PrimeLife

Data Privacy Management (DPM'09)

September 24, 2009

Obligation Language and Framework to Enable Privacy-aware SOA

L. Bussard

European Microsoft Innovation Center

(joint work with M. Ali and U. Pinsdorf)

A research project funded by the European Commission's 7th Framework Programme

Outlines

- PrimeLife
- Privacy in Service Oriented Architectures
- Shortcoming of State of the Art
- **Our Solution**
  - **Specifying Obligations**
  - **Enforcing Obligations**
- Future Work

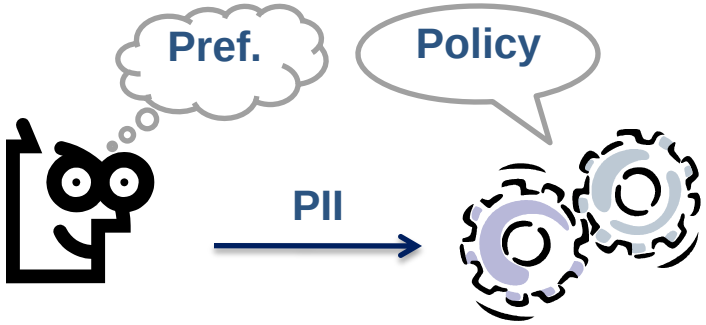
PrimeLife in a Nutshell

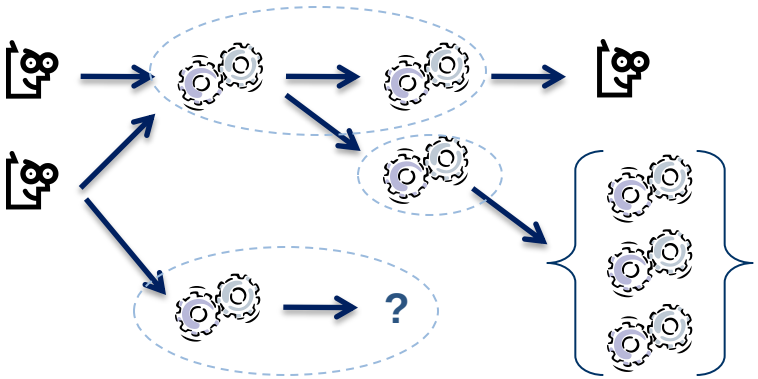
- Partners

- Technical Goals
  - Privacy policies and preferences
  - Anonymous credentials
  - User experience
- http://www.primelife.eu/

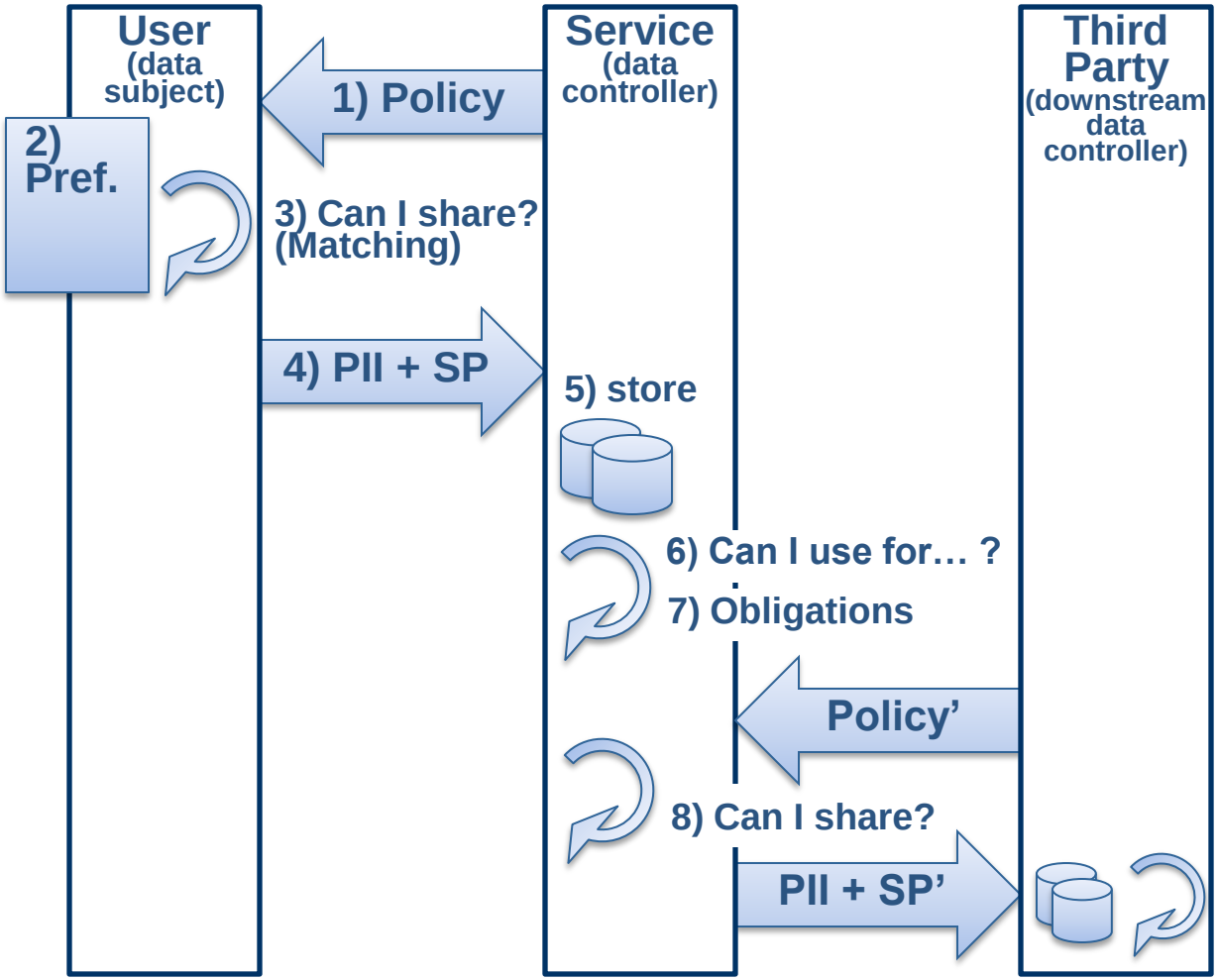
Privacy in SOA

- Variety of technologies: mash-ups, workflows, orchestrations
- Multi-hop data sharing
- Multiple trust domains
- Data from multiple users may be combined.
- Dynamic discovery and binding
- Persons may consume PII

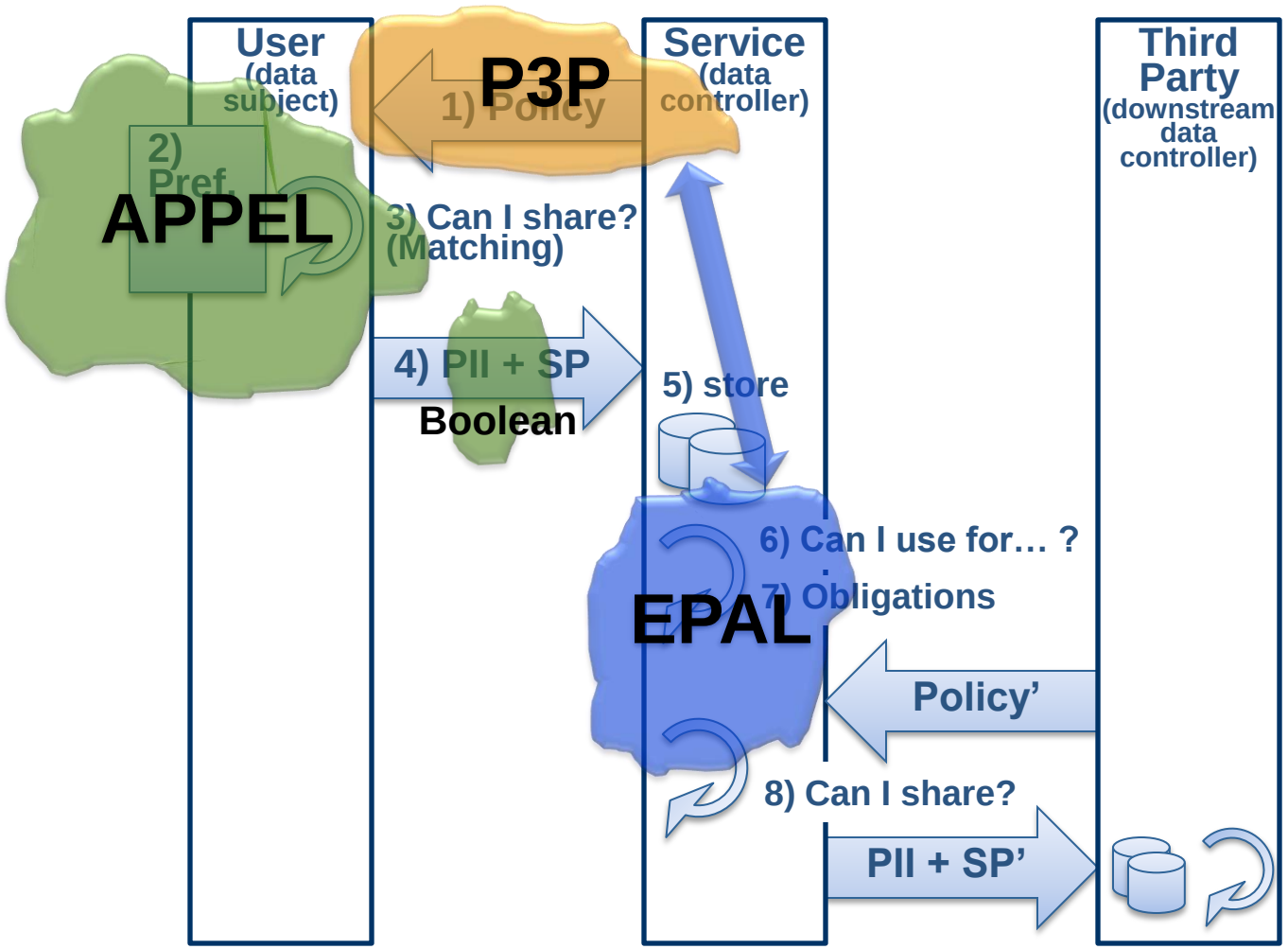




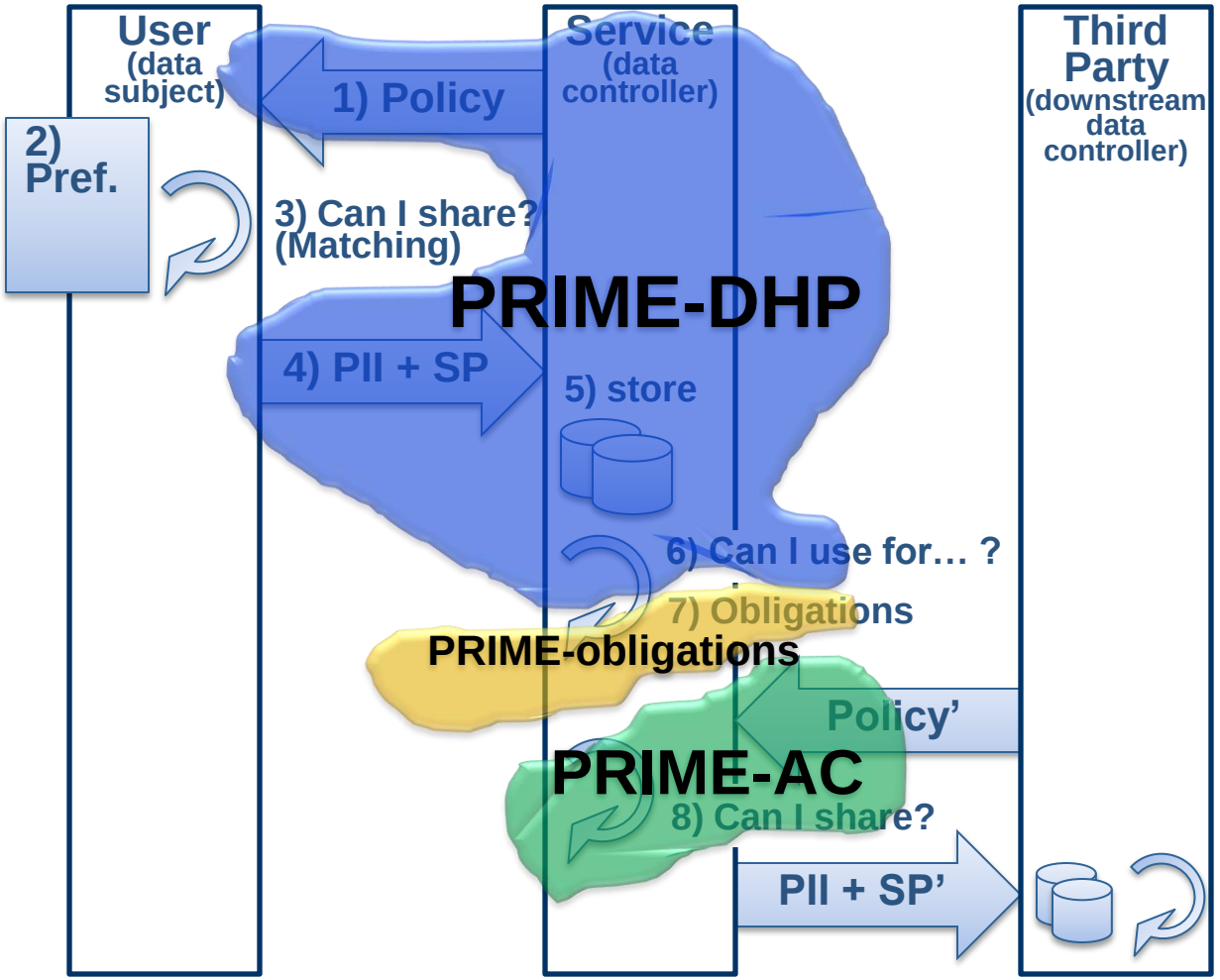
General Scenario



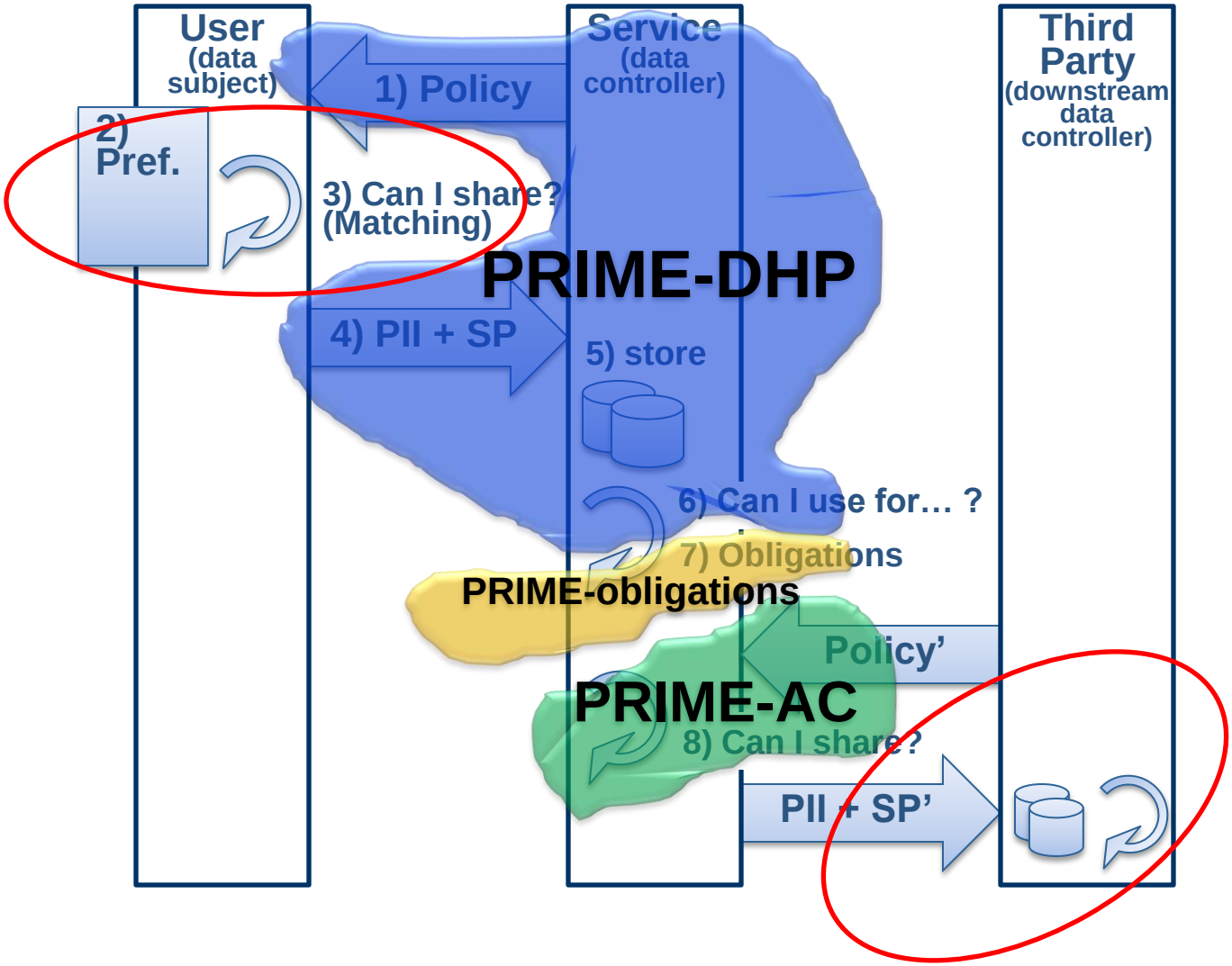
state of the art: APPEL + P3P + EPAL



state of the art: PRIME

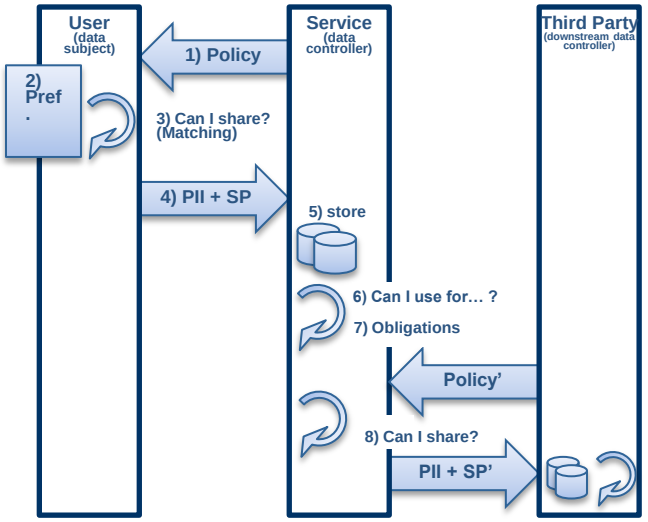


Shortcoming of state of the art

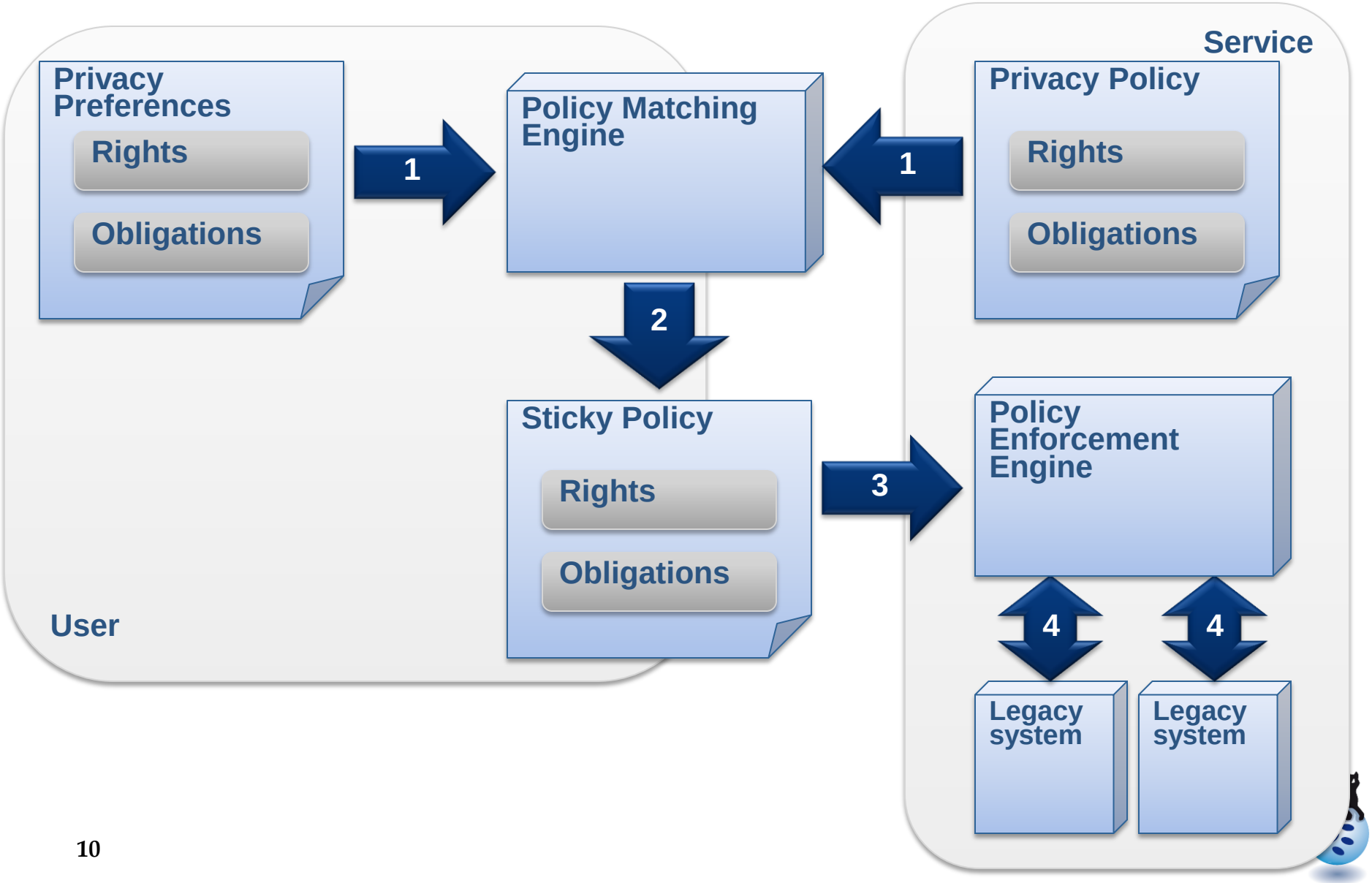


Our Approach

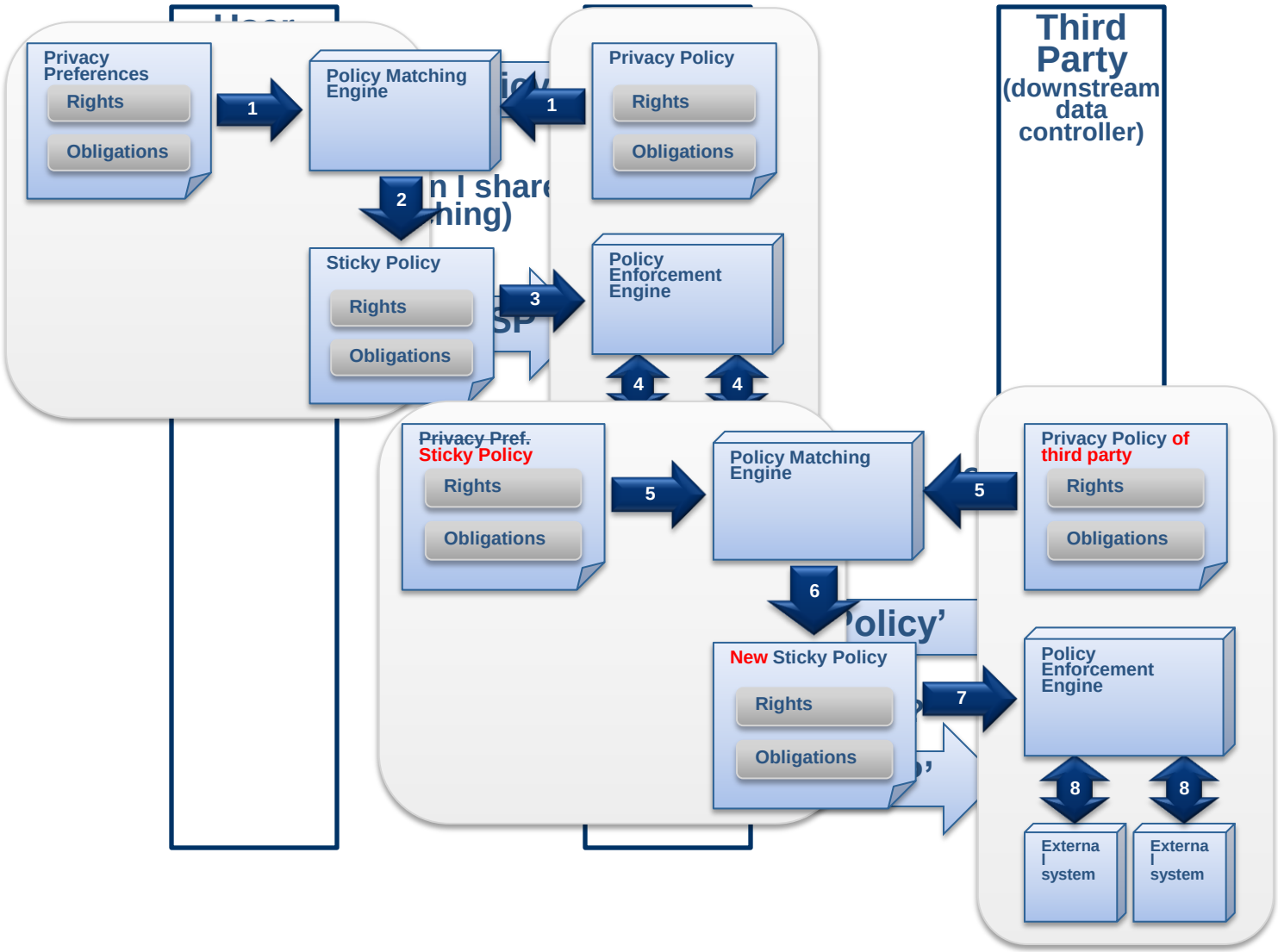
- Policy Language
  - Rights
    - Data usage (purpose, etc.)
    - Data sharing (Access Control, etc.)
  - Obligations
    - Triggers + Actions
    - Examples: Retention, Notification, Log, etc.
- Policy Matching
  - Similar language for policies, preferences, and sticky policies
- Policy Enforcement



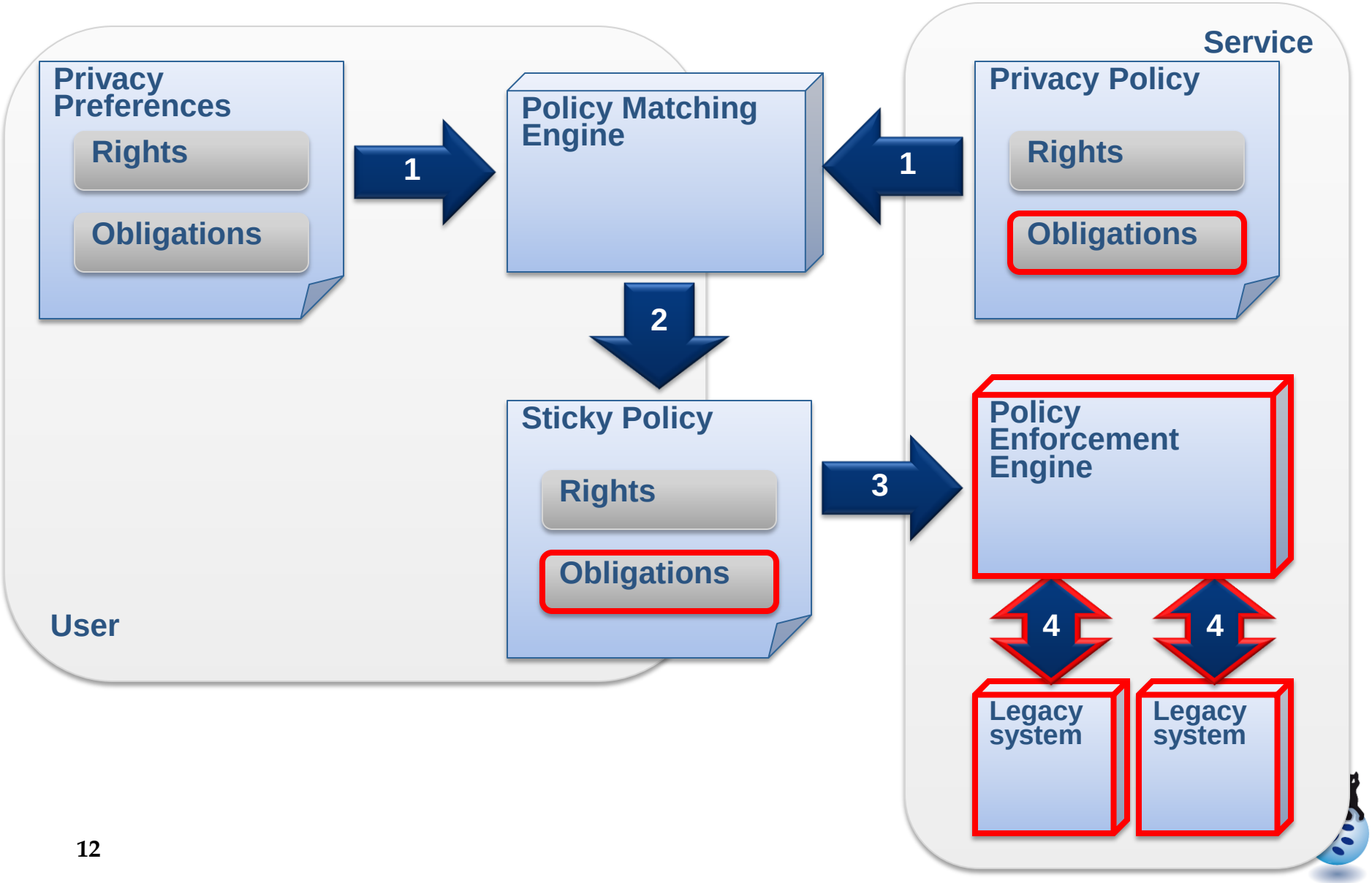
Our Solution



Applying our solution



Our Solution



Obligation

- We define obligation as:

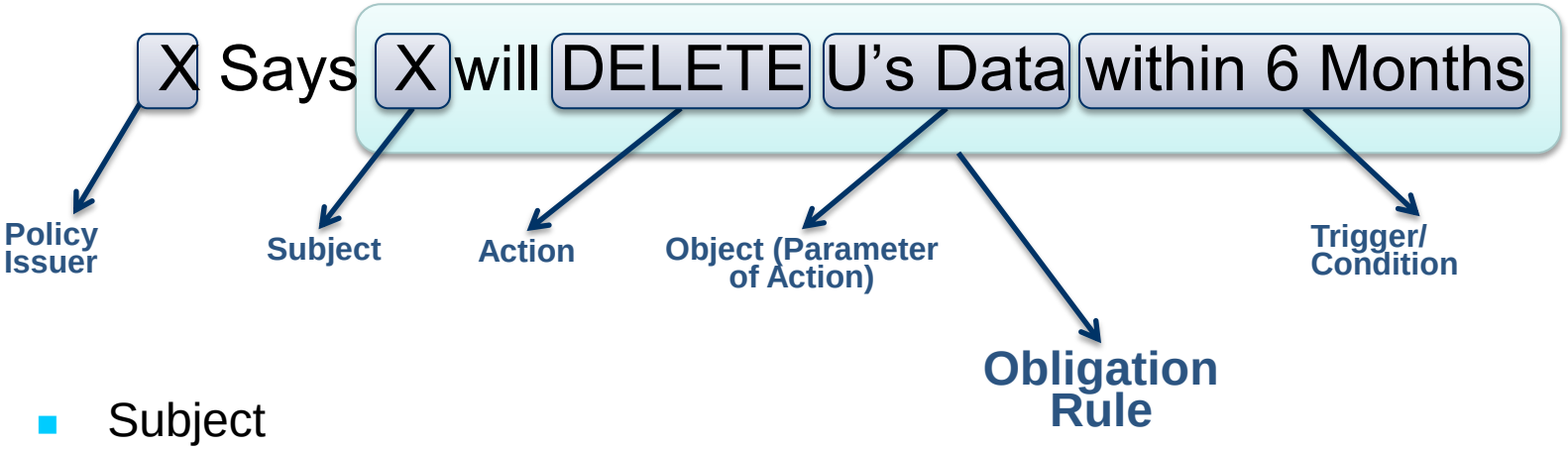
  "Promise made by a SUBJECT to be fulfilled through some ACTION under defined TIMELINES and CONDITIONS"

- Example:
  - X Says X will DELETE U's Data within 6 Months

Obligation Requirements

- Independence from Transport / Storage
  - Independence from policy language
  - Independence from data storage
  - Independence from communication protocols
- Extensibility
  - Support for common obligations
  - Support for domain specific obligations
- Abstraction
  - Support for abstraction of actions
  - Support for preventive obligations
  - Support for abstraction of triggers
- Deployments
  - Support for distributed deployment
  - Support for different trust models
- Auditability
  - Transparency of data handling
- Matching

Obligation Structure

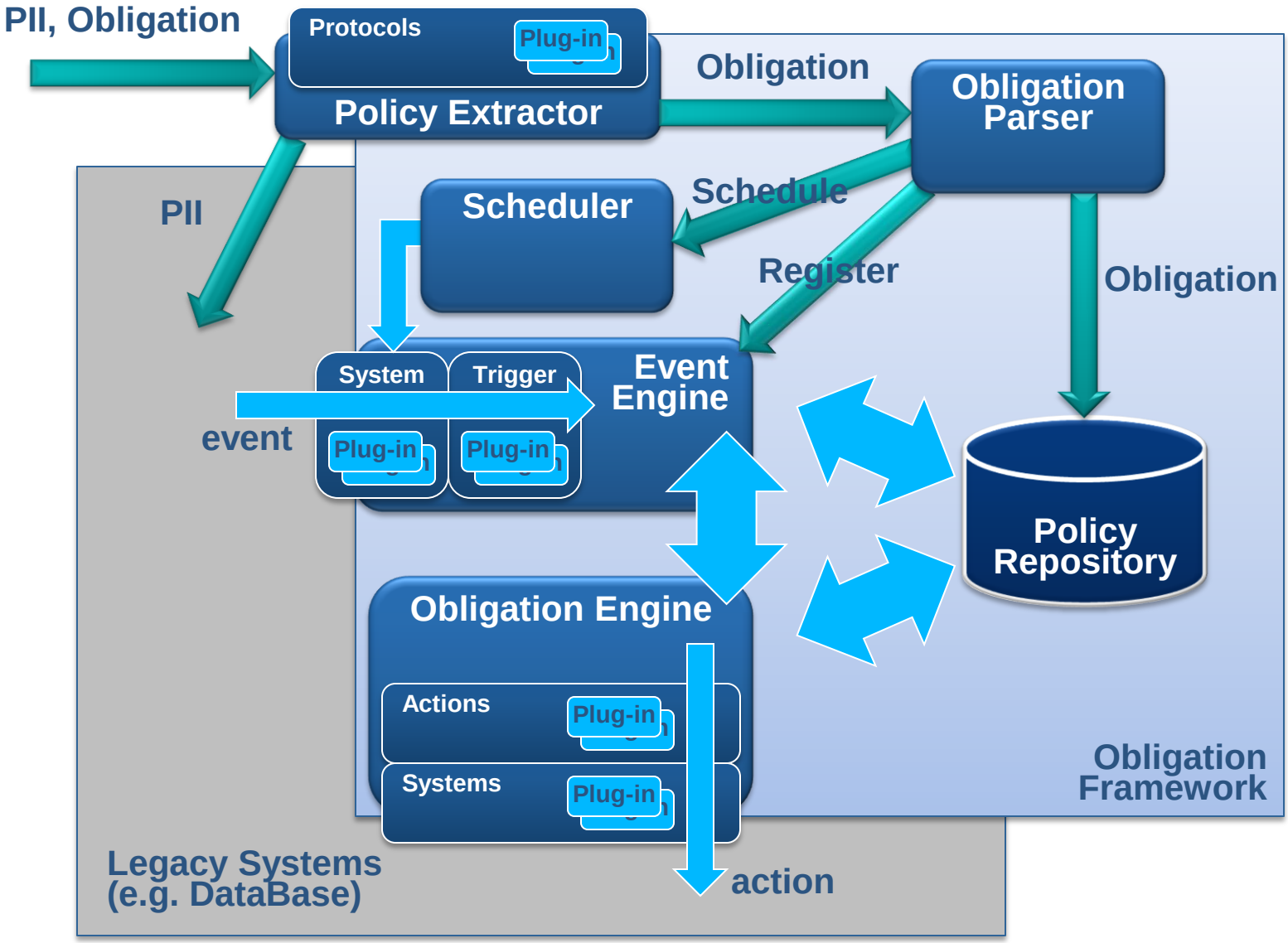


- Subject
  - The entity liable to fulfill obligation
    (i.e. the subject of the obligation not the data subject)
- Action
  - The activity (or sequence of activities) executed to fulfill obligation
- Conditions (Temporal constraints, generic conditions etc)
  - Constraints on the obligation rule
- Triggers
  - Inward event to trigger execution of obligation rule
- Outward Events
  - Outward notification events

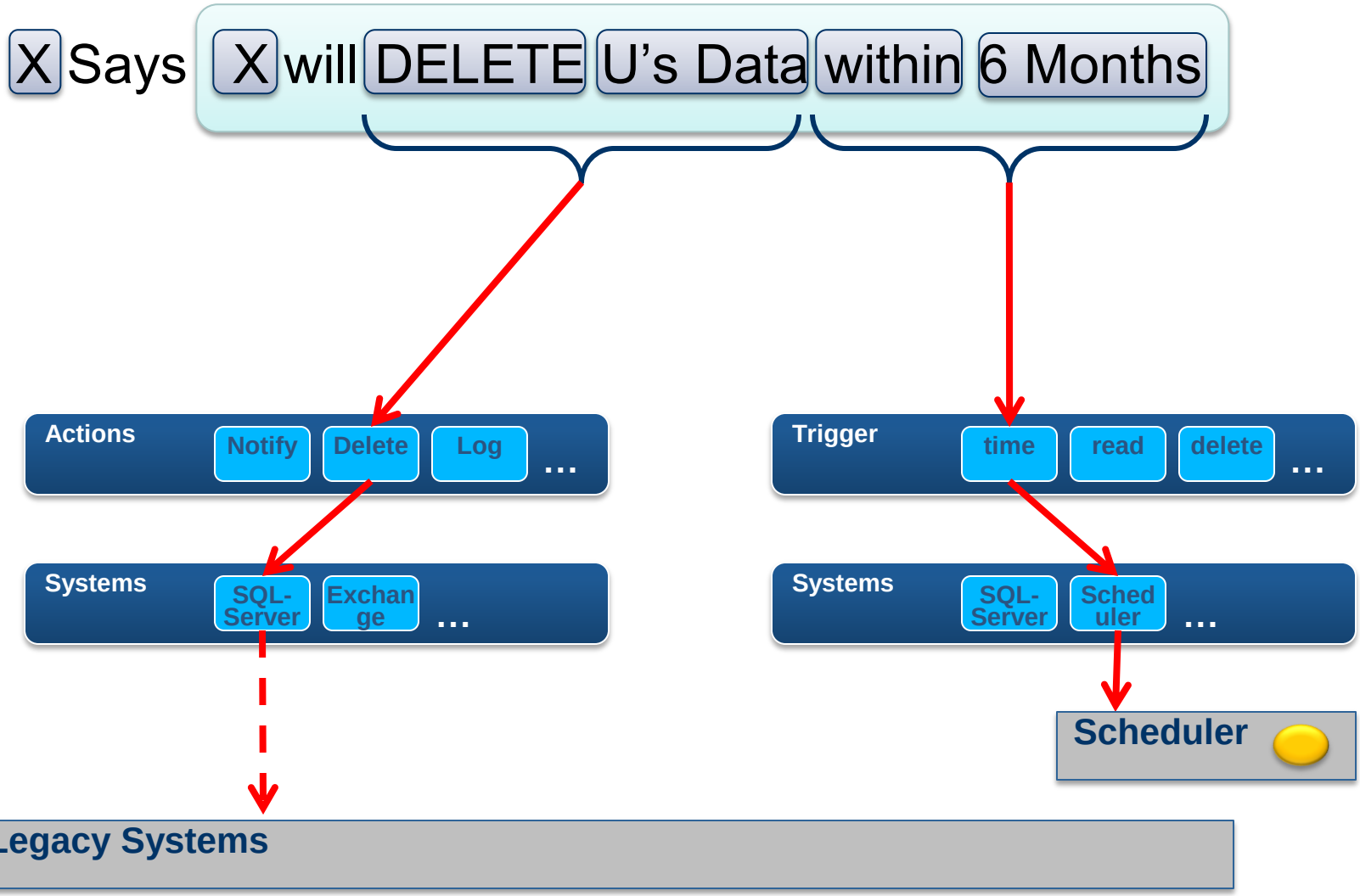
Obligation Classification

- Trigger-related
  - Conditional vs. Mandatory
  - Repeating vs. Non-Repeating
- Action-related
  - Proactive vs. Preventive
  - Observable vs. Non-observable
- Subject-related
  - Collective vs. Individual
  - Self Obligation vs. Third Party Obligations

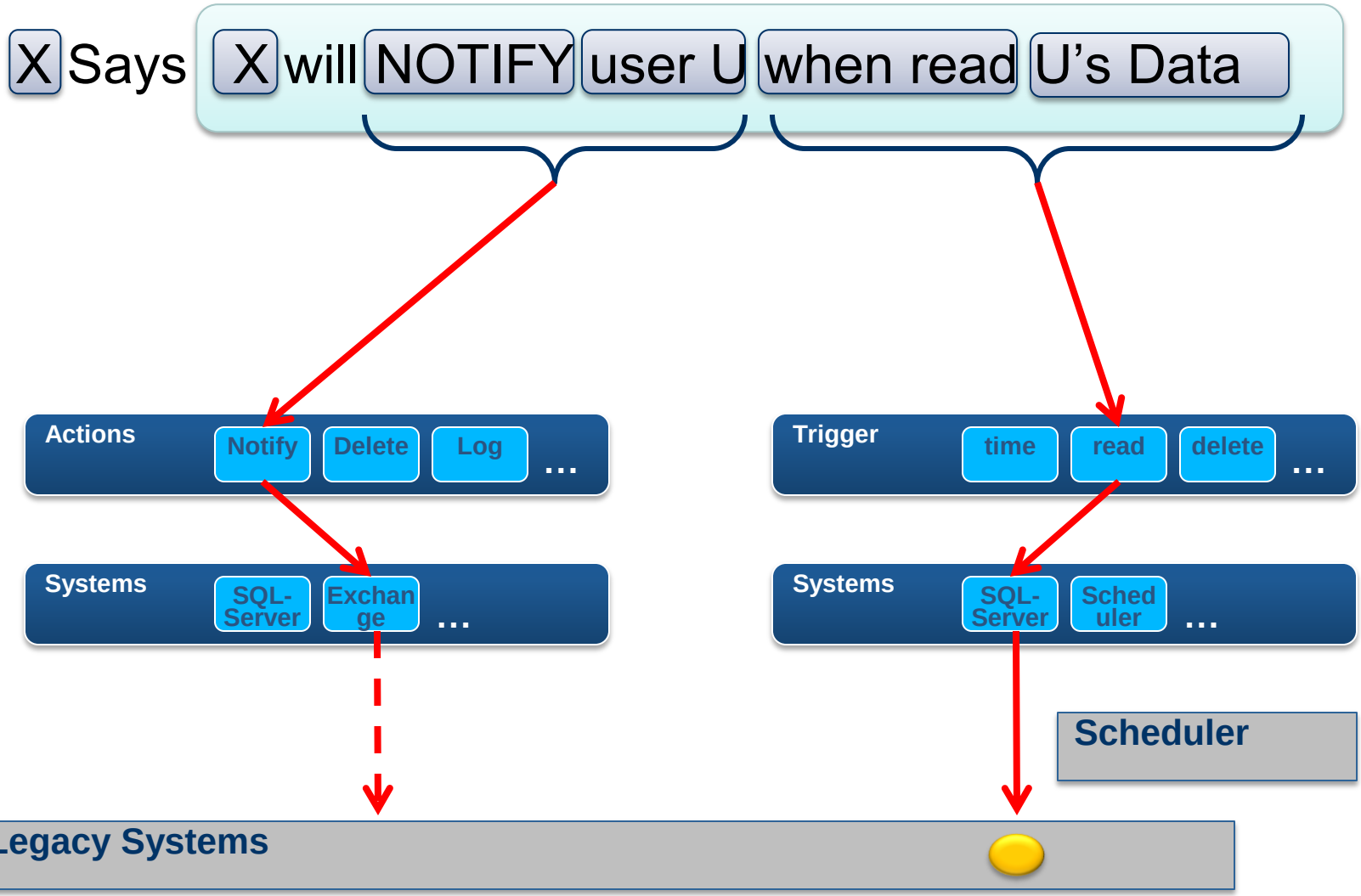
Proposed Framework Architecture



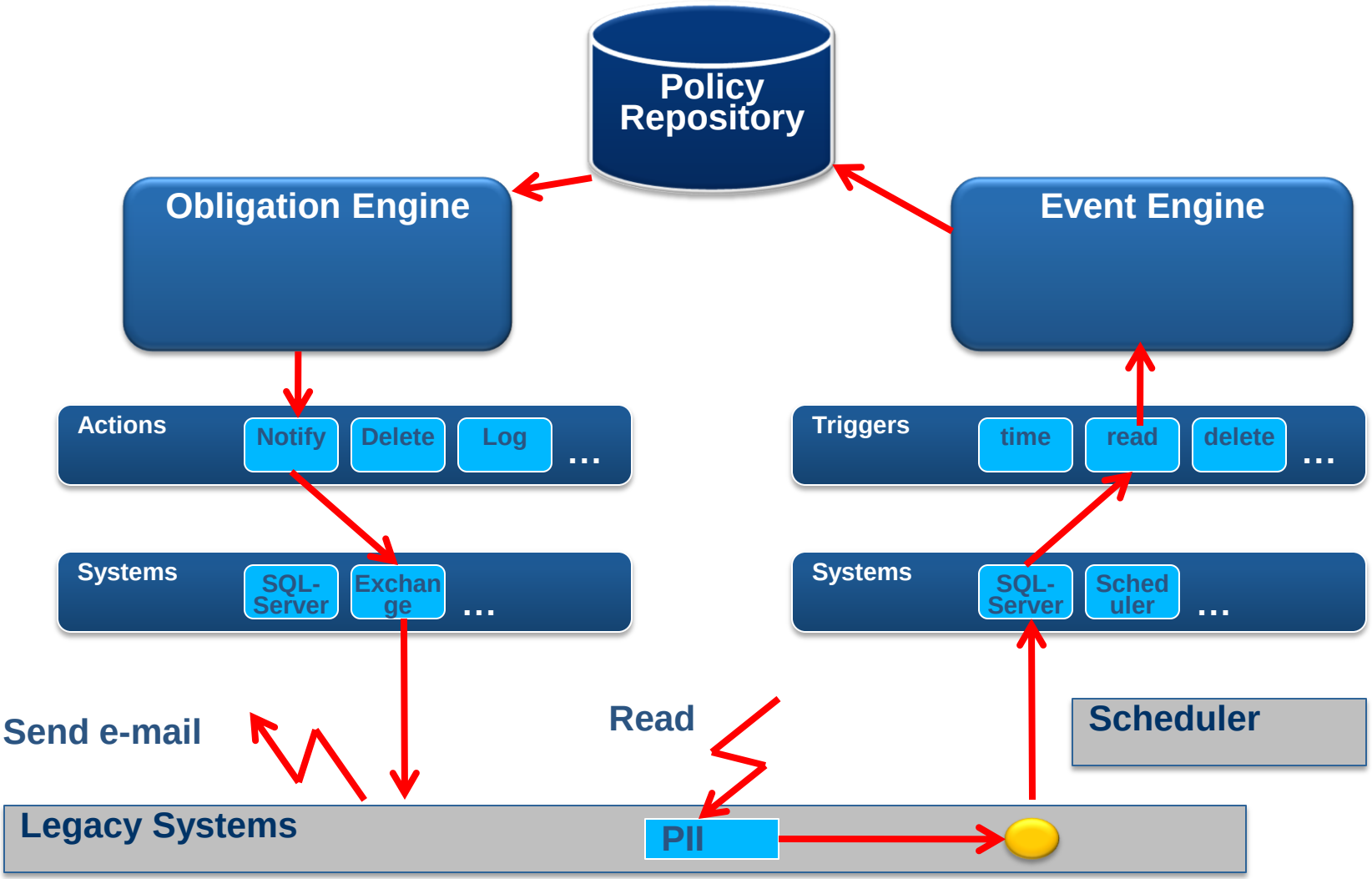
Setting plug-ins



Setting plug-ins



Plug-ins in action



Results

- A language to describe obligations
  - Definition of Triggers
  - Definition of Actions
- Implementation of an enforcement framework
- Mechanisms to extend language and enforcement with domain specific obligations

Ongoing and Future work

- Matching obligations
  - Semantics of actions and triggers
  - *Policy* is-less-permissive-than *Preference* ?
    (to appear: PrimeLife document H5.3.2)
- Integration with policy languages
  - XACML-based data handling
    (to appear: PrimeLife H5.3.2)
  - SecPAL for Privacy
    (September'09 MSR report: MSR-TR-2009-128)
- Matching behavior (traces) and policies
- Checking enforceability of policies

Questions?

Laurent Bussard

European Microsoft Innovation Center

lbussard@microsoft.com

http://research.microsoft.com/en-us/people/lbussard/