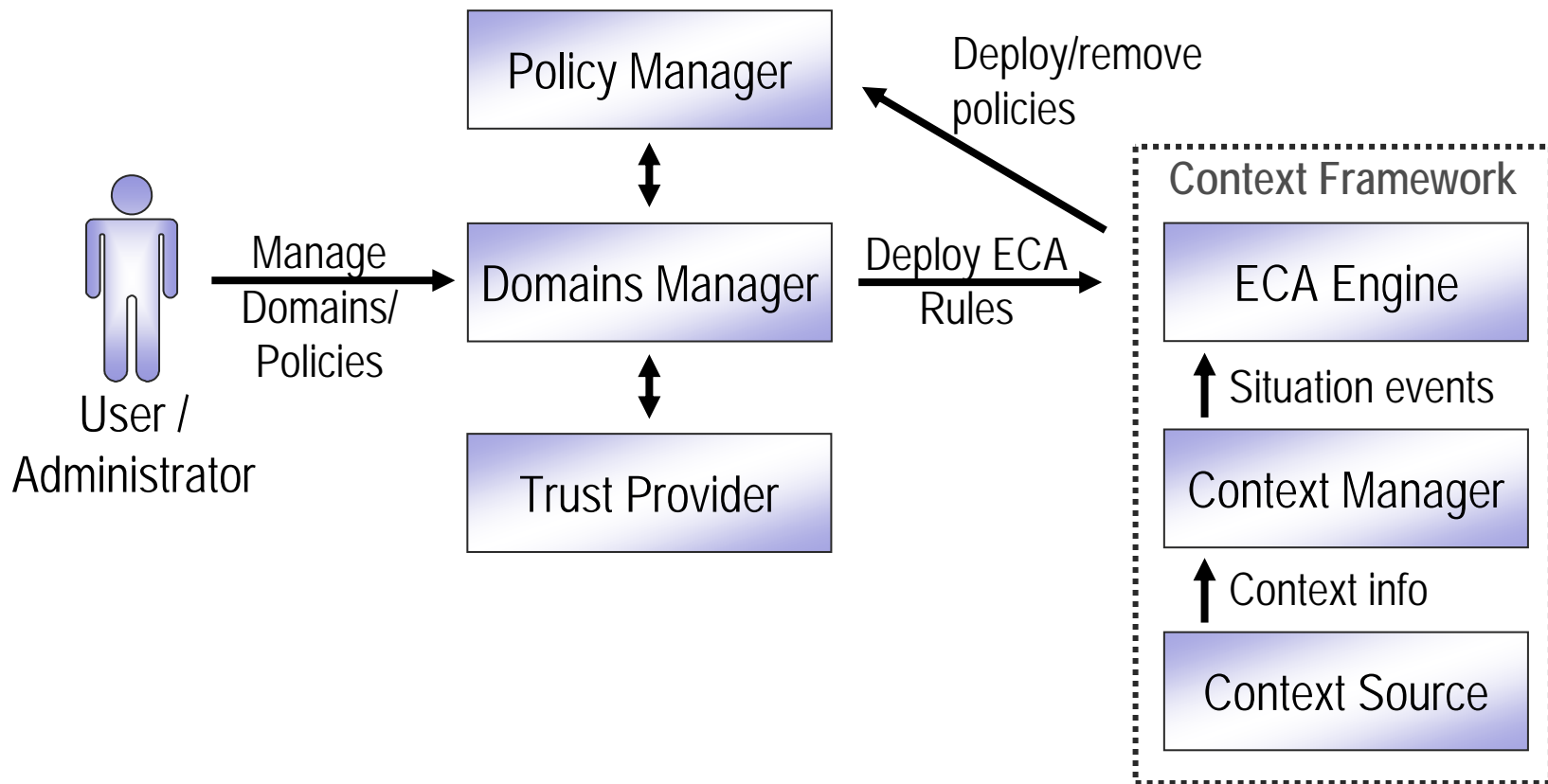
Airport Scenario – Fire Situation



Information Model



Architecture



Domains Management/Visualization

Context-Aware Domains Management

Description:

Context situation: ▼

Time:

Policy **Visualization**

Policy subject: ▼

Policy target: ▼

Policy action: ▼

Evaluation

- Use case scenarios for the different types of policies and situations of interest
 - Is it useful for users?
- Performance of rule execution versus policy deployment time
 - How much time does it take to detect situations, deploy/remove policies?
 - Are the context/situation changes very dynamic?
- Scalability evaluation with large number of users, policies and very dynamic context information
 - What is the impact of the dynamicity in the scalability of the system?
 - Event detection window size