Fraunhofer
IESE

Usage Control in the Industrial Data Space

Authors:
Andreas Eitel
Christian Jung
Christian Haas
Christian Mader
Gerd Brost
Julian Schütte
Jaroslav Pullmann
Johannes Zrenner

IESE-Report No. 056.17/E
Version 1.0
December 2017

A publication by Fraunhofer IESE

Fraunhofer IESE is an institute of the Fraunhofer Gesellschaft.
The institute transfers innovative software development techniques, methods and tools into industrial practice, assists companies in building software competencies customized to their needs, and helps them to establish a competitive market position.

Fraunhofer IESE is directed by
Prof. Dr.-Ing. Peter Liggesmeyer
(Executive Director)
Prof. Dr. Dieter Rombach
(Director Business Development)
Fraunhofer-Platz 1
67663 Kaiserslautern
Germany

Abstract

In the age of Industry 4.0, data exchange between different organizations is an essential prerequisite to add more value to data and to develop modern business models. However, we have to solve several challenges to facilitate a secure and trustworthy data exchange between different organizations. In addition, data sovereignty is a key success factor for data-driven business models. In the Industrial Data Space, we provide solutions to realize a secure and trustworthy data exchange as well as data sovereignty.

In this report, we focus on data usage control, a conceptual and technological solution to cope with data sovereignty challenges. We introduce a common scenario for the Industry 4.0 age, in which a supplier and an original equipment manufacturer (OEM) are exchanging data to mitigate risks in the supply chain management. We describe the concepts of usage control and align them with this application scenario.

In addition, we introduce the different technologies for implementing usage control in the Industrial Data Space. In doing so, we present the Integrated Distributed Usage Control Enforcement (IND²UCE), Label-based Usage Control (LUCON) and Data Provenance. We align every technological solution to our accompanying application scenario.

Finally, we present building blocks to realize data usage control in the Industrial Data Space such as the Information Model, Trusted Connector, and a data flow interceptor for Apache Camel.

Keywords: IND²UCE, Data Usage Control, Industrial Data Space

Table of Contents

1 Introduction

The Industrial Data Space is about creating a data space where businesses can exchange and exploit data in a secure manner. For the Industrial Data Space as well as other data-driven businesses, data sovereignty is a key success factor. Data sovereignty has the goal to provide a Data Owner [1] with full control over her data. This includes being able to control the usage of her data by the Data Consumer.

In this document, we present data usage control as technical solution to cope with data sovereignty. In doing so, we introduce an application scenario, in which a supplier and an OEM exchanges sensitive data to improve their business. We describe all data usage control technologies used within the Industrial Data Space consortium and align them with our usage control enabled application scenario. In this context, usage control is used to enforce the company policies (such as "delete data after 14 days") that are attached to the data exchanged between both parties.

1.1 Structure

We divide our document into five chapters:

Chapter 1 motivates usage control in the Industrial Data Space and illustrates a real world application scenario including natural language policies about data usage restriction. We use the scenario throughout the entire document.

In Chapter 2, we describe the difference between access and control and usage control. In more detail, we present basic concepts of usage control such as enforcement, decision-making, specification, management and negotiation. In addition, we present related concepts to usage control.

Chapter 3 presents different building blocks and technological solutions within the IDS consortium. In doing so, we present the Information Model, the Trusted Connector, and the Apache Camel Interceptor. We conclude by presenting the involved roles in the usage control process.

In Chapter 4, we present the existing usage control technologies in the Industrial Data Space: Integrated Distributed Usage Control Enforcement (IND²UCE), Label-based Usage Control (LUCON) and Data Provenance. We align every technological solution to our accompanying application scenario.

Chapter 5 discusses capabilities, limitations, and implications of usage control in the Industrial Data Space. We conclude the chapter with discussing different stages of expansion for data usage control.

1.2 Motivation

Nowadays, business is spurred by continuously exchanging information between business partners. However, data is typically secured by access control mechanisms only. After access to data has been granted by these mechanisms, data can be arbitrarily altered, copied and disseminated by the recipient. Data usage control offers possibilities to control future data usages beyond the initial access (also known as obligations).

In the age of Industry 4.0, there are more critical and sensitive data exchanged between business partners (see Figure 1). In general, companies have intrinsic and extrinsic motivations to apply usage control: On the one hand, companies may use usage control to prevent misuse of their own data, to protect their intellectual property, and to preserve the data value (intrinsic motivation). On the other hand, companies have to comply with legal obligations such as the European Union General Data Protection Regulation EU-GDPR (extrinsic motivation). Hence, companies have to prevent misuse of other persons or companies data.

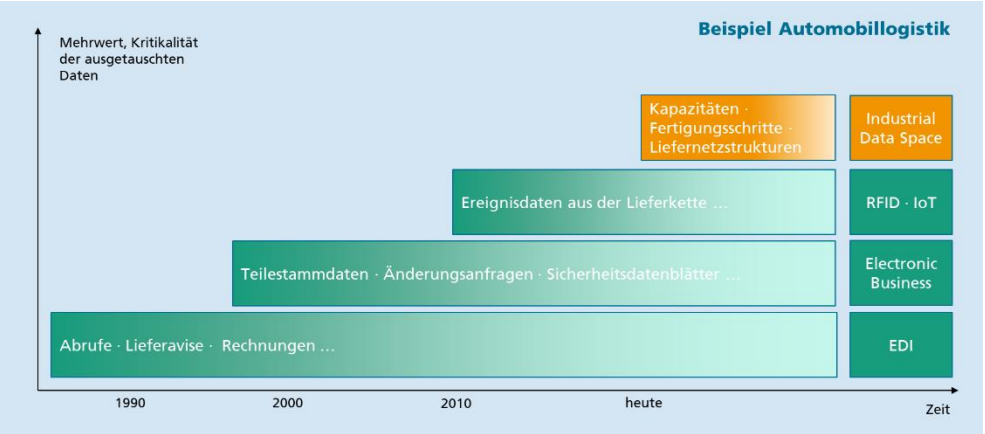

Figure 1 Today's business is spurred by exchanging critical and sensitive information

1.3 Accompanying Application Scenario

In the age of globalization and high cost pressure, supply networks of automotive original equipment manufacturers (OEM) are complex and interference-prone for risks (e.g., earthquake, fire, war). For that reason, supply chain risk management (SCRM) becomes more and more important for a high supply reliability.

Figure 2 illustrates the data exchange between a supplier and the OEM in a collaborative SCRM. On the one hand, there is data flowing from the suppliers to the OEMs such as affected parts and sub-supplier information, which the OEMs use in their supplier management system. On the other hand, the OEMs send data such as part demands or inventory range to the suppliers, which the suppliers process in their risk management system.

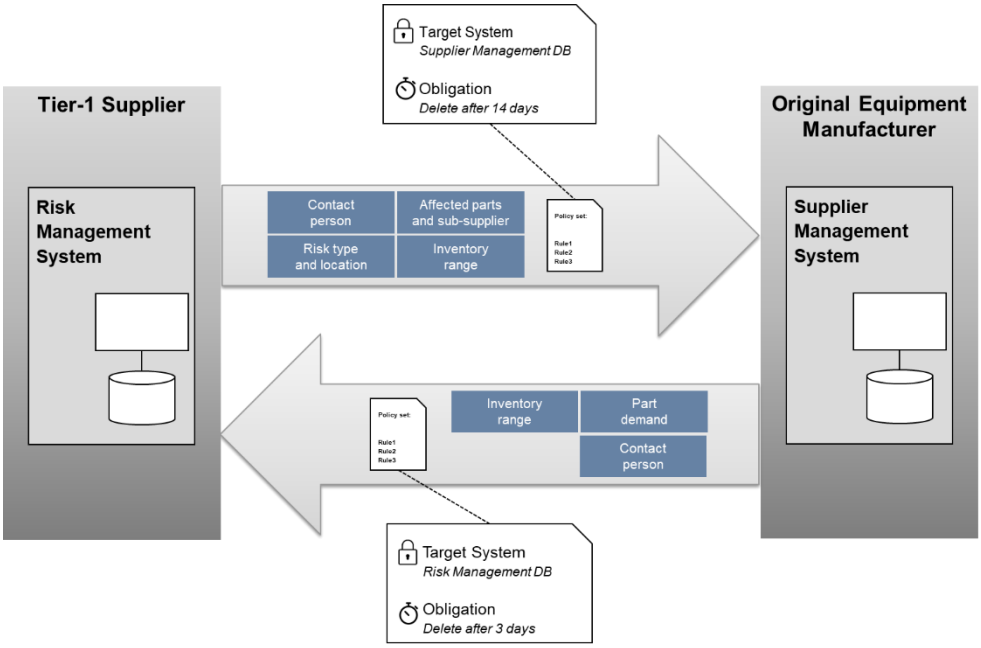

Figure 2 Data Exchange in the Collaborative Supply Chain Risk Management

Nowadays in the SCRM processes, most of the communication between the OEMs and the suppliers is done via phone, email, or web conferences. Table 1 shows the attributes of the data exchange from the supplier and OEM perspective (supplier as data provider and OEM as data provider).

Table 1 SCRM data exchange between OEM and supplier

Data from supplier to OEM	Data from OEM to supplier
• **Risk type and location** • **Affected parts and sub-supplier** • **Inventory range** • **Contact person**	• **Part demand** • **Inventory range** • **Contact person**

In the process, there is sensitive and valuable data provided by the supplier as well as by the OEM: For example, data about the sub-supplier is very sensitive

for the supplier. With such data, the OEM could skip the supplier and purchase directly form the sub-supplier. The part demand and inventory range are sensitive data for the OEM, because they make the production volume and warehouse transparent.

An automation of the data exchange in the SCRM process would lead to time and money savings for suppliers and OEM. In this case, the systems must ensure that the exchanged data is compliant with the company policies. This is where usage control can be used as technical extension to a contract to technically enforce the policies of the respective data provider. In fact, usage control improves security by controlling the data usage on the target system. Examples for appropriate policies in natural language and a first refinement are:

- The OEM can only use supplier data for risk or bottleneck management, but not for purchasing or sales purposes.
    - Data Provider: Supplier
    - Data Consumer: OEM
    - Enforcement Environment: Data Consumer
    - Usage Restrictions:
        - Target System=Supplier Management Database
        - Purpose=Risk Management, Bottleneck Management
- The OEM has to delete all exchanged data in the SCRM process after 14 days.
    - Data Provider: Supplier
    - Data Consumer: OEM
    - Enforcement Environment: Data Consumer
    - Usage Restrictions:
        - Time to Live=14 days
- The supplier has to delete all exchanged data in the SCRM process after three days.
    - Data Provider: Supplier
    - Data Consumer: OEM
    - Enforcement Environment: Data Consumer
    - Usage Restrictions:
        - Time to Live=3 days
- The supplier can only import data from the OEM into the system "risk management".
    - Data Provider: OEM
    - Data Consumer: supplier
    - Enforcement Environment: Data Consumer
    - Usage Restrictions:
        - Target System=Risk Management Database

We will use the aforementioned scenario and policies throughout the document to illustrate how usage control copes with such challenges.

Access control can cope with some aspects of the presented policies. For example, only users with roles related to risk or bottleneck management will be able to access the sensitive data, but no users with roles related to purchasing or sales. Alternatively, only specific target systems will get access to the exchanged data. The future obligation "deletion of data after a certain amount of time" can be solved by revoking the access rights. However, using access control to solve such data usage restrictions seems inappropriate. We will present further security requirements that traditional access control cannot achieve in Section 2.2.

2 Usage Control Concepts

Usage control is an extension to traditional access control (see Figure 3). It is about the specification and enforcement of restrictions regulating what must (not) happen to data. Thus, usage control is concerned with requirements that pertain to data processing (obligations), rather than data access (provisions). Usage control is relevant in the context of intellectual property protection, compliance with regulations, and digital rights management.

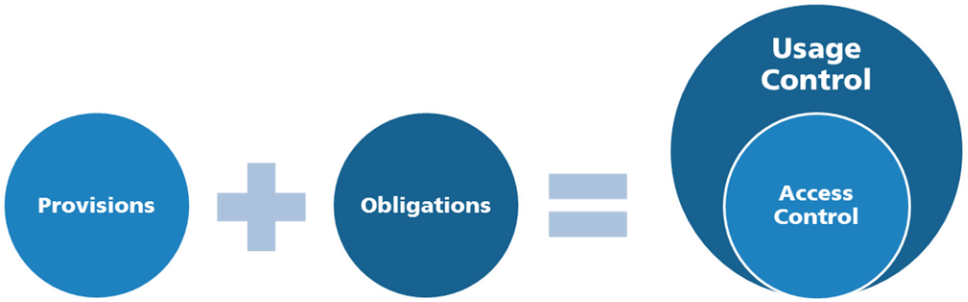

Figure 3 Usage control – an extension to traditional access control

2.1 Access Control

In information security, access control restricts access to resources. The term authorization is the process of granting permission to resources. Several access control models exist, such as Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-based Access Control (RBAC), Attribute-based Access Control (ABAC), etc. Although such a plethora of access control models exists, RBAC and ABAC are most commonly used.

We will use the XACML (eXtensible Access Control Markup Language) Standard [2] to introduce commonly used terms in the field of access control. XACML is a policy language to express ABAC rules. The main building blocks of the language are *subject*, *action*, *resource* and *environment*. The *subject* describes who is accessing a data asset (e.g., a user). The *action* describes what the subject wants to perform on the data asset (e.g., read, write). The *resource* describes the data asset. Finally, the *environment* specifies the context (e.g., time, location).

Figure 4 illustrates the data-flow model of XACML and the main actors or components to implement it: Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Information Point (PIP), and Policy Administration Point (PAP).

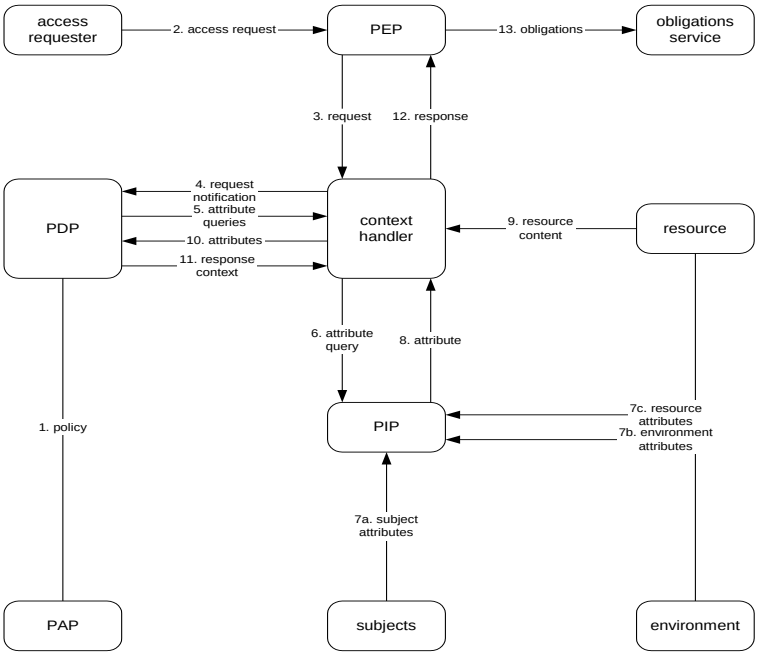

Figure 4 XACML Data-flow diagram [2]

In general, attributes can describe anything and anyone, but tend to split into four categories:

- Subject attributes
  Attributes that describe the user by e.g. age, role or clearance.
- Action attributes
  Attributes that describe the action attempted e.g. read, delete or view.
- Resource (or object) attributes
  Attributes that describe the resource itself e.g. object type, location or classification.
- Contextual (environment) attributes
  Attributes that address time, location or other dynamic aspects.

In the IDS, access control is a resource-centric regulation of access requests from subjects (i.e., IDS participants) to resources (i.e., data services). Resource owners define attribute-based access control policies for their endpoints and define the attribute values a subject must attest in order to grant access to the resource. These attributes may include:

- Specific identity of connector(s) (only access requests from a specific connector / specific connectors will be granted)
- Connector attributes (only access requests from a connector that possesses specific attributes will be granted)

- Security profile requirements (only access requests from a connector that fulfills specific security feature requirements will be granted, e.g., having a TPM >= 1.2 and doing application isolation)

The actual access control decision has to be taken within the connector and can be realized using technologies such as XACML or JAAS, depending on the implementation of the connector. The IDS security architecture does not dictate a specific access control enforcement language or implementation.

2.2 Usage Control

In contrast to access control, where access to specific resources (e.g., a service or a file) is restricted, the IDS architecture also supports data-centric usage control. In general, the overall goal is to enforce usage restrictions for data after access has been granted. Therefore, the purpose of usage control is to bind policies to data being exchanged and to continuously control the way how messages may be processed, aggregated, or forwarded to other endpoints. This data-centric perspective allows the user to continuously control *data flows*, rather than *accesses to services*. At configuration time, these policies support developers and administrators in setting up correct data flows.

At runtime, the usage control enforcement prevents IDS connectors from treating data in an undesired way, for example by forwarding personal data to public endpoints. Thus, usage control is both a tool for system integrators to ensure they are not building an architecture that violates security requirements, and an audit mechanism, which creates evidence of a compliant data usage.

The following examples illustrate security requirements that cannot be achieved using traditional access control, but rather require data-centric usage control:

- **Secrecy**
  Classified data must not be forwarded to nodes which do not have the respective clearance.
- **Integrity**
  Critical data must not be modified by untrusted nodes as otherwise their integrity cannot be guaranteed anymore
- **Time to live**
  A prerequisite for persisting data is that it must be deleted from storage after a given period of time
- **Anonymization by aggregation**
  Personal data must only be used as aggregates by untrusted parties. A sufficient number of distinct records must be aggregated in order to prevent deanonymization of individual records
- **Anonymization by replacement**
  Data which allows a personal identification (e.g., faces in camera images)

must be replaced by an adequate substitute (e.g., pixelized) in order to guarantee that individuals cannot be deanonymized from the data.

- **Separation of duty**
  Two data sets from competitive entities (e.g., two automotive OEMs) must never be aggregated or processed by the same service.
- **Usage scope**
  Data may only serve as input for data pipes within the connector, but must never leave the connector to an external endpoint.

It is important to note that the purpose of usage control is to allow the specification of such constraints and enforcing them in the running system. It is a prerequisite to usage control that the enforcement mechanism itself is trusted, i.e. usage control itself does not establish trust in an endpoint. It rather builds upon an existing trust relationship and facilitates the enforcement of legal or technical requirements such as service level agreements (SLA) or data privacy regulations. Thus, users must be aware that usage control will only provide certain enforcement guarantees if applied on highly trusted platforms, such as Trusted Connectors in the Industrial Data Space (see Section 3.2).

Technical enforcement, organizational rules and legal contracts

Usage control should be seen as a machine-readable contract, which is expected to be fulfilled by a party. It is a way to track and trace data as it is used within different systems and to collect evidence of the violation of agreed usage constraints. With that in mind, solutions range from organizational rules or legal contracts to a completely technical enforcement of usage restrictions. For example, an organizational rule could state that the company rules forbid using removable storages such as USB sticks. Similarly, a technical enforcement such as group policies by the windows operating system can prevent the employees from using removable storage media. In some scenarios, we can interchangeably use organizational rules/legal contracts and technical rules. In other scenarios, we can use both enforcement forms to complete each other. In the long term, we assume a substitution of organizational rules/legal contracts by technical enforcement (as illustrated in Figure 5).

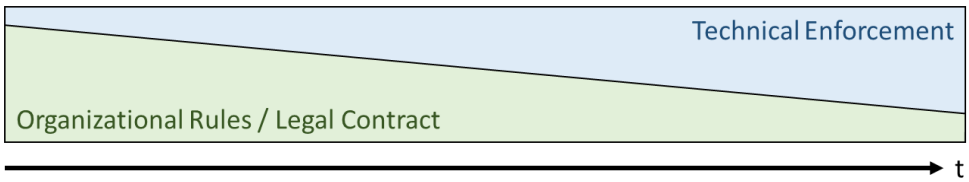

Figure 5 Technical Enforcement vs. Organizational Rules / Legal Contract

We can characterize and implement the enforcement of data usage restrictions in different shapes. Organizational rules or legal contracts can be substituted or

at least accompanied by technical solutions, which introduce a new level of security. Vice versa, technical solutions can be accompanied by organizational rules or legal contracts to support the overall goal achievement (e.g., to compensate missing capabilities of the technical solution).

Although it is a commonly used solution to solve usage control restrictions with organizational rules, we will focus on the technical enforcement in this document.

2.3 Enforcement

For enforcing usage restrictions, system actions need to be monitored and potentially intercepted by control points (i.e., PEPs). These actions are given to the decision engine (i.e., the PDP) for requesting permission or denial of the action. In addition to just allowing or denying the system action, the decision can also require a modification of the action. A PEP component encapsulates the enforcement.

Regarding our accompanying scenario, OEM and supplier demand the deletion of data after a certain time or that only a limited audience can access the sensitive data. Hence, we have to intercept the data flow and check which audience (i.e., processing system) is using the data. For example, the supplier demands the OEM that only the supplier management system can use the data.

2.4 Decision and Information

The enforcement relies on a decision. A Policy Decision Point (PDP) takes the responsibility to answer incoming requests (e.g., system actions) from a PEP with a decision (see Figure 6). The decision-making based on usage restrictions is also called (policy) evaluation. There are several evaluation possibilities such as event- (see Section 4.1) or flow-based (see Section 4.2) approaches.

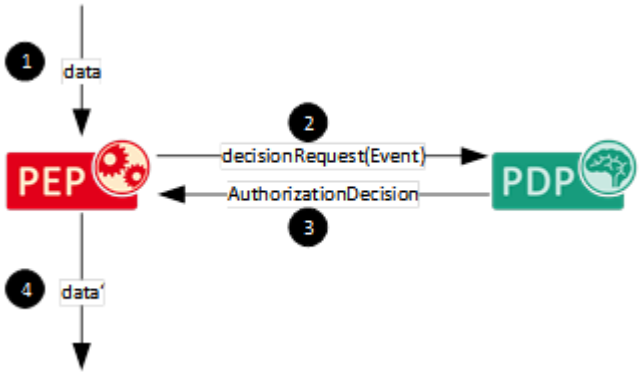

Figure 6 PEP <-> PDP communication

For event-based systems, data usage occurrences are represented as events including attributes to characterize the data usage. The event processing can be differentiated in simple processing (e.g., event-condition-action paradigm) and stream processing (e.g., sliding window) of events. The terms "event stream processing" and "complex event processing" are often used interchangeably.

In our accompanying scenario, we can model the transition of data as event with attributes about the data itself and the recipient. The attributes contain metadata and the target system (e.g., supplier management system). Taking our example from the previous section, the decision engine would draw a deny decision if the target system does not correspond to the expected supplier management system.

The policy decision may also depend on additional information that is not present in the intercepted system action itself. This includes information about contextual information such as data flows or the geographical location of an entity. There is also the possibility for pre- or post-conditions that have to hold before (e.g., integrity check of the environment) and after (e.g., data item is deleted after usage) the decision-making. In addition, there is the possibility to define on-conditions that have to hold during usage (e.g., only during business hours). These conditions usually specify constraints and permissions that have to be fulfilled before, during, and after using the data (see Figure 7).

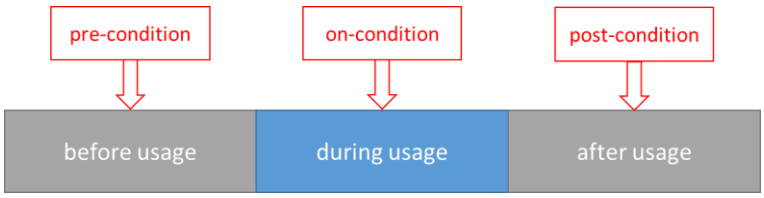

Figure 7 Pre-, On-, and Post-Conditions

A Policy Information Point (PIP) provides missing information for the decision-making. In addition, we can use such a component to get contextual information for or about the intercepted system action (e.g., data flow information, geolocation of the requesting device).

Regarding our accompanying scenario, we may transform the D-U-N-S number [3] of a supplier to a concrete supplier name and address information. For example, if we want to limit the use of data depending on the geolocation of the supplier, a PIP can resolve the D-U-N-S number to a postal address and finally the postal address to GPS coordinates. Supplier and OEM are usually using different part numbers. Therefore, another example for a PIP is the translation of supplier part number to OEM part number and vice versa.

2.5 Specification, Management, and Negotiation

Another important aspect of usage control is the specification and management of usage restrictions. Data providers have to express their restrictions on their data in a more or less formal way. For a technical enforcement, the specification must produce a machine-readable output. The Policy Administration Point (PAP) is the entry point for specification of usage policies, often via a user friendly graphical interface.

In our accompanying scenario, the Collaborative Supply Chain Risk Management (CSCRM) App takes the role of the PAP. There is a version for the supplier and a version for the OEM to specify their data usage restrictions.

A Policy Management Point (PMP) administers the usage restrictions. Hence, the component is concerned with the policy life cycle. This includes the instantiation, negotiation, deployment, and revocation of usage restrictions, as well as conflict detection and resolution.

There are two ways where usage restrictions are placed. First, usage restrictions can be adhered to the data, which is also called sticky policy [4]. Sticky policies are one way to cope with the distribution of the usage restrictions. In this approach, machine-readable usage restrictions (policies) stick to data when it is exchanged. There exist different realization possibilities. Usually, data is encrypted and can only be decrypted when the adherence to the usage restrictions are guaranteed. Second, policies can be stored independently from the data, for instance, in a central component (i.e., a PMP/PRP). In this case, the management component has to take responsibility to exchange the usage restrictions between different systems.

The management of usage policies becomes especially important when exchanging data across system boundaries. Every time data crosses system boundaries, the target system must be prepared for the protection of incoming data, that is, the corresponding policies need to be deployed. The resulting negotiation of policies is also part of the policy management. As enforcement mechanisms can work differently (e.g., work on different system actions) on different systems or technologies, abstract policies can have different instantiations. Hence, usage policies must be instantiated on the target system.

2.6 Related Concepts

There are related concepts to cope with data sovereignty challenges. We present them next:

Data Leak/Loss Prevention

Data Leak/Loss Prevention (DLP) technologies detects and prevents potential data breaches by monitoring sensitive data. Commonly used are Endpoint DLP solutions that run on the client's operating system (e.g., as extension or feature of a security suite). In addition, there are also DLP solutions available that are monitoring the network or access to central storage devices.

Digital Rights Management

The term Digital Rights Management (DRM) is frequently used in the area of protecting digital content from unintended use, modification, and distribution. Different DRM technologies exist to protect multimedia content such as movies (e.g., DVD, Blu-ray), music (e.g., Audio CDs, Internet music), television, or E-books. In addition, there exist DRM technologies to protect digital documents (e.g., MS Word, PDF) within enterprises. This kind of DRM is also known as enterprise rights management (ERM) or information rights management (IRM) and aims to control of access and use of corporate documents.

Windows Information Protection

Microsoft introduced several technologies to establish a comprehensive information protection in their operating system and software such as Microsoft Office (e.g., BitLocker, Windows Information Protection (WIP), Office 365 and Azure Information Protection) [5]. WIP, for instance, is an integral part of Windows 10. Goals of the WIP are to protect data on own devices, to separate private and business data (data separation), to prevent unauthorized access and use (data leakage protection), and to protect data when shared. WIP-protected documents can only be used in WIP-compliant apps. For example, WIP prevents pasting sensitive information (e.g., by using ctrl+c → ctrl+v) to non WIP-compliant apps.

3 Usage Control building blocks in the Industrial Dataspace

This chapter explains which components the Industrial Data Space uses to integrate usage control technologies. The first section presents the important usage control parts of the IDS information model, which adds additional context to the data to enhance decision-making. The following sections are about the Trusted Connector and the Apache Camel interceptor technology. In the last section, we discuss capabilities and limitations of usage control within the Industrial Data Space.

3.1 Information Model

The Industrial Data Space Information Model is a modular metadata-model (ontology) describing, e.g., the capabilities of Industrial Data Space infrastructure components, such as Connectors and the Data Endpoints they expose. Descriptions of data provided by the Data Endpoints are published at dedicated Broker-registries allowing potential Data Consumers to search for and identify data that is relevant (semantics) and applicable (quality) for her particular tasks and to asses in advance its affordability (price) and usability (restrictions).

Extending upon the W3C standard Open Digital Rights Language (ODRL) the Usage Control module provides a machine-readable specification of usage control policies. These specify actions that a party is prohibited (redistribute) or permitted (store) to perform with regard to given a data asset. In addition, they codify any potentially involved duties. Despite a simple core model depicted in Figure 8, ODRL policies are a formal way to declaratively express usage contracts at specification level. In that way, the Information Model achieves a technology-agnostic, consistent representation of usage control policies across the overall Industrial Data Space.

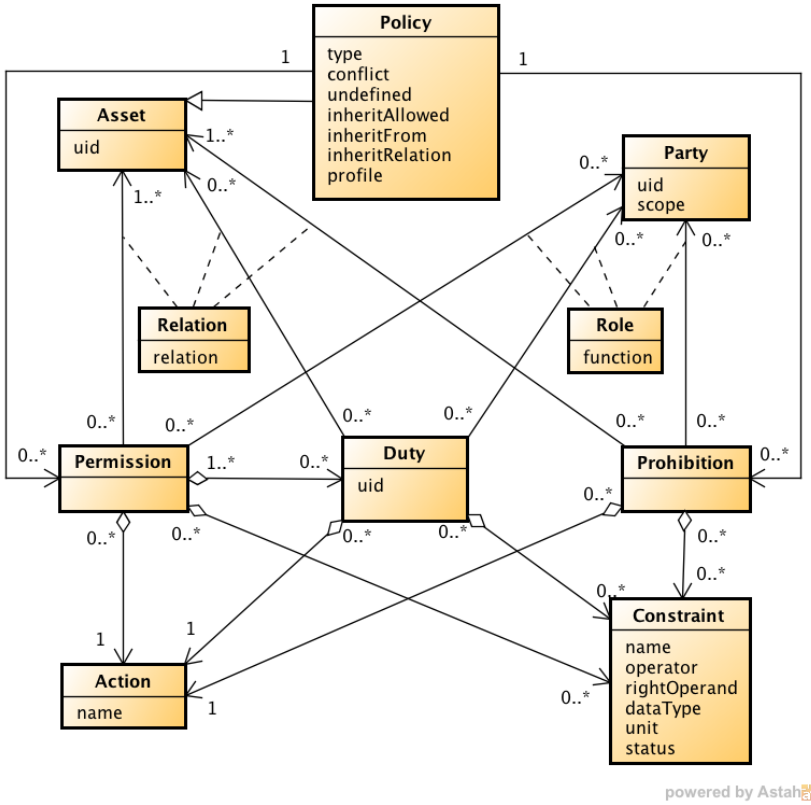

Figure 8 ODRL Core Model 2.1 [6]

In order to implement and enforce the specification-level policies within individual target environments a mapping to organizational and technical measures is required. While the former are out of scope here, the latter approaches involve a variety of additional information sources (PIP) and a tight integration with the host environment (PEP). Here the Information Model enhances ODRL constructs via predefined extension "hooks" to support mapping towards lower level, implementation-oriented policy languages (e.g., IND²UCE XML [7]).

As an example, the ODRL Constraint class expresses logical conditions that govern the applicability of a Rule. Herein an Operator (*eq*) relates the Left Operand (a predicate like *absolutePosition*) to a Right Operand (dynamic or predefined value). On the one side, the Information Model extends the group of predefined predicates [8] in order to support decision-making in particular scenarios of the IDS such as data residency [9], on the other side, it defines a configuration overlay (b) to tie the abstract predicates (a) to operable programming logic supplied by the respective target environment (c), as illustrated by Figure 9.

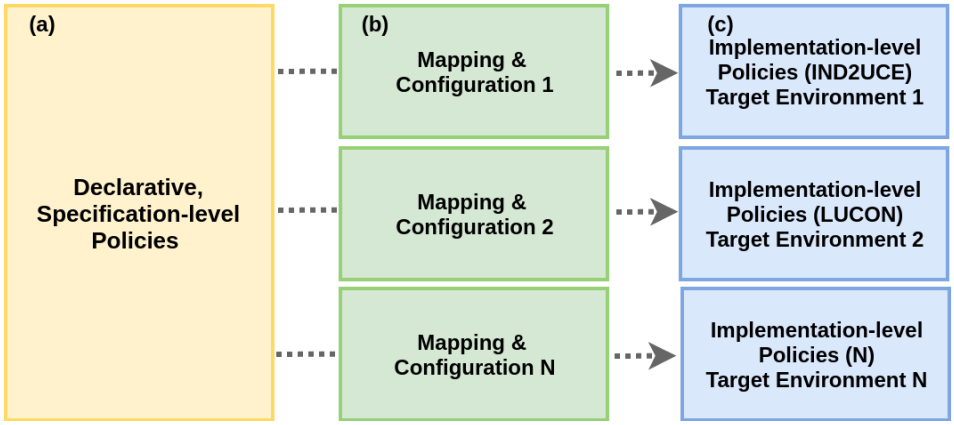

Figure 9 Examples of mapping among policy language levels

Regarding our accompanying scenario, the policy of the supplier "The OEM can only use supplier data for risk or bottleneck management, but not for purchasing or sales purposes" can be represented in ODRL (see Figure 10) and IND²UCE (see Figure 11).

```
<http://example.com/policy:1111>
    a odrl:Offer ;
    odrl:permission [
        odrl:action odrl:use ;
        odrl:target <http://example.com/SpecificOperationOfADataService> ;
        odrl:assigner <http://example.com/Supplier> ;
        odrl:assignee <http://example.com/OEM> ;
        odrl:constraint [
            odrl:purpose ids:RiskManagement;
            odrl:purpose ids:BottleNeckManagement;
        ];
  ];
    odrl:prohibition [
        odrl:action odrl:use ;
        odrl:target <http://example.com/SpecificOperationOfADataService> ;
        odrl:assigner <http://example.com/Supplier> ;
        odrl:assignee <http://example.com/OEM> ;
        odrl:constraint [
            odrl:purpose ids:Purchasing;
            odrl:purpose ids:Sales;
        ];
    ].
```

Figure 10 ODRL Representation

For IND²UCE, we assume to have an event that contains different parameters that represent the target system and the purpose of use.

```
<policy id='urn:policy:ids:1111' description=''>
  <preventiveMechanism>
    <event action='urn:action:ids:enforceExchange'>
      <param:string name='target' value='http://example.com/SpecificOperationOfADataService'/>
      <param:int name='dataID' value='19'/>
    </event>
    <condition>
      <and>
        <or>
          <function:equals>
            <param:string name='purpose'/>
            <constant:string value='ids:RiskManagement'/>
          </function:equals>
          <function:equals>
            <param:string name='purpose'/>
            <constant:string value='ids:BottleNeckManagement'/>
          </function:equals>
        </or>
        <not>
          <or>
            <function:equals>
              <param:string name='purpose'/>
              <constant:string value='ids:Sales'/>
            </function:equals>
            <function:equals>
              <param:string name='purpose'/>
              <constant:string value='ids:Purchasing'/>
            </function:equals>
          </or>
        </not>
      </and>
    </condition>
    <authorizationDecision>
      <allow/>
    </authorizationDecision>
  </preventiveMechanism>
</policy>
```

Figure 11 IND²UCE Representation

In LUCON the message is marked with labels indicating its purpose. A data flow is only allowed, if the purpose is *riskManagement* or *bottleneckManagement* and not either of *sales* or *purchase*.

```
flow_rule {
    id oemSupplierData
    description "The OEM can only use supplier data for risk or bottleneck management,
                 but not for purchasing or sales purposes"
    when {
        endpoint "http://example.com/SpecificOperationOfADataService"
    }
    receives {  purpose(riskManagement) or purpose(bottleneckManagement)
                and not(
                    purpose(sales) or purpose(purchase)
                )
    }
    decide allow
}
```

Figure 12 LUCON Representation

3.2 Trusted Connector

Usage Control only makes sense in an ecosystem where a certain level of trust in the participants can be realized. To enable the establishment of trust relationships, the central technological components used for data processing and

exchange need to be trustworthy. The IDS connector is the central component for data exchange and data processing over the Industrial Dataspace and thus a central component that needs to be trusted (and, for that, it needs to be secure).

The Trusted Connector focusses on security and delivers a trusted platform, incorporating crucial building blocks:

- Identity & Trust management for authenticating communicating parties (e.g., other connectors) and shaping trust relationships between partners.
- A trusted platform as a baseline for secure processing of data.
- Trustworthy communication based on authenticated and encrypted connections.
- Access & Usage control foundations for flexible access control and hooks for usage control frameworks.

Trusted Connector instances enable remote integrity verification, so the integrity of the deployed software stack can be guaranteed before granting access to data.

The Trusted Connector can guarantee a controlled execution environment for data services and can support the creation of trust relationships. A general constraint is one that remains for all deployed IT systems: As long as physical or logical access is granted to administrators, protection against data theft by malicious partners is almost impossible to prevent. We rather see the Industrial Data Space as a network of partners that are provided with the technical means to fulfill their obligations and support in deciding what partners to trust and to define reasonable access conditions.

3.3 Apache Camel Interceptor

A basic element of the Industrial Data Space is the connector [1], which interconnects the different partners and enables data sharing between them. The Trusted Connector (see Section 3.2) is an instance of the base connector that focuses on security aspects such as encryption or remote attestation. Internally, it uses Apache Camel to coordinate the data flow between different systems and applications. From a technical point of view, the developer does this by using pipelining, which is a dominant paradigm of Apache Camel for connecting different nodes in a route definition. The basic idea of a pipeline is that Apache Camel passes the output of one node to the next node as input. Every node in such a route is a processor, except for the initial endpoint (as shown in Figure 13).

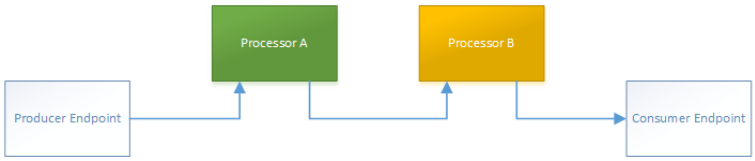

Figure 13: Apache Camel pipeline example

In order to control the usage of data, one approach can be to intercept the flow of data between the services and applications. Figure 14 shows exemplarily how developers can do this. Apache Camel offers the possibility to integrate interceptors that it executes every time before and after a processor is working.

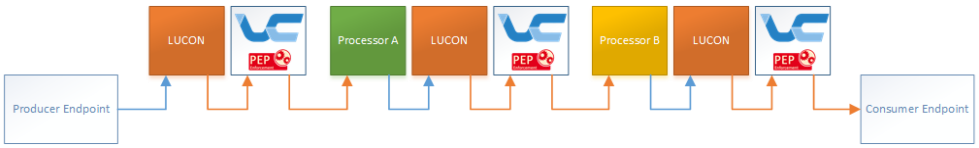

Figure 14: Intercepting Apache Camel data flows

As the Industrial Data Space provides an information model (see Section 3.1), additional metadata enhances the data transferred via the route, which therefore enables a better usage control enforcement. The connector attaches the metadata to the data package, as we explain in Section 3.1. In addition, a PIP is able to resolve more metadata during the decision-making process when necessary.

This paradigm also works across company borders, as data flows always through the IDS connector and Apache Camel respectively (as shown in Figure 15). When reaching the receiving connector, the respective policy to protect the data is automatically instantiated.

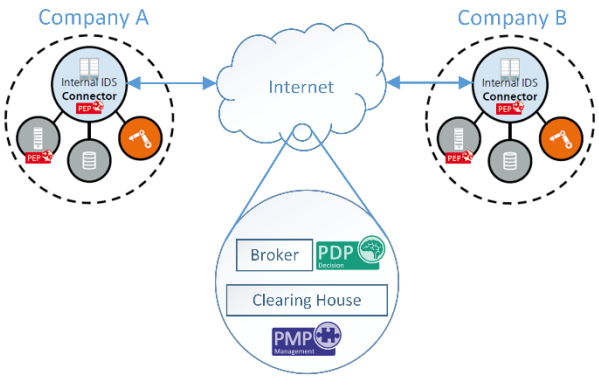

Figure 15: Data flow between throughout company borders

Depending on the policies available, this way of enforcement is not enough to cover all possible use cases and full usage control. We refer the reader to Section 5 for more details and future expansions of usage control.

3.4 Involved Roles in the Usage Control Process

Usage control is a cross cutting technology which involves several roles in the Industrial Data Space.

Broker
In addition to the information who provides which data, the broker might also know the usage restrictions for the respective data, as they are part of the metadata.

Connector
The connector is the main component for implementing usage control. Hence, usage control enhanced connectors such as the Trusted Connector, contain relevant components to perform usage control enforcement (PEPs such as the Apache Camel Interceptor, PDP, PMP). However, PMP and PDP need not to be part of the connector (see Section 4.1). In addition, connectors should provide the technology-dependent policies to the data they provide – for all kind of systems and enforcement technologies that are part of the ecosystem.

Clearinghouse
Provenance Tracking as described in Section 4.3 can track the usage of data and the enforcement of their restrictions. The Clearinghouse is able to use this data later on.

App Store
As we explain in Section 5.4, it is possible that apps take advantage of the usage control technology. The IDS App Store needs to be able to show users, if an app implements usage control.

App Provider
The previous paragraph about the App Store indicates that apps can take advantage of the usage control technology. To do so, App Providers need to implement certain components such as control points (i.e., PEPs) into their application.

4 Usage Control Technologies in the Industrial Data Space

In the Industrial Data Space, several technologies for implementing usage control exist that we present next.

4.1 Integrated Distributed Data Usage Control Enforcement (IND²UCE)

The policy enforcement product IND²UCE (Integrated Distributed Data Usage Control Enforcement) is a technical solution for usage control, developed by Fraunhofer IESE. It provides all mandatory components for providing a comprehensive enforcement of data usage policies in different technical infrastructures and application domains.

In addition to the access- and usage control components already introduced in Section 2, IND²UCE brings another component into use: A PXP. A PDP uses a Policy Execution Point (PXP) during the decision-making process to execute additional actions that not directly relate to an application which implements usage control technology. General usages include issuing a log message, starting a service, triggering a physical alarm lamp or simply executing code to do something.

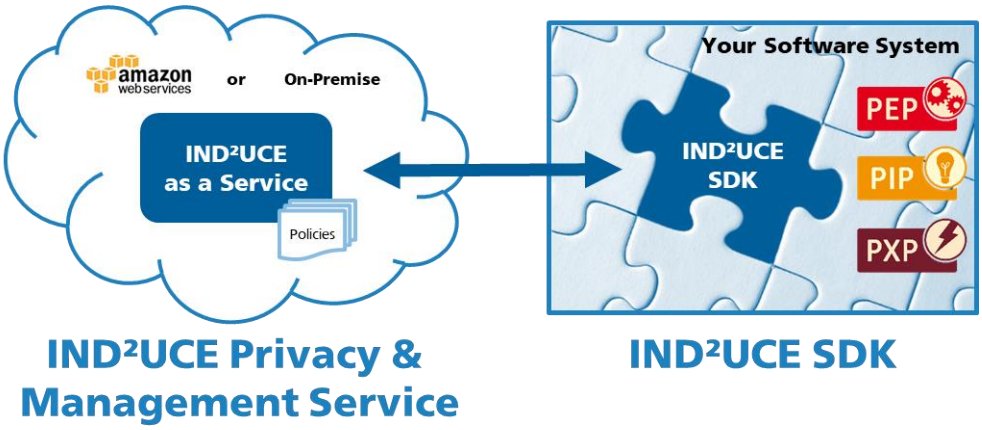

Figure 16 IND²UCE as a Service

Figure 16 presents the two offerings of IND²UCE. The IND²UCE Privacy & Management Service can be used as a cloud service running on Amazon Web Services (AWS) or as On-Premise installation. The service package contains a horizontally scalable decision engine (PDP), a policy and component management (PMP), and a policy editor (PAP) for software developers as target group. The

management component is a multi-tenant web application. The administration is done via a web interface that supports different roles and views.

In addition to the IND²UCE Privacy & Management Service, an open source SDK and examples exist for an easy integration of PEP, PXP und PIP components. It allows software providers to integrate usage control capabilities into their existing software products. For further information about the SDK and the policy language, please refer to the developer's webpage (https://dev.ind2uce.de/).

There is also the possibility to run the IND²UCE Privacy and Management Service on premise within the IDS connector.

IND²UCE in the accompanying scenario

For implementing usage control with the IND²UCE framework, we develop a PEP that is controlling the data flow of nodes communicating along an Apache Camel route in the IDS connector (see also Section 3.3). The PEP intercepts all message flows, extracts information from the data (i.e., metadata [see also Section 3.1]) and creates an event representation that is sent to the PDP. As the PEP can detect, to which endpoint the data is transferred (see also Section 3.3), we are able to enforce the policy, which constraints the selection of an eligible target system. The Apache Camel Interceptor (i.e., the PEP) enforces the decision of the PDP.

In addition, we need a policy execution point (PXP) to initiate the deletion of data. In our scenario, the PXP has to interact with the supplier management system DBMS of the OEM to fulfill the obligation to delete data after fourteen days. Therefore, the PDP supplies the component with the data location and the time to delete. When the time is reached, it automatically deletes the data. It is assumed that the connected DBMS supports this operation and that it provides an interface to delete all representations of the data subjected to a usage policy.

Figure 17 illustrates the integration of IND²UCE components into the application scenario. There is a PEP in the IDS connector, a PXP to interact with the DBMS and the management (PMP) and decision (PDP) component as part of the connector. Both can also be outsourced to a cloud service as mentioned earlier.

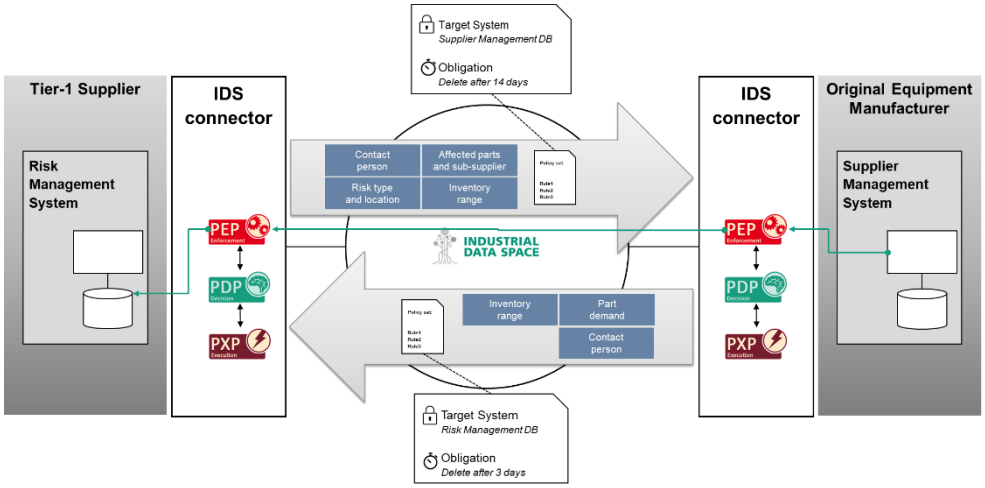

Figure 17 Exemplary integration of IND²UCE components in the accompanying scenario

4.2 Label-based Usage Control (LUCON)

LUCON is an alternative approach on data flow control, which is supported by the Industrial Data Space and is the default mechanism delivered with the open source version of the Trusted Connector. While IND²UCE is a comprehensive policy framework focusing at central management and enforcement of obligations, LUCON is a lightweight usage control framework, which focuses on controlling the flow of data between different endpoints. This is relevant in several scenarios, e.g. when data from competing companies must not be mixed (separation of concern), when privacy constraints must be enforced (anonymization of personal data) or when obligations must be bound to data flows.

By attaching "labels" to messages and modifying them along the message route, it is possible to classify data based on its origin and content and to capture data flows in the connector and between endpoints. Policies determine valid and invalid data flows and bind messages to obligations. They are evaluated against the service descriptions from the semantic Information Model so that policies can be written against any kind of service property that is already available in the model. When obligations are bound to remote messages, LUCON requires the remote peers to enforce them and builds upon the trust relationship established by remote attestation.

Besides policy enforcement at runtime, LUCON also allows to upfront validate usage control policies against Apache Camel message routes. When users install message routes in the connector, LUCON checks if and under which conditions the route may violate a policy (which would block the respective message at runtime) and shows an explanation to the user. This supports users in setting up correct routes and policies, but it also allows to generate auditable proofs that the systems complies with a policy.

LUCON in the accompanying scenario

In the accompanying scenario, LUCON can enforce usage policies. Example policies are:

- Policy 1 (Supplier to OEM): Any message labeled as "*sub-supplier information*" must be sanitized before being sent to "*external*" endpoints (i.e., connectors not owned by the supplier). Sanitizing in this context means removal of all information exposing the identity of the sub-supplier.
- Policy 2 (OEM to supplier): Any message labeled as "*part demand*" or "*inventory range*" must:
  - Not be sent to any external party other than the requested supplier (identified by its certificate). That is, suppliers must not forward this request to outside of their trust domain.
  - Not be sent into a services which are described by type "storage" or "persisting". That is, suppliers may use the critical data to answer a request on the fly, but must not persist it in a message queue, data base, or any other type of storage. The service description refers to properties stated in the semantic Information Model.

These policies combine the trust level established by remote attestation, the service descriptions from the semantic Information Model, and the Apache Camel message routes. A Camel interceptor and a PDP enforce them at runtime. In addition, users at both sides can confirm a priori that their Camel routes will not violate the policies.

4.3 Data Provenance, Transparency and Accountability

The Scientific Computing Community has introduced data provenance tracking approaches in order to make the derivation of research results traceable. Later on, legal scholars from the areas of compliance and data protection have recognized the benefit of these approaches, since the provenance of (personal) data can also be used to verify the adherence to business processes, procedural instructions, and data protection regulations.

Implementing provenance tracking requires that data flows between entities of interest are being monitored. Depending on the granularity of these entities of interest, different points of integration and different technologies for monitoring come into question. For example, fine-grained state-based data flow tracking as required for gapless usage control or also more coarse-grained approaches such as LUCON (see section 4.2).

In any case, data provenance tracking collects and stores metadata about local data items, their source entity and destination entities at each entity of interest. By this means, as illustrated in Figure 18, a partial tree of the descent of each

data item is persisted on each entity. In order to obtain the entire history of a data item being monitored, a provenance collection step pulls and aggregates these partial trees from all systems within a provenance domain, such as the Industrial Data Space. Authorized users, i.e., data owners, can trigger provenance collection from within a provenance dashboard, which will subsequently visualize all transactions of the considered data item and which can also be extended to correlate actual data flows with expected flows according to business processes, regulations, etc.

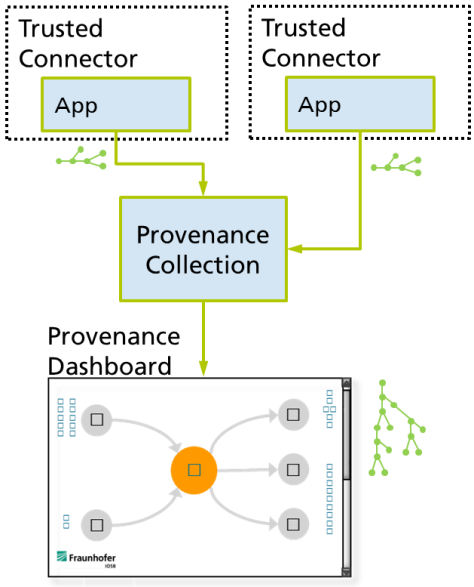

Figure 18 Conceptual integration of data provenance tracking

Provenance Tracking in the accompanying scenario

In the accompanying scenario, the provenance dashboard could serve as an attestation service, which shows that the data provided by the supplier only flowed into the OEM's supply chain risk management (SCRM) and nowhere else. This information could be shared with any party with rightful interests without revealing the sensitive data itself.

5 Discussion of Usage Control in the Industrial Data Space

This section discusses what usage control in its current state can achieve in the IDS and therefore summarizes the capabilities and limitations already mentioned in the previous sections. In addition, usage control has also implications for the IDS and affects various components, which is part of the discussion in the last sections.

5.1 Capabilities

By using the current state of usage control that is implemented into the IDS it can support developers and administrators in setting up correct data pipes that comply with policies and do not leak data via side effects. For example, usage control prevents IDS connectors from treating data in an undesired way like forwarding personal data to public endpoints. In addition, usage control in the IDS can also be used as an audit mechanism, which creates evidence of a compliant data usage. For instance, usage control mechanisms can monitor and log usages of data.

With usage control, it is possible to modify the messages exchanged between endpoints to comply with a policy. For example, personal data can be removed or data can be aggregated. It is furthermore possible to change the route of the package or drop it completely if demanded by a policy. Moreover, apps running on the connector can implement PEPs, which connect the usage control infrastructure and further enhance the functionality by allowing a more detailed control and data flow tracking.

5.2 Limitations

Usage control does only work within its ecosystem where it has the full control over the data. Achieving full control does also mean that there are cases that expect developers to integrate usage control components (such as PEP, PIP, PXP) into their application or services to fulfill usage restrictions. In most cases, developers have to integrate at least the PEP component.

Although usage control uses several abstraction layers, there will always be a possibility to circumvent the system. One of the best-known examples for that is media disruption. For example, a usage control system may control taking screenshots and printing, but it cannot prevent a person to take a photo from the screen displaying the sensitive data. That said, if data leaves the ecosystem, it needs additional protection (such as encryption) in order to keep control over the data.

Usage control is no hard security feature such as cryptography, which one can mathematically prove. It is rather a complementary solution to have more fine-grained control over data flowing in a system and goes well together with organizational rules (see Section 2.2). In addition, it is rather an extension to access control than replacing it.

Implementing a usage control technology does not automatically establish trust in an endpoint. It rather builds upon an existing trust relationships such as existing contracts and a secure computing environment like highly trusted platforms (like the IDS Trusted Connector, described in section 3.2).

When physical access is granted to administrators, protection against data theft by persons with malicious intents is almost impossible to prevent. It is part of future work to evaluate possible countermeasures.

5.3 Implications

Implementing usage control into an existing system has various implications. Creating events, the decision-making and the transfer of events between the affected components takes extra time as well as some computational power. Besides, all usage control components need memory to persist information or to perform the computation. In sum, it will reduce the performance of the overall system and demands machines with more power.

As already stated, the basic idea of usage control is to control the dataflow. In a case where a developer enhances an application with usage control technology, he needs to integrate at least one PEP. Depending on the complexity of the enforcement, he needs to integrate even more than one PEP within one or several applications. As all of those integrations also need planning and testing, it increases the development and testing time and effort in comparison to a system without usage control.

In addition to the enforcement components, the system needs policies. Therefore, a policy specification process needs to be established. During this process, the policy experts of the data owner have to collect information about how others should use the data. This process costs additional time and communication effort for the data owners and leads in the end to higher costs.

5.4 Stages of Usage Control Enforcement

Usage control can be implemented in different ways. The solutions range from organizational rules or legal contracts to complete technical enforcement of usage restrictions (cf. Section 2.2). Intermediate levels may contain parts of both enforcement manifestations. We will describe a transition of enforcing usage

restrictions from organizational rules/legal contracts to a complete technical enforcement that we align to our accompanying application scenario.

No technical Usage Control

We can express usage restrictions as organizational rules or as part of a legal contract between two companies. In this case, we have no technical measure, but may enforce some violations by disciplinary penalty or lawsuit. Regarding our accompanying scenario, there is a legal contract between the two companies stating that the exchanged data can only be used in the specific target systems (i.e., supplier management or risk management) and that data must be deleted after a certain time. For violations, the contract states fines that are a multiple of the total contract value. In this case, we have only organizational measures to enforce usage restrictions.

Simple technical Usage Control (without 3rd Party Software)

We accompany these organization rules and legal contracts by technical measures. Regarding the legal contract, the companies may grant each other the right to perform a security audit, which includes checking the data usages. By using the IDS infrastructure, we can control to which target system the data is flowing by using the PEP in the IDS connector. In addition, a PXP demands the deletion of data after a certain time. We demand from the risk management system that it provides an interface for triggering the deletion of specific data items. Figure 19 presents the data-flow from the supplier management system to the risk management system database, which is controlled by the PEP in the connector and the data deletion request to the database, which is triggered by a PXP from the IDS connector.

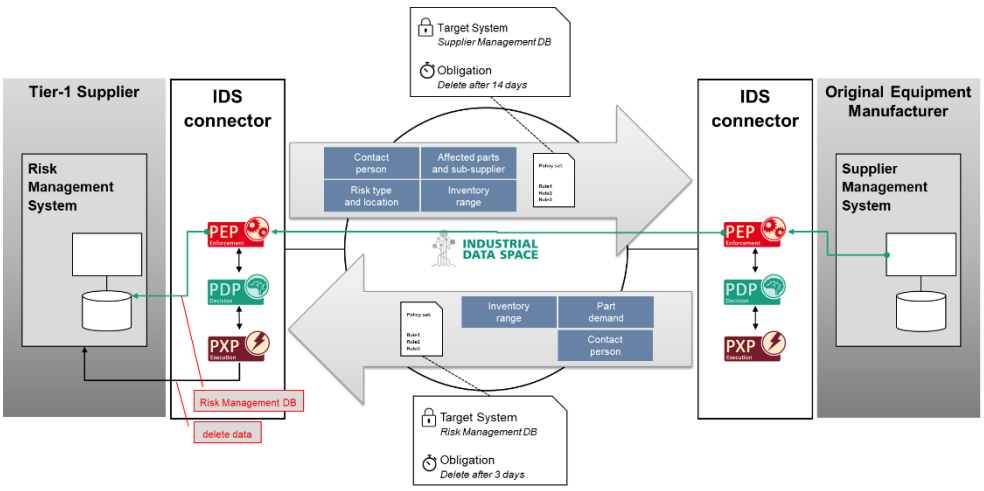

Figure 19 Data Exchange (PEP and PXP in the IDS Connector)

Usage Control integrated in Storage Infrastructure (3rd Party Software)

We further elaborate our technical solution and integrate a PEP into the databases. In doing so, we can control data-flows to and from the database. It allows us to control the flow of information to specific applications rather than only controlling the flow to the database. In addition, we can restrict the data usage based on time constraints (e.g., data is only allowed to be used three days after the insert operation). Moreover, we place the PXP for deletion directly into the database and do not demand a dedicated interface for fulfilling this obligation. Figure 20 presents our extensions and the additional enforcement components that we placed at the database (i.e., storage infrastructure).

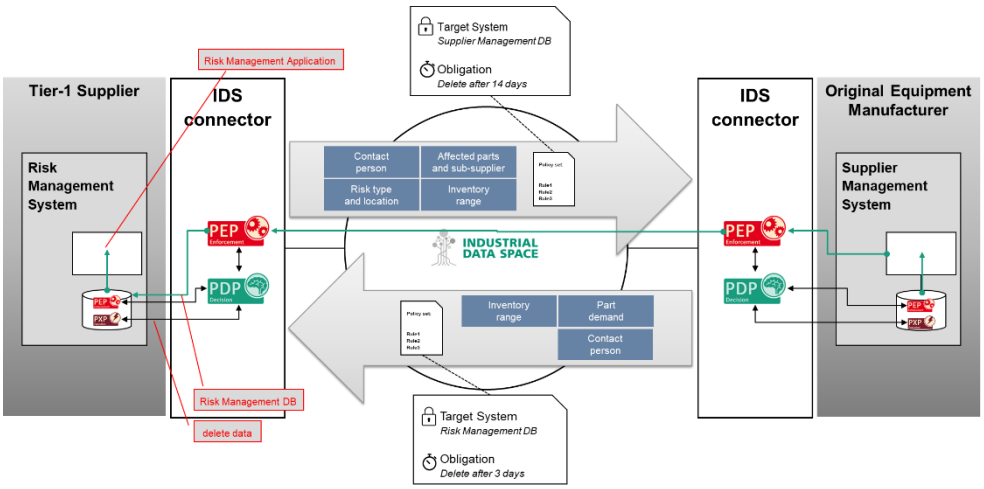

Figure 20 Storage Infrastructure (i.e., additional PEP and PXP in the DB)

Usage Control integrated in Storage Infrastructure and App API

However, there are points in the system, where data may leak. A user interacting with the risk management system may use the system to export or dump the data into a file. Hence, we have to integrate additional enforcement points that are controlling the interfaces of the risk management system. They allow us to provide fine-grained data-flow control on the application level. For example, the PEP can modify data in transit or limit the amount of displayed information controlled by security policies. Figure 21 presents our extension at the risk management system API (i.e., application level).

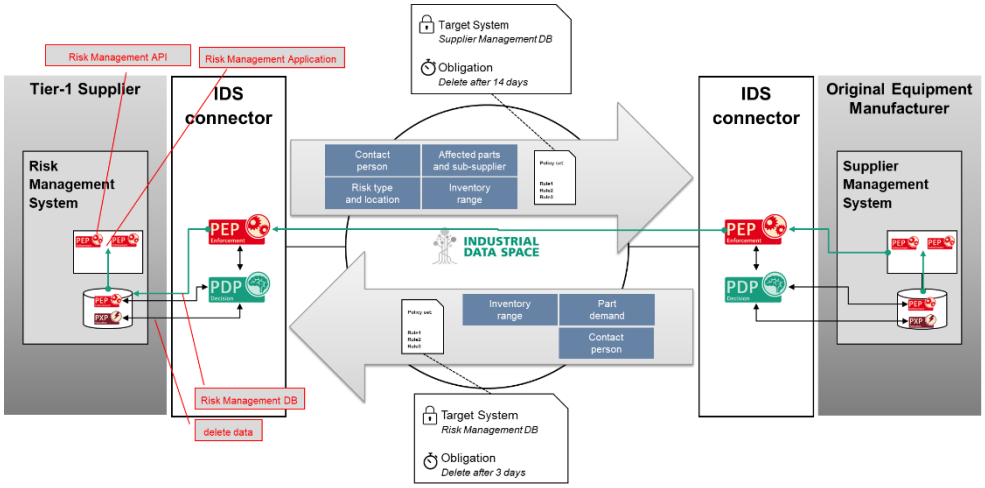

Figure 21 Application (i.e., additional PEPs in the application)

Usage Control integrated in Storage Infrastructure, App API, and Clients

However, there are still possibilities how data can may leak. The system users may print the content or they create a screenshot of the data. Usage control technologies can also prevent such behavior by controlling the operating system. For example, a PEP may prevent taking screenshots or printing of sensitive data. Figure 22 presents the integration of enforcement components into client systems such as a desktop computer.

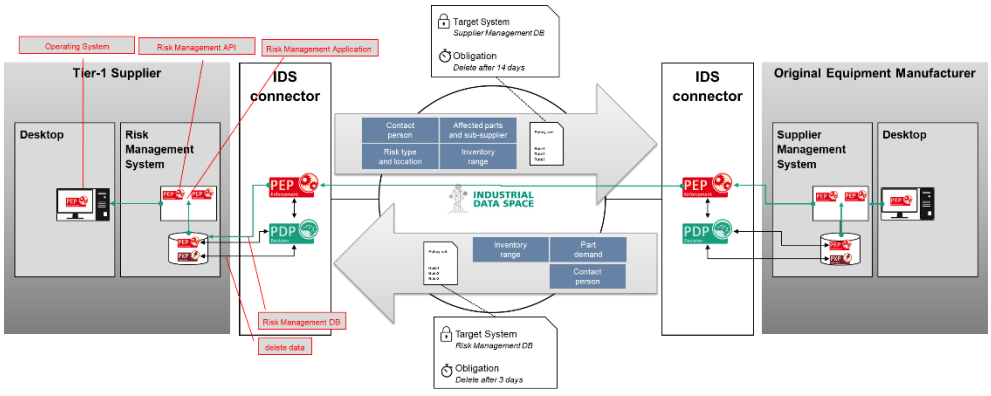

Figure 22 Client Systems (additional PEPs in the client's operating system)

Remaining Risk: side channels with no possible technical control

Finally, there are still possibilities how data may leak For example, instead of making a screenshot, the users could take a picture of the screen by using a mobile device (i.e., external system or media disruption). Hence, we cannot

achieve a perfect and comprehensive protection of data, but we can put controls to the system to reduce the possibilities for potential misuses. Regarding the last improvement stage, an organizational rule could state that taking any picture within the company is prohibited.

Conclusion

To conclude, the enforcement of data usage restrictions can be characterized and implemented in different shapes. Organizational rules or legal contracts should be accompanied by technical solutions; and vice versa, technical solutions should be accompanied by organizational rules or legal contracts to support the overall goal achievement.

To implement a comprehensive usage control, we have to integrate control points into different systems and abstraction layers (see Figure 23) that are working together to achieve the overall goal of data sovereignty.

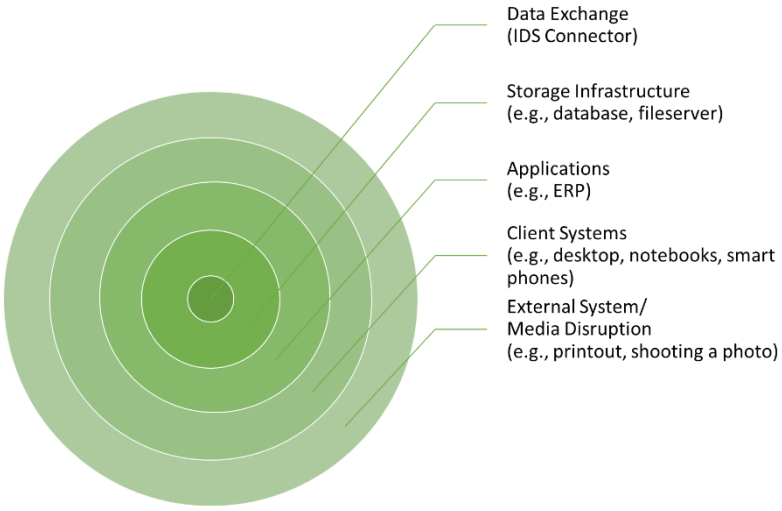

Figure 23 Comprehensive Usage Control across Systems and Abstraction Layers

References

[1] B. Otto, S. Lohmann, S. Auer, G. Brost, J. Cirullies, A. Eitel, T. Ernst, C. Haas, M. Huber, C. Jung, J. Jürjens, C. Lange, C. Mader, N. Menz, R. Nagel, H. Pettenpohl, J. Pullmann, C. Quix, J. Schon, D. Schulz, J. Schütte, M. Spiekermann and S. Wenzel, "Reference Architecture Model for the Industrial Data Space," Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. AND Industrial Data Space e.V., München, 2017.

[2] Standard OASIS and O. Standard, "eXtensible Access Control Markup Language (XACML) Version 3.0," 22 January 2013. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html. [Accessed 25 October 2017].

[3] 27 09 2017. [Online]. Available: https://en.wikipedia.org/wiki/Data_Universal_Numbering_System.

[4] M. C. Mont and S. Pearson, "Sticky Policies: An Approach for Managing Privacy across Multiple Parties," *Computer,* pp. 60-68, 09 September 2011.

[5] N. Mercer, "Introducing Windows Information Protection," Microsoft, 29 June 2016. [Online]. Available: https://blogs.technet.microsoft.com/windowsitpro/2016/06/29/introducing-windows-information-protection/. [Accessed 25 October 2017].

[6] R. Iannella, S. Guth, D. Paehler and A. Kasten, "ODRL Version 2.1 Core Model," 05 03 2015. [Online]. Available: https://www.w3.org/community/odrl/model/2.1/. [Accessed 28 09 2017].

[7] Fraunhofer IESE, "IND2UCE Policy Language Documentation - Version 3.0.25," 26 04 2017. [Online]. Available: http://dev.ind2uce.de/3.1.33/policyLanguage/language.html. [Accessed 28 09 2017].

[8] R. Iannella, M. Steidl, S. Myles and V. Rodríguez-Doncel, "ODRL Vocabulary & Expression - 3.14.9 Left Operand," 26 09 2017. [Online]. Available: https://www.w3.org/TR/odrl-vocab/#term-LeftOperand. [Accessed 28 09 2017].

[9] The Object Management Group, "Data Residency Working Group," [Online]. Available: http://www.omg.org/data-residency/. [Accessed 28 09 2017].

Document Information

Title:	Usage Control in the Industrial Data Space
Date:	December 2017
Report:	IESE-056.17/E
Status:	Final
Distribution:	Public Unlimited