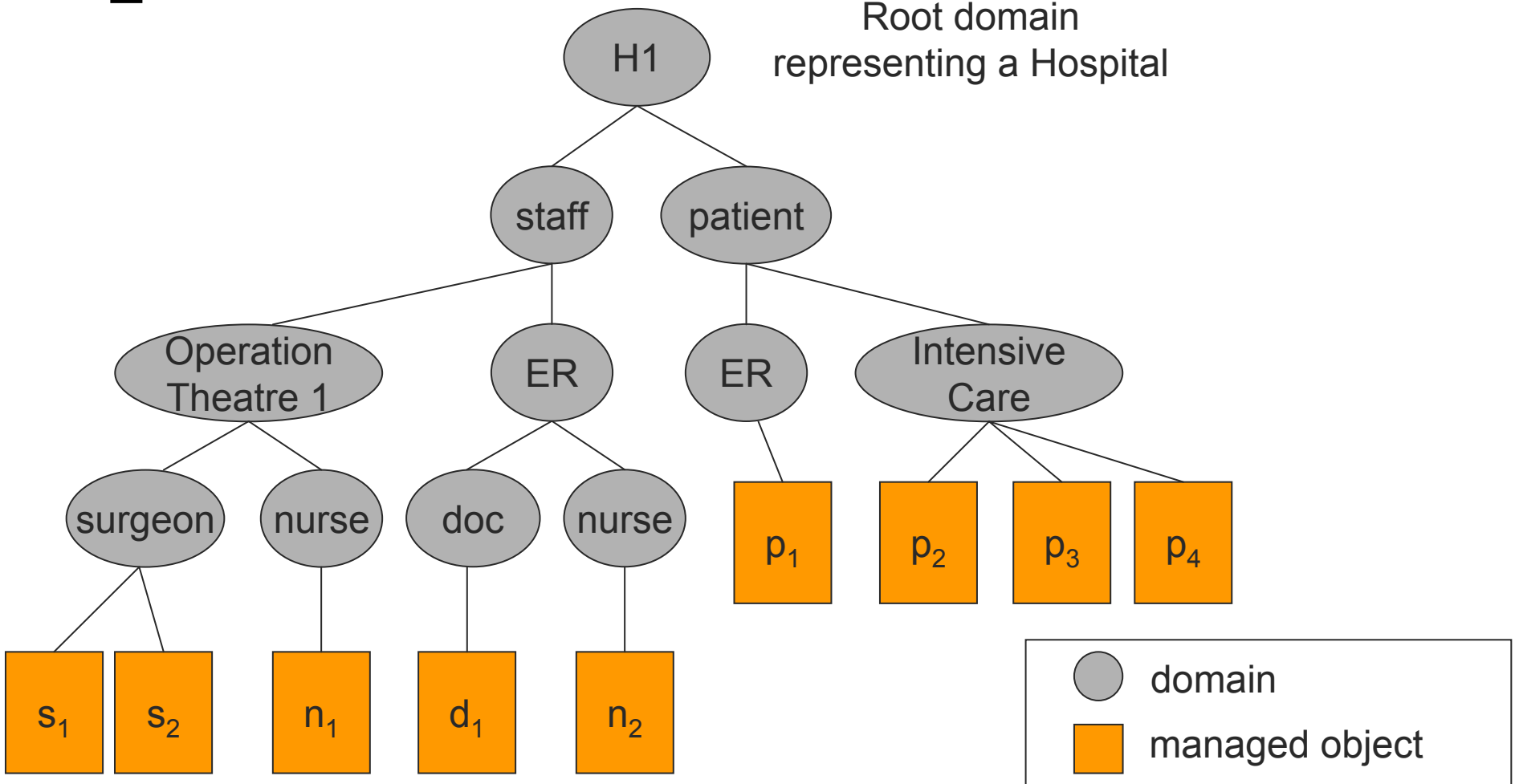
Authorisation and Conflict Resolution for Hierarchical Domains

Giovanni Russello
Changyu Dong
Naranker Dulay

Ponder in a nutshell

- A policy language
- A policy interpreter that:
  - Operates on *Managed Objects (MO)*
  - Organizes MOs in a hierarchical domain structure
  - Enforces policies defined on MOs

A Domain Structure Example



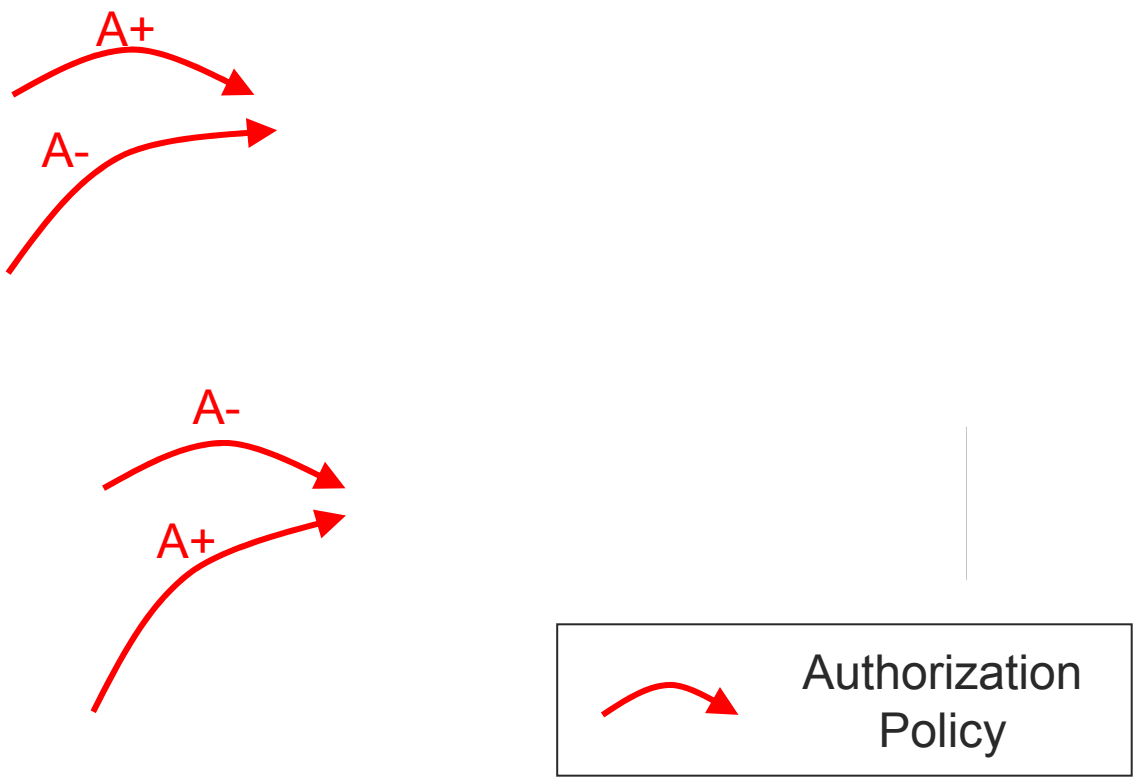
Ponder Authorisation Policies

Control over the execution of a MO's actions

auth+/- subject, action, target

when condition

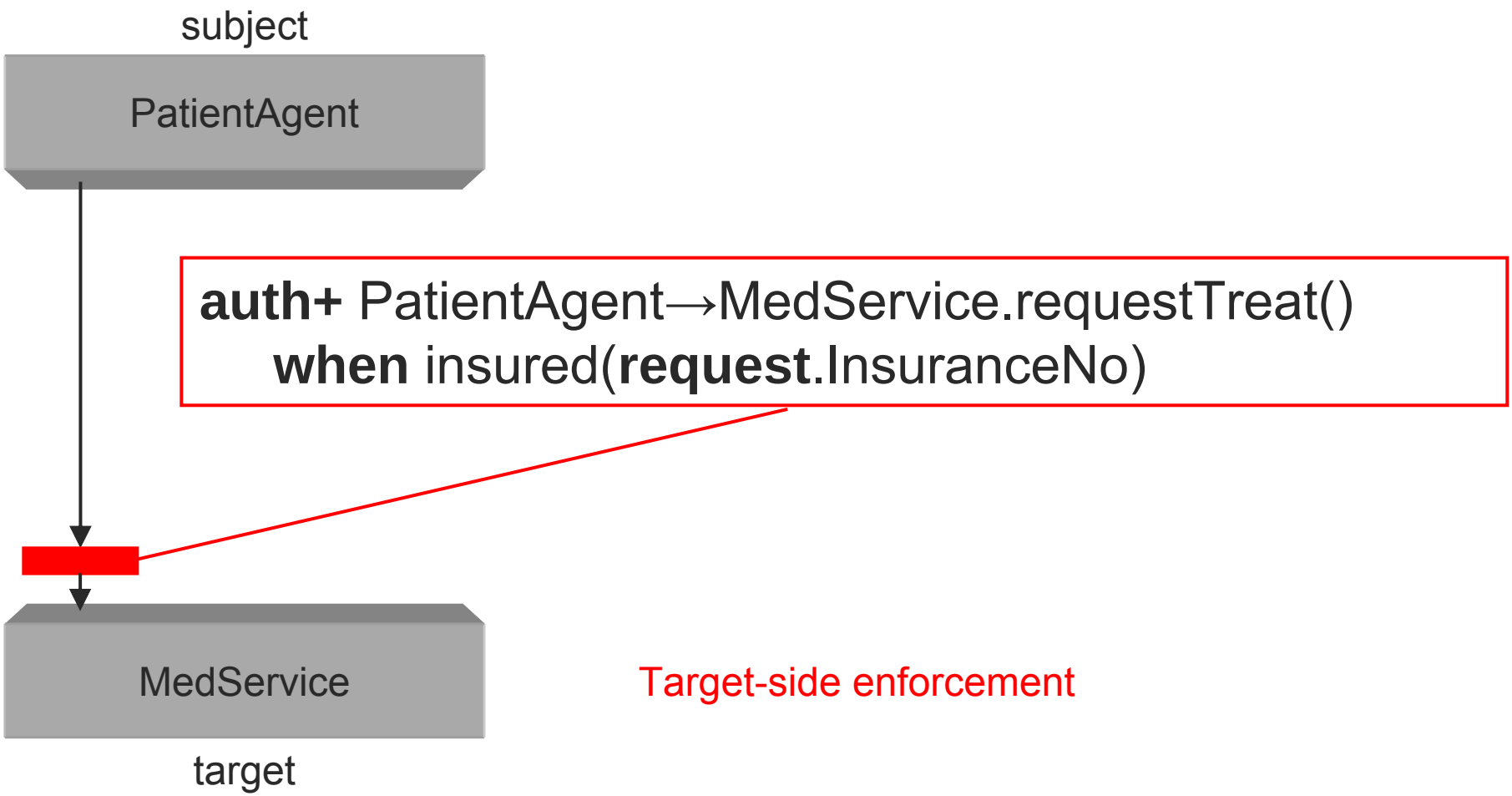
From Domains to Privileges



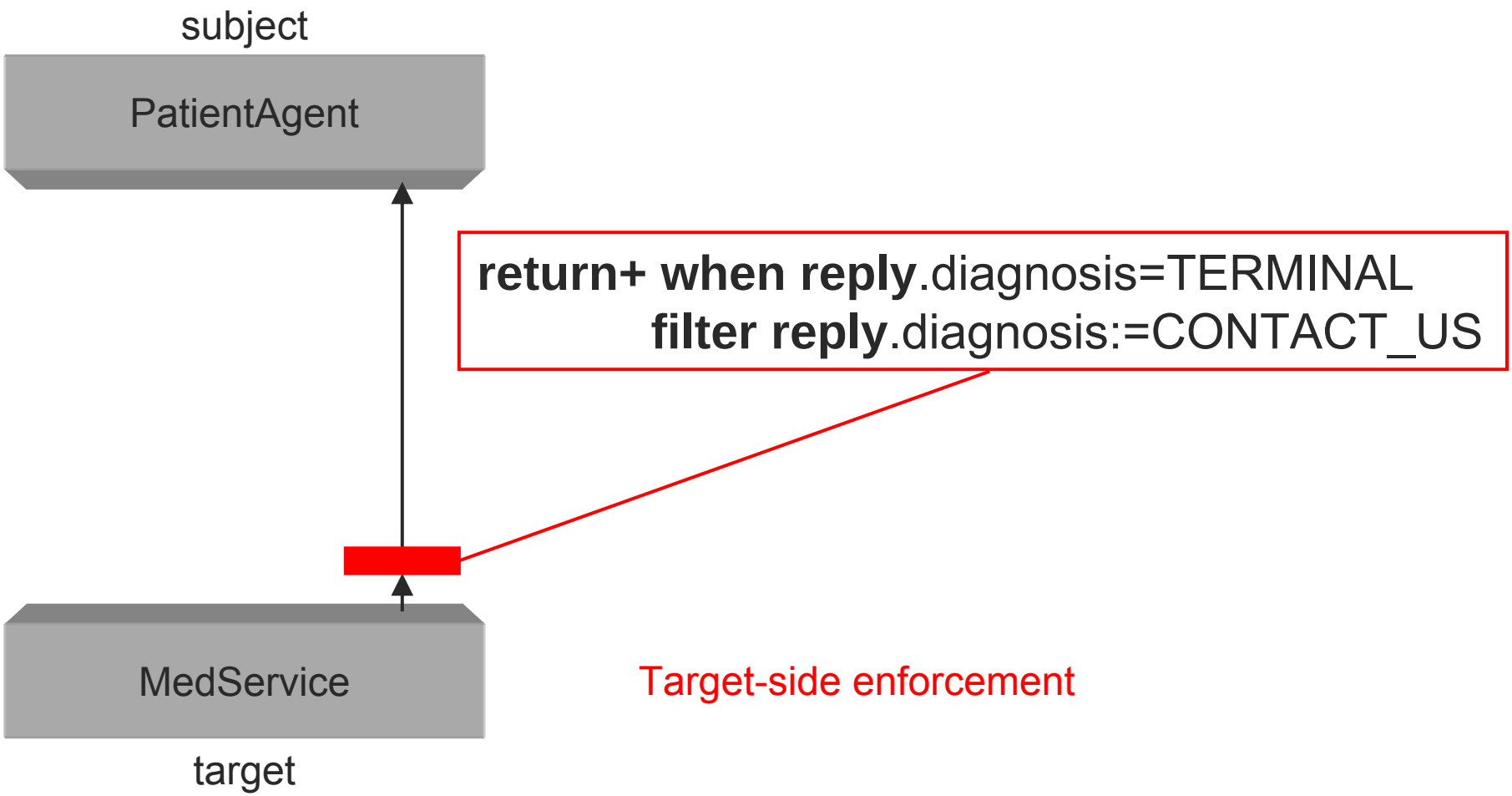
Authorisation Policy Extension

- Traditionally auth policies are used to protect the target side of an action
- A finer grained control can be achieved if the policy enforcement points (PEP) are applied also at the subject side

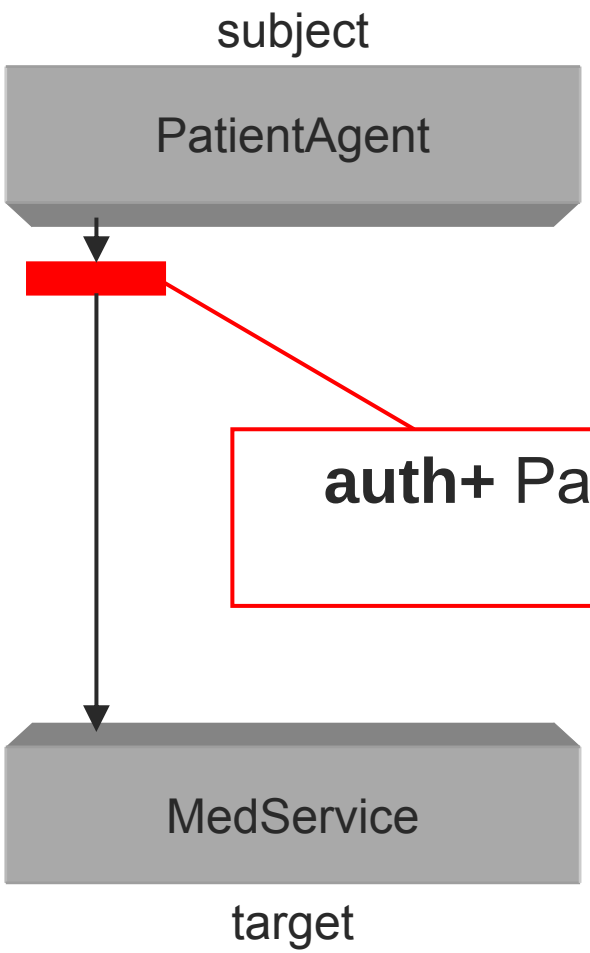
Authorisation Policy Enforcement



Authorisation Policy Enforcement



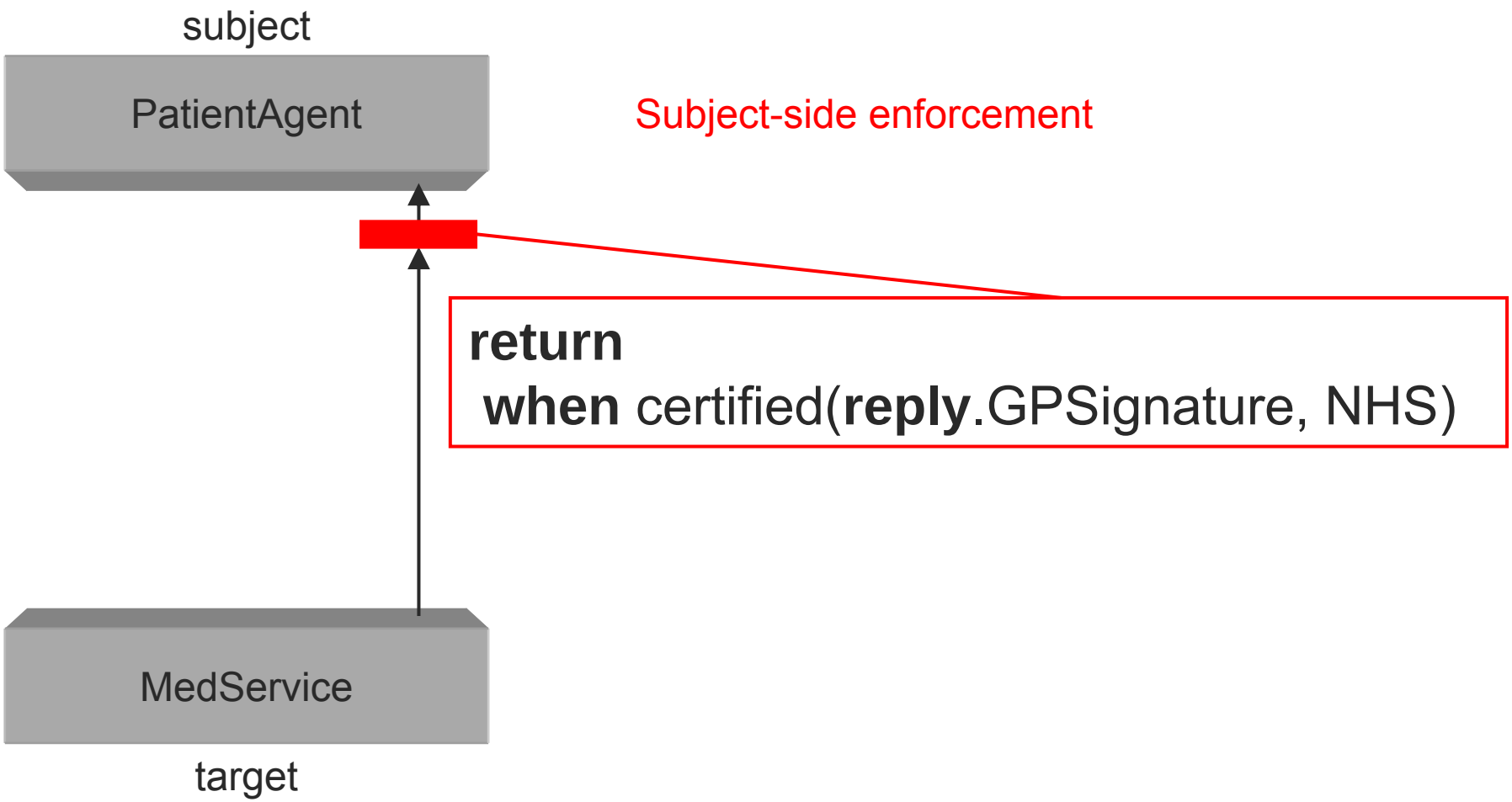
Authorisation Policy Enforcement



Subject-side enforcement

auth+ PatientAgent.requestTreat()→MedService
when certified(MedService, NHS)

Authorisation Policy Enforcement



Conflicts

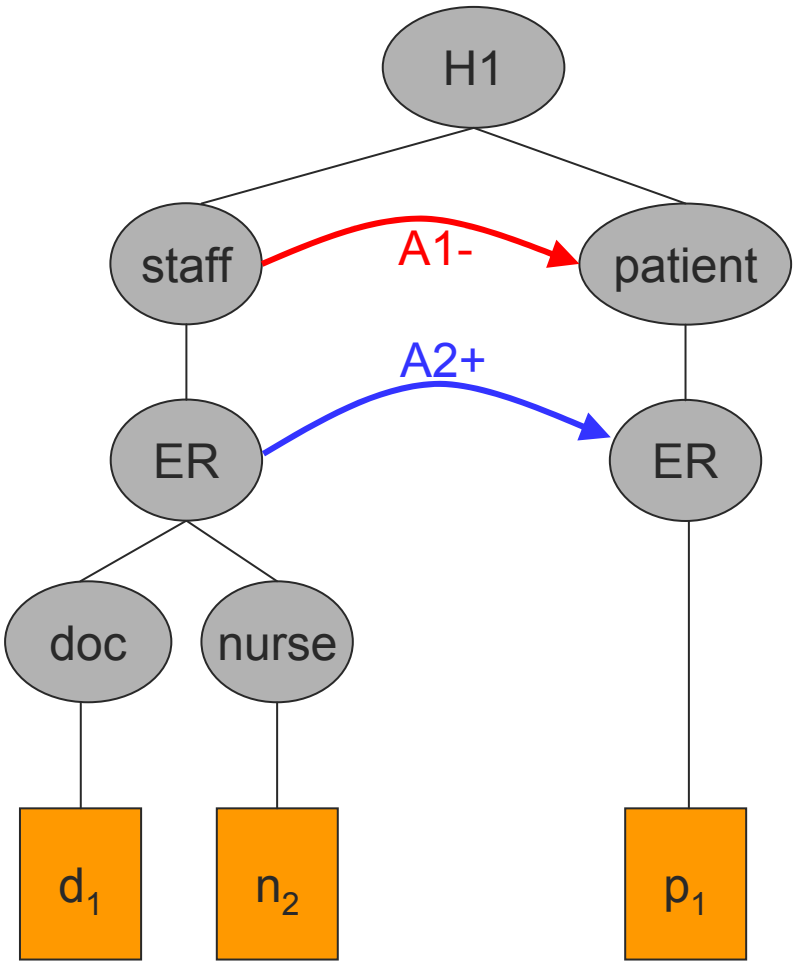
- Modal Conflicts: two or more policies with opposite sign apply to the same (subject, target action)-tuple
- Application-specific Conflicts: the entities defined in the policy conflict with external criteria (conflicts of interest, conflicts of duties, conflicts of resource allocation,…)

We focus on Modal Conflicts only!

Modal Conflict Resolution

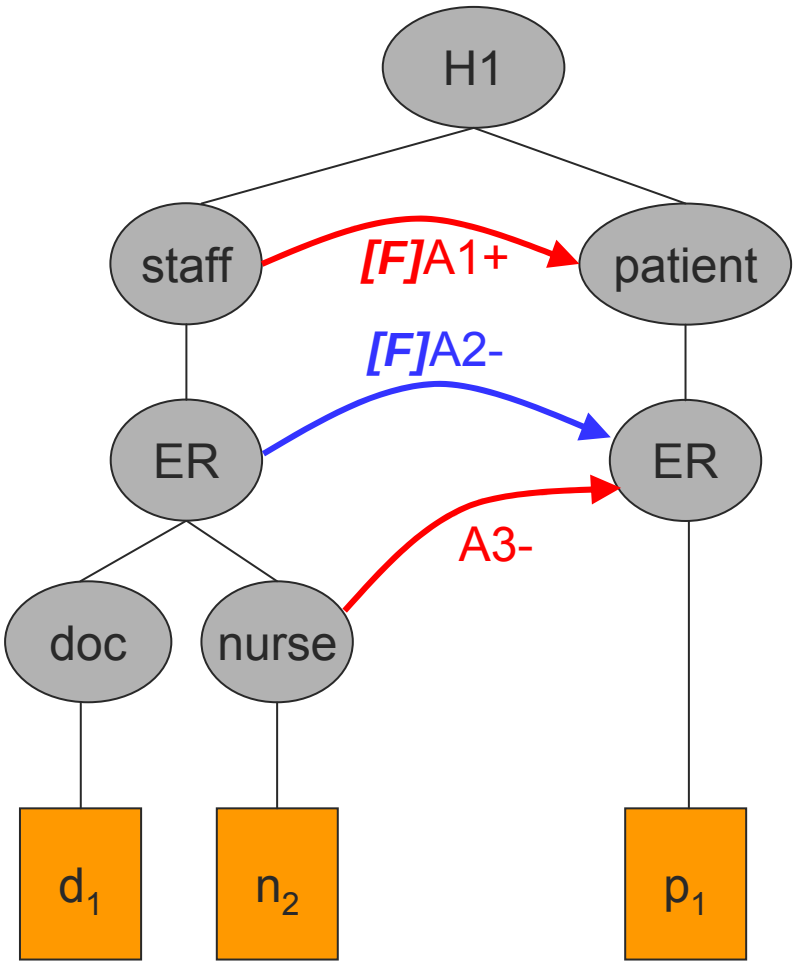
- Based on domain hierarchy
- Set of rules applied during the execution of operations where:
  - Most specific authorisation policies get priority
  - It is possible to define *final* authorisation policies that overwrite any policies down the domain structure

Conflict Resolution Example



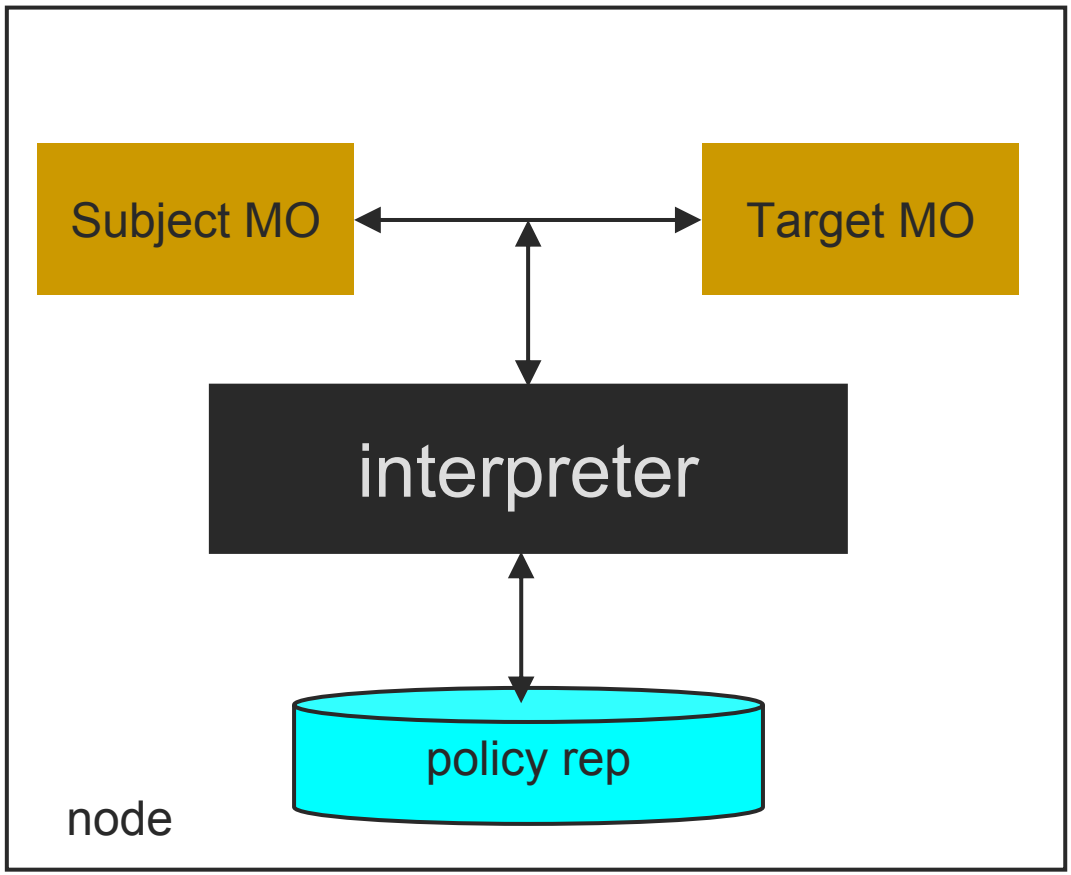
Most specific authorisation policies get priority

Conflict Resolution Example

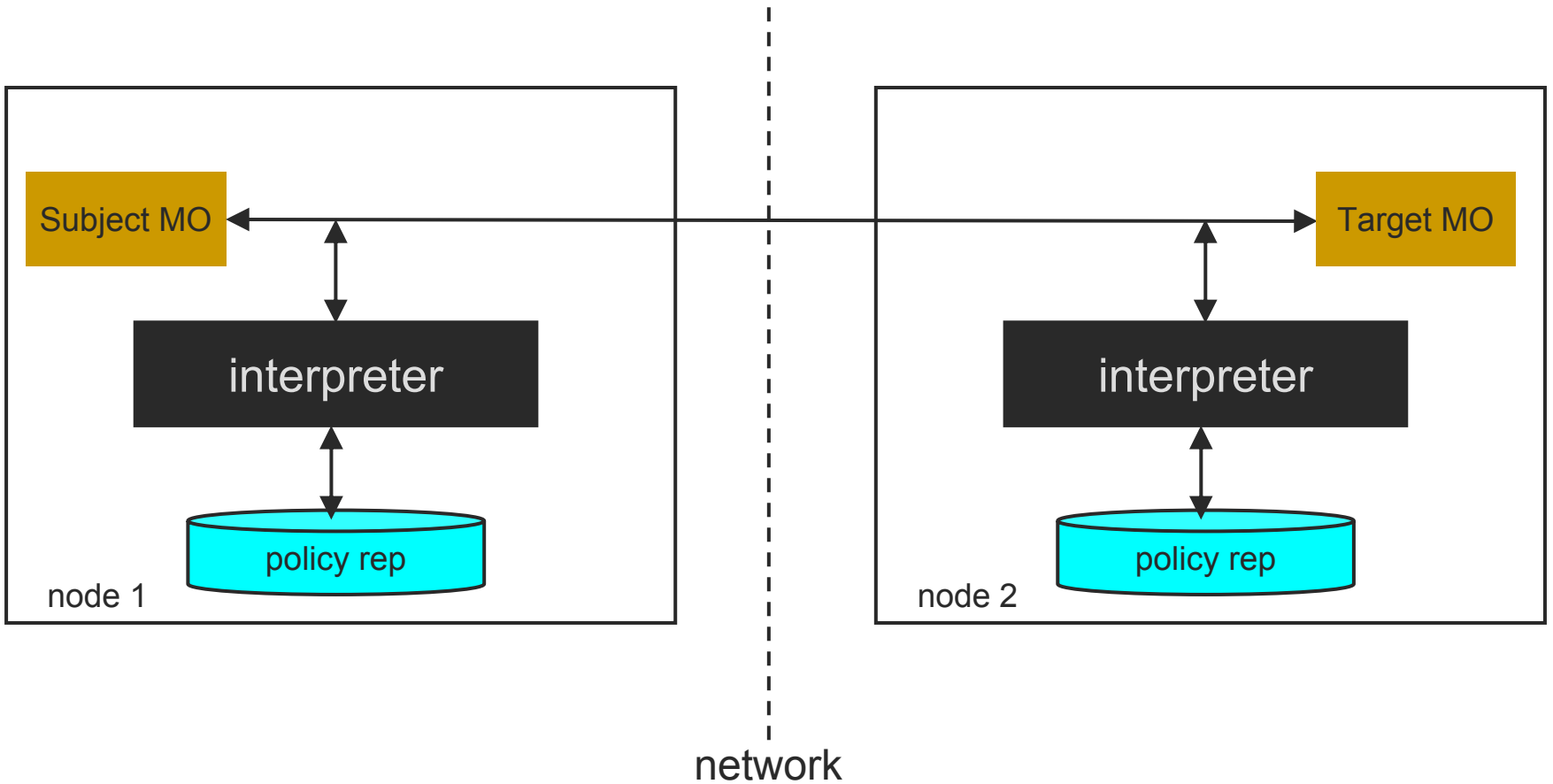


Most generic ***Final*** authorization policies get priority

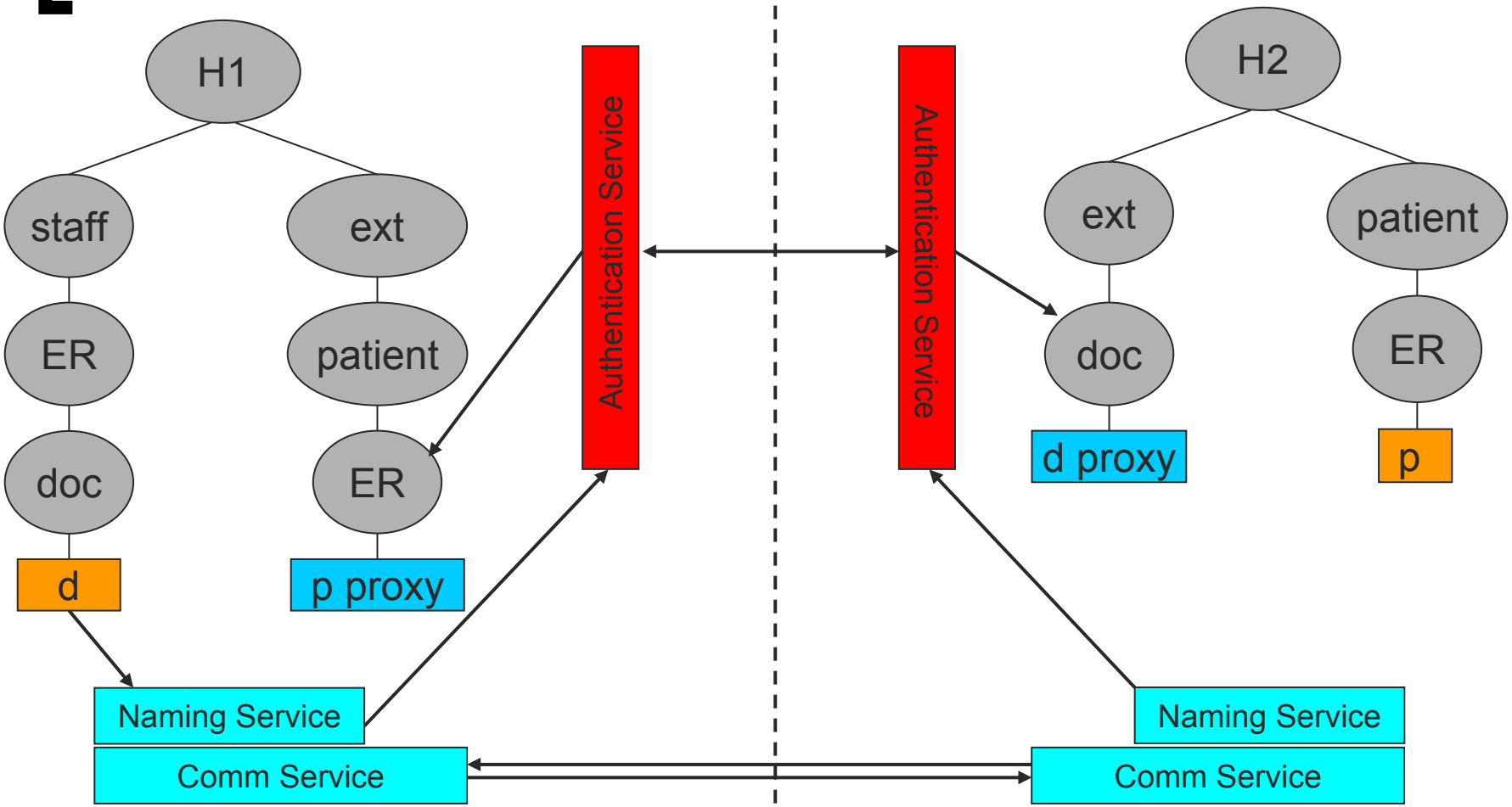
Policy Enforcement – Local Scenario



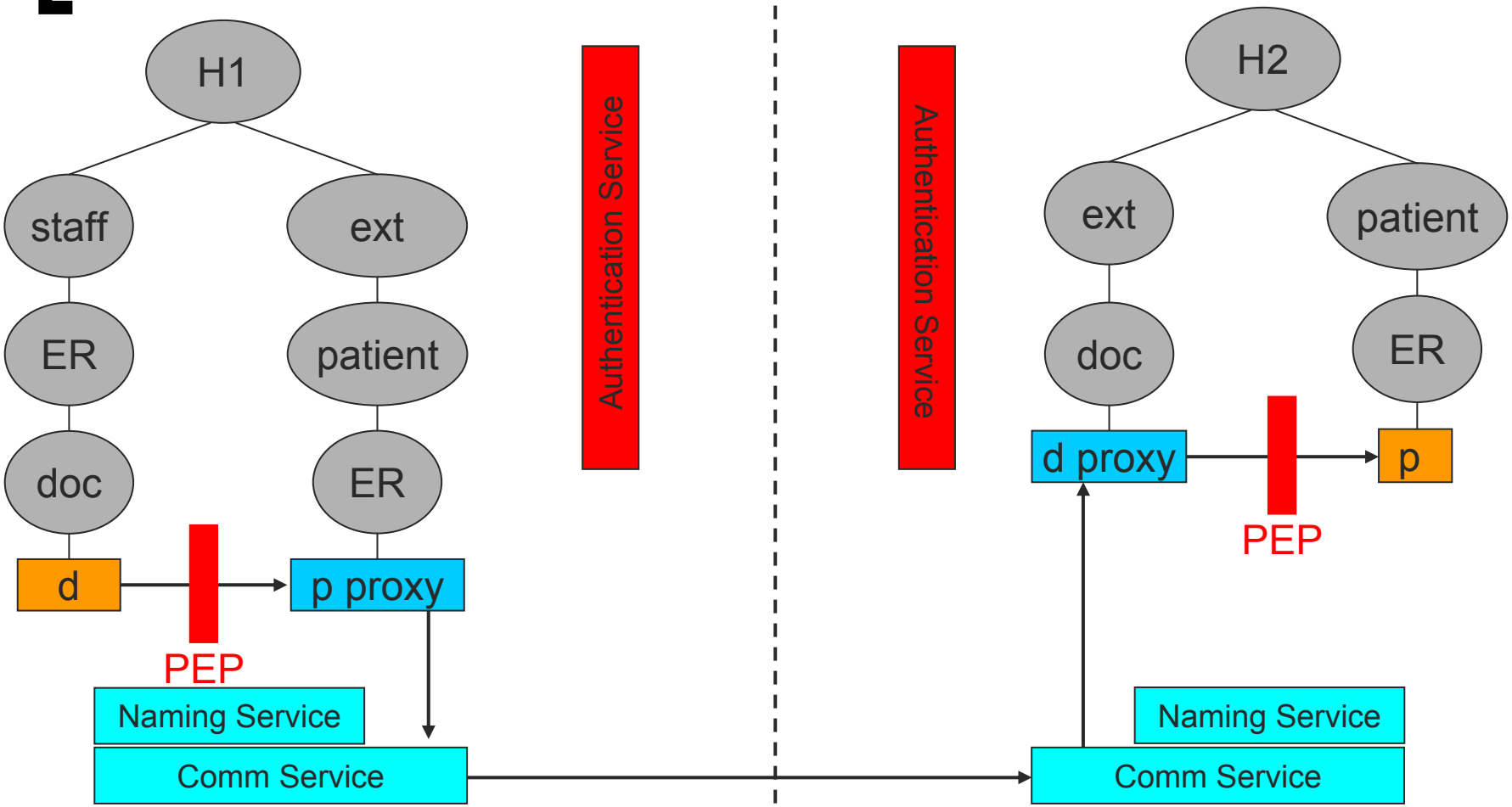
Policy Enforcement – Remote Scenario



MO Remote Invocation



MO Remote Invocation



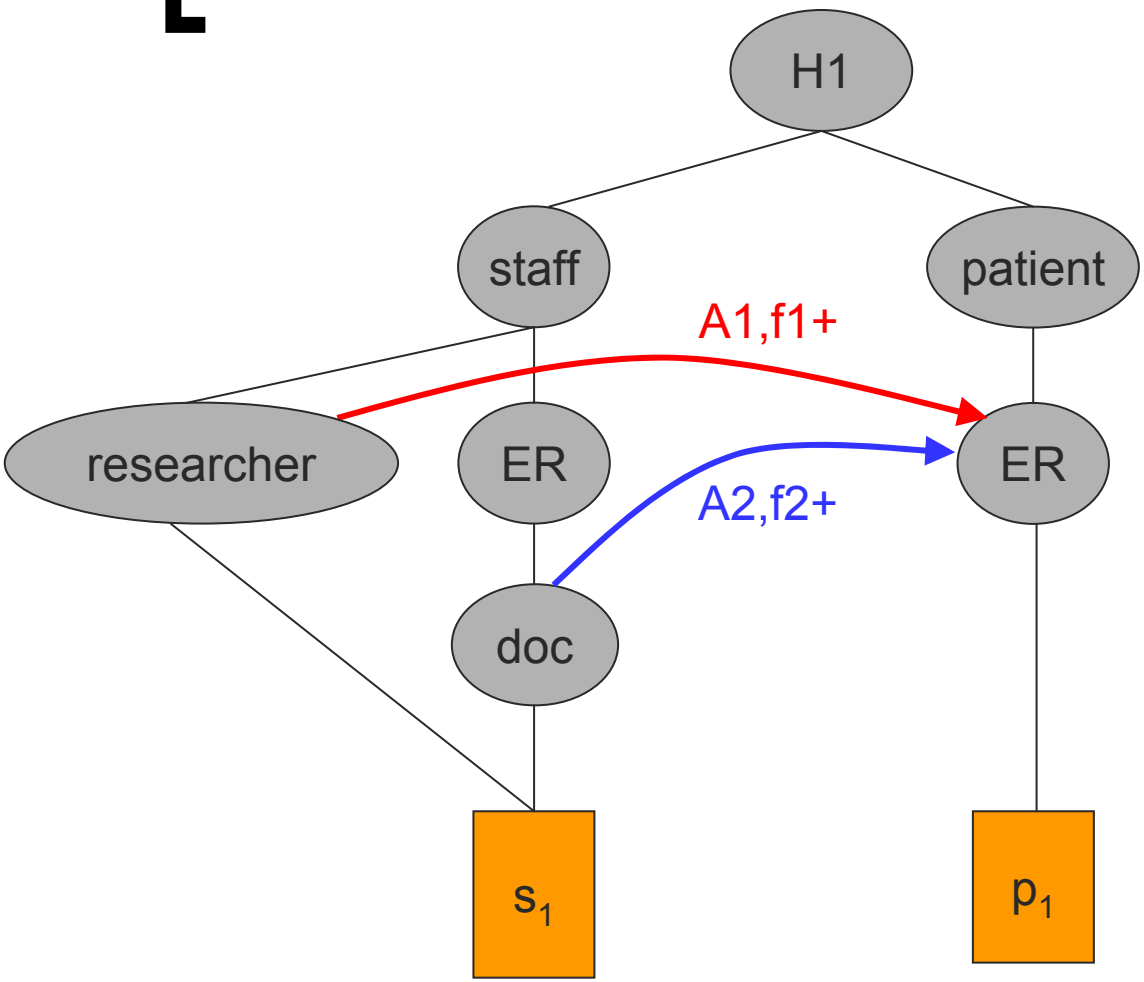
Conclusions & Future Work

- Uniform framework that caters for both subject and target protection
- Deterministic resolution of conflicts

Future directions:

- Logging of conflicts and resolution steps for off-line investigation
- Synchronization of policies after domain structure changes

Filter Conflict Resolution Example



Patient record:
<name,age,address,symptoms>

Filter Conflict Resolution Example

A1
return+
researcher→patient.readRec()
filter f1 **reply**.name:=**NULL**

A2
return+
doctor→patient.readRec()
filter f2 **reply**.address:=**NULL**

f1(<*name,age,address,symptoms*>)=<*NULL,age,address,symptoms* >

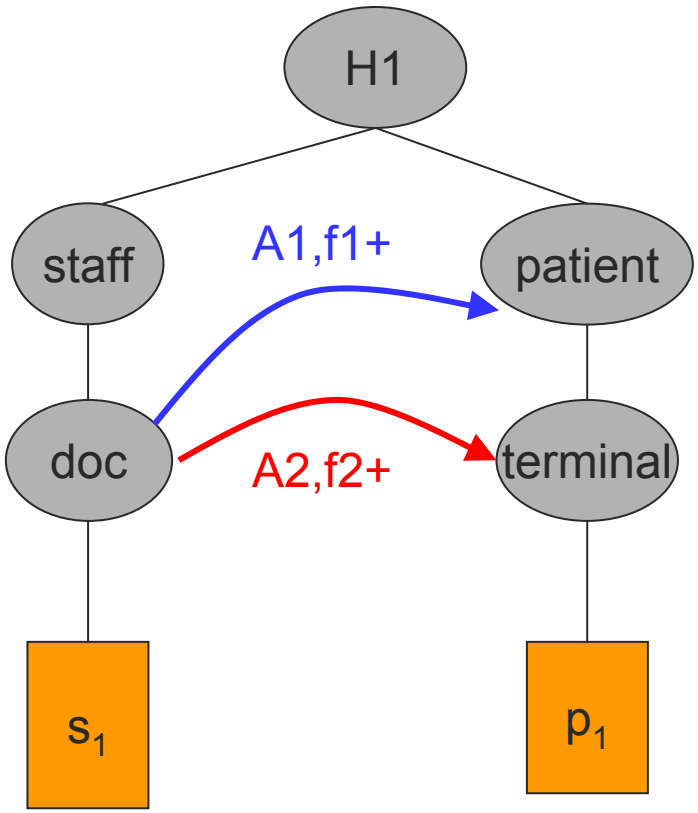
∧

f2(<*name,age,address,symptoms*>)=<*name,age,NULL,symptoms* >

↓

<*name,age,address,symptoms* > + logging of the output result

Filter Conflict Resolution Example



Patient record: <name,age,address,symptoms>

Terminal Patient record: <name,age,address, symptoms,LE>

Filter Conflict Resolution Example

A1
return+
doc→patient.readRec()
filter f1 **reply**.name:=**NULL**

A2
return+
doctor→terminal.readRec()
filter f2 **reply**.LE:=**NULL**

f1(<*name,age,address,symptoms*>)=<*name,age,NULL,symptoms*>

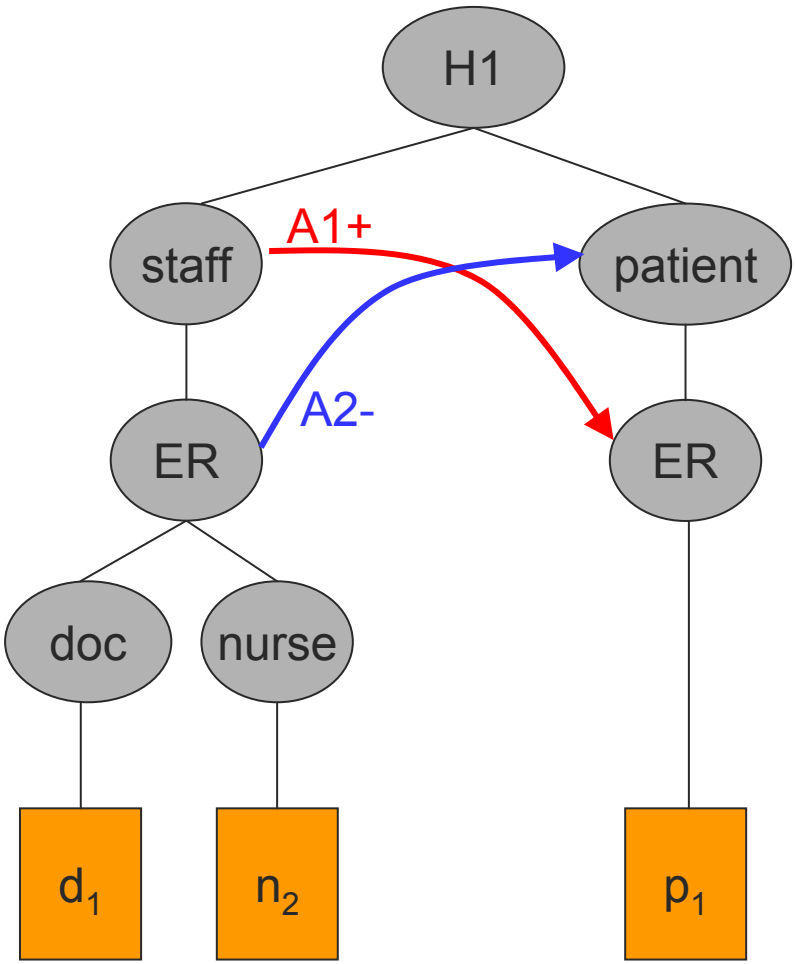
V

f2(<*name,age,address,symptoms,LE*>)=<*name,age,address,symptoms,NULL*>

↓

<*name,age,NULL,symptoms,NULL*> + logging of the output result

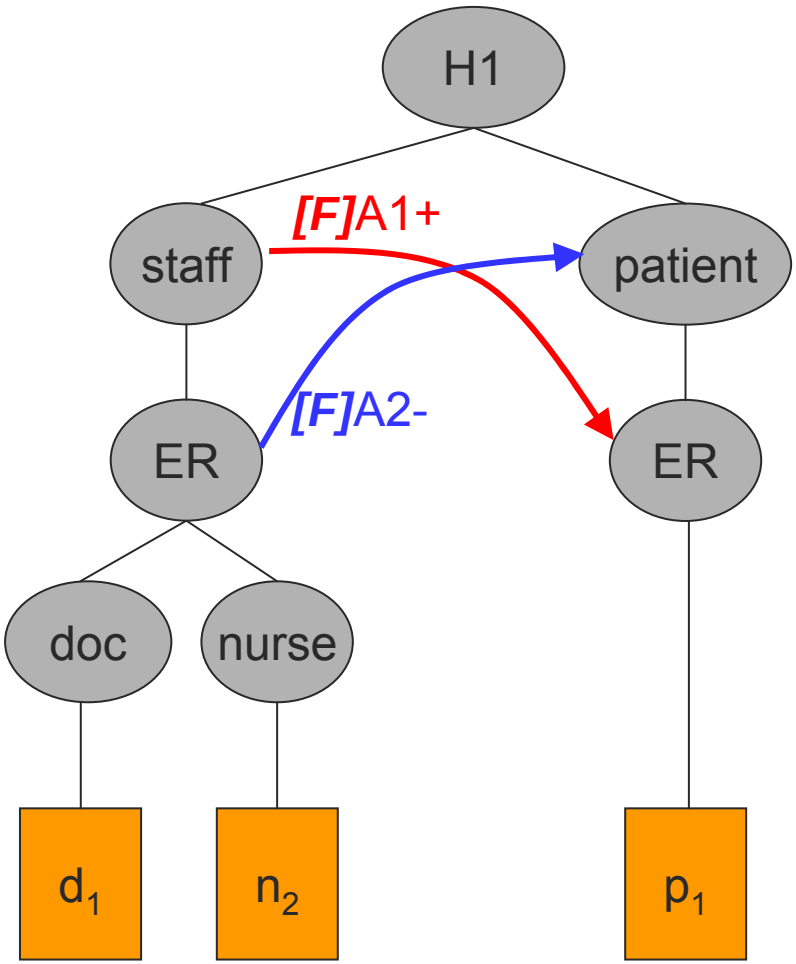
Modal Conflict Resolution Rules VI



(d) Currently, we use the subject path for discerning the priority of the policies.

i.e.: most specific policy = A2

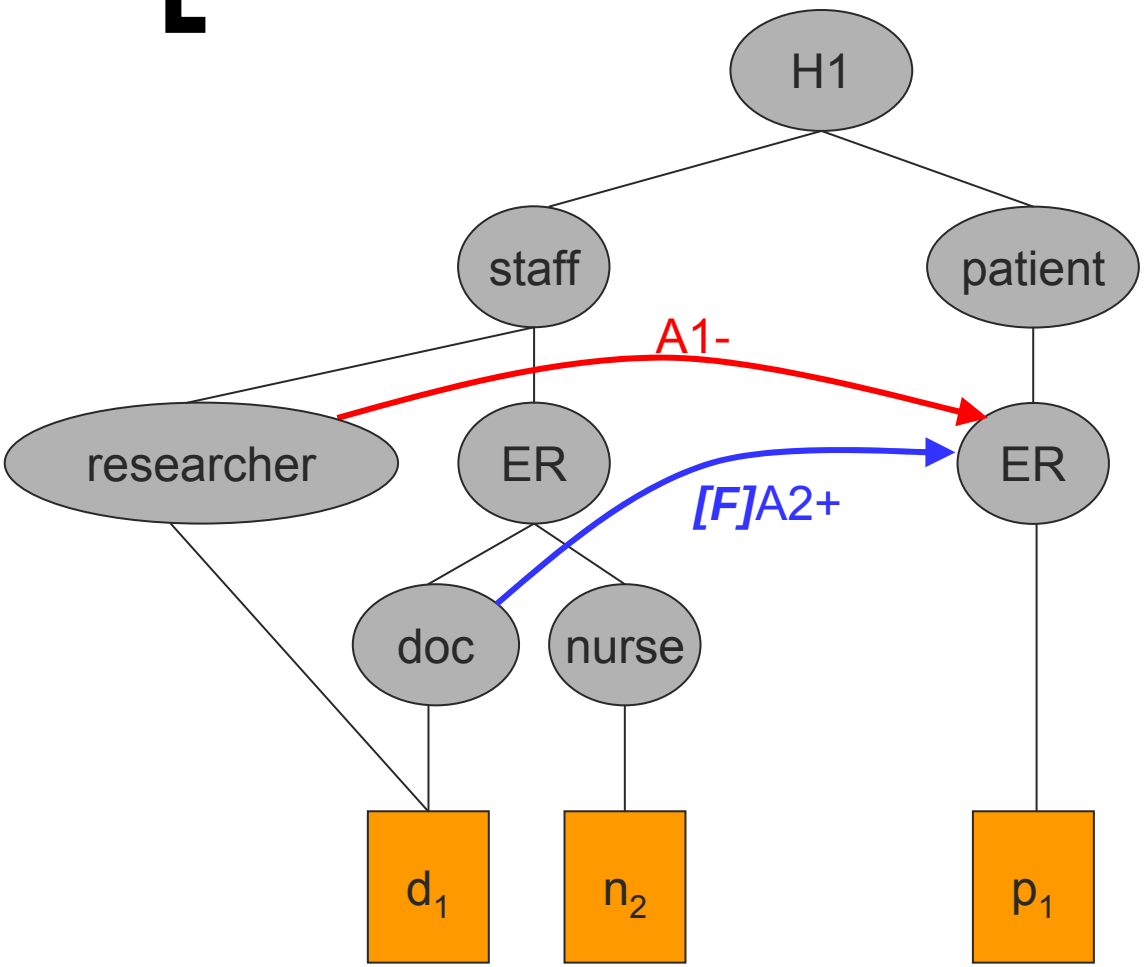
Modal Conflict Resolution Rules VI



(d) Currently, we use the subject path for discerning the priority of the policies.

i.e.: most general Final policy = A1

Modal Conflict Resolution Rules V

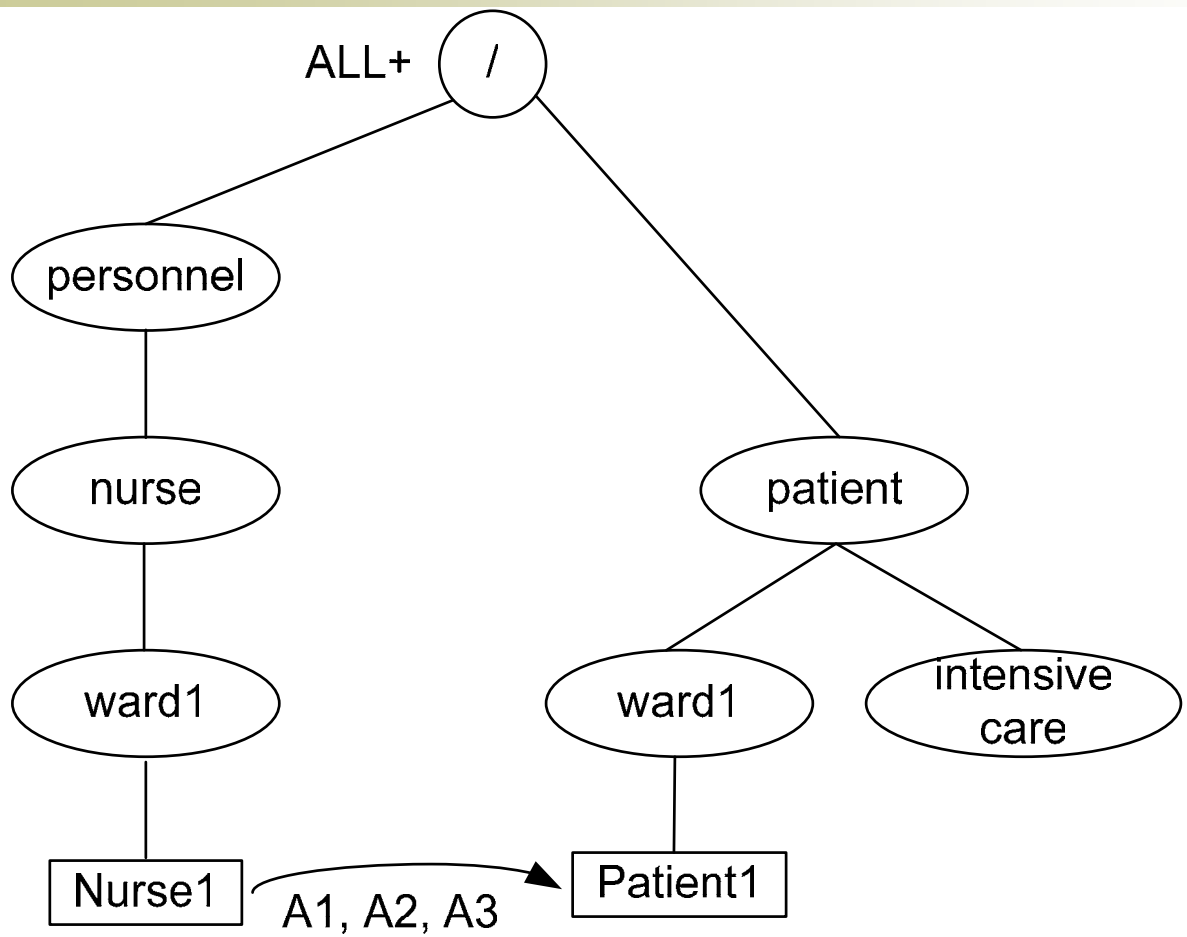


(e) After identifying for each path an applicable policy, resolve the conflict as if policies are defined at the same level (rule (a)).

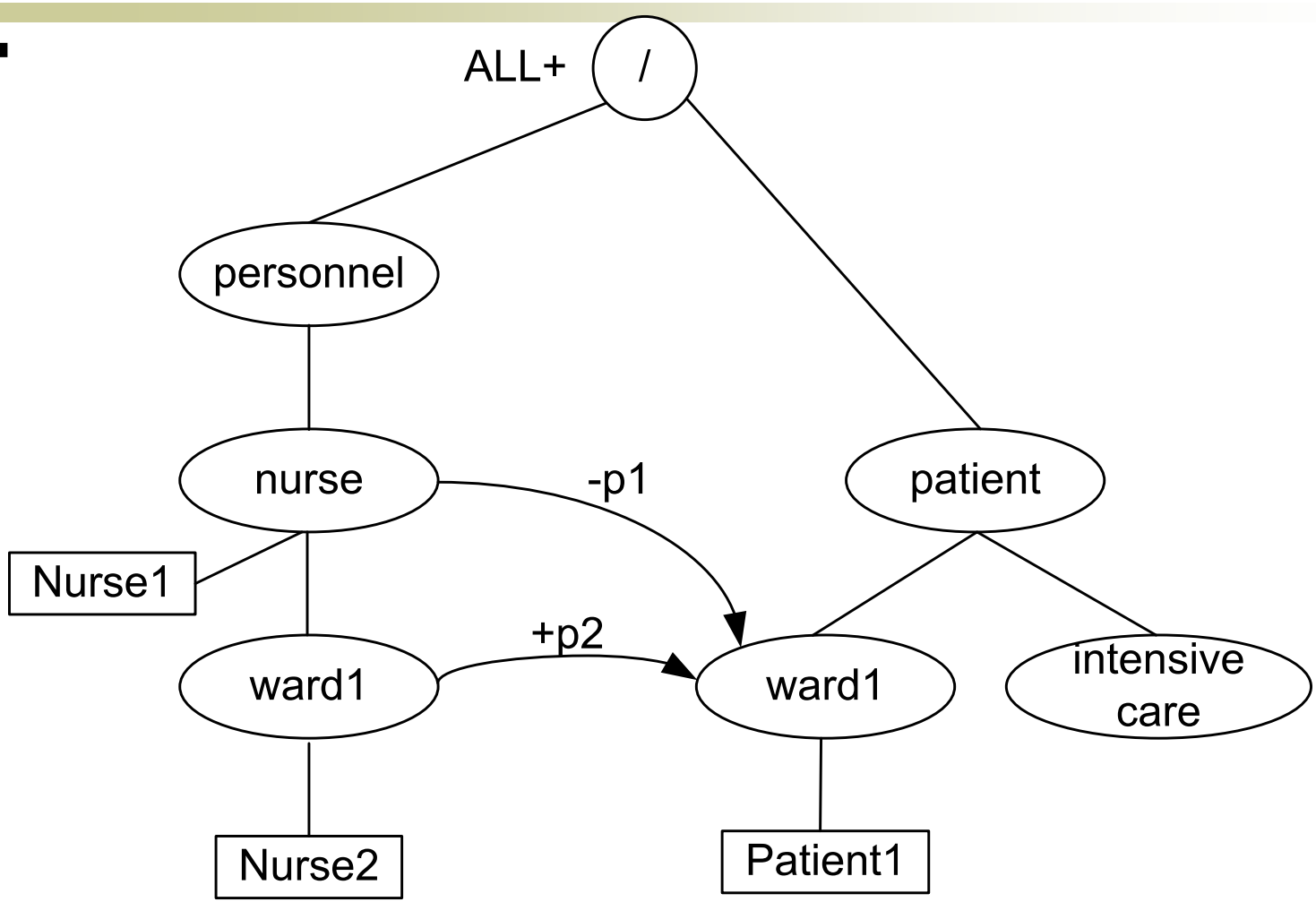
Demo

- Hospital domain structure
- Basic scenario
- Scenario 1 to 3 with different conflicts

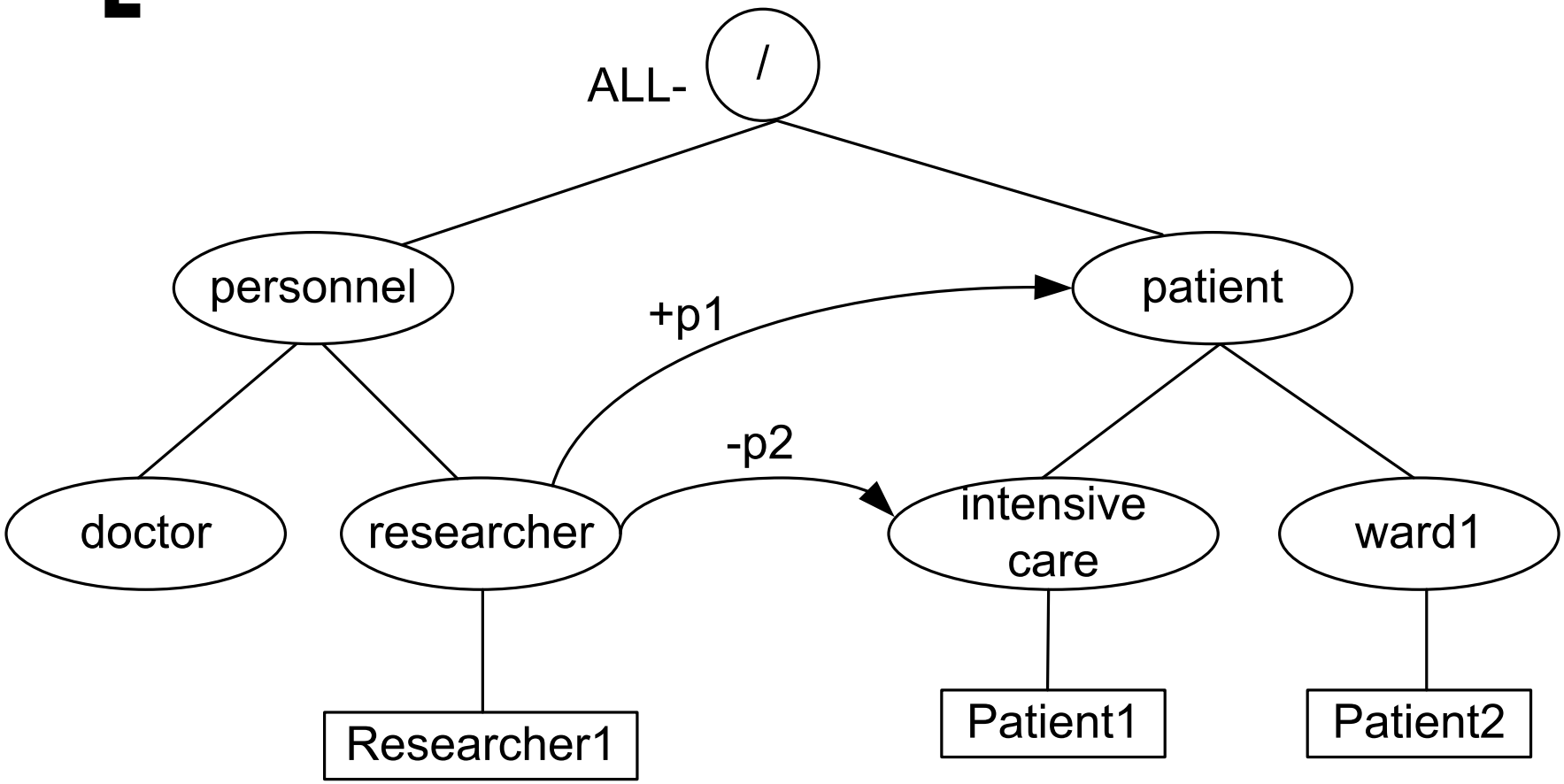
Demo – Basic Scenario



Demo – Conflict Scenario 1



Demo – Conflict Scenario 2



Demo – Conflict Scenario 3