Policy-driven distributed data exchange processes

EPI Closing Event

L. Thomas van Binsbergen

Informatics Institute, University of Amsterdam

March 7, 2024

Regulated data exchange:

Data exchange systems governed by regulations, agreements and policies

as an instance of

Regulated systems:

software systems with embedded regulatory services derived from norm specifications that monitor and/or enforce compliance

NWO-funded: SSPDDP – Secure and scalable, policy-driven data exchange

Regulated data exchange:

Data exchange systems governed by regulations, agreements and policies

as an instance of

Regulated systems:

software systems with embedded regulatory services derived from norm specifications that monitor and/or enforce compliance

NWO-funded: DL4LD – Data Logistics for Logistics Data

Data Logistics for Logistics Data
TKI DINALOG Dutch Institute for Advanced Logistics
NWO Netherlands Organisation for Scientific Research
Commit 2 Data connecting business and science
THALES BiZZdesign ciena evofenedex Gemeente Amsterdam
AIRFRANCE KLM TRANSFIDES ORACLE
Universiteit van Amsterdam
TNO innovation for life

Regulated data exchange:

Data exchange systems governed by regulations, agreements and policies

as an instance of

Regulated systems:

software systems with embedded regulatory services derived from norm specifications that monitor and/or enforce compliance

NWO-funded: EPI – Enabling Personalized Interventions

Regulated data exchange:

Data exchange systems governed by regulations, agreements and policies

as an instance of

Regulated systems:

software systems with embedded regulatory services derived from norm specifications that monitor and/or enforce compliance

EFRO-funded: AMDEX Fieldlab – neutral data-exchange infrastructure

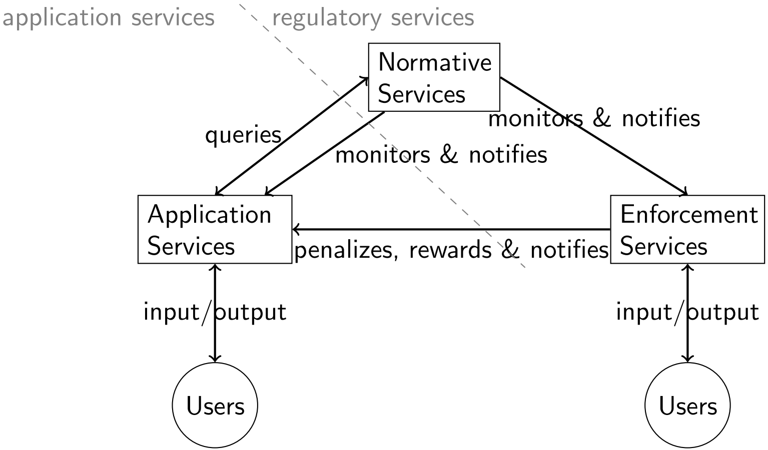
Amsterdam Economic Board
SURF
amsix
Universiteit van Amsterdam
deXes
European Union
European Regional Development Fund
Investing in your future
Provincie Noord-Holland

Section 1

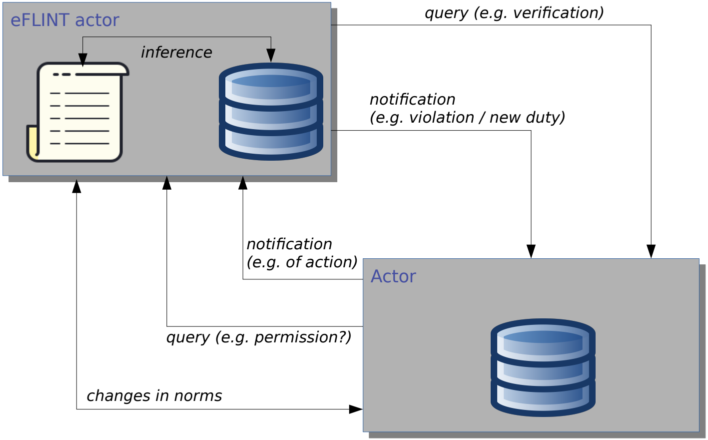
Policy-driven data exchange @ UvA

Joint with: Tom van Engers

Our approach to regulated systems



Policy reasoning with eFLINT DSL



L. Thomas van Binsbergen and Lu-Chi Liu and Robert van Doesburg and Tom M. van Engers. "eFLINT: a domain-specific language for executable norm specifications". In: *Proceedings of the 19th ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences*. ACM, 2020, pp. 124–136. DOI:

Policy Administration and Enforcement

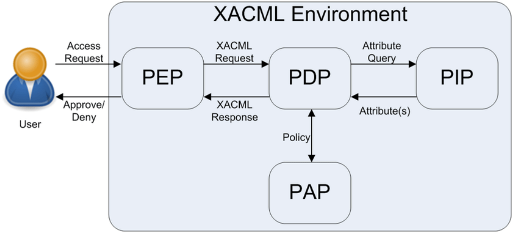


Figure: Simplified XACML architecture – M.S. Ferdous. "User-controlled identity management systems using mobile device". PhD thesis.

Requirements on Administration

- Links between legal text and policy
- **Layered policies**
- Versioning
- **Reuse**
- Usability: registration, selection, ...

Requirements on policy language

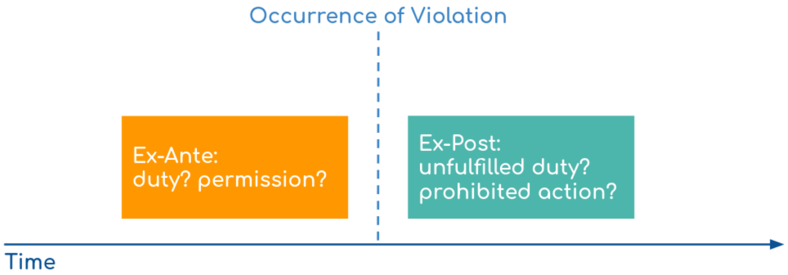
- Connects legal primitives and computational primitives
- Compositional and extensible specifications
- Supports simulation, scenario checking, verification

Policy Administration and Enforcement

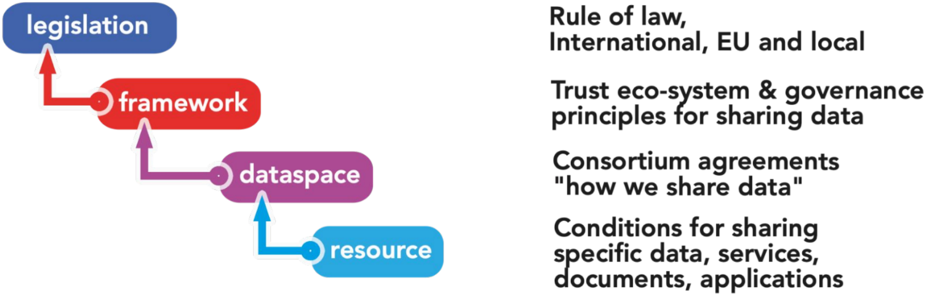
Requirements on Enforcement

- Occurs at all stages:
  "before, during and after processing"
- Ex-ante and **ex-post enforcement**
- Legal obligations
- Accountable
- **Explainable**
- Pre- and post-conditions
- Human-in-the-loop

Ex-post dynamic enforcement



Layered policy specification

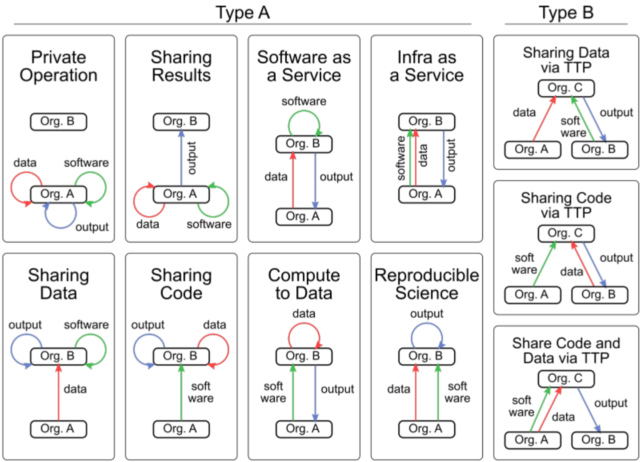


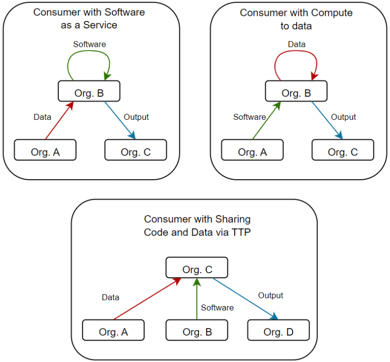
Experiments

- GDPR $\longrightarrow$ Financial sharing agreement $\rightarrow$ Organisational policy
- GDPR $\longrightarrow$ Medical consortium regulatory document $\rightarrow$ Resource-level access control

L. Thomas van Binsbergen et al. "Dynamic generation of access control policies from social policies". In: *The 11th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2021)*. Vol. 198. Procedia Computer Science. Elsevier, 2021, pp. 140–147. DOI: 10.1016/j.procs.2021.12.221

Reuse – Data exchange archetypes





https://gitlab.com/eflint/data-exchange-templates (Nina Verheijen)

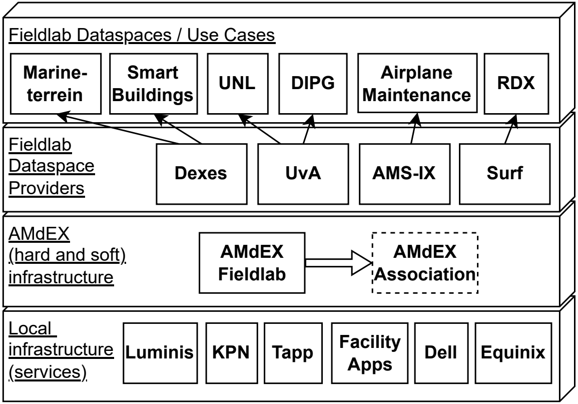
Sara Shakeri, Lourens Veen, and Paola Grosso. "Evaluation of Container Overlays for Secure Data Sharing". In: *2020 IEEE 45th LCN Symposium on Emerging Topics in Networking (LCN Symposium)*. 2020, pp. 99–108. DOI: 10.1109/LCNSymposium50271.2020.9363266

Section 2

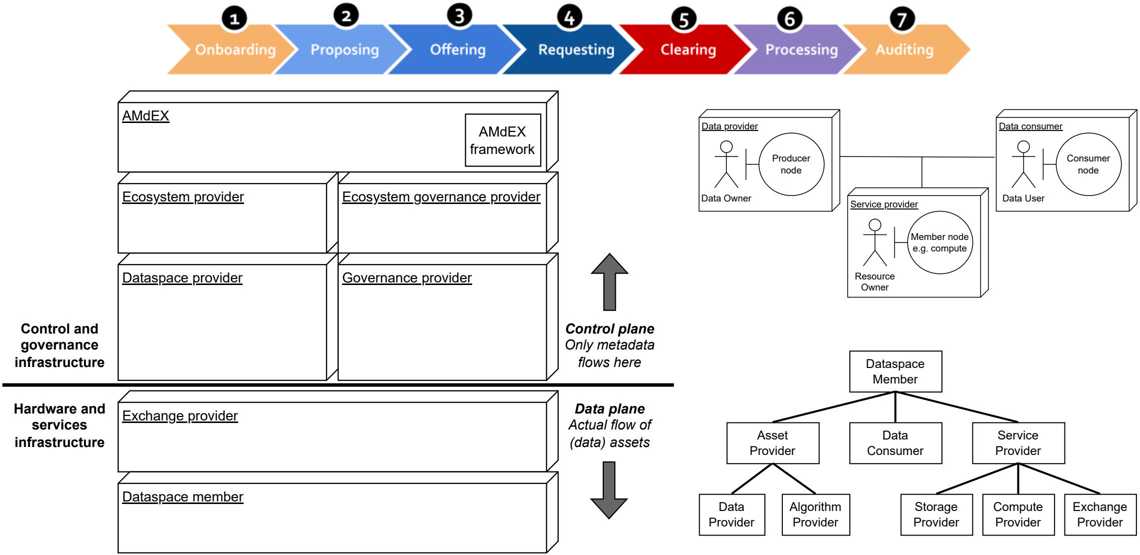
AMdEX fieldlab

Joint with: AMdEX partners

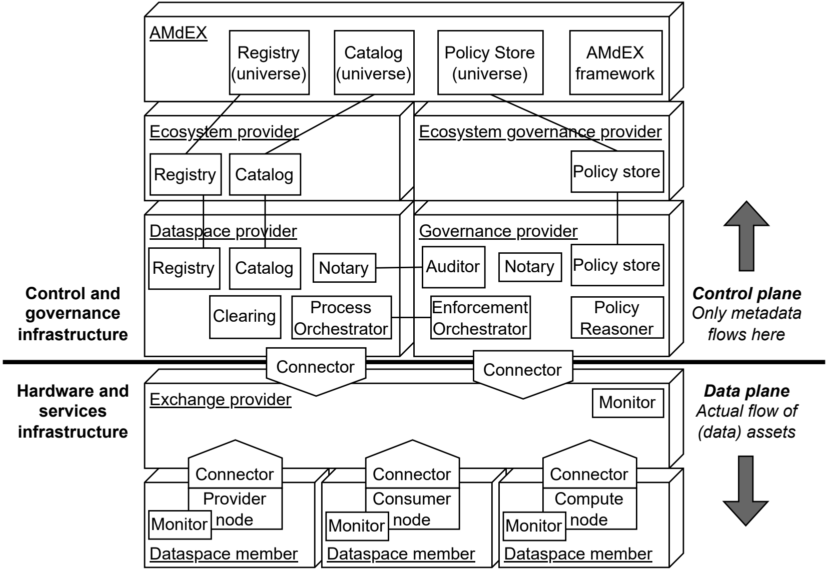
AMdEX fieldlab overview



AMdEX Reference Architecture



Architecture with Components



AMdEX fieldlab – main results

Main results and insights

- High-level reference architecture
- Main selling points: genericity (archetypes), integrated governance
- Implemented components: Catalog, Secure Analysis Environment, Policy Reasoner, Orchestrator
- Lab experiments: Policy Store, Notary/auditor,

Next steps

- Consolidation and standardisation
- Interoperability with EU initiatives, IDSA in particular
- **AMdEX-DMI** project: scaling up use cases, researching auditing
- **Targeted use cases** with specific service providers:
  synthetic data, secure multi-party computation, federated ML, differential privacy, ...

Section 3

Policy-enhanced Access Control

Joint with: Milen G. Kebede

Back to basics: Access control and XACML architecture

An access request typically consists of:

- An actor
- An action (e.g., read/write)
- A resource / asset
- Optionally: A context identifier

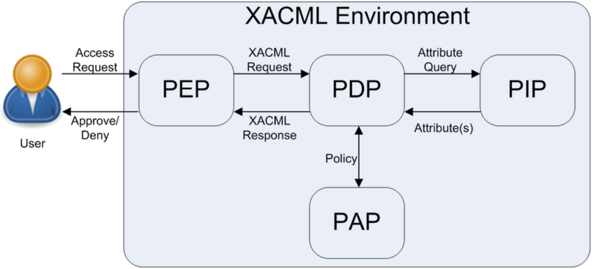


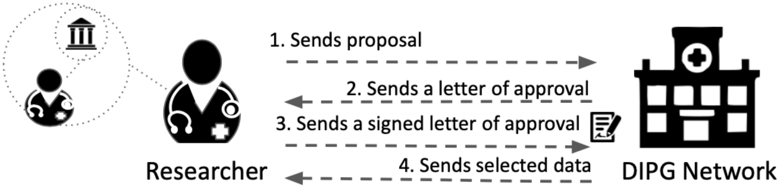
Figure: Simplified XACML architecture – M.S. Ferdous. "User-controlled identity management systems using mobile device". PhD thesis.

```
Fact actor
Fact asset

Act read  Actor actor Related to asset
Act write Actor actor Related to asset
```

DIPG use case

- Diffuse Intrinsic Pontine Gliomas(DIPG) registry: rare disease repository that allows researchers to access patient data that can lead to discovering new treatment.



Dynamic generation of access control policies from social policies

L. Thomas van Binsbergen[1,a], Milen G. Kebede[a], Joshua Baugh[b], Tom van Engers[a], Dannis G. van Vuurden[b]

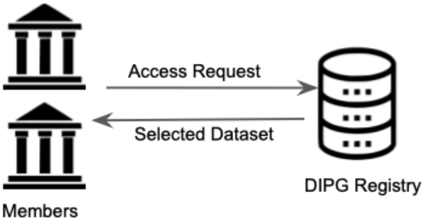
[a]*Informatics Institute, University of Amsterdam, 1090GH Amsterdam, The Netherlands*
[b]*Princess Maxima Center for Pediatric Oncology, Department of Neuro-oncology, Utrecht, The Netherlands*

The DIPG case – Compliance questions

According to the GDPR (1) and the DIPG regulatory document (2):

1. What conditions need to be fulfilled by a member before making data available?

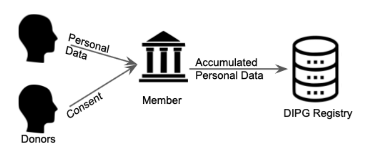
2. What conditions need to be fulfilled when accessing (3) data from the registry?



eFLINT reasoner as Policy Decision Point

Question 1

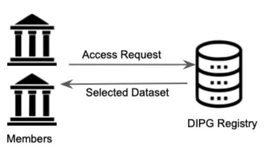
What conditions need to be fulfilled before making data available?



`?Enabled(write(<X>,<Y>))`

Question 2

What conditions need to be fulfilled when accessing data from the registry?



`?Enabled(read(<X>,<Y>))`

L. Thomas van Binsbergen et al. "Dynamic generation of access control policies from social policies". In: *The 11th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2021)*. Vol. 198. Procedia Computer Science. Elsevier, 2021, pp. 140–147. DOI: 10.1016/j.procs.2021.12.221

Compliance Question 1 – GDPR Rules

GDPR – Article 6(1)(a):

Personal data can be collected for a specific purpose if consent has been given for that purpose

GDPR – Article 5(1)(d):

Data must be accurate for purpose specified

```
Act collect-personal-data
  Actor controller
  Recipient subject
  Related to data, processor, purpose Where subject-of(subject,data)
  Creates processes(processor, data, controller, purpose)
  Conditioned by accurate-for-purpose(data, purpose)
  Holds when consent(subject, controller, purpose, data)
```

Compliance Question 1 – regulatory document

DIPG Regulatory document – Article 4(2):

Members should transfer data to the DIPG registry in a coded form only

```
Fact coded Identified by dataset

Act make-data-available
  Actor institution
  Related to dataset
  Conditioned by coded(dataset)
  Holds when member(institution)
```

Compliance Question 1

```
Extend Act make-data-available Syncs with (Foreach donor:
  collect-personal-data(controller = institution
                       ,subject    = donor
                       ,data       = dataset
                       ,processor  = "DCOG"
                       ,purpose    = "DIPGResearch")
    When subject-of(donor, dataset))
```

An institution can make a dataset available when (for each donor (subject) in the dataset):

- The institution is a member (DIPG Regulatory Document – Article 4(2))
- Data is coded (DIPG Regulatory Document – Article 4(2))
- Consent is given by *each* donor for data processing by the DCOG for the purpose of DIPGResearch (GDPR – Article 6)
- Data should be accurate for the purpose DIPGResearch (GDPR – Article 5)

```
Extend Act write Holds when Enabled(make-data-available(member, asset))
                          && affiliated-with(actor, member)
```

Compliance Question 2

```
Extend Act read Holds when (Exists project, institution:
  approved(project,institution) &&
  selected(asset,project) &&
  affiliated(actor, institution))
```

An actor can *read* an asset when (there exists a project and an institution for which):

- The project is approved for the institution
- The asset is selected for the project
- The actor is affiliated with the institution

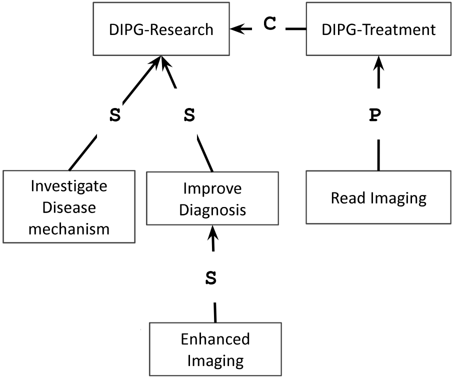
Subsection 1

Purpose-based Access Control

Purpose graph

Purpose graph (V, S, P, C) is a directed acyclic graph (DAG) with purposes in V labelling nodes and with three sets of edges S, P, and C corresponding to the specific-of, prerequisite-of and compatible-with relations respectively.

Example: DIPG Purpose graph




```
+compatible-with(DIPGTreatment, DIPGResearch).
+specific-of(Investigate, DIPGResearch).
+specific-of(ImproveDiagnosis, DIPGResearch).
+specific-of(EnhancedImaging, ImproveDiagnosis).
+prerequisite-of(ReadImaging, DIPGTreatment).
Fact asset Identified by DIPGData
Fact subject Identified by Subject.
+subject-of(Subject, DIPGData).
```

Evaluating Access requests

An access request is a tuple (S, A, O) where A is an action, S is the actor performing the action and O is the asset on which the action will be performed and is evaluated using two approaches

(1) Action A is expected to be a node in the purpose graph.

(2) Action A corresponds to a program submitted by a user to perform some processing on the asset.

  - Purpose is computed by analyzing the source code of the program.

```
Physical enhance-imaging
    Syncs with process(actor, EnhancedImaging, DIPGData)
Physical read-imaging
    Syncs with process(actor, ReadImaging, DIPGData).
```

The physical actions are ‘qualified’ as being an instance of the institutional action and inherit its pre- and post-conditions.

Action Matching

Given a triple (S,A,O), which forms instances of process and an access request, a path of edges in the purpose graph is sought that links the action A to one of the obliged or consented purposes (for all subjects, in the case of consent).

```
+consent(Subject, Member, DIPGData, DIPGResearch).
enhance-imaging(Member). // Lawful:
// EnhancedImaging -s-> ImproveDiagnosis -s-> Consented(DIPGResearch)
read-imaging(Member). // Lawful:
// ReadImaging -p-> DIPGTreatment -c-> DIPGResearch
// +must-inform(Member, Subject, DIPGTreatment)
```

(1) **enhance-imaging** action is lawful because it is more specific than the consented purpose DIPGResearch

(2) **read-imaging** action is considered lawful by invoking the prerequisite-of and compatible-with relations

compatible-with relation generates **must-inform duty** which is then added to the eFLINT knowledge base.

Section 4

Discussion

Some open questions

- How general is our approach? How realistic is it to support generic archetypes?
  Can we sufficiently standardize to include many types of service providers?
  Howto secure multi-party computation (sMPC) and federated machine learning (FML)?

Some open questions

- How general is our approach? How realistic is it to support generic archetypes?
  Can we sufficiently standardize to include many types of service providers?
  Howto secure multi-party computation (sMPC) and federated machine learning (FML)?

- How realistic is our approach to policy administration and construction?
  Requires collaboration between legal and software expert?
  Many interpretations and versions across layers, how to prevent inconsistencies?

Some open questions

- How general is our approach? How realistic is it to support generic archetypes?
  Can we sufficiently standardize to include many types of service providers?
  Howto secure multi-party computation (sMPC) and federated machine learning (FML)?

- How realistic is our approach to policy administration and construction?
  Requires collaboration between legal and software expert?
  Many interpretations and versions across layers, how to prevent inconsistencies?

NGF-funded: AMdEX-DMI project

- How to trace and audit exchange processes without access to data or algorithms?
  Solutions involving encrypted-storage providers?

Some open questions

- How general is our approach? How realistic is it to support generic archetypes?
  Can we sufficiently standardize to include many types of service providers?
  Howto secure multi-party computation (sMPC) and federated machine learning (FML)?
- How realistic is our approach to policy administration and construction?
  Requires collaboration between legal and software expert?
  Many interpretations and versions across layers, how to prevent inconsistencies?

NGF-funded: AMdEX-DMI project

- How to trace and audit exchange processes without access to data or algorithms?
  Solutions involving encrypted-storage providers?
- What information is needed for auditing, and are service providers willing to share?
  Can we handle logging information as 'just another' sensitive data asset?
  Can we identify 'levels of auditability' to become part of consortium agreements?

Policy-driven distributed data exchange processes

EPI Closing Event

L. Thomas van Binsbergen

Informatics Institute, University of Amsterdam

March 7, 2024

Section 5

Data Exchange Processes

1. Onboarding

- Onboarding of a dataspace member, a use case, an external ecosystem/dataspace
- Involves: technical connection, registration, possible certification, archetype selection

- **Registry**: Registers AMdEX participants and dataspace members with their roles; can be used for finding possible new dataspace members

1. Onboarding

- Onboarding of a dataspace member, a use case, an external ecosystem/dataspace
- Involves: technical connection, registration, possible certification, archetype selection

Member	User	Role	Component
UNL	analyst(UNL)	data consumer / algorithm provider	consumer node
Surf	resource owner	compute provider	compute node
University X	analyst(X) custodian(X)	data consumer asset provider / compute provider	consumer node compute node

Table: Onboarded dataspace members of UNL use case. Agreement: equal schema, horizontal split

- **Registry**: Registers AMdEX participants and dataspace members with their roles; can be used for finding possible new dataspace members

2. Proposing

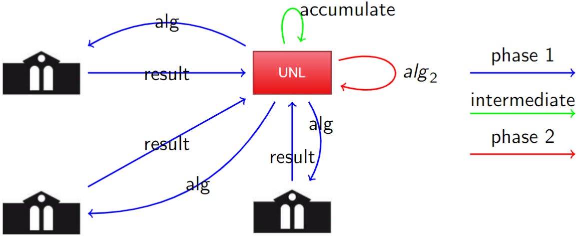
- Discuss the inclusion of (additional) archetypes, members, or resource
- May result in additional onboarding steps and/or in offers made

2. Proposing

- Discuss the inclusion of (additional) archetypes, members, or resource
- May result in additional onboarding steps and/or in offers made

UNL Scenario 1 (Compute to data):

Compare the difference in average salary between male and female academics at various function levels (UD, UHD, HL)

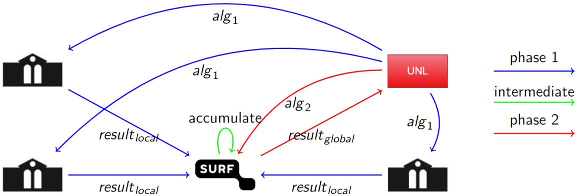


2. Proposing

- Discuss the inclusion of (additional) archetypes, members, or resource
- May result in additional onboarding steps and/or in offers made

UNL Scenario 2 (Sharing data via TTP):

> *How long does it take men and women on average to become full professor, independent of whether they stayed at the same university?*



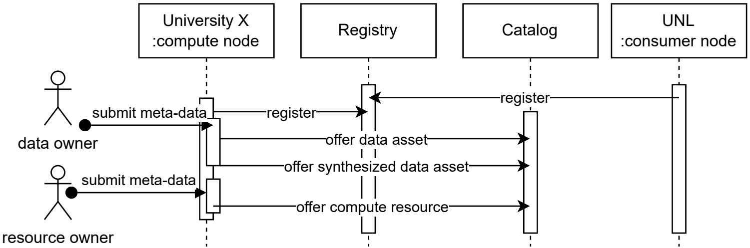
3. Offering

- Offer (data) assets and resources under certain pre- and post-conditions
- Offer should be checked for consistency with consortium agreement

Catalog: Holds meta-data about assets and resources offered, including policies (conditions)

3. Offering

- Offer (data) assets and resources under certain pre- and post-conditions
- Offer should be checked for consistency with consortium agreement



Catalog: Holds meta-data about assets and resources offered, including policies (conditions)

4. Requesting

- Abstract archetypes and execution plans become concrete, e.g. from which universities is data requested? Which query is used? Is the TTP involved?

4. Requesting

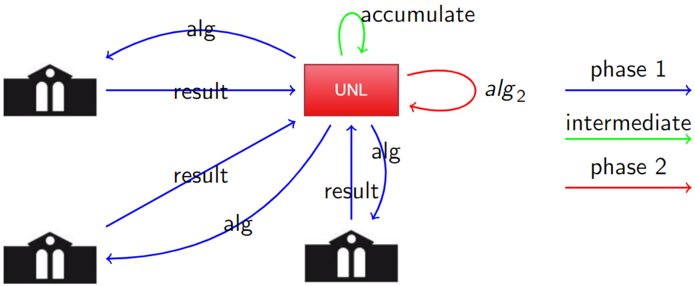
Data Request

Status: pending

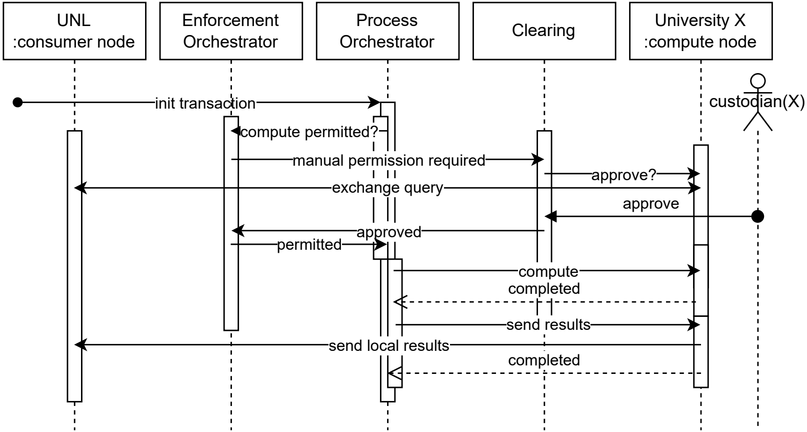
Consumer	UvA - Marten Steketee
Providers	VU UU
Description	Demo
Query	SELECT `Salschal`, `Taakomv` FROM arbeid;
Responses:	
VU	Status: pending
UU	Status: accepted

5. Clearing

- Processes through which pre-conditions are checked and enforced, e.g. do the resource conditions allow the selected archetype, did custodians approve the request?



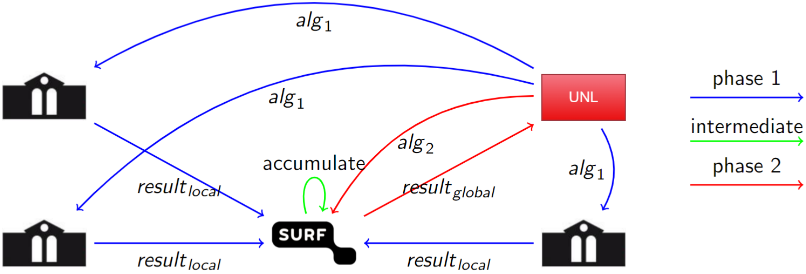
5. Clearing



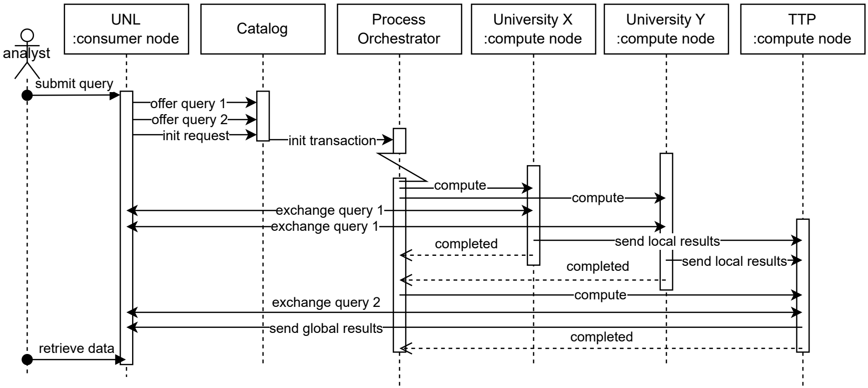
- **Clearing** modules: handling pre-conditions that (may) require human action
- **Enforcement Orchestrator**: ensures **Policy Reasoner** receives the required policy (from **Policy Store**) and policy information to make policy decisions

6. Processing

- The execution of data exchange process steps
- May be manual or automatic, may involve centralized coordination



6. Processing



- **Process Orchestrator**: *drives* the step-by-step execution of exchange processes

Lesson learnt

Centralized control not necessary; Decentralized control at odds with accountability

7. Auditing

- Determining the compliance of processing, including post-conditions, after the fact
  - against new versions (interpretations) of policies
  - with new information relevant to policy
- Examples:
  - Did all approving members make their data available? (requires tracing)
  - Was the data of the expected quality? And appropriately synthesized? (requires resource)
  - Did the third party processor use a secure analysis environment? (requires logging)

- Enforcement and process **notary** components keep record of exchange processes

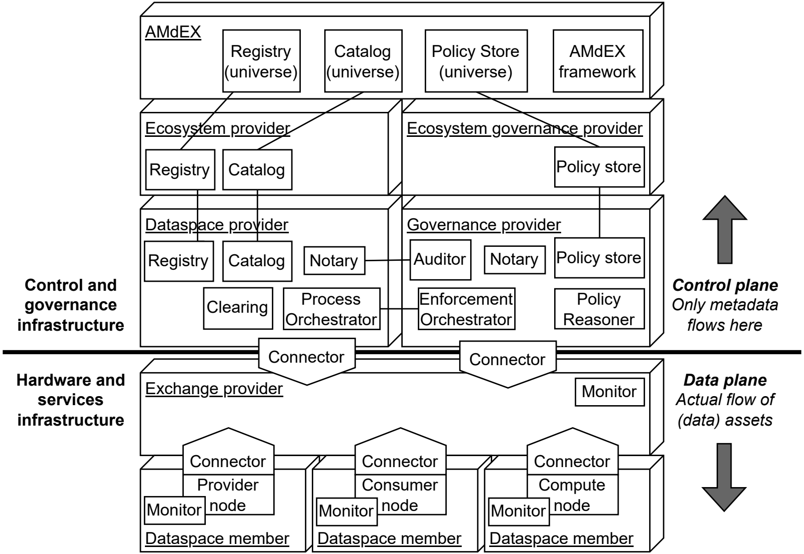
7. Auditing

- Determining the compliance of processing, including post-conditions, after the fact
  - against new versions (interpretations) of policies
  - with new information relevant to policy
- Examples:
  - Did all approving members make their data available? (requires tracing)
  - Was the data of the expected quality? And appropriately synthesized? (requires resource)
  - Did the third party processor use a secure analysis environment? (requires logging)

Lessons learnt

AMdEX 'meta-data' principle at odds with auditing

Architecture components (overview)



Section 6

The eFLINT language

Toy example – knowledge representation

(Toy Article 1) a natural person is a legal parent of another natural person if:

- *they are a natural parent, or*
- *they are an adoptive parent*

```
Fact person Identified by String
Placeholder parent    For person
Placeholder child     For person

Fact natural-parent   Identified by parent * child
Fact adoptive-parent  Identified by parent * child

Fact legal-parent     Identified by parent * child
  Holds when adoptive-parent(parent,child)
          || natural-parent(parent,child)
```

Toy example – powers and duties

(Toy Article 2) a child has the power to ask a legal parent for help with their homework, resulting in a duty for the parent to help.

```
Act ask-for-help
  Actor       child
  Recipient   parent
  Creates     help-with-homework(parent,child)
  Holds when  legal-parent(parent,child)

Duty help-with-homework
  Holder         parent
  Claimant       child
  Violated when  homework-due(child)

Fact homework-due Identified by child

Act help
  Actor       parent
  Recipient   child
  Terminates  help-with-homework(parent,child)
  Holds when  help-with-homework(parent,child)
```

Toy example – scenario / case

‘Domain of discourse’ specification:

```
Fact person Identified by Alice, Bob, Chloe, David
```

Initial state:

```
+natural-parent(Alice, Bob).
+adoptive-parent(Chloe, David).
```

Scenario:

```
ask-for-help(Bob, Alice).                      // permitted: Alice is Bob's legal parent
+homework-due(Bob).                            // homework deadline passed
?Violated(help-with-homework(Alice,Bob)).      // query confirms duty is violated
help(Alice,Bob).                               // duty terminated
```