# Normative Reasoning in Distributed Systems and GDPR-based Access Control

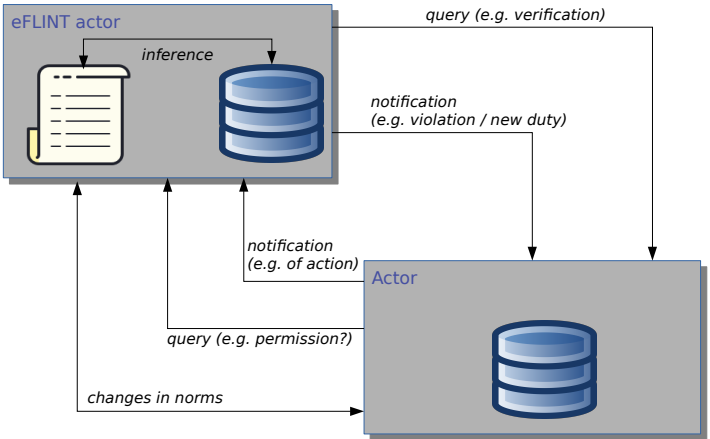
L. Thomas van Binsbergen, Tom van Engers

Informatics Institute, University of Amsterdam
ltvanbinsbergen@acm.org

January 28, 2025

# Normative Specification Language eFLINT

- Domain-specific language coupling normative/legal to computational concepts
- Based on logic programming and inference; Captures transitions between Knowledge bases
- Enables modelling and simulation; Enables integration into running systems.



# Foundational, normative & computational concepts

computational

**state**

$parent(A, B) = true$

$\ldots$

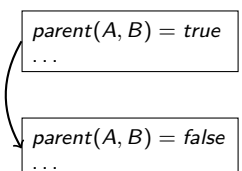
# Foundational, normative & computational concepts

computational

**state**

$parent(A, B) = true$
...

**transitions**



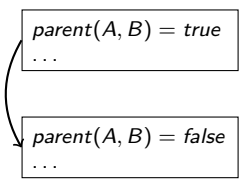
# Foundational, normative & computational concepts

computational

**state**

$parent(A, B) = true$
...

**transitions**




- Violations of state and transition

# Foundational, normative & computational concepts

deontic

**prohibitions**

**permissions**

**obligations**

computational

**state**

$parent(A, B) = true$
...

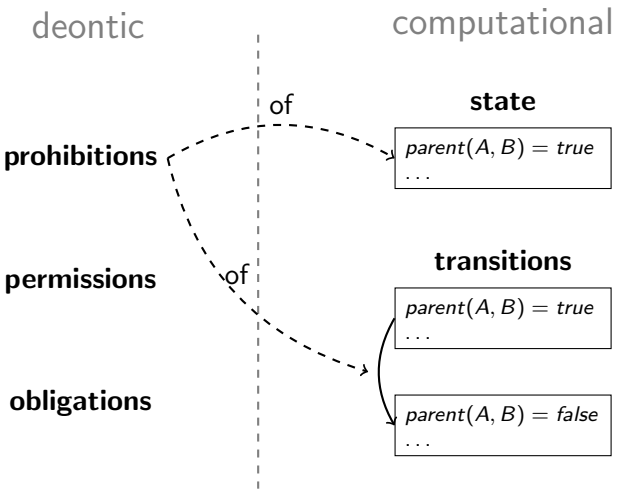
**transitions**

$parent(A, B) = true$
...

$parent(A, B) = false$
...

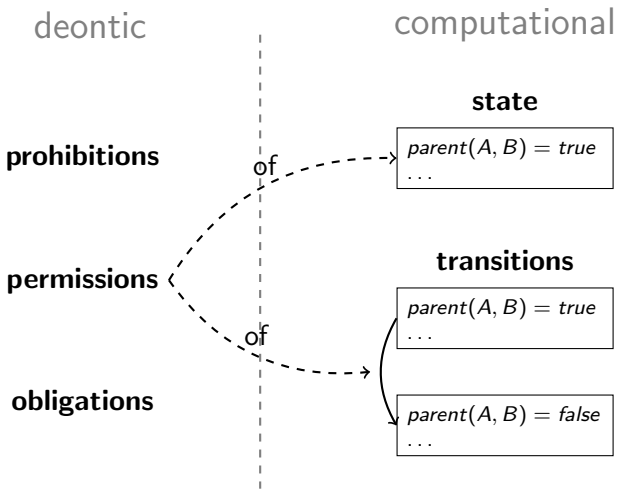
- Violations of state and transition

# Foundational, normative & computational concepts



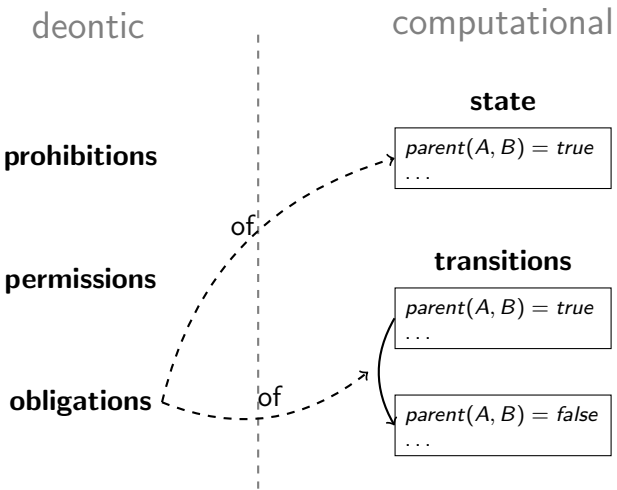

- Violations of state and transition

# Foundational, normative & computational concepts



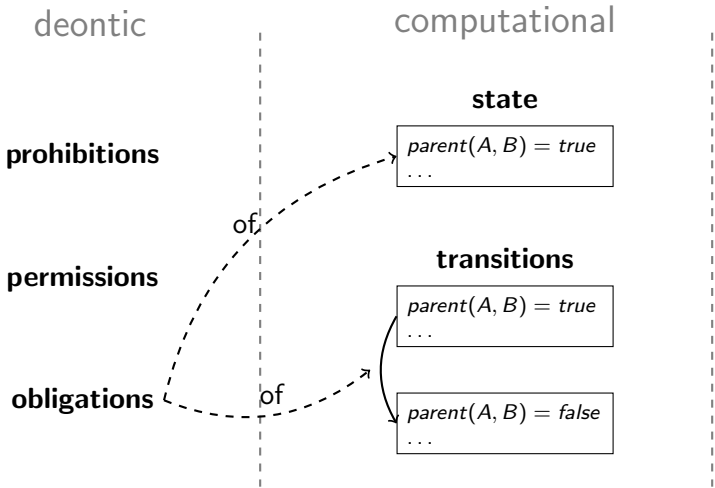

- Violations of state and transition

# Foundational, normative & computational concepts



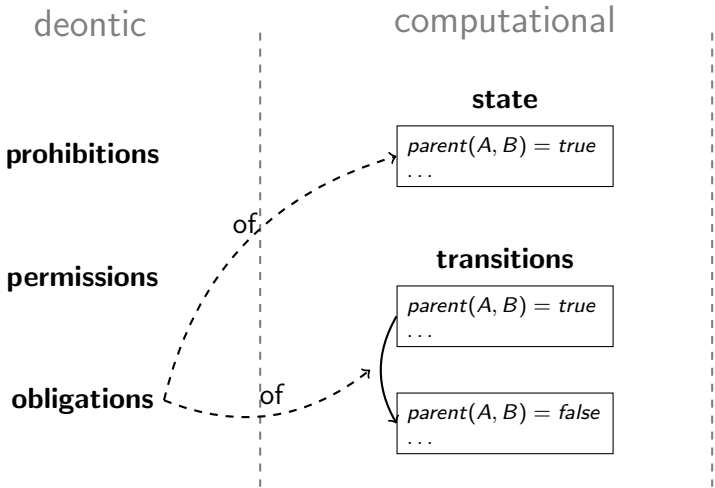

- Violations of state and transition

# Foundational, normative & computational concepts




- Violations of state and transition

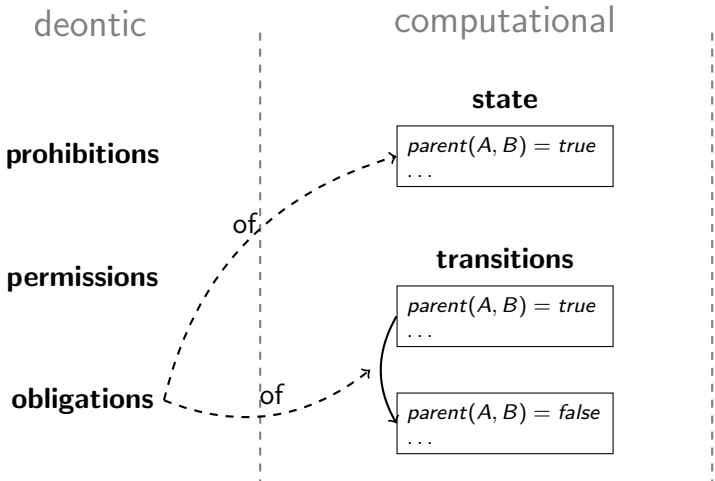
# Foundational, normative & computational concepts




- Powers have (normative) consequences

- Violations of state and transition

# Foundational, normative & computational concepts

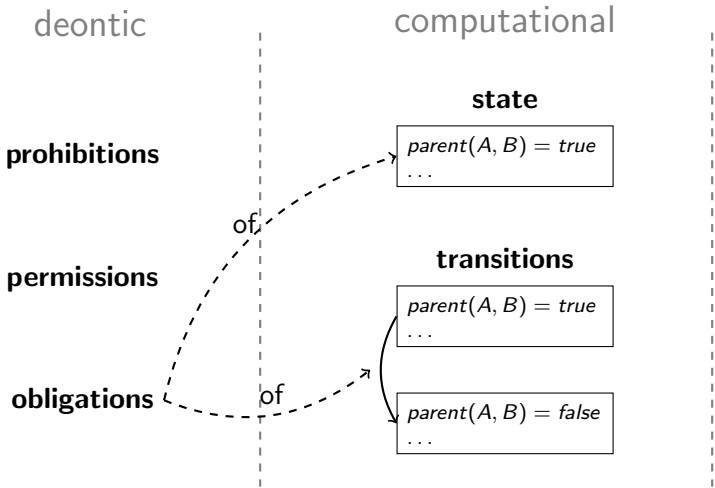



- Violations of state and transition

**powers**

- Powers have (normative) consequences
- Deontic and potestative terms are first-class
- Powers modify truth-assignments to variables

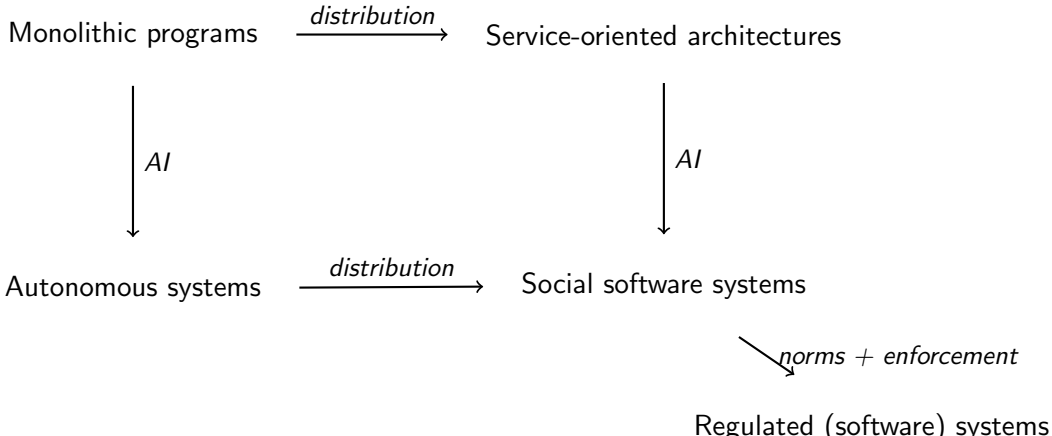
# Foundational, normative & computational concepts



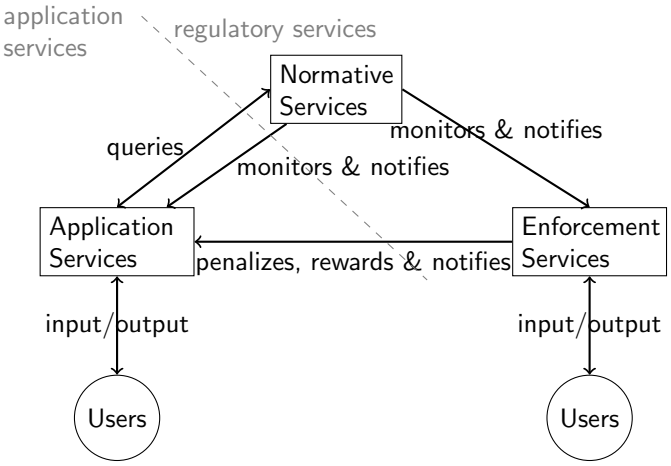

- Violations of state and transition

- Powers have (normative) consequences
- Deontic and potestative terms are first-class
- Powers modify truth-assignments to variables
- Duties capture expected actions

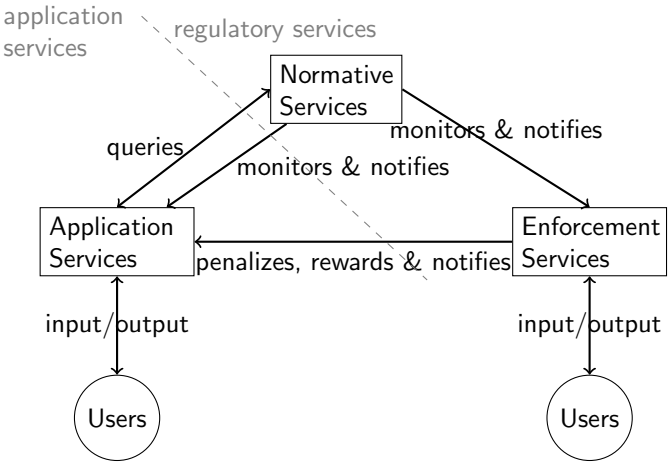
# Towards regulated systems



# Regulated system = application services + regulatory services



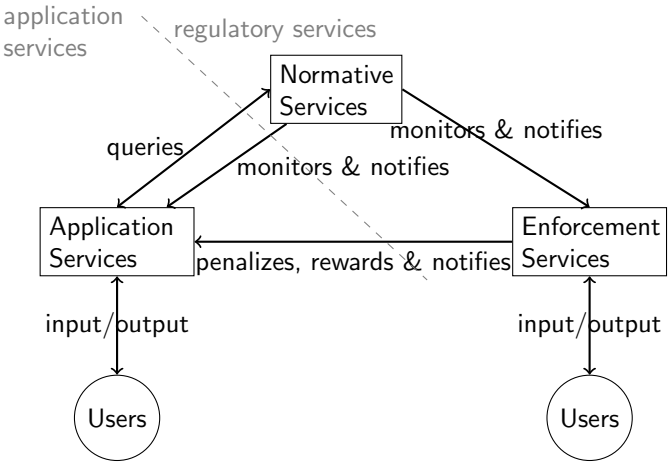
# Regulated system = application services + regulatory services




Enforcement strategies

- Static, Ex-ante: orchestration and planning

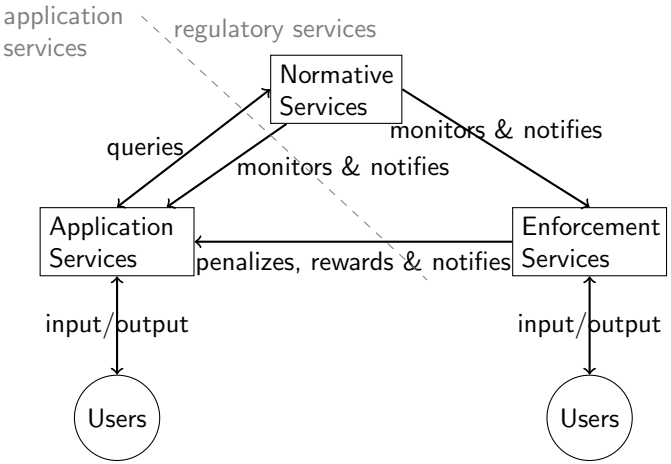
# Regulated system = application services + regulatory services




Enforcement strategies

- Static, Ex-ante: orchestration and planning
- Dynamic, Ex-ante: access control

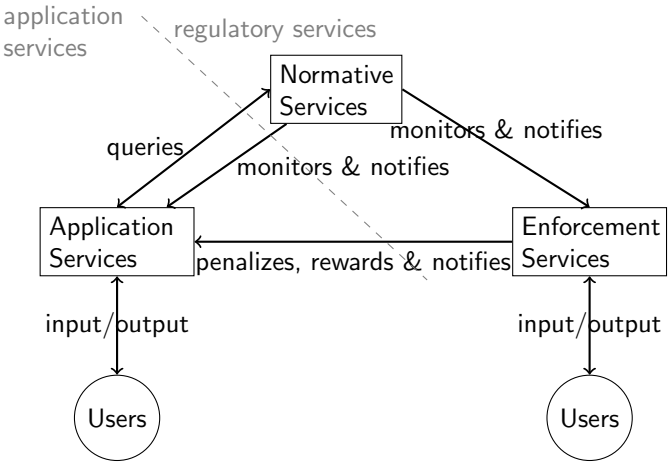
# Regulated system = application services + regulatory services




Enforcement strategies

- Static, Ex-ante: orchestration and planning
- Dynamic, Ex-ante: access control
- Dynamic, Ex-post: usage control, runtime verification and adaptation

# Regulated system = application services + regulatory services




Enforcement strategies

- Static, Ex-ante: orchestration and planning
- Dynamic, Ex-ante: access control
- Dynamic, Ex-post: usage control, runtime verification and adaptation
- Static, Ex-post: accountability and auditing

## SSPDDP – Secure and scalable, policy-driven data exchange

CWI · VU University Amsterdam · Air France KLM · ABN·AMRO · ING

## DL4LD – Data Logistics for Logistics Data

TKI Dinalog · NWO · Commit 2 Data · Thales · BiZZdesign · ciena · evofenedex · Gemeente Amsterdam · Universiteit van Amsterdam · TNO innovation for life · Airfrance KLM · Transfides · Oracle

## EPI – Enabling Personalized Interventions

NWO · St Antonius ziekenhuis · parnassia groep · UMC Utrecht · Universiteit van Amsterdam · CWI · KPMG · Prinses Máxima Centrum · Philips · white box · SURF · Commit 2 Data · santeon · Ministerie van Economische Zaken · ZonMw · EPI

## AMdEX – neutral data-exchange infrastructure

DMI·ECOSYSTEEM · Amsterdam Economic Board · SURF · amsix · Universiteit van Amsterdam · deXes · European Union, European Regional Development Fund, Investing in your future · Provincie Noord-Holland

# Normative Reasoning in Distributed Systems and GDPR-based Access Control

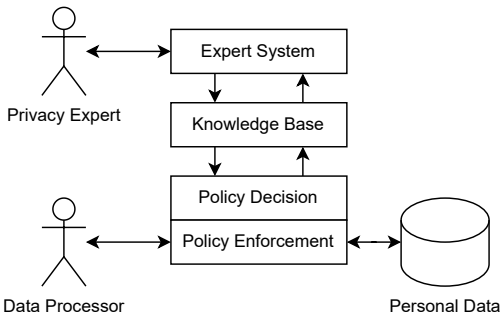
L. Thomas van Binsbergen, Tom van Engers

Informatics Institute, University of Amsterdam
ltvanbinsbergen@acm.org

January 28, 2025

# GDPR-based Access Control

**Goal**: Develop a knowledge-based, expert system for reasoning with GDPR-compliance and generating authorisations in distributed access and usage control implementations.




*Is a given processing action lawful with respect to claimed legal bases in the GDPR?*

# GDPR-based Access Control

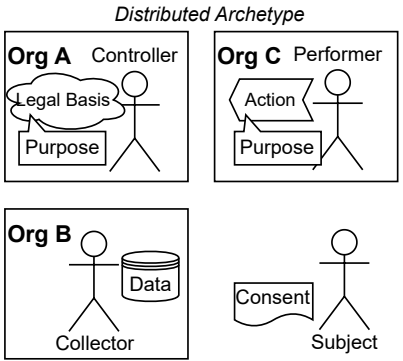
*Is a given processing action lawful with respect to claimed legal bases in the GDPR?*

# GDPR-based Access Control

*Is a given processing action lawful with respect to claimed legal bases in the GDPR?*

## Processing Action

- Identified by actor, asset, and *processing purpose*
- An action is always executed for one purpose



# GDPR-based Access Control

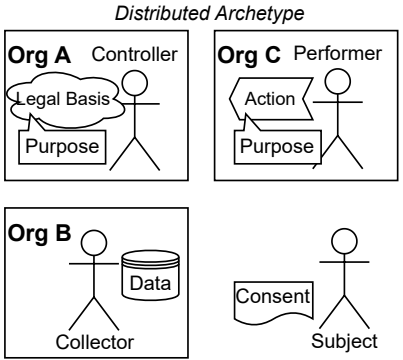
*Is a given processing action lawful with respect to claimed legal bases in the GDPR?*

## Processing Action

- Identified by actor, asset, and *processing purpose*
- An action is always executed for one purpose

## Legal basis

- Refers to Art. 6(1)(a-f), e.g., consent, legal obligation, legitimate interest, ...
- Identified by article (member) and *intended purpose*
- One or more legal bases can be *claimed*



# GDPR-based Access Control

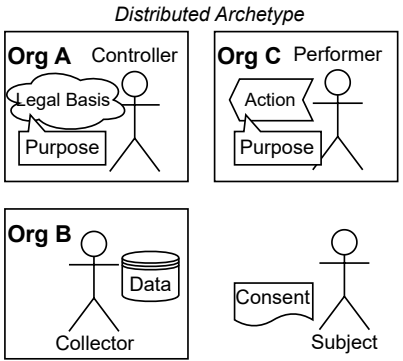
*Is a given processing action lawful with respect to claimed legal bases in the GDPR?*

## Processing Action

- Identified by actor, asset, and *processing purpose*
- An action is always executed for one purpose

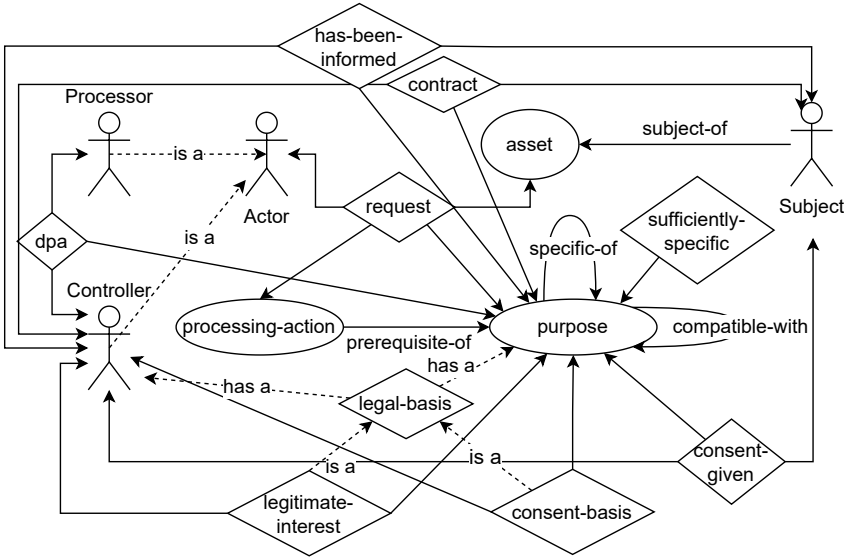
## Legal basis

- Refers to Art. 6(1)(a-f), e.g., consent, legal obligation, legitimate interest, ...
- Identified by article (member) and *intended purpose*
- One or more legal bases can be *claimed*




*Is a given collect action lawful with respect to claimed legal bases in the GDPR?*

# Ontology of lawful processing concepts



# Knowledge base (ontology instantiations) and semantics

## Case-generic rules for determining:

- Are the claimed legal bases valid?, e.g.
  - Is the intended purpose considered *sufficiently specific*?
  - Have the subjects been informed?
  - Have the subjects given consent (if legal basis is 'consent')?

$$\frac{\textit{legitimate-interest}(C, P) \quad \textit{sufficiently-specific}(P) \quad \forall_S(\textit{subject-of}(S, D) \rightarrow \textit{has-been-informed}(S, C, P))}{\textit{legal-basis}(C, P, D)}$$

# Knowledge base (ontology instantiations) and semantics

## Case-generic rules for determining:

- Are the claimed legal bases valid?, e.g.
  - Is the intended purpose considered *sufficiently specific*?
  - Have the subjects been informed?
  - Have the subjects given consent (if legal basis is 'consent')?
- Can the processing purpose and the lawful purpose be united? i.e.,
  - The processing purpose is *identical to* or *more specific than* the intended purpose
  - The processing purpose is *not incompatible* with the intended purpose

$$\frac{\textit{request}(U, A, P, D) \quad \textit{prerequisite-of}(A, P) \quad \textit{processor-for}(U, C, P') \quad \textit{specific-of}(P, P') \quad \textit{legal-basis}(C, P', D)}{\textit{lawful-request}(U, A, P, D)}$$

# Knowledge base (ontology instantiations) and semantics

## Case-generic rules for determining:

- Are the claimed legal bases valid?, e.g.
  - Is the intended purpose considered *sufficiently specific*?
  - Have the subjects been informed?
  - Have the subjects given consent (if legal basis is 'consent')?
- Can the processing purpose and the lawful purpose be united? i.e.,
  - The processing purpose is *identical to* or *more specific than* the intended purpose
  - The processing purpose is *not incompatible* with the intended purpose

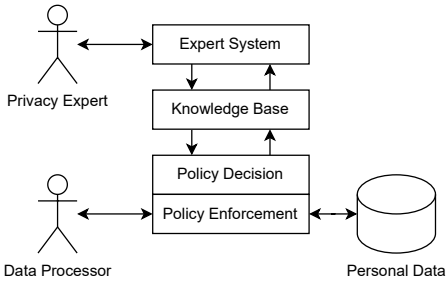
$$\frac{\textit{request}(U, A, P, D) \quad \textit{prerequisite-of}(A, P) \quad \textit{sufficiently-specific}(P) \quad \textit{processor-for}(U, C, P') \quad \textit{compatible-with}(P, P') \quad \textit{legal-basis}(C, P', D) \quad \forall_S(\textit{subject-of}(S, D) \rightarrow \textit{has-been-informed}(S, C, P))}{\textit{lawful-request}(U, A, P, D)}$$

# Knowledge base (ontology instantiations) and semantics

## Case specific statements

Expert drives the inference process by *claiming*:

- One or more legal bases
- Whether the intended purposes are sufficiently specific
- Whether (all) data subjects have been informed
- ... have given consent...
- etc.

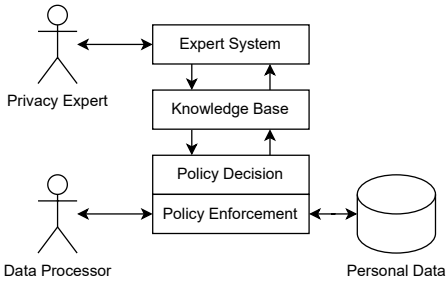


# Knowledge base (ontology instantiations) and semantics

## Case specific statements

Expert drives the inference process by *claiming*:

- One or more legal bases
- Whether the intended purposes are sufficiently specific
- Whether (all) data subjects have been informed
- ... have given consent...
- etc.

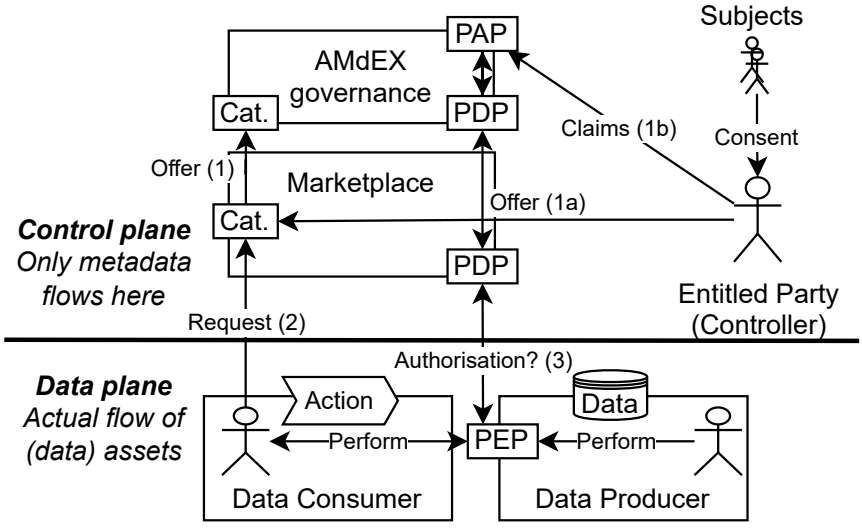



## General approach

1. Encode case-generic rules in eFLINT and apply to all processing requests
2. Convert input by domain-expert into case-specific eFLINT statements
3. Assemble policy per request, make decision, and record inputs and outputs

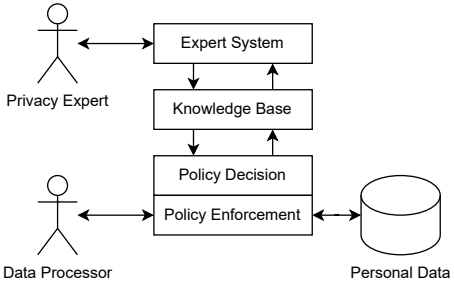
# Operationalisation within AMdEX-DMI

DMI·ECOSYSTEEM



# GDPR-based Access Control

**Goal**: Develop a knowledge-based, expert system for reasoning with GDPR-compliance and generating authorisations in distributed access and usage control implementations.




**Contributions**:

- Raising the level of *abstraction* of policy specification to the level of the *domain-expert*.
  *Before*: System administrator sets (low-level) access policies
  *After*: Privacy expert submits claims regarding purposes and legal bases
- Authorisations are generated only when processing of legal data is lawful (according to the GDPR) in a *certifiable* and *accountable* manner
- Case-generic specification is *adaptable*, *extensible*, and *transparent*

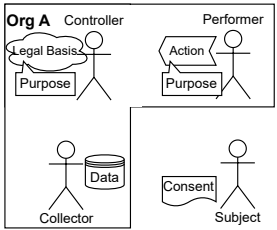
# Normative Reasoning in Distributed Systems and GDPR-based Access Control

L. Thomas van Binsbergen, Tom van Engers

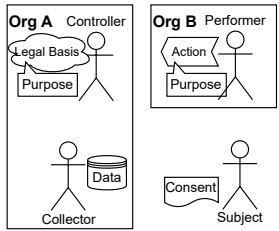
Informatics Institute, University of Amsterdam
ltvanbinsbergen@acm.org

January 28, 2025

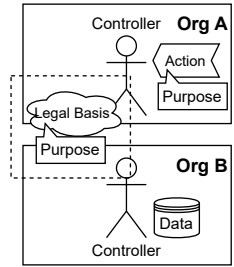
*No Delegation Archetype*



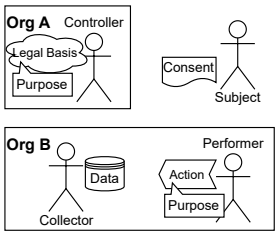

*Delegated Action Archetype*



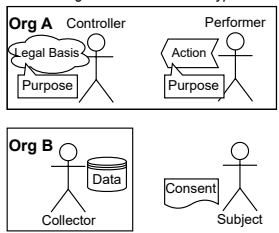

*Joint Controllers Archetype*



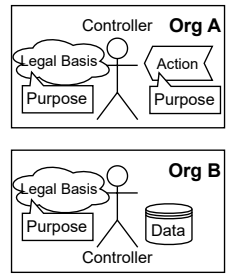

*Delegated Processing Archetype*



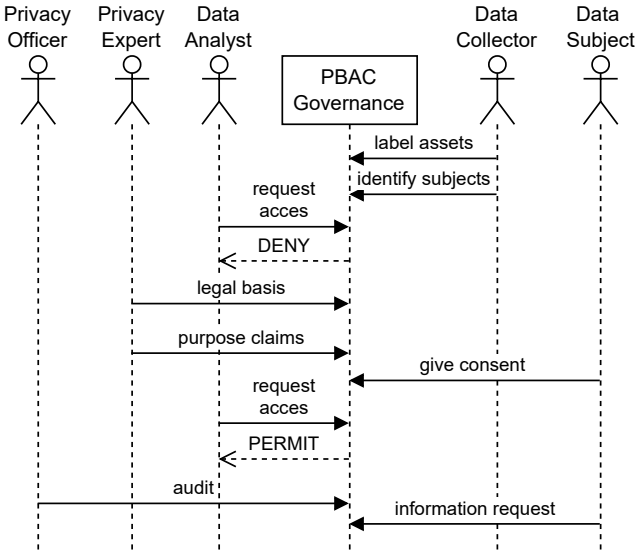

*Delegated Collection Archetype*



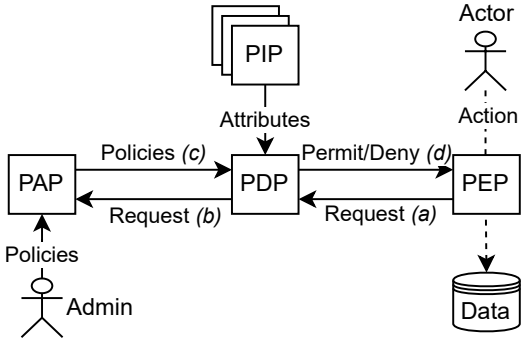

*Independent Controllers Archetype*



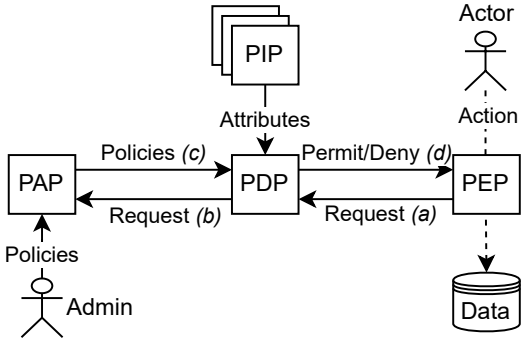
# System interactions



# Access Control (AC) revisited

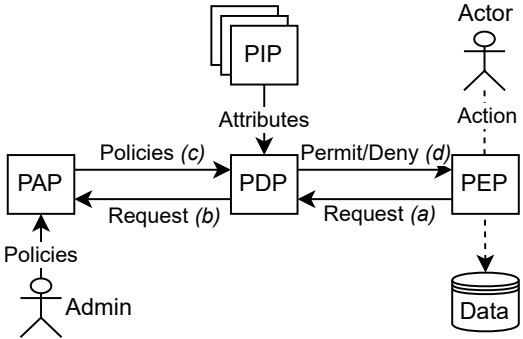


# Access Control (AC) revisited



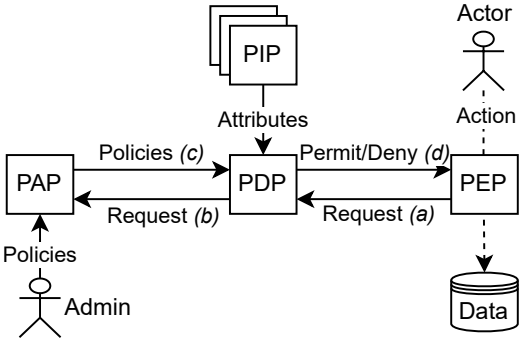

- Request consists of *Actor*, *Action*, *Asset*

# Access Control (AC) revisited



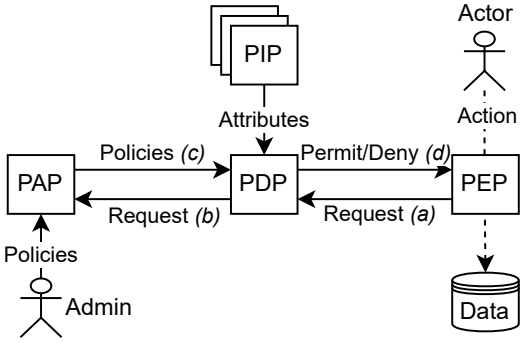

- Request consists of *Actor*, *Action*, *Asset*
- Role-Based AC: $Role(Actor) \in RolePermitted(Action, Asset)$

# Access Control (AC) revisited




- Request consists of *Actor*, *Action*, *Asset*
- Role-Based AC: $Role(Actor) \in RolePermitted(Action, Asset)$
- Purpose-Based AC: $Purpose(Role(Actor), Action) \in PurposePermitted(Asset)$

# Access Control (AC) revisited




- Request consists of *Actor*, *Action*, *Asset*
- Role-Based AC: $Role(Actor) \in RolePermitted(Action, Asset)$
- Purpose-Based AC: $Purpose(Role(Actor), Action) \in PurposePermitted(Asset)$
- GDPR-Based AC: $Purpose(Actor, Action) \lesssim Purpose(LegalBasis(...))$

# Overview

1. Legal analysis
2. Ontology
3. Semantic specification (inference rules)
4. Semantic implementation (eFLINT)
5. Policy specification (purpose details, consent)
6. System integration (XACML, AMdEX)
7. Reflections

# Legal Analysis (1)

## Definition

A controller can claim a *legal basis* for processing for a specific intended purpose if the processing is lawful according to the GDPR (Art. 6), in which case one of the following applies:

- the data subject has given consent (Art. 6(1)(a)), or
- the processing is necessary for:
  - the performance of a contract with the data, or subject (Art. 6(1)(b)), or
  - compliance with a legal obligation (Art. 6(1)(c)), or
  - the vital interest of subject or natural person (Art. 6(1)(d)), or
  - public interest or vested authority (Art. 6(1)(e)), or
  - the controller has a legitimate interest (Art. 6(1)(f)).

And all data subjects involved must be informed about the legal basis and purpose, prior to the processing.

# Legal Analysis (2)

## Definition

A *purpose-based processing request* connects an actor (a processor or controller) to a processing action, performed on an asset for a prescribed processing purpose. The request is considered lawful if:

- the action is prerequisite of the processing purpose, and
- the processing purpose is *sufficiently specific*, and
- the processing purpose:
  - coincides with a purpose that has a lawful legal basis, or
  - is more specific than a purpose that has a lawful legal basis, or
  - is not incompatible with a purpose that has a lawful legal basis.

## Definition

A purpose is a *specific-of* of another purpose if it concretises a more abstract purpose without including elements not contained in the more abstract purpose.

# Examples of semantic specification rule

$$\frac{\textit{legitimate-interest}(C, P) \quad \textit{sufficiently-specific}(P) \quad \forall_S(\textit{subject-of}(S, D) \rightarrow \textit{has-been-informed}(S, C, P))}{\textit{legal-basis}(C, P, D)} \tag{1}$$

# Examples of semantic specification rule

$$\frac{\textit{legitimate-interest}(C, P) \quad \textit{sufficiently-specific}(P) \quad \forall_S(\textit{subject-of}(S, D) \rightarrow \textit{has-been-informed}(S, C, P))}{\textit{legal-basis}(C, P, D)} \tag{1}$$

$$\frac{\textit{request}(U, A, P, D) \quad \textit{prerequisite-of}(A, P) \quad \textit{specific-of}(P, P') \quad \textit{legal-basis}(C, P', D) \quad \textit{processor-for}(U, C, P')}{\textit{lawful-request}(U, A, P, D)} \tag{2}$$

# Examples of semantic specification rule

$$\frac{\textit{legitimate-interest}(C, P) \quad \textit{sufficiently-specific}(P) \quad \forall_S(\textit{subject-of}(S, D) \rightarrow \textit{has-been-informed}(S, C, P))}{\textit{legal-basis}(C, P, D)} \tag{1}$$

$$\frac{\textit{request}(U, A, P, D) \quad \textit{prerequisite-of}(A, P) \quad \textit{specific-of}(P, P') \quad \textit{legal-basis}(C, P', D) \quad \textit{processor-for}(U, C, P')}{\textit{lawful-request}(U, A, P, D)} \tag{2}$$

$$\frac{\begin{array}{c}\textit{request}(U, A, P, D) \quad \textit{prerequisite-of}(A, P) \quad \textit{sufficiently-specific}(P) \\ \textit{compatible-with}(P, P') \quad \textit{legal-basis}(C, P', D) \quad \textit{processor-for}(U, C, P') \\ \forall_S(\textit{subject-of}(S, D) \rightarrow \textit{has-been-informed}(S, C, P))\end{array}}{\textit{lawful-request}(U, A, P, D)} \tag{3}$$

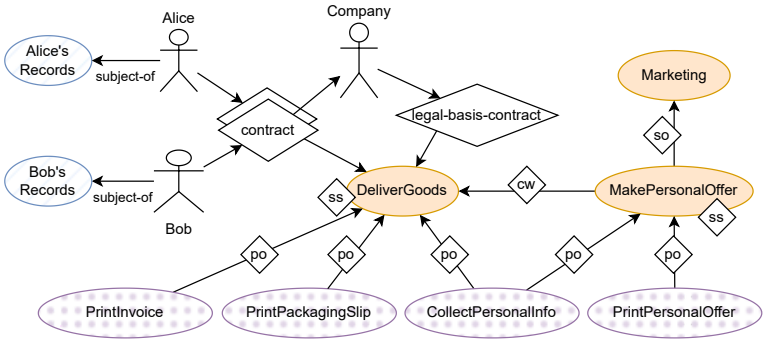
# Example eFLINT fragments implementing semantics

```
Fact lawful-request
  Identified by actor * processing-action * purpose * asset
  Conditioned by request() // only considers created requests
```

```
Extend Fact lawful-request
  Holds when prerequisite-of(processing-action, purpose)
         && specific-of(purpose, purpose')
         && legal-basis(controller, purpose', asset)
         && processor-for(actor, controller, purpose')
```

```
Extend Fact lawful-request
  Holds when prerequisite-of(processing-action, purpose)
         && sufficiently-specific(purpose)
         && compatible-with(purpose, purpose')
         && legal-basis(controller, purpose', asset)
         && processor-for(actor, controller, purpose')
         && has-been-informed(subject, controller, purpose)
```

# Example purpose graph and scenarios




(a.) The processing actions that are prerequisites of delivering goods are lawful, for each individual subject, if a contract exists with that subject and for that purpose.

(b.) The further processing of the data to print and include a personal offer may be lawful depending on whether this purpose is considered to be incompatible with the delivery.

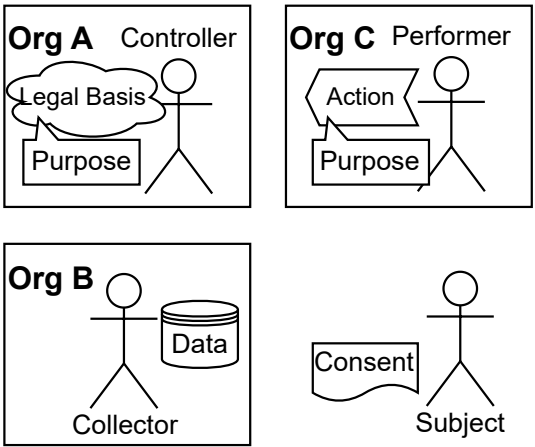
(c.) If, instead, the company asks for consent as a legal basis, the consent needs to state 'making a personal offer' and not 'marketing' as the latter is not deemed to be sufficiently specific.

# Overview

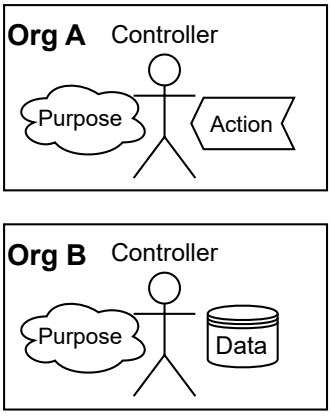
1. Legal analysis
2. Ontology
3. Semantic specification (inference rules)
4. Semantic implementation (eFLINT)
5. Policy specification (purpose details, consent)
6. System integration (XACML, AMdEX)
7. Reflections

# Archetypical patterns of processing activities

*Distributed Archetype*




*Independent Controllers Archetype*



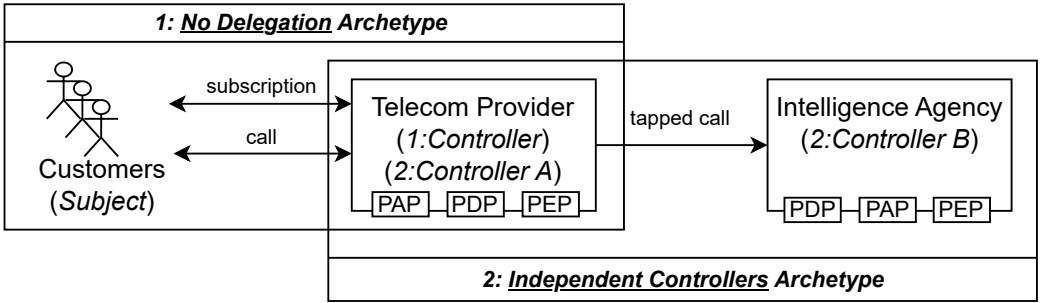
# Policy administration capabilities and roles

| **Capability** | **Policy (purpose-graph) contributions** | **Assigned to** |
|---|---|---|
| Control | legal-basis, dpa, has-been-informed, contract(s) (if applicable) | Controller, Authority |
| Qualify | prerequisite-of, compatible-with, specific-of, sufficiently-specific | Controller, Authority |
| Collect | asset(s), subject-of | Collector |
| Perform | request | Performer Collector |
| Consent | consent-given (including withdrawal of consent) | Subject |

# Policy administration capabilities and roles

| Processing Archetype | Organisation | Policy Administration Roles |
|---|---|---|
| No Delegation | Controller | Controller, Collector, Performer |
| Delegated Action | Controller | Controller, Collector |
| | Performer | Performer |
| Delegated Processing | Controller | Controller |
| | Performer | Collector, Performer |
| Delegated Collection | Controller | Controller, Performer |
| | Collector | Collector |
| Distributed | Controller | Controller |
| | Collector | Collector |
| | Performer | Performer |
| Independent Controllers | Controller A | Controller, Collector |
| | Controller B | Controller, Performer |

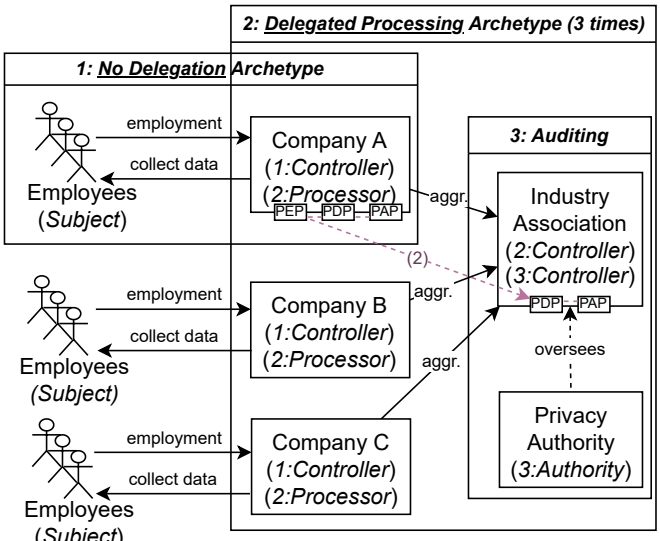
# Example case: KPN and wiretapping




Scenario 2 checks:

- Upon sending: KPN's PEP confers with KPN PDP for *collecting*
- Upon receiving: Agency's PEP confers with Agency PDP for *performing*

# Example case: industry benchmarking



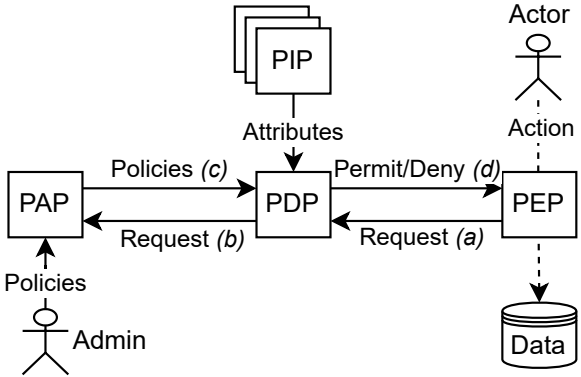

Scenario 1 checks:

- Company's PEP confers with local PDP for both collecting and performing (e.g., 'pay salary')

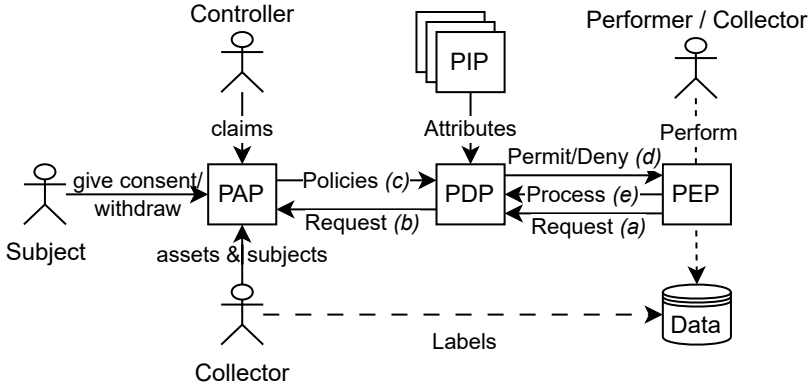
Scenario 2 checks:

- Company's PEP confers with Association's PDP for both collecting and performing (e.g., 'total salary, employee count')

# Simplified XACML architecture (technical roles)

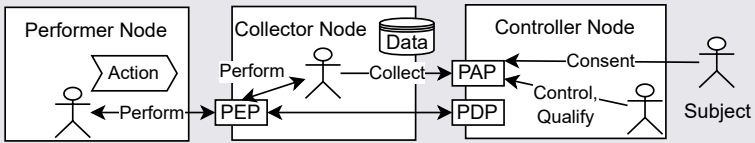


# Simplified XACML architecture with PBAC policy administration



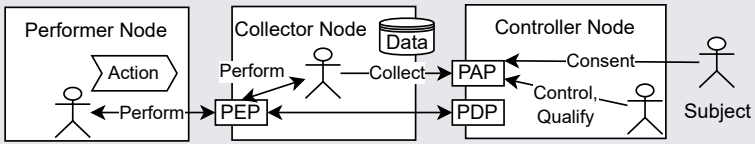
# Mapping roles unto data exchange systems

## Self-governed peer-to-peer system (distribution archetype)

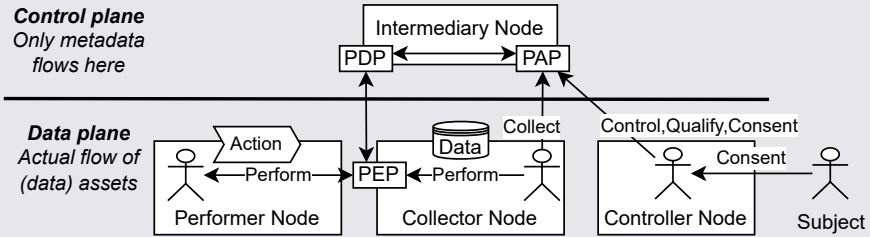


# Mapping roles unto data exchange systems

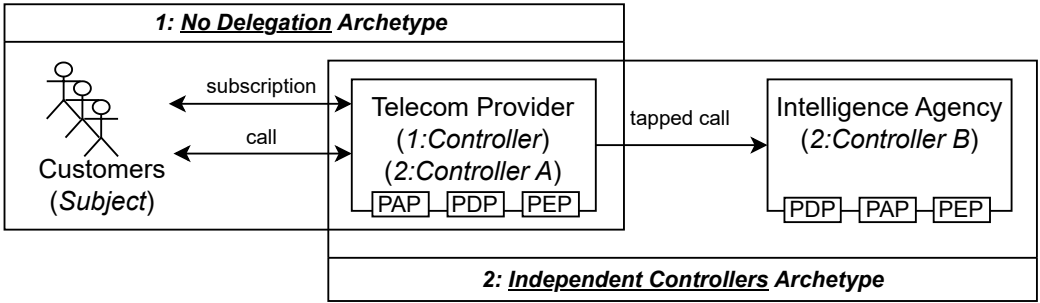
## Self-governed peer-to-peer system (distribution archetype)




## Peer-to-peer system governed by intermediary (AMdEX)



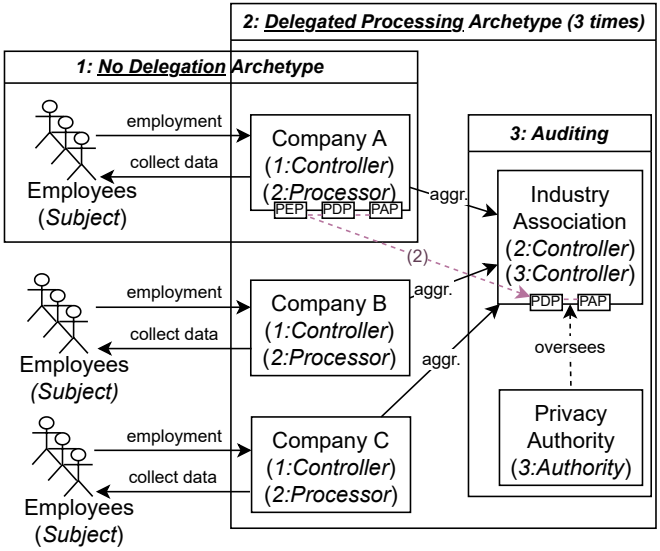
# Example case: KPN and wiretapping




Scenario 2 checks:

- Upon sending: KPN's PEP confers with KPN PDP for *collecting*
- Upon receiving: Agency's PEP confers with Agency PDP for *performing*

# Example case: industry benchmarking




Scenario 1 checks:

- Company's PEP confers with local PDP for both collecting and performing (e.g., 'pay salary')

Scenario 2 checks:

- Company's PEP confers with Association's PDP for both collecting and performing (e.g., 'total salary, employee count')

# Reflections on accountability and explainability

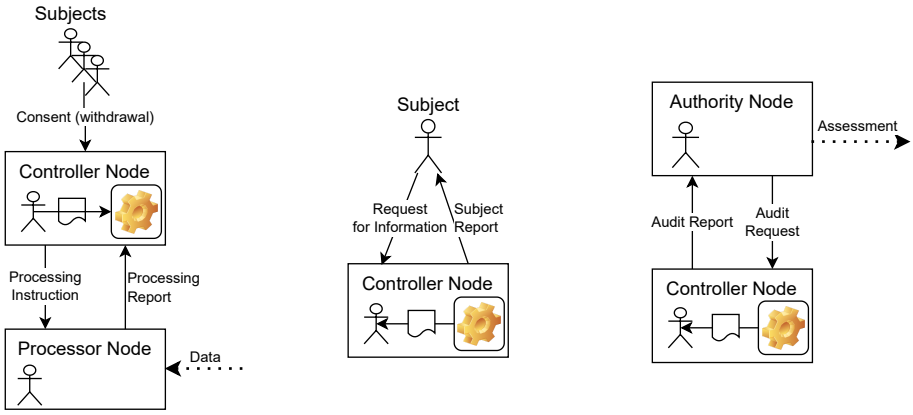



Figure: Different reasoning scenarios with different stakeholders.

# Model Evolution

## Reflected in current solution

- Original and further processing purposes need to be *sufficiently specific*
- Requirement to *inform subjects* of legal bases, prior to processing
  $\hookrightarrow$ which in some cases can be inferred
- Requirement to specify *processing purpose*

## Necessary updates to be made

- Cases with two or more *independent controllers* (Control vs Perform capability)
- Cases with *joint controllership*

# Future Work – AMdEX integration

We aim to show feasibility within the current AMdEX-DMI project.