gaia-x

Leveraging ODRL for Compliance as Code

A Comprehensive Framework

May 03, 2023

Frédéric BELLAICHE, Ph.D
VP Technology & Research - Dawex

Dawex at a glance: a European scale-up recognized worldwide for its expertise and achievements in data exchange

Company profile

Founded in 2015

Offices: Paris, Lyon, Montreal, Tokyo (2023)

Global reach

- France & Europe
- Japan (2nd largest market)
- North America
- Middle East

Recognized as a pioneer and innovator

- **9 awards** US, EU, ME
- **Tech Pioneer** at the World Economic Forum
- Speaker in **Davos**
- **Speaker at G7 Summit** and other global events
- **Leads Gaia-X Data Exchange** Working Group

Customer references

in more than **15 strategic sectors**

What is Compliance as Code ?

Compliance as code refers to the practice of using software **code to automate compliance processes** and ensure that data exchange activities meet **relevant policies**.

It involves writing **code that specifies the requirements** for compliance and then **executing** that code to validate compliance automatically.

- Started in January 2000 with **Lawrence Lessig** “Code Is Law” article: **code acts as the law** since it dictates what the users can do or not
- From “**Code is Law**” to “**Law is Code**”: regulations define the set of actions of what can / must be done in the digital space
- In the context of **Data Exchange**, compliance as code can be used to **ensure** that data is exchanged securely and in accordance with regulations and **usage policies**.

What is Compliance as Code ?

Policy as Code and Compliance as Code

- **Policy as Code:** Policy as code is a concept that involves representing policies as machine-readable code that can be **evaluated** and **enforced** automatically. Typically, in **Open Policy Agent** (OPA), policy as code refers specifically to the practice of using OPA's declarative language, **Rego**, to express policies and then enforcing those policies
- **Compliance as Code:** Compliance as code involves using a language to express **policy modeling** for compliance checking. The **Open Digital Rights Language** (ODRL) is a standard providing a flexible and extensible language for describing the rights and permissions associated with digital content (who can access it, under what conditions, and for what purposes), including regulatory requirements and business policies.
- **Both** are needed, they do not oppose

What is ODRL ?

ODRL is a **model** for describing the **usage of content**, including authorized, prohibited, and mandatory actions, the resources on which the actions apply, the actors and participants involved in the actions, usage conditions, and additional information such as responsibilities and regulations.

It allows the implementation of **usage control features** that complement the access controls usually in place. These usage controls come from licenses and are applied before using data, so they are dependent on each data and its usage environment.

Some examples of usage control include authorization of usage for a specific period, anonymization of data before processing, and prohibition of data transfer under certain conditions.

Links:

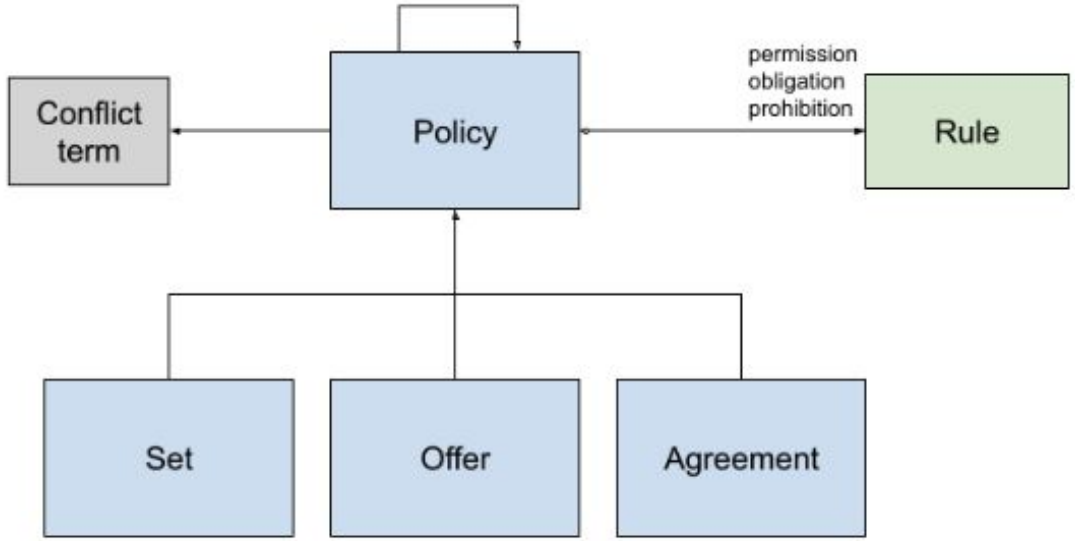
- ODRL Information Model 2.2: https://www.w3.org/TR/odrl-model/
- ODRL Vocabulary & Expression 2.2:https://www.w3.org/TR/odrl-vocab/
- ODRL Implementation Best Practices:https://w3c.github.io/odrl/bp/
- ODRL Profile Best Practices: https://w3c.github.io/odrl/profile-bp
- Market Data Profile :https://www.w3.org/2021/md-odrl-profile/v1/

Main concepts of ODRL

Policies

A policy is a group of rules (which can be permissions, prohibitions, or obligations). It has three child classe

- **Set**: a generic collection of rules.
- **Offer**: a collection of rules that are offered by an actor designated by the "assigner" property
- **Agreement**: a collection of rules that have been agreed upon by an actor designated as "assignee" and another actor designated as "assigner"

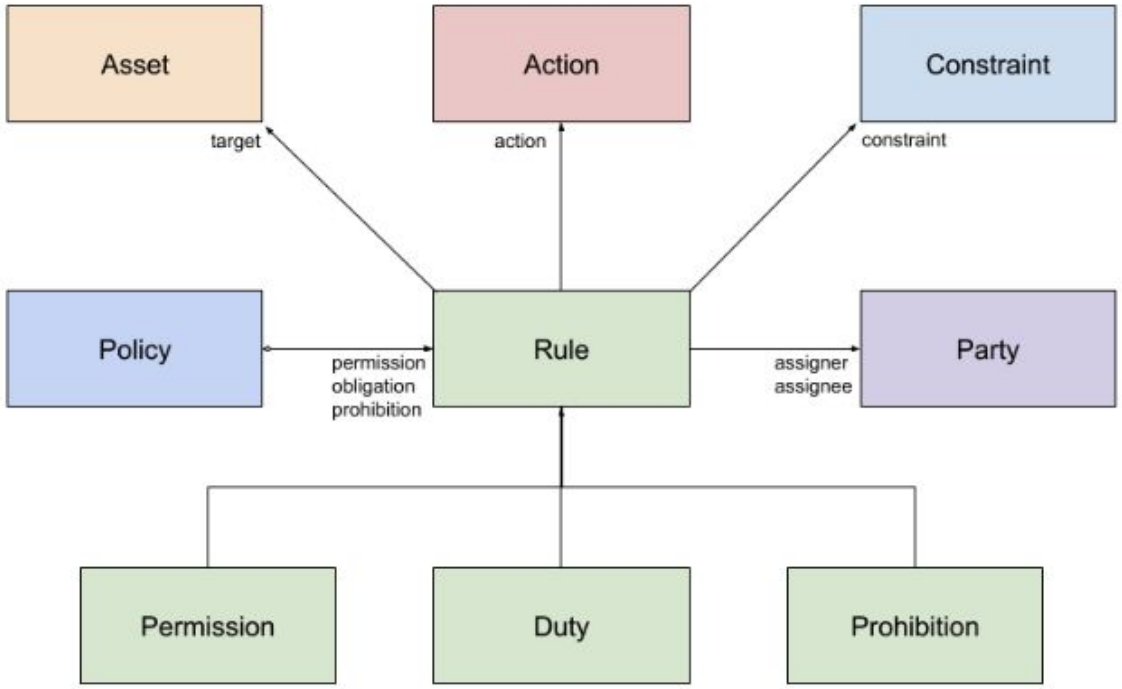


Main concepts of ODRL

Rules

Rule is the base class defining the rules of policies. It has the following properties: asset, which defines the target resource, action, which defines the operation to be performed on the resource, party, which defines the actors involved in the rule, and constraint, which defines the conditions for the rule's validity (e.g., date > 2020). It has three child classes

- **Permission:** This rule grants permission for the actor to perform the action.
- **Obligation:** This rule obliges the actor to perform the action.
- **Prohibition:** This rule prohibits the actor from performing the action

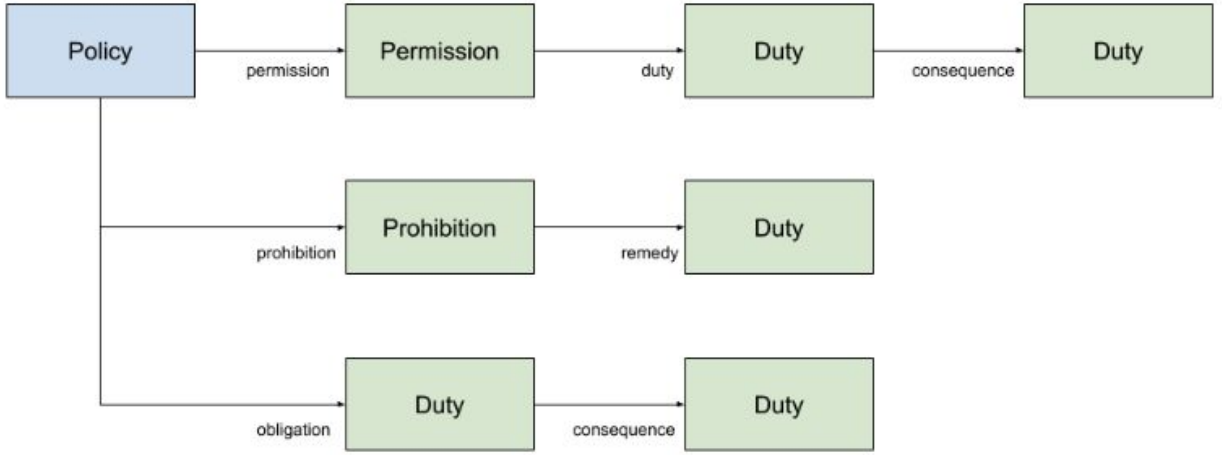


Main concepts of ODRL

Duty

The Duty class can also be used to specify a rule:

- **In a permission**, it specifies pre-conditions that must be met before granting permission.
- **In an obligation**, it specifies an action to be performed if the obligation is not fulfilled.
- **In a prohibition**, it specifies an action to be performed if the prohibition is not respected.

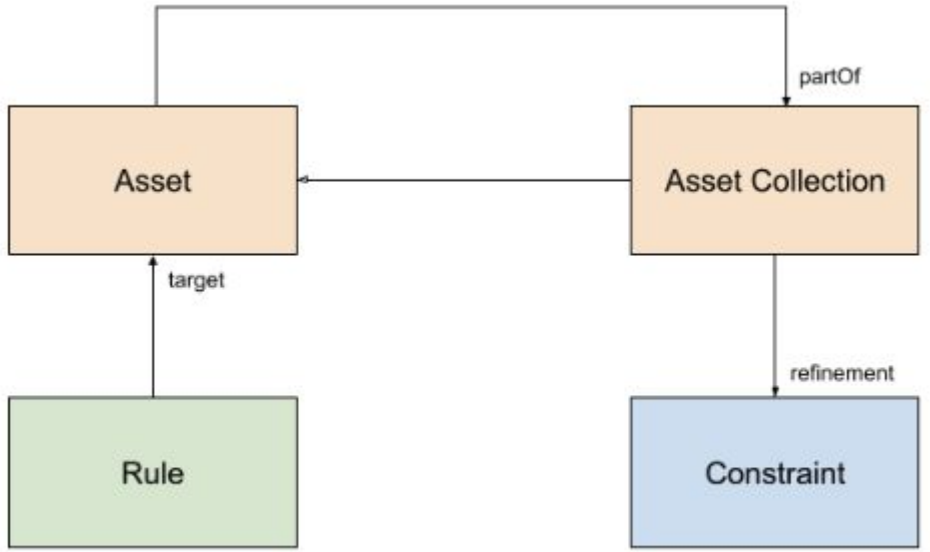


Main concepts of ODRL

Asset

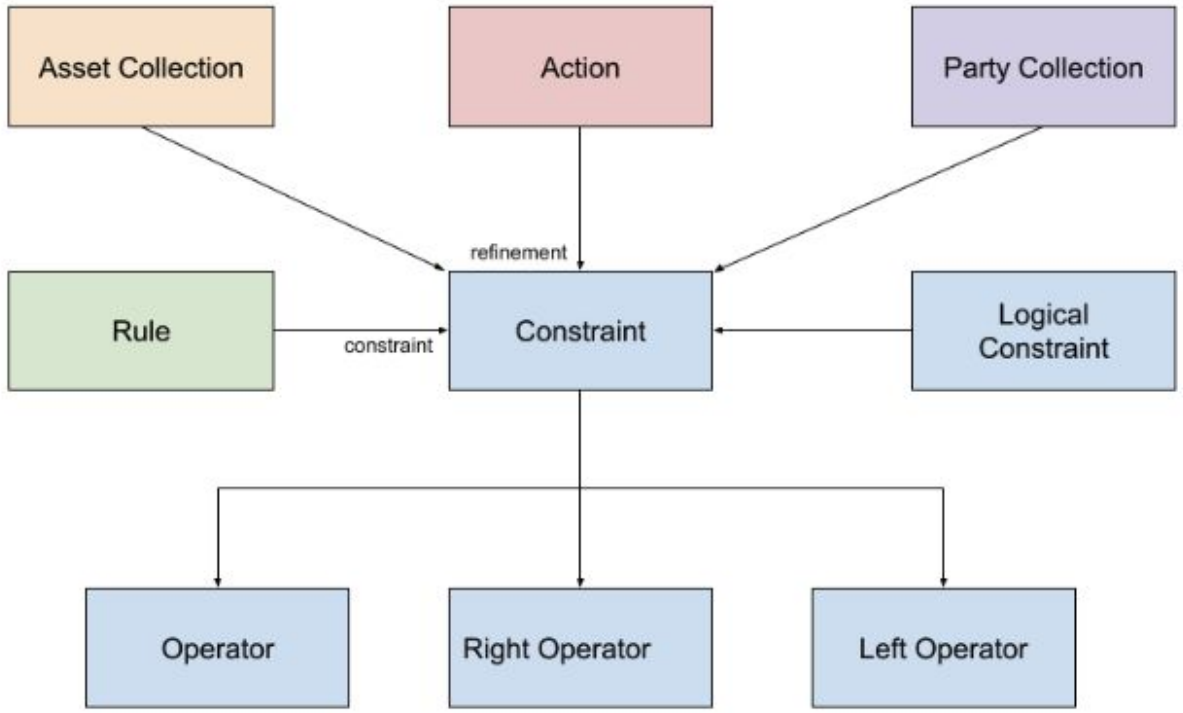
An asset is a resource or a collection of resources. An asset is the target of a rule to which it applies.

A **collection** can have constraints that allow filtering of the elements of the collection to which a rule applies



Main concepts of ODRL

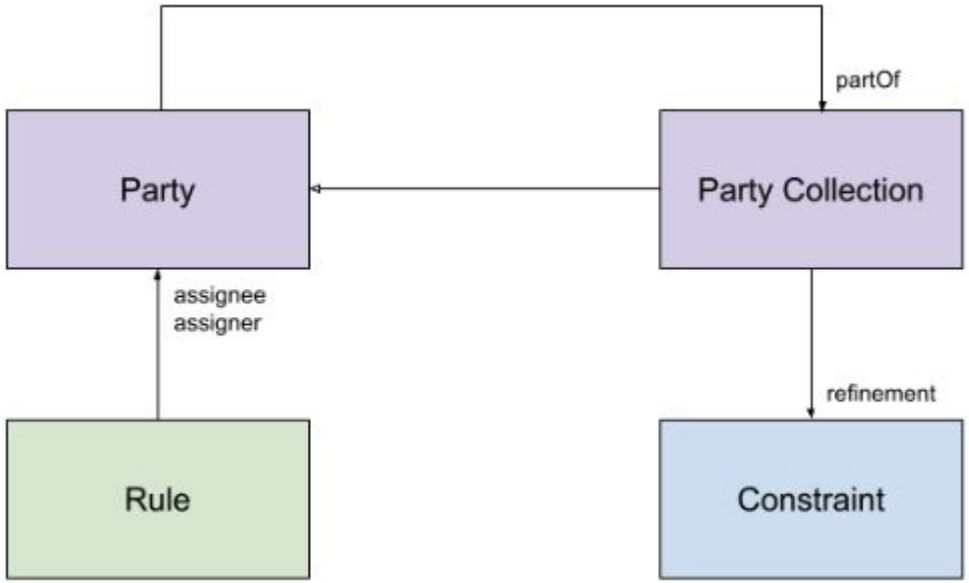
- **Action:** An action is an operation that can be performed on an asset. ODRL defines, among others, the two main actions: "**use**" and "**transfer**". Other actions can be defined in a specific vocabulary
- **Constraint**: Constraints are logical expressions that allow filtering of different collections. ODRL defines the following logical operators: "**or**", "**xone**", "**and**", and "**andSequence**".



Main concepts of ODRL

- **Party:** A party is an actor or collection of actors who have a functional role in a rule.

  The actor can have the role of "assignee" (recipient of the rule) or "assigner" (issuer of the rule). A collection can have constraints that allow filtering of the elements of the collection to which a rule refers.



A few examples

Licence

Exclusivity: The term "**ensureExclusivity**" allows to specify an exclusive license

```
"@context": "http://www.w3.org/ns/odrl.jsonld",
"@type": "Set",
"uid": "http://[redacted].com/policy:1010",
"obligation": [
  {
    "target": "http://[redacted].com/dataset/123",
    "assignee": "Data Provider",
    "action": "ensureExclusivity"
  }
]
}
```

A few examples

Territories

The term **'spatial'** is used to specify territories
Restrict the use to the specified territories
using the operator **isAnyOf**:

```
{
 "@context": "http://www.w3.org/ns/odrl.jsonld",
 "@type": "Set",
 "uid": "http://[redacted].com/policy:1010",
 "permission": [
   {
     "target": "http://[redacted].com/dataset/123",
     "assignee": "Data Acquirer",
     "action": "use",
     "constraint": [
       {
         "leftOperand": "odrl:spatial",
         "operator": "isAnyOf",
         "rightOperand": {
           "@list": [ "fr", "es" ]
         }
       }
     ]
   }
 ]
}
```

A few examples

Industries

The term "**industry**" is used to specify business sectors.
Restrict the use to specified business sectors
using the "**isAnyOf**" operator

```
{
 "@context": "http://www.w3.org/ns/odrl.jsonld",
 "@type": "Set",
 "uid": "http://[redacted].com/policy:1010",
 "permission": [
   {
     "target": "http://[redacted].com/dataset/123",
     "assignee": "Data Acquirer",
     "action": "use",
     "constraint": [
       {
         "leftOperand": "odrl:industry",
         "operator": "isAnyOf",
         "rightOperand": {
           "@list": [ "automotive" ]
         }
       }
     ]
   }
 ]
}
```

A few examples

Sub-licensing

The term **grantUse** allows managing the possibilities of sub-licensing

Example: no sub-licensing right:

```
{
 "@context": "http://www.w3.org/ns/odrl.jsonld",
 "@type": "Set",
 "uid": "http://[redacted].com/policy:1010",
 "prohibition": [
   {
     "target": "http://[redacted].com/dataset/123",
     "assignee": "Data Acquirer",
     "action": "grantUse"
   }
 ]
}
```

What’s next ?

The Gaia-X **Data Exchange Services** Working Group is working on the usage of **ODRL** as a mean towards **Compliance as Code for Data Exchange**

- **Exploring** ORDL aspects, expressing policies
- **Evaluating** ORDL profiles & extensions
- **Connecting** ORDL concepts to Data Products / DCAT, Participants and Services

Thank you !

frederic.bellaiche@dawex.com