

HORIZON EUROPE PROGRAMME

TOPIC HORIZON-CL5-2021-D3-03

Demonstration pilot lines for alternative and innovative PV technologies

(Novel c-Si tandem, thin film tandem, bifacial, CPV, etc.)

GA No. 101084046

Digitalised pilot lines for silicon heterojunction tunnel interdigitated back contact solar cells and modules



PILATUS

PILATUS - Deliverable report 5.1

Report on the governance and design choices for a data space suitable for collaboration along the PV value chain



Funded by the
European Union

Deliverable No.	PILATUS D5.1	
Related WP	5	
Deliverable Title	Report on the governance and design choices for a data space suitable for collaboration along the PV value chain	
Deliverable Date	2023-10-31	
Deliverable Type	REPORT	
Dissemination level	Public (PU)	
Author(s)	Renaud Langou (CSEM), Baptiste Schubnel (CSEM), Mathias Klinger (MBG), Andreas Waltinger (MBG), Christian Guhr (MBG), Christopher Berge (ISRA VISION), Damien Lachenal (MBR), Rainer Grischke (MBCH)	
Checked by	Baptiste Schubnel	2023-10-25
Reviewed by (if applicable)	All consortium members	2023-10-16
Approved by	Anna Molinari (UNR) – Project coordinator	2023-10-26
Status	Final	2023-10-25

Document History:

Version	Date	Editing done by	Remarks
V01	09/10/2023	All authors	First version ready for review
Final	25/10/2023	Baptiste Schubnel (CSEM)	Incorporated review feedbacks from Christopher Berge (ISRA VISION, technical reviewer), Piter Miedema (UNR), Christian Guhr (MBG)

Deliverable Background:

The present deliverable is part of PILATUS WP5 entitled *Digitalization, Automation, Industry 4.0* whose objective is to drive the implementation of Industry 4.0 concepts in the context of PILATUS by developing the right digital infrastructure and processes. This deliverable deals specifically with the first objective of WP5 – to set up a secure infrastructure for data collection and exchange for carrying out Industry 4.0 and automated process optimization tasks.

Publishable summary

This document elaborates on the design, implementation, and testing of a data space connector tailored for the solar industry within the PILATUS European project.

Deliverable Context

The PILATUS project involves several stakeholders from the European PV industry, including wafer manufacturers, cells and modules producers, service providers, and applied research institutes. The project's objective is to set up pilot lines for heterojunction interdigitated back contact (IBC) cells and modules in Europe, integrating cutting-edge analysis and Industry 4.0 features. The implementation, training and testing of Industry 4.0 and causal analysis tools necessitates extensive data exchange among participants, spanning manufacturers, testing facilities, research institutes, and service providers. It therefore necessitates the setup of an adequately secured data exchange infrastructure.

Deliverable Achievements

The present deliverable documents the outcomes of task 5.1 entitled “Set up of digital infrastructure for secure data exploitation”. In this task, CSEM has led the definition, implementation, deployment and testing with PILATUS partners of a data space tailored for the project needs and the wider solar industry. The deliverable describes the business scenarios involved in the project, and explains the design choices made based on them following the International Data Spaces Association (IDSA) reference architecture [1]. It defines a suitable vocabulary for data exchange related to cell and module manufacturing, defects, and performance measurements. It also defines a suitable implementation of the IDSA data space based on IDSA data space connectors, and a series of tests and standard files and protocols for connector deployment and testing (docker compose files, python scripts, testing protocols). Deployment tests results are documented as well as lessons learned, and challenges encountered by project partners in deploying the connectors.

Deliverable Structure

The introductory section emphasizes the need of a data space as an alternative to other data exchange solutions for the PILATUS project.

Section 2 documents the design choices made in the PILATUS project for the data space following standard IDSA reference architecture and its five-layer hierarchical approach: the business layer (roles, rules, and identities), the functional layer (trust and security, interoperability, apps), the process layer (data flow), the information layer (in particular vocabulary for data exchange), and the system layer (software and hardware requirements).

Section 3 dives into the practical implementation of the connectors (tests and choice of the most relevant available open-source connector), the tests performed, and the deployment of connectors on partners infrastructure. It outlines the lessons learned for future projects involving data spaces in the solar industry or in a similar setting.

Contents

1	Introduction.....	7
1.1	PILATUS data sharing needs	7
1.2	Data exchange: common methods and limitations	7
1.3	Overview of the IDSA reference architecture standard and the connectors.....	8
2	Design choices for the PILATUS data space.....	10
2.1	Business layer	10
2.1.1	Overview.....	10
2.1.2	PILATUS project	10
2.1.3	Generalization to the Solar industry	12
2.2	Functional layer	12
2.2.1	Overview.....	12
2.2.2	PILATUS project	12
2.2.3	Generalization to the solar industry.....	13
2.3	Process layer.....	13
2.3.1	Overview.....	13
2.3.2	PILATUS project	14
2.3.3	Generalization to the solar industry.....	15
2.4	Information layer.....	15
2.4.1	Overview.....	15
2.4.2	PILATUS Project: Data models and metadata	15
2.4.3	PILATUS Project: Common Vocabulary	16
2.5	System layer	26
2.5.1	Overview.....	26
2.5.2	PILATUS project	26
2.5.3	Generalization to the solar industry.....	27
3	Data space implementation and tests.....	28
3.1	Connector choice: deployments and tests at CSEM	28
3.1.1	Available connectors and pre-selection	28
3.1.2	Deployment and tests at CSEM	30
3.1.3	Tests results and final connector choice	32
3.2	Implemented features on top of the dataspace connector.....	32
3.2.1	GUI.....	32
3.2.2	Scripts and APIs for automated transfer	33
3.3	Partners' deployment.....	34
3.3.1	Exchange scenarios.....	34
3.3.2	Providing code and guidance to partners	35
3.3.3	Testing phase with partners: lessons learned.....	35
3.4	Contribution to project objectives	36
3.5	Contribution to major project exploitable result.....	36
4	Conclusion and future steps.....	37
5	Risks and interconnections.....	38

5.1	Risks/problems encountered	38
6	Deviations from Annex 1	39
7	References.....	40
8	Acknowledgement.....	41

List of Figures

Figure 1: Illustration of the core implementation concepts of the data space for the solar industry: Data sovereignty (represented by the broker agreement and the identity trust store), Decentralization (connectors are running locally by each partner) and the connectors (software that permits data exchange). 9

Figure 2: Industry 4.0 scenario. Data from a MES or from a machine are provided to a research institute for service development via connectors. Data are used for training, evaluation, and testing. The developed solution can then be directly deployed on site in case of low latency requirements..... 11

Figure 3: Reliability/process optimization scenario. Data are exchanged between more than two actors (for instance, EURAC, CSEM and Meyer Burger). There are no stringent latency requirements, and results of the analyses can be provided back to the manufacturer via connectors. 11

Figure 4: Broader scope for data exchange. The service provider develops a service that monitors field performance data and production line metrology data. It provides services for reliability assessments (both to asset owner and manufacturer), and services for reliability/ manufacturing process improvement to the manufacturer based on metrology and performance field data..... 12

Figure 5: PILATUS dataflow for Industry 4.0 applications. Arrows direction indicates a “get” operation. Blue coloured arrows outline the flow from the user GUI to the results of the analysis. 14

Figure 6: PILATUS Dataflow for cross linking performance field measurements and production line metrology. Arrows direction indicates a “get” operation. Blue coloured arrows outline the flow from the user GUI to the results of the analysis. The two arrows from “data integration and merging” to the Research Institute connectors mean that the analysis is using both manufacturer and Research Institute data. 14

Figure 7: Typical API response format..... 15

Figure 8: JSON template file for metadata description derived from OPC UA standard. 16

Figure 9: Real Technical layer implemented in the PILATUS project, with security enhancement via IP restrictions and DMZ (Demilitarized Zone, a subnetwork presenting external-facing services online) deployment. 26

Figure 10: First data exchange test: data exchange without central authentication 31

Figure 11: Second data exchange test: data exchange with central authentication enabled. 31

Figure 12: GUI for the PILATUS Connectors 32

Figure 13: Python scripts to add and retrieve data from the data space via connectors..... 33

Figure 14: Typical exchange scenario between Meyer Burger and CSEM 34

List of Tables

Table 1: Partners roles and responsibilities in The PILATUS project.....	10
Table 2: PV systems general terms	17
Table 3: Databases and MES general terms.....	18
Table 4: Cell manufacturing terms.	18
Table 5: Modules manufacturing terms.....	20

Table 6: Vision systems and inspections terms.....	21
Table 7: Field measurements and performance assessment terms.	23
Table 8: Description of the Eclipse dataspace connector (Source: [10])	28
Table 9: Description of the Dataspace Connector (Source: [10]).....	29

Abbreviations & Definitions

Abbreviation	Explanation
(s)FTP	(secured) Files Transfer Protocol
API	Application Programming Interface
AWS	Amazon Web Services
DAPS	Dynamic Attribute Provisioning Service
DMZ	Demilitarized Zone (Networks)
DoS	Denial of Service
DSO	Distribution System Operators
EDC	Eclipse dataspace connector
FTP	File Transfer Protocol
FZU	Fyzikální ústav (Institute of Physics of the Czech Academy of Sciences)
GUI	Graphical User Interface
IBC	Interdigitated back contact
IDS	International Data Spaces
IDSA	International Data Spaces Association
IP	Internet Protocol
IPR	Intellectual Property Rights
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
MES	Manufacturing Execution System
MITM	Man in the Middle
ODBC	Object Database Connectivity
OPC UA	Open Platform Communications Unified Architecture
REST	Representational State transfer
SaaS	Software as a service
SFTP	Secure FTP (File Transfer Protocol)
SQL	Server Query Language

1 Introduction

1.1 PILATUS data sharing needs

The PILATUS project aims to establish digitalized pilot lines for interdigitated back contact (IBC) solar cells and modules. Within WP4 and WP5, CSEM and EURAC are developing analytical tools and machine learning algorithms that will be integrated into the cell and module pilot lines, as well as to the outdoor setups. The developments include:

- *Outdoor Performance and reliability analysis*: data analysis and automated diagnostics of all facilities (EURAC, FhG, FZU) outdoor performance data. This work is carried out in T4.4
- *Data analytics for cell processing optimization*: development and deployment of causal analytics methods for optimizing cell materials and cell processing inputs. This work is carried out in T5.2.
- *Data analytics for module line optimization*: development and deployment of algorithms for module production, including equipment predictive maintenance and parameter optimization. This work is carried out in T5.3.
- *Large scale analysis from cell manufacturing to PV system performance*: development of algorithms to link outdoors performance tests and production line quality tests. This work is carried out in T5.4.

These developments require data exchange between data owners (Meyer Burger, ISRA VISION, EURAC, FhG, FZU) and data users (EURAC, CSEM) within the PILATUS Consortium. Task T5.4 is particularly demanding in terms of data exchange, as outdoor field measurements and production data must be treated together. Another important point is that, although the production line algorithms from T5.2 and T5.3 might operate locally and might not need data transfer between CSEM and Meyer Burger once deployed (for instance, on a dedicated computing machine at the production line without internet connectivity for the cell and module analyses), data sharing is essential for all algorithms during the development and testing phases. To ensure this exchange is done in a secure and sovereign manner, T5.1 consists in the set-up of a digital infrastructure for secured data exploitation and exchange. The infrastructure is based on the IDSA reference architecture. Before coming to the principal characteristics of the architecture, we first review common data exchange protocols and their drawbacks in Section 1.2.

1.2 Data exchange: common methods and limitations

We give below a short overview of common data exchange protocols and highlight their limitations in terms of security and data exchange capabilities.

(S)FTP ((Secured) file transfer protocol): Data is directly transferred from a server (data provider) to a client (data consumer). The secured protocol version uses SSH (Secure SHell) to ensure communications and data are encrypted during transfer. Default FTP is not encrypted and hence vulnerable to many security issues, like Man in the Middle Attack (MITM) [2]. Its secure version suffers known lack of standardization and compatibility issues and are often blocked by firewall or proxies. Moreover, FTP lacks scalability for big data scenarios and is by design made for file exchange, not for dynamic real-time/ near real time data sharing or database data sharing.

Emails: Data are sent to one (multiple) partner(s) via email service providers. This way of sharing contents limits the size, frequency, standardization, flexibility, and type of data that can be exchanged (no-real time). Moreover, it is sensitive to data breaches (e.g., via server hacking) and to wrong

manipulations that can lead to disclosure of sensitive data (e.g., sending sensitive content to a wrong recipient).

Cloud Storage: Data are stored and/or duplicated on a central cloud storage solution like Google Drive, Dropbox, Microsoft Cloud, Amazon S3 or one of the databases provided by cloud services. The host partner allows the authorized PILATUS project partners to access these data. This solution is centralized and depends on third-party providers, which creates risks of lock-in and uncertainty due to the fluctuating policies of such providers for data encryption [3].

APIs (Application Programming Interfaces): Real-time or batch data exchange between different computer programs. Public-facing APIs are vulnerable when not properly secured (SQL injection, DoS, stolen authentication [4]) and are commonly attacked by hackers for data theft [5].

Web portals: Web interfaces where data can be uploaded and downloaded securely. Examples of such platform are SharePoint from Microsoft or Rubrik. These solutions do not allow real-time/ database data exchange and are centralized by essence. Web portals like SharePoint are also vulnerable to security exploits that may lead to data leaks [6].

Physical storage: Data is exchanged on hard disks or USB drives. This way of exchanging data is non-reproducible and impractical if exchanges happen multiple times. Moreover, data theft might happen if the device is stolen, and malware and viruses can be transmitted from one company to the other when plugging in the devices.

VPNs (Virtual Private Networks): Data providers set up an access to their internal network via VPN to the data consumers. This way, the latter gain direct access to databases or servers where the shared data are stored. Risks in providing access to internal networks to external companies include loss of control and data breaches: when many partners get access to the network, it becomes harder to manage access rule and to ensure that partners have only access to the data or server that one wants to expose them. If not well configured, VPN could lead partners to gain more permission and data access than wanted. VPN access can also enable the propagation of malware.

As the above discussion should make clear, all the existing protocols possess inherent limitations, making them unsuitable (or at least, painful to use) for multi-partner data sharing initiatives like required in PILATUS project. These shortcomings range from security concerns (lack of encryption or flawed access controls) to a lack of standardization (absence of a uniform interface) and issues with scalability (challenges in replication or scaling). Additionally, the centralized data storage approach of most methods compromises data sovereignty and requires ad-hoc bilateral agreements. In contrast, the data space connectors, coupled with the IDSA reference architecture employed in the PILATUS project, address and rectify these concerns.

1.3 Overview of the IDSA reference architecture standard and the connectors

We give a short summary on the IDSA reference architecture model for data exchange, and why its main technical components, the data space connectors, offer an interesting tentative solution for secure data exchange in the multiple-actor scenario from the PILATUS project. The IDSA reference architecture implementation is based on the following three core concepts (see also Figure 1):

Data sovereignty: Data providers have control over their data, even when it's being shared or utilized by third-party entities: in any practical implementation of the IDSA architecture, the data owner defines who can access their data, how long and for what purpose. Policies are put in place by the data provider and must be agreed by the data consumer before any download.

Decentralization: IDSA focuses on a decentralized approach for data exchange: data are not duplicated centrally, as done with more traditional data exchange mechanisms like cloud storage or web platforms. Moreover, each participant can take decisions based on their policies and business needs when they share data.

Connectors: They are the essential technical tools in the IDSA architecture and act as gateways for data exchange. Connectors implement the rules and policies set by data providers and ensure that data sovereignty is maintained. Data space connectors ensure end-to-end encryption of data during exchange. They allow dynamic and real-time data exchanges while providing data-specific access controls. Unlike web application tools like SharePoint, they are designed primarily for data exchange and ensure scalability and integration across diverse platforms.

However, as opposed to “point and click” SaaS storage and exchange solutions like AWS or Rubrik services, the open-source connectors implemented in the PILATUS project are based on containers and require specific IT administration skills for deployment; this requirement will be discussed further in Section 3.3. Many connectors are in development within open-source projects, and a pre-selected list was tested by CSEM in T5.1; see also Section 3.1.

Besides, the IDSA reference architecture also defines a blueprint on how to structure the data exchange ecosystem, identifying the main players, functions, and processes. To carry out this task, it introduces the concept of distinct “layers” that must be specified for each data space application needs: the business layer, the functional layer, the process layer, the information layer, and the technical layer. The PILATUS project establishes the first data space architecture for the solar industry and the present report hence specifies in detail the design of each of these layers in this particular use case; see Section 2.

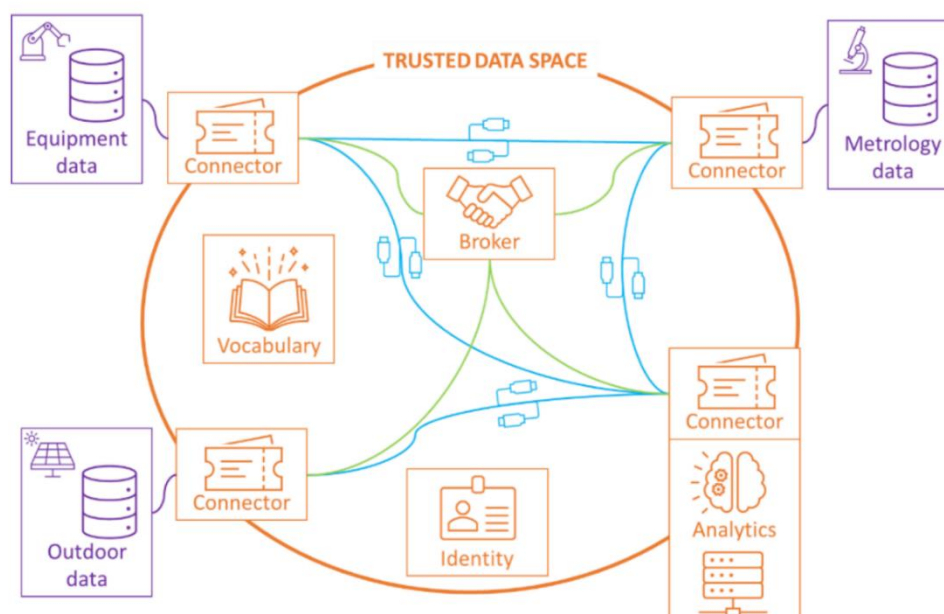


Figure 1: Illustration of the core implementation concepts of the data space for the solar industry: Data sovereignty (represented by the broker agreement and the identity trust store), Decentralization (connectors are running locally by each partner) and the connectors (software that permits data exchange).

2 Design choices for the PILATUS data space

For each IDSA layer, we outline its content, focusing on specifications and design for the PILATUS project. When relevant, we also discuss how specifications would be modified for data spaces embracing more actors in the solar industry.

2.1 Business layer

2.1.1 Overview

This layer identifies stakeholders, defines their roles, highlights data exchange scenarios and business models, and sets agreements rules for data transactions.

2.1.2 PILATUS project

Roles and responsibilities

The PILATUS project involves 19 partners. Research institutes (Fraunhofer CSP, Fraunhofer ISE, TNO, FZU, Liège University, EPFL, CSEM, EURAC), manufacturers of (materials for) encapsulants, wafers, solar cell, and modules manufacturers (Padanoplast, CPT, Norwegian Crystal, 5 Meyer Burger entities), and service and machine providers active in the solar manufacturing industry (Exateq, ISRA VISION). Out of these partners, four partners are actively involved in the data space deployment in T5.1 (Meyer Burger, EURAC, ISRA VISION and CSEM), but the architecture is in principle open to any other partner willing to participate. [Table 1](#) summarizes the roles of each involved partner in WP5 and their roles in the data exchange process. The Meyer Burger group appears through three of its entities: MBG, MBCH and MBR.

Table 1: Partners roles and responsibilities in The PILATUS project

Partner	Activity	Data consumer	Data provider	Service consumer	Service provider
MBR	R&D facility	No	Yes	Yes	No
MBCH	Module manufacturer	No	Yes	Yes	No
MBG	Cell and module manufacturer	No	Yes	Yes	No
CSEM	R&D institute, Industry 4.0 developer	Yes	No	Yes [EURAC KPIs]	Yes
EURAC	R&D institute, outdoor tester	Yes	Yes	No	Yes
ISRA VISION	Machine vision systems provider	No	Yes	No	Yes

The IDSA standard governance comes with a list of further roles and responsibilities: the broker, that helps consumers to find providers, the clearing house, that manages contracts and usage logging, and the identity provider. In the limited scope of the PILATUS project, these responsibilities are all with CSEM. Identity authentication is crucial and is ensured by the Dynamic Attribute Provisioning Service (DAPS) running at CSEM; see also Section 2.5.

Agreements for data transactions and services utilization

Partners must know how to request data and services from others. In PILATUS, partners use a shared Excel document on the project platform to share with others their needs in terms of data and services. This document, also available on the data space and shared by CSEM, requires data consumers to list dataset names, potential owners, and sharing duration needs. During monthly meetings, this table is reviewed on demand, and data owners detail sharing sources and sharing policies constraints. The policy agreement is then specified directly in the connectors of the data provider and agreed between provider and consumers prior to transaction. The same applies to service sharing.

Business scenarios

Two main use case scenarios are foreseen in PILATUS for data exchange and are depicted on Figure 2 below: the development of Industry 4.0 services (i.e., services meant to run on production lines), and the development of services to improve field performance and reliability of the modules (services addressing the operational phase of the product and bringing the information back to the manufacturing line). The first kind only involves two actors (the manufacturer and the service/R&D provider) and requires a high level of confidentiality, security, as well as a very low latency. In most cases, the final version of the service will be deployed directly on site to ensure low latency, and connectors will be used for data exchange in the design, implementation, and testing phase of the service; see also Figure 2.

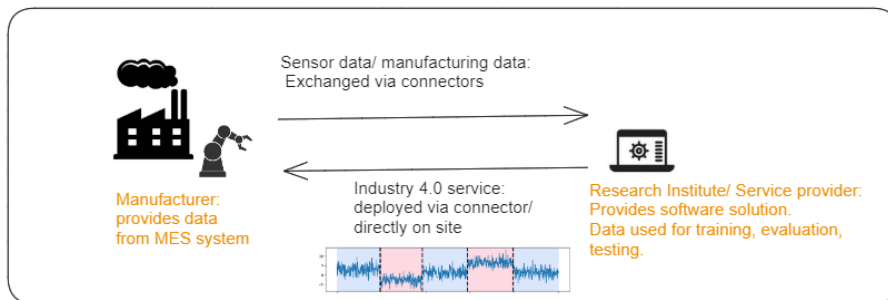


Figure 2: Industry 4.0 scenario. Data from a MES or from a machine are provided to a research institute for service development via connectors. Data are used for training, evaluation, and testing. The developed solution can then be directly deployed on site in case of low latency requirements.

The second kind involves multiple actors (more than two, in PILATUS WP5, T5.4 there are two Meyer Burger entities involved, EURAC, CSEM, and possibly ISRA VISION), and has lower latency requirements. In that case, results of the analysis or the application can be exchanged between the service provider and the manufacturer directly via connectors; see Figure 3.

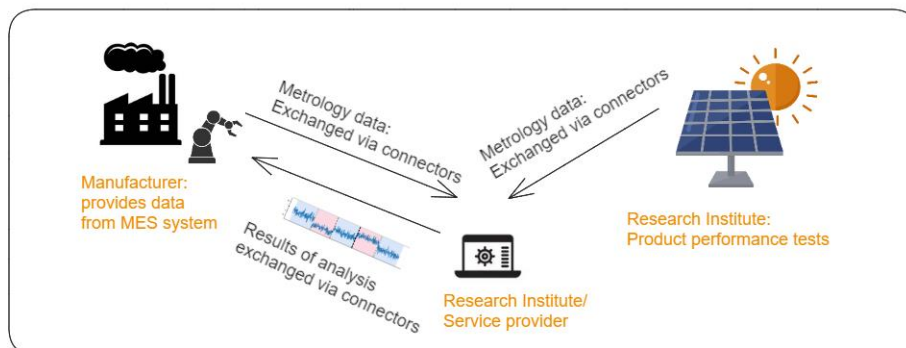


Figure 3: Reliability/process optimization scenario. Data are exchanged between more than two actors (for instance, EURAC, CSEM and Meyer Burger). There are no stringent latency requirements, and results of the analyses can be provided back to the manufacturer via connectors.

The principal goal of both (business) case is an increase in the performance of the solar cells and modules over their whole life cycle: during the production phase (lower production losses), and in operation (improved operational reliability). Both, the manufacturer and the research institute, benefit from the service: indirect financial benefit via lower manufacturing losses or higher reliability for the manufacturer, direct benefit for the service provider/research institute via the creation of a (potentially after project) paid SaaS service. As PILATUS is a collaborative project, no financial transactions take place when data or services are shared during the project (no financial transactions attached).

2.1.3 Generalization to the Solar industry

Outside PILATUS, solar asset owners or DSOs should join the data space platform. This would position them to collaborate with research institutes/service providers for services like predictive maintenance and production forecasting, and with solar manufacturers to improve product quality and ensure that reliability agreements are met. They will serve as data providers and service consumers; see also Figure 4. In this broader scope, financial transactions might be attached to data and service sharing.

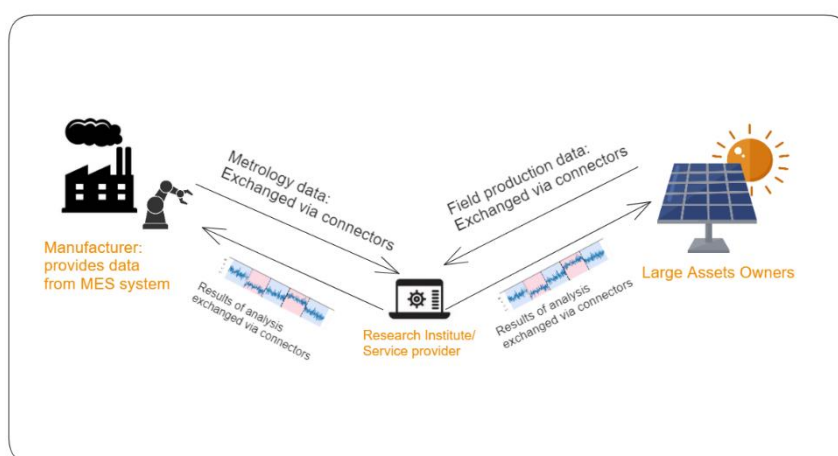


Figure 4: Broader scope for data exchange. The service provider develops a service that monitors field performance data and production line metrology data. It provides services for reliability assessments (both to asset owner and manufacturer), and services for reliability/ manufacturing process improvement to the manufacturer based on metrology and performance field data.

2.2 Functional layer

2.2.1 Overview

This layer details capabilities and services aligned with the business layer goals. It specifies core functions for connectors and components, including data validation, policy enforcement, and data transformation.

2.2.2 PILATUS project

In PILATUS, connectors are used primarily for data transfer, policy enforcement and interoperability. Other functions, ensured by developed services or the connectors themselves, are outlined below.

Functions executed by connectors

Data transfer & policy application:

Data transfer, access and sharing control is ensured by the connectors together with the DAPS (see

System layer, Section 2.5). They are used to read files or databases directly and expose them to the authorized data consumers. Policies are encoded in the data space connectors directly. The governance rules and data needs are registered in the dedicated Excel file available on the connectors.

Functions executed outside connectors

Data storage & validation:

Data are stored in partners dedicated infrastructure. Internal processes are run to ensure data validation. Data are stored in databases or standard format files that can be read easily by automated scripts (.parquet, .csv, .txt, .json, .pkl, .h5, ...). Validation is done all along the data flow chain to ensure smooth running of the services (at algorithmic level).

Data processing & integration:

Data processing and integration (e.g., integration of field performance data with production data from manufacturing line) is done at the algorithmic level by the service providers. Cells and modules manufacturers ensure that their products have a unique identifier that can be used to identify each cell/ module in the field performance testing phase.

Predictive maintenance applications and advanced analytics:

Predictive maintenance applications and advanced analytics (Performance analysis from field data, causal analysis, and optimization) to improve production lines yields are developed by the research institutes/ service providers and implemented with general-purposed programming languages (e.g., typically, Python/C/C++).

Data insights & visualization:

GUI (graphical user interfaces) are created by the research institutes / service providers to visualize the advanced analytics results and Industry 4.0 related tasks. These are also developed with interoperable languages (python, C++). If the manufacturer uses third-party visualization software, algorithm integration to commercial solutions is possible if these tools have open interfaces with general-purposed programming languages (Python, C++...).

Alerting:

In a 24/7 production, facility alerting of abnormalities, trends, outliers, etc. pp., is a key feature for keeping the yield high. With a throughput in a gigawatt-scale production capacity in the range of several thousand solar cells per minute, which each have dozens of key process parameters, manual monitoring of visualizations or process control charts is impossible. All developed models, GUIs or predictive applications need to alert the 24/7 production team accordingly.

2.2.3 Generalization to the solar industry

In the wider solar industry, functionalities would be expanded to include advanced analytics such as production forecasting, contract reliability checks, and lifecycle cost planning for new installation projects. The services will be offered to large assets owners by research institutes or service providers.

2.3 Process layer

2.3.1 Overview

This layer describes the sequences of operations and workflows essential for data exchange. It maps out how different functionalities (defined in the Functional Layer) come together to realize specific data exchange scenarios.

2.3.2 PILATUS project

The dataflow lifecycle is represented in Figure 5 and Figure 6.

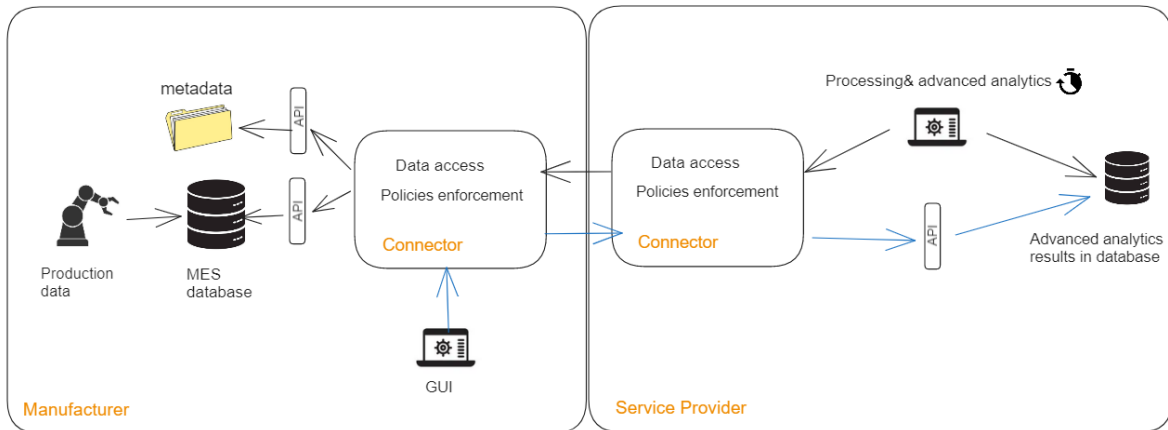


Figure 5: PILATUS dataflow for Industry 4.0 applications. Arrows direction indicates a “get” operation. Blue coloured arrows outline the flow from the user GUI to the results of the analysis.

We separated the two cases: development of an Industry 4.0 application, and a tripartite development of an advanced analytics for improving cell and module production reliability based on feedbacks from field performance tests. In both cases, the process is similar. Manufacturers generate data that they store in their MES database and are responsible for data-persistence strategies. Connectors handle data transfer and policy enforcement. Service providers clean and process the data, returning algorithmic results to manufacturers through connectors. Visualization uses a commercial or custom-made GUI. When low latency is needed, the GUI and algorithms run on the manufacturer's infrastructure. Data is only remotely shared with the research institute during initial testing. A similar process occurs in the tripartite situation, shown in Figure 6.

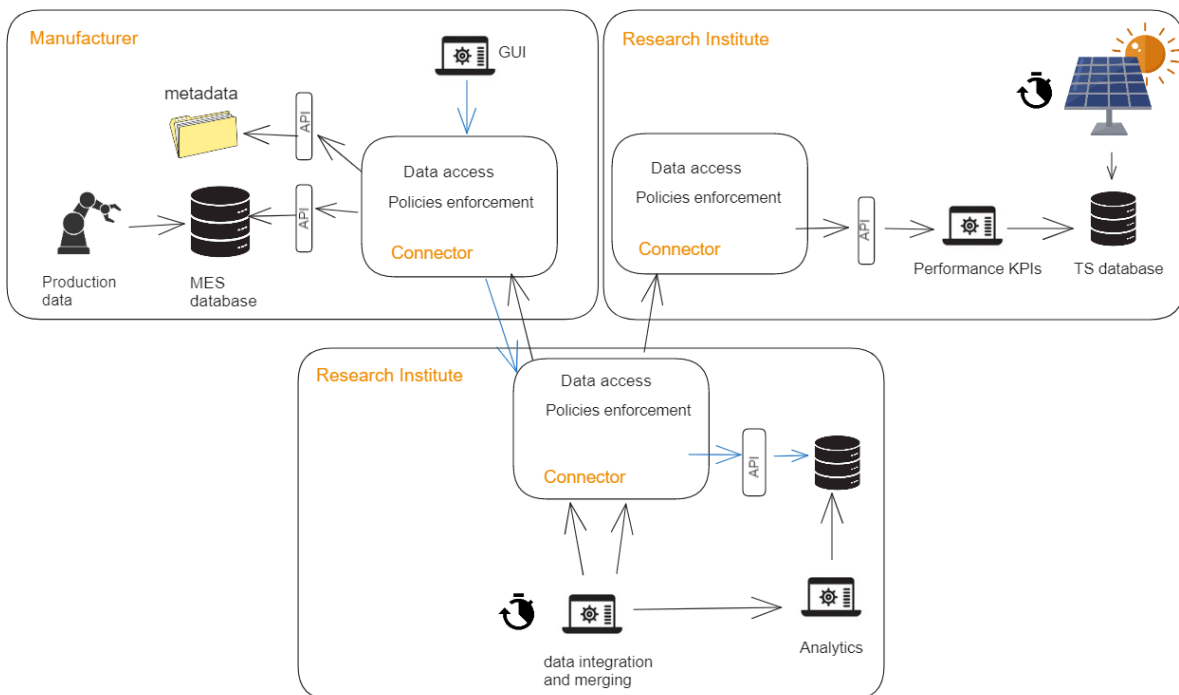


Figure 6: PILATUS Dataflow for cross linking performance field measurements and production line metrology. Arrows direction indicates a “get” operation. Blue coloured arrows outline the flow from the user GUI to the results of the analysis. The two arrows from “data integration and merging” to the Research Institute connectors mean that the analysis is using both manufacturer and Research Institute data.

2.3.3 Generalization to the solar industry

Similar dataflows pertain to business cases outlined in Section 2.1.3 involving large asset owners or DSOs. These actors replace the research institute in KPI field data collection and may be interested in performance KPI and reliability analyses provided by third parties. They own data, consume services, and provide services to others.

2.4 Information layer

2.4.1 Overview

The Information layer tackles the semantics of data exchange and defines data models, metadata descriptions, and standardized terminologies relevant to the domain of application.

2.4.2 PILATUS Project: Data models and metadata

Data model

Manufacturing data are stored in the manufacturers' MES systems. The project will adopt the existing MES data models from participating solar cells and modules manufacturers. MES systems use relational tables with unique identifiers (primary keys) for cells, strings, and modules. Tables are split logically by manufacturing line stations and testing units (see also Vocabulary in Section 2.4.3). A typical example of how two SQL tables for manufacturing production lines share primary keys is displayed below (the exact fields from the manufacturer are not disclosed for confidentiality reasons):

1. Table 1: CellSample
 - CellID (Primary Key)
 - ManufacturingDate
 - Series (e.g., Series A)
2. Table2: EfficiencyTest
 - CellID (Primary Key)
 - MeasurementTimestamp
 - Isc
 - Voc
 - Pmax
 - FF

To cope with heterogeneous fields and structures in the SQL tables, a metadata description will be shared by data providers for each shared dataset to ensure that service providers have the correct information for data retrieval and processing (see below). As described in Section 3.2.2, the preferred way to exchange data will be via APIs that directly do the "select" (and potentially merge) queries on the databases. Data exposed via API will be available in. Json formats and use a RESTful API endpoint. A standard response to a "get" will be a list of dictionaries with the table fields name and the corresponding fields values. On the example entitled "Table 1" above, this would lead to a response of the type:

```
[
  { "CellID ": 1, " ManufacturingDate": "2023-01-01", "Series":"xy" }
  { "CellID ": 2, " ManufacturingDate": "2023-01-01", "Series":"xy" }
]
```

Figure 7: Typical API response format

Performance field data that are time dependent are usually stored in databases specialized for time series (InfluxDB, anoSQL database, or TimeScaleDB, an SQL time series database based on PostGres). Data points will also be exposed via API and follow a similar structure as above, timestamps being often in that case the first entry field.

Metadata description

The metadata description will be available for each dataset. Each data provider will provide a file description compliant with [OPC UA building types](#). The following JSON file structure is suggested, inspired from OPC UA standard (shown here for a random cell tester on a line):

```

{
  "Node": {
    "NodeID": "StationMeasurement123",
    "Name": "EfficiencyTestStation",
  },
  "Properties": {
    "MachineType": "Cell Tester-123",
    "Location": "Line 2, Station 4",
    "LastCalibrationDate": "2023-05-10T14:10:00Z",
  },
  "Variables": {
    "CellID": {
      "DataType": "CHAR",
      "Description": "Unique Cell Identifier"
    },
    "Isc": {
      "DataType": "Double",
      "Description": "Short Circuit Current (Amps)"
    },
    "Voc": {
      "DataType": "Double",
      "Description": "Open Circuit Voltage (Volts)"
    },
    "Pmax": {
      "DataType": "Double",
      "Description": "Maximum Power (Watts)"
    },
    "FF": {
      "DataType": "Double",
      "Description": "Fill Factor (Ratio)"
    },
    "MeasurementTimestamp": {
      "DataType": "DateTime",
      "Description": "Timestamp of the last measurement"
    }
  },
  "References": {
    "IsPartOf": "SolarCellLine2",
    "UsesEquipment": ["SolarCellLoader", "IVTester"]
  }
}

```

Figure 8: JSON template file for metadata description derived from OPC UA standard.

2.4.3 PILATUS Project: Common Vocabulary

Two types of data are exchanged in the PILATUS project within WP4 and WP5: metrology data from the production lines (cells, modules) and metrology data from field performance tests. A vocabulary for each case is established below to facilitate data exchange and common understanding. The vocabulary is split in the following categories: general PV systems terminology, general terminology for databases, cells, modules and vision systems vocabulary, and performance field measurement vocabulary. When describing production lines, the order of the machines and processes follows the standard order of the production line (from inputs to outputs). Some of the terms are taken from standards like IEC TS 61836 [7], reports from the IEA [8] and internal vocabulary used by PILATUS partners. Data stored in databases might have composite names coming from the tables, e.g. “BrightFlakes_Count” can be used to design the number of Bright Flakes (see Table 6) on an inspected cell.

General terms

Table 2: PV systems general terms

Terms	Abbreviation/ Symbol	Definition	Unit
GENERAL TERMS FOR PV SYSTEMS			
Wafer		Thin slice of semiconductor material, typically silicon, used as the base for manufacturing solar cells.	
Solar cell		Device that converts sunlight into electricity through the photovoltaic effect.	
Half cell		A wafer cut in half creating two working solar cells out of one wafer.	
Sample		A single wafer, cell, or other specimen	
Encapsulating material		The material with which photovoltaic cells are laminated.	
String		Series of interconnected photovoltaic cells.	
Cell matrix		Interconnected strings.	
Laminate		Prepared unit comprising cell matrix, encapsulating material, and glass plate(s).	
Solar module		Laminate with fitted electrical connection.	
Junction box	<i>J-Box</i>	Enclosure on the solar PV module where the PV strings are electrically connected.	
Backsheet		The protective outer layer on the backside of a solar module.	
Coating		A functional layer on the surface of a cell/sample, used to achieve certain optical or electronic behaviour.	
Chamfer		Symmetrical sloping surface at an edge or corner, typically applied to eliminate sharp edges and to improve the aesthetic appeal, light-absorption properties, or efficiency of the solar cell.	
Fingers		Thin, conductive metal lines on the cell surface that collect the current generated by the cell and transport it to the busbars, playing a crucial role in the cell's electrical performance.	
Ribbon		conductive strips used to interconnect solar cells in the cell matrix	

Table 3: Databases and MES general terms

Terms	Abbreviation/ Symbol	Definition	Unit
GENERAL TERMS FOR DATABASES AND MES			
Table		Basic storage units in a relational database	
Identifier	<i>ID</i>	Unique value used to recognize and distinguish each record.	
Query		A request to retrieve, modify, or add data within a database.	
Data pipeline		Series of operations performed on data to convert them from one form to another.	
Manufacturing execution system	<i>MES</i>	Computerized system that monitors, controls, and optimizes production processes in real-time within a manufacturing facility.	
SQL/NoSQL		Different types of database systems based on structured (SQL) or flexible schema-less structures (NoSQL (Not Only SQL)).	

Vocabulary for cell manufacturing

Table 4: Cell manufacturing terms.

Terms	Abbreviation/ Symbol	Definition	Unit
MANUFACTURING LINE			
Texturing station		Modifies the surface of the silicon wafer to maximize light absorption.	
Dopant diffusion furnace		Oven that diffuses specific impurities into the silicon wafer to develop the desired semiconductor properties.	
Plasma-enhanced chemical vapor deposition machine	<i>PECVD</i>	Used to deposit thin film layers, like anti-reflective coatings and passivation layers, under plasma-enhanced conditions.	
Metallization station		Deposits metal contacts onto the cell surface, facilitating the gathering of electrons during operation.	
Photolithography		Employs light to transfer a geometric pattern onto the substrate, essential for detailed patterning of the metal contacts.	
Anti-reflective coating machine	<i>ARC</i>	Adds a layer on the wafer's surface to decrease reflection and enhance light absorption.	
Cell tester & sorter	<i>CTS</i>	Tests each cell's performance and categorizes them based on efficiency or power output.	

Laser doping & ablation		Uses lasers for dopant introduction or material removal from the cell.	
Edge isolation machine		Ensures the cell's edges are devoid of conductive material, preventing unintended electrical paths.	
Cleaning station	<i>Clean</i>	Employs chemical and physical treatments to ensure a contaminant-free wafer surface.	
Cell inspection & quality control		Tools and procedures for visual cell examination to detect performance-impacting defects (see also table on image-based inspections).	
SOLAR CELLS MEASUREMENTS			
Open-circuit voltage	V_{oc}	Maximum voltage a solar cell produces when it's not connected to a load (i.e., no current flowing).	V
Short-circuit current	I_{sc}	Maximum current a solar cell produces when its terminals are shorted. It indicates the maximum current a cell can provide under full sunlight.	A (or mA)
Maximum power point	P_m	Point on the current-voltage curve where the cell's output power is maximized.	W (or mW)
Fill factor	FF	Ratio of the maximum power point (P_m) on the open circuit voltage (V_{oc}) and the short circuit current (I_{sc}) product.	%
Cell efficiency	H	The ratio of the electrical power a cell can produce to the sunlight power it receives.	%
Current-voltage (I-V) curve:	$I-V$	Graphical representation of the relationship between the current output of the solar cell and its operating voltage.	
Lifetime	LT	Average time a minority carrier can exist in a high-energy state in semiconductor substrate before recombining.	s
Parallel resistance	R_p, R_{sh}	Sames as shunt resistance; see below.	Ω
Reflectivity	$REFL$	Fraction of sunlight reflected off a solar cell's surface instead of being absorbed.	%
Resistivity	R	Measure of resistance of wafers.	Ω cm
Series resistance	R_s	Series resistance in the equivalent circuit diode model of a solar cell.	Ω
Sheet resistance	R_{sheet}	Sheet resistance of a material.	Ω/sq
Shunt resistance	R_{sh}, R_p	Resistance that models current leakage across the cell in the equivalent circuit diode model of a solar cell.	Ω
Wafer thickness	WTK	Thickness of a wafer used in solar cell production.	μm
SOLAR CELLS DEFECTS			
Wrong colour		The colour of the solar cell deviates from the specification.	RGB

Cracks		Cracks on the solar cell surface.	
Dislocations		Crystallographic defects where irregularities occur in the periodic structure of the crystalline solar cell.	
Fracture		Fracture of a wafer or half-cell.	
Grain boundaries		Interfaces where crystals of different orientations meet, which can reduce a cell's efficiency.	
Contaminations		Unintended substances or particles introduced during manufacturing.	
Edge isolation failures		Inadequate isolation of the cell edges, leading to shunting.	
Shunts		Undesired short circuits between regions with different polarity inside the cell, leading to partial or complete power loss.	

Vocabulary for module manufacturing

Table 5: Modules manufacturing terms.

Terms	Abbreviation/ Symbol	Definition	Unit
MANUFACTURING LINE			
Cell connection station	<i>CCS</i>	Part of the production line where individual solar cells are electrically connected to form strings.	
Lay-up station		Arranges the cell strings onto a glass substrate in preparation for lamination.	
Lamination system	<i>LAMI</i>	Encapsulates the cell strings between layers of protective materials, typically using EVA (ethylene-vinyl acetate) or other encapsulants.	
Framing		Attaches aluminium frames to the laminated panels.	
Junction box station		Affixes the junction box, which houses the electrical connections, to the solar module.	
Flasher		Exposes the module to a flash of light mimicking sunlight to measure its electrical performance.	
EL tester		Inspects module using EL (see image-based inspections).	
Insulation & grounding test		Ensures that the module's electrical components are properly insulated and grounded.	
Glass cleaning station		Cleans the top glass layer of the module.	
Barcode & labelling station		Applies necessary labels, barcodes, or QR codes to the finished module for tracking.	
MODULES DEFECTS (manufacturing, see operation in field performance table)			

Cracks		Physical fractures or breaks in the cell material.	
Inactive cells		Cells that do not produce electricity when exposed to sunlight.	
Black spots		Localized darkened areas, often indicative of material defects.	
Ribbon shift		Misalignment or displacement of the conductive strips used to interconnect solar cells.	
Cell/ matrix misalignment		Misalignment of the cells on the cell matrix.	
Broken cells		Parts of the cell are cut/ broken (and missing).	
Bubbles		Trapped pockets of air or gas between the layers of a solar module.	

Vocabulary for cell optical inspections

Table 6: Vision systems and inspections terms.

Terms	Abbreviation/ Symbol	Definition	Unit
GENERAL TERMS			
Class/ Category		Specific category or label to which data points can be assigned in classification tasks.	
Segmentation mask		Binary or multi-label map that indicates the category or object label of each pixel.	
Electroluminescence	<i>EL</i>	Diagnostic technique where a solar cell is excited by an external forward current, causing it to emit light.	
Photoluminescence	<i>PL</i>	Non-contact imaging method that captures the light emitted by a solar cell when it is excited by photons.	
Visual inspection		Inspection by eye/optical camera in visible domain.	
Probe region		A defined region on the wafer/cell where a local measurement is carried out.	
Border region		A region with a defined width, running around the full wafer contour. Defining a border region allows for setting different sensitivity and defect tolerance in border and interior regions.	
Interior (region)		The inner part of a wafer that is not border or edge.	
Print		A visible pattern on the cell, often formed by screen printing (e.g., metal contacts of a	

		cell) but often also used for non-printed patterns (e.g., laser patterns).	
CELL OPTICAL DEFECTS			
Intrusion		Manufacturing anomaly where part of the cell surface indents inward (typically small edge breakage).	
Protrusion		Manufacturing anomaly where part of the cell surface extends outward.	
VBreak		A special V-shaped form of breakage on mono-Si, very sensitive to mechanical stress and often leading to fracture.	
(Cell) Breakage		Structural damage or fractures at the edge of the cell material.	
Busbar Break		Disruption or breakage in the conductive strips (Busbars) used to collect and transport electric current, adversely impacting the cell's ability to efficiently conduct electricity.	
Chamfer break		Defect where the intended bevelled edge is damaged or malformed.	
Cracks, Microcracks		Optically invisible structural damage of fracture within the cell material, detectable only by special methods (PL, EL, or other IR-based inspection)	
Dot		Small defects on the surface (or coating) that exhibit a strong contrast.	
Bright flake		Coating defect, usually caused by distorted plasma field lines (e.g., due to a bigger particle on the surface). Bright flakes are brighter than the background.	
Dark flake		Similar to bright flakes, but the field line distortion caused the defect to appear darker than the background.	
Inhomogeneities		Areas of uneven material properties or composition, which can lead to irregular light absorption and conversion.	
Etching mark		Defect characterized by visible marks or lines usually caused by the chemical process used to remove layers or sections of the material.	
Finger interruptions		Break or discontinuity in the conductive finger lines.	
Finger slubs		Imperfections or irregularities in the conductive finger lines, such as excess material or deformations.	
Finger thinning		A finger thinning is a defect where the finger is thinner than desired, leading to an increase of series resistance.	
Stains		Unwanted discolorations or marks on the surface or within the structure of the cells.	

Closing interruption		Break or discontinuity in the closing elements or contacts of the cell.	
Rings		Rings are a defect type characterized by a ring shape.	
Shortcuts		Shortcuts between fingers (IBC cell designs).	
Dust particles		Defect where dust particles are present on the cell surface.	
Uncategorized		The uncategorized defects occur for surface and contour analysis.	
CELL OPTICAL QUALITY ATTRIBUTES			
Placement quality		Quality result for sample positioning below camera.	
Cell quality		Quality result for the cell quality (result of print, colour, surface, contour, geometry, and/or others). Usually, top level result: Derivates are e.g., CellQuality_PrintGeometry.	
Colour quality		Quality result for colour sorting (usually top level result): Derivates are, e.g., ColourQuality_Border, ColourQuality_ProbeRegion,...).	
Hue, saturation, value	<i>HSV</i>	Colour model that represents colour more in terms of how people perceive and interpret colours, where Hue defines the colour itself, Saturation represents the dominance or intensity of the colour, and Value corresponds to the brightness of the colour.	

Vocabulary for performance field measurements

Table 7: Field measurements and performance assessment terms.

Terms	Abbreviation/ Symbol	Definition	Unit
GENERAL TERMS			
Sample		Data acquired from a sensor or measuring device.	
Sampling interval		Time between samples.	
Record		Data recorded and stored in data log.	
Recording interval		Time between records.	
Reporting period		Time between reports.	

Standard test conditions	STC	Reference values of G_{POA} (1000 W/m ²), PV cell junction temperature (25 °C) & reference spectral irradiance defined in IEC 60904-3.	
WEATHER-RELATED PARAMETERS			
Irradiance	G	Incident flux of radiant power per unit area	W/m ²
In-plane irradiance	G_{POA}	Sum of direct, diffuse and ground reflected irradiance incident upon inclined surface parallel to plane of modules in PV array.	W/m ²
Global horizontal irradiance	GHI	Direct plus diffuse irradiance incident on horizontal surface.	W/m ²
Direct normal irradiance	DNI	Irradiance emanating from the solar disk and from the circumsolar region of the sky within a subtended full angle of 5° falling on a plane surface normal to the sun's rays.	W/m ²
Diffuse horizontal irradiance	DHI	Global horizontal irradiance excluding the portion emanating from the solar disk and from the circumsolar region of the sky within a subtended full angle of 5°.	W/m ²
Irradiation	H	Irradiance integrated over a specified time interval.	kWh/m ²
Albedo		Fraction of sunlight that is reflected by the surface on a scale of 0 to 1.	
Soiling ratio	SR	Ratio of the actual power output of the PV array under given soiling conditions to the power that would be expected if the PV array were clean and free of soiling.	
Ambient temperature	T_{amb}	Ambient air temperature.	°C
Module temperature	T_{mod}	PV module temperature, measured at the backside of the PV module.	°C
PV-SYSTEM OPERATION PARAMETERS			
Active power	P	Instantaneous product of current and voltage.	W
Power DC side	P_{DC}	Instantaneous power measured on DC side.	W
Power AC side	P_{AC}	Instantaneous power measured on AC side.	W
Energy	E	Power integrated over a specified time interval.	kWh
Energy DC side	E_{DC}	DC power integrated over a specified time interval.	kWh
Energy AC side	E_{AC}	AC power integrated over a specified time interval.	kWh

Module or string voltage	V_{MP}	Module or string voltage at maximum power point.	V
Module or string current	I_{MP}	Module or string current at maximum power point.	A
Open circuit voltage	V_{OC}	Voltage of the PV module or string at open-circuit conditions.	V
Short circuit current	I_{SC}	Current of the PV module or string at short-circuit conditions.	A
Nominal power	P_{DC0}	Nameplate DC rating.	
Module temperature coefficient	γ	Parameter describing relation between module temperature and PV module power at maximum power point.	1/°C
PV-SYSTEM OPERATIONAL FAILURES			
Hot spots		Areas on the module that have significantly higher temperatures.	
Glass breaking		Physical damage to the module's front or back glass.	
Discoloration		Change in the module's colour.	
Backsheet cracking		Fractures or breaks in the protective outer layer of a solar module.	
Snail trails		Visible dark lines or patterns resembling the trail left by a snail, caused by moisture and certain material degradation.	
Potential induced degradation	PID	Degradation of module performance due to voltage, ion migration, and environmental conditions.	
Junction box failures		Problems associated with the junction box, including corrosion.	

2.5 System layer

2.5.1 Overview

This layer specifies the physical and software infrastructure required for data exchange. For more details on the exact implementation of the connectors and the tests realized within the PILATUS project, we refer the reader to Section 3, where all tests and the final connectors implementation and deployment are presented. This section gives a broad overview of the components and physical requirements.

2.5.2 PILATUS project

The main software components are the databases and files, connectors, the DAPS, and the additional tools (GUI and scripts for automated data Exchange). The main hardware components are production machines, sensors, and Linux (eventually Windows) servers. The summary of the layer specification is displayed on Figure 9; see also Section 3.

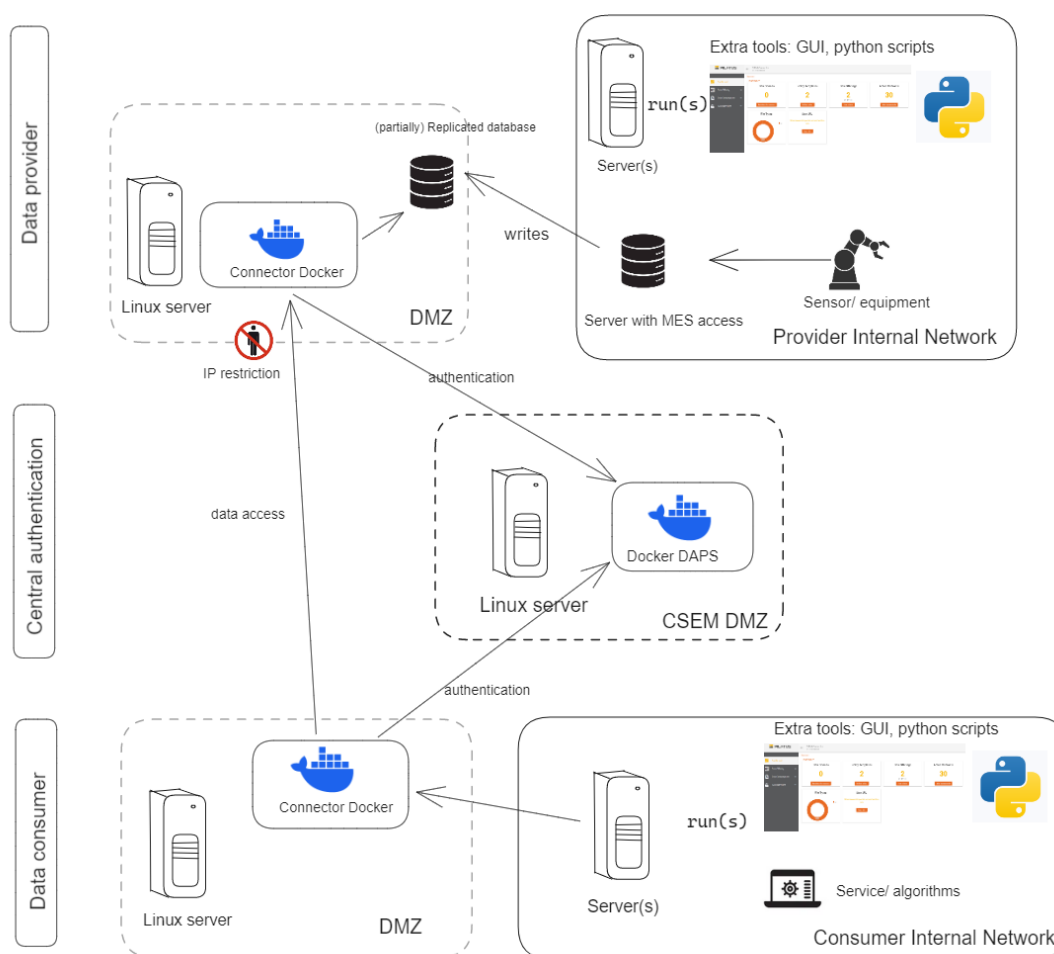


Figure 9: Real Technical layer implemented in the PILATUS project, with security enhancement via IP restrictions and DMZ (Demilitarized Zone, a subnetwork presenting external-facing services online) deployment.

Databases and files

Data from production facilities and machines are stored in files (e.g., .csv) and/or to MES systems from manufacturers (usually physical storage on servers). MES systems mostly use in background commercial relational databases (e.g., Oracle) that can be accessed automatically via standard APIs like ODBC. Time series alike performance field measurements are usually stored in time series

databases (like [Influx DB](#), or [Timescale DB](#)) or document based databases like [Mongo DB](#). Research institutes like EURAC and CSEM use these open-source solutions that are easy to automatically access with programming languages like Python, R, C, or java (see Paragraph **Additional tools** below). In both cases, databases and MES systems run on servers with sufficient storage capacity (hard disk). Persistence strategies are put in place individually by manufacturers to ensure smooth running of their infrastructure within the physical storage capacities.

Connectors

Connectors are deployed with docker on Linux, Mac OS, or Windows servers and CSEM prefers deployment on Linux-based OS for simplicity of use. Connectors are deployed in one of two modes (data consumer/data provider) on each partner's infrastructure in a decentralized fashion. For security, partners should deploy connectors in their DMZ (demilitarized zone), a subnetwork presenting external-facing services online, acting as a buffer against external cyber threats to their internal network.

DAPS and security

The DAPS ensures central authentication of each data space participant. The DAPS runs on a Linux server at CSEM and ensures that the certificates provided by connected partners are correct and allow connectors transactions. To ensure a maximum-security level, data provider connectors can put in place further IP security restrictions for data consumer connectors connection.

Additional tools (GUI, RESTful APIs, scripts)

A GUI to visualize data offering, debug data exchange, and help set up policy for data exchange is put in place in the PILATUS project. It is a multiplatform GUI and can run on any local laptop or server; see also Section 3.2.1. Scripts to serve automated data download and data exposure are provided by CSEM in Python, see Section 3.2.2.

2.5.3 Generalization to the solar industry

In a broader scope, the DAPS could run outside CSEM, e.g., at Fraunhofer AISEC (Fraunhofer Institut für Angewandte und Integrierte Sicherheit) [9] that provides a DAPS for data space projects. Furthermore, the clearing house and broker components could be operated by third-party service providers and not rely on CSEM.

3 Data space implementation and tests

3.1 Connector choice: deployments and tests at CSEM


3.1.1 Available connectors and pre-selection

Connectors have various implementations. A list is given by IDSA [10]. In its initial assessment, CSEM leaned towards the most mature implementations of connectors that matched our requirements: the Eclipse dataspace connector from the Eclipse Dataspace Components (EDC) and the Dataspace connector (most other implementations have low TRLs and hence are not suitable for the PILATUS project).

Eclipse dataspace connector

The *Eclipse dataspace connector* is a promising implementation that might represent the future of connectors. It is written in Java, and it is configured through code extension. The tables below are taken from [10].

Table 8: Description of the Eclipse dataspace connector (Source: [10])


Name of the connector	Eclipse Dataspace Components
Logo of the connector or company logo	
Maintainer (company name)	Committer Group in Eclipse Foundation
Type of connector	Data connector framework
Short description	Whatever the individual setup is – on-premises bare-metal, different cloud vendors, hybrid, even single end-user machines – the EDC can be customized to work within any environment at scale. The connector’s added value is achieved through the separation of control and data plane, which enables a modular and thereby customizable way to build data spaces. Due to common interfaces and mapping of existing standards, the connector adds capabilities of contract negotiating and policy handling in an interoperable manner. As an open-source project hosted by the Eclipse Foundation, it provides a growing list of modules for many widely deployed cloud environments “out-of-the-box” and can easily be extended for more customized environments, while avoiding any intellectual property rights (IPR) headaches.
Maturity level	TRL 8-9
License type	Apache 2.0
Features	<ul style="list-style-type: none"> • Modular and highly extensible framework • Separate control and data planes • System is asynchronous and highly available

	<ul style="list-style-type: none"> • Policy negotiation and data transfer orchestration • Transfer processes are fully auditable • Eliminate single points of failure • Cloud aware policy enforcement and projection • Default implementations and blueprints available
--	---

Dataspace Connector

The *Dataspace connector* is the first connector implementation made by Fraunhofer Institute and maintained by Sovity. It is written in Java. Given the ambiguity of the term "*Dataspace connector*," we consistently italicize it when we refer to this connector in the rest of the document. The *Dataspace connector* is ready-to-use and configured via a file (no code/ low code approach). It can be deployed using Docker and comes with a pre-implemented management GUI. For simplified deployment, Docker-compose templates are also available.

Table 9: Description of the Dataspace Connector (Source: [10])

Name of the connector	<i>Dataspace Connector</i>
Logo of the connector or company logo	
Maintainer (company name)	Sovity
Type of connector	Generic open-source solution
Short description	The Dataspace Connector is an IDS connector that is currently being maintained by Sovity GmbH. The connector was originally developed at the Fraunhofer ISST. With the help of the Dataspace Connector, existing software can easily be extended by IDS connector functionalities in order to integrate them into an IDS data ecosystem. Furthermore, it is possible to use the Dataspace Connector as a basis for the development of own software that is to be connected to an IDS data ecosystem.
Maturity Level	IDS-Ready and part of the IDS Graduation Scheme
License type	Open-source software
Features	The Dataspace Connector integrates the IDS Information Model and uses the IDS Messaging Services for IDS functionalities and message handling. The core component in this repository provides a REST API for loading, updating, and deleting resources with local or remote data enriched by its metadata. It supports IDS conform message handling with other IDS connectors and components and implements usage control for selected IDS usage policy patterns.

3.1.2 Deployment and tests at CSEM

Tests were conducted at CSEM using the preselected connectors to guarantee a smooth deployment and seamless data exchange. The tests (a, b, and c) are detailed below.

a. Dataspace connector docker-compose deployment.

IDSa provides docker-compose template examples for the connectors, in particular for the so-called *Dataspace Connector*. A docker-compose template defines a multi-container Docker application (see also [11]). CSEM drew inspiration from these repositories to create its own docker-compose template, which includes the Dataspace connector docker container, the PostgreSQL DB docker container, and the connector UI docker container. The final docker compose template file used in the PILATUS project is given below:

```
services:
  postgres:
    image: postgres:13
    environment:
      - POSTGRES_USER=${DB_USER}
      - POSTGRES_PASSWORD=${DB_PASSWORD}
      - POSTGRES_DB=connector_db
    volumes:
      - ./connector_db_data:/var/lib/postgresql/data
    networks:
      - local

  connector:
    image: ghcr.io/international-data-spaces-association/dataspace-connector:8.0.2
    ports:
      - 8262:8080
    environment:
      - CONFIGURATION_PATH=/config/config.json
      - DAPS_URL=https://daps.portal.csem.ch
      - DAPS_TOKEN_URL=https://daps.portal.csem.ch/auth/token
      - DAPS_KEY_URL=https://daps.portal.csem.ch/auth/jwks.json
      - DAPS_INCOMING_DAT_DEFAULT_WELLKNOWN=/jwks.json
      - SERVER_SSL_KEY-STORE=file:///conf/keystore.p12
      # Define the PostgreSQL setup
      - SPRING_DATASOURCE_URL=jdbc:postgresql://postgres:5432/connector_db
      - SPRING_DATASOURCE_PLATFORM=postgres
      - SPRING_DATASOURCE_DRIVERCLASSNAME=org.postgresql.Driver
      - SPRING_DATASOURCE_USERNAME=${DB_USER}
      - SPRING_DATASOURCE_PASSWORD=${DB_PASSWORD}
      - SPRING_JPA_DATABASE_PLATFORM=org.hibernate.dialect.PostgreSQLDialect
      - CONFIGURATION_FORCE_RELOAD=false
      - SERVER_SSL_ENABLED=true
      - SPRING_SECURITY_USER_NAME=${CONNECTOR_USER}
      - SPRING_SECURITY_USER_PASSWORD=${CONNECTOR_PASSWORD}

    volumes:
      - ./config/config.json:/config/config.json
      - ./config/keystore.p12:/conf/keystore.p12
      - ./config/truststore.p12:/config/truststore.p12
    networks:
      - local
    ulimits:
      nofile: 65535
    depends_on:
      - postgres

  csemconnectorui:
    build:
      context: .
      dockerfile: ./Dockerfile
    environment:
      - CONNECTOR_URL=https://cv50111:8080
      - CONNECTOR_USER=${CONNECTOR_USER}
      - CONNECTOR_PASSWORD=${CONNECTOR_PASSWORD}
    ports:
      - 8263:8083
    networks:
      - local
```

```
ulimits:
  nofile: 65535

networks:
  local:
    driver: bridge
```

b. Tests without central authentication (test mode)

The aim of this first test was to verify the connectors’ ability to seamlessly communicate and exchange files in “test mode”, without central secure authentication.

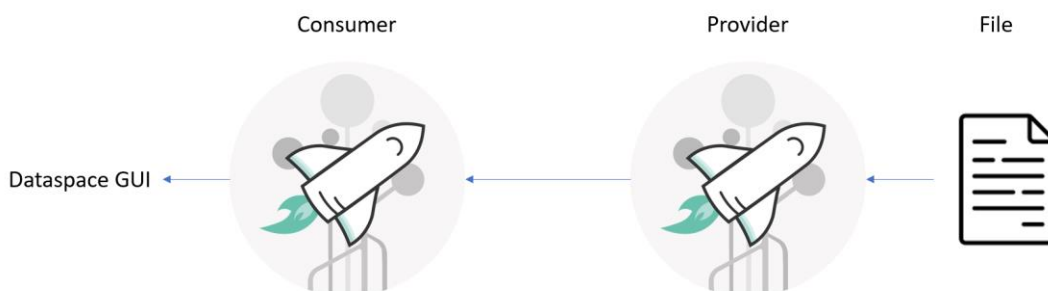


Figure 10: First data exchange test: data exchange without central authentication

In this first test, the consumer connector requests a file from the provider connector, bypassing all security checks. Both connectors must be in test mode to communicate together. For this propose, CSEM deployed two dataspace connector instances in test mode on two different servers with an open port on the same network. CSEM then attempted to exchange files with the Connector.

c. Tests with central authentication (production mode)

In order to put the dataspace on internet and ensure secure exchange, the next step is to enable security features (DAPS authentication) and to put connectors in production mode. This requires SSL certificates and an authentication server, the DAPS. The DAPS is an authentication server that recognize a connector via its SSL certificates and checks that the connector is into a dataspace. The DAPS is provided by IDSA, see [12].

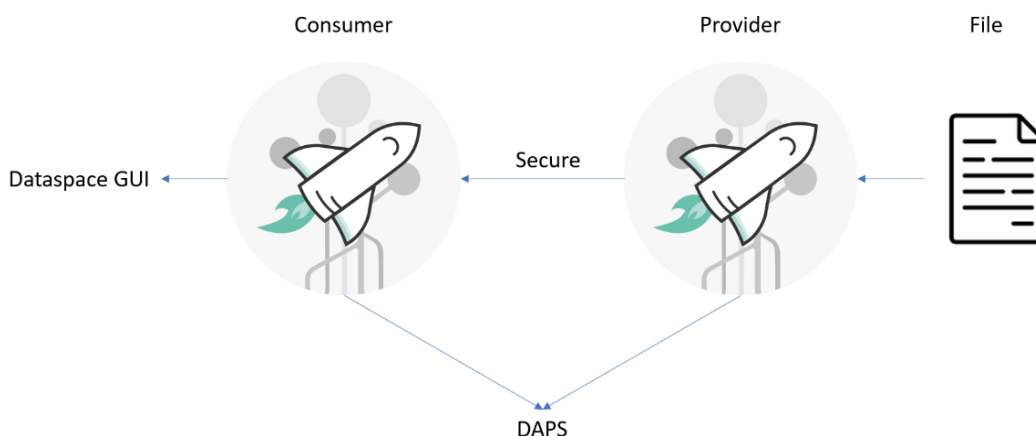


Figure 11: Second data exchange test: data exchange with central authentication enabled.

In that exchange scheme, the consumer connector requests a file from the provider connector. First, it seeks an agreement from the provider connector to access the file. This agreement is given if both

connectors recognize each other and if the DAPS confirms the connectors' identities. To carry out this first secured exchange, the following is needed.

One has to:

- deploy a DAPS server
- put test connectors in production mode
- generate SSL certificate for each connector
- register each connector on the DAPS server
- generate the trust store file (a file that contains the certificates of each connector)
- install the trust store file on every connector.

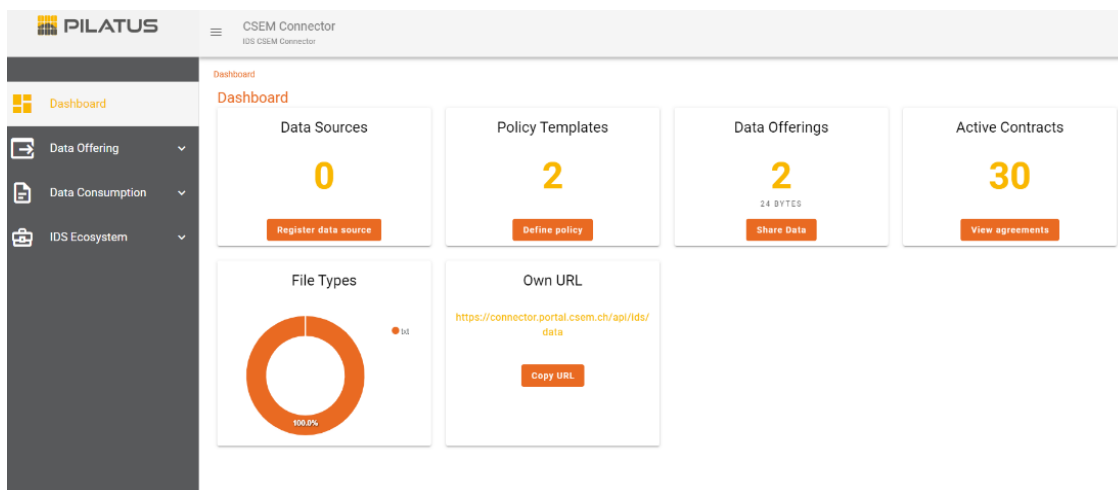


Figure 12: GUI for the PILATUS Connectors

3.1.3 Tests results and final connector choice

The *Eclipse dataspace connector* (Table 8) requires a predominantly code-based configuration (java), making it hard to deploy with partners where a no-code/ little code approach is needed. CSEM found that the *Dataspace connector* (Table 9) was the easiest to deploy with basic Linux/Docker knowledge and simple file configuration. Furthermore, CSEM encountered issues in testing the *Eclipse dataspace connector* and was failing to carry out the two tests b) and c) mentioned above due to insufficient documentation (many tickets opened on GitHub repository for documentation clarification). The connector could be connected to the DAPS (installation successful), but data exchange was hard to set up (the API was not integrated to the connector, no documentation around it). CSEM did not encounter the same issues with the *Dataspace connector* (Table 9), and the two tests b and c were passed without blocking issues.

3.2 Implemented features on top of the dataspace connector.

The *Dataspace connector* natively support sharing files, APIs, and databases. When a file is uploaded to a connector, it retains that specific version. If the local file changes, the connector doesn't auto-update. However, for APIs or databases, the connector fetches the current data dynamically. We describe here some upper layers that were implemented and deployed by CSEM to enable these functionalities.

3.2.1 GUI

One other useful feature of the *Dataspace Connector* (Table 9) is the pre-implementation of a GUI for tests and data space management purposes. It should only be served on a local network. With the GUI, CSEM and partners can monitor, manage, and debug connector connections. Actual data exchange

runs without this GUI. The GUI has been adapted by CSEM to comply with the PILATUS theme and colours; see Figure 12.

3.2.2 Scripts and APIs for automated transfer

The GUI cannot be used for automatic data transfer (it is, however, made for static files exchange and policy agreements). CSEM has implemented Python scripts to transfer data via RESTful APIs.

Script to upload data & set policies on connectors.

CSEM has prepared Python scripts that can be run by a data provider. By running these scripts, the data provider can use the connector API to upload data to its connector and to set policies, thereby determining which connector can access specific data.

Script to download dynamic data through connectors.

CSEM has developed a minimal API that fetches data from the provider's local network.

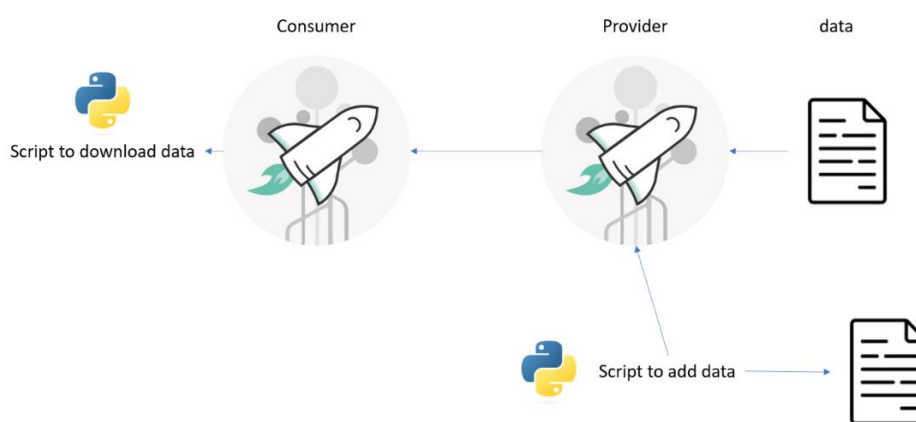


Figure 13: Python scripts to add and retrieve data from the data space via connectors.

Here, the consumer connector queries the API resource. The connector then invokes the API and returns its response (see Figure 13).

The objective of querying APIs for data download (and not directly the database) is to secure the database, ensuring that consumers cannot perform unintended write, read, or delete operations.

We give below a simple Python code using Fast API to share a folder's file list and allow file downloads. With this example, one can monitor folder events, file changes, etc., via the connectors:

Python script:

```
import os
from datetime import datetime
from pathlib import Path
from typing import Optional

from fastapi import FastAPI, HTTPException
from starlette.requests import Request
from starlette.responses import FileResponse

app = FastAPI()

@app.get("/")
async def get_file(request: Request, filename: Optional[str] = None):
    base_dir = Path(__file__).resolve().parent
    file_dir = base_dir / "data"
    if filename:
        file_path = file_dir / filename
        if os.path.isfile(file_path):
            return FileResponse(file_path, filename=filename)
```

```

    return HTTPException(status_code=404)
files = [
    {
        "file_name": f,
        "size_octet": os.path.getsize(file_dir / f),
        "update_date": datetime.fromtimestamp(os.path.getmtime(file_dir / f)).isoformat(),
        "creation_date": datetime.fromtimestamp(os.path.getctime(file_dir /
f)).isoformat(),
        "url": f"{request.base_url}?filename={f}",
    }
    for f in os.listdir(file_dir) if os.path.isfile(file_dir / f)
]
return files

if __name__ == "__main__":
    import uvicorn

    uvicorn.run(app, host="0.0.0.0", port=8000)

```

3.3 Partners' deployment

3.3.1 Exchange scenarios

As described in the Business layer model (see Section 2.1), a typical exchange scenario for Industry 4.0 applications involves a research institute and a manufacturer. Figure 14 below illustrates this with CSEM as research institute and Meyer Burger as data provider. According to Table 1, partners are either data providers and/or data consumers. Provider and consumer cases outlined below must be implemented and tested with partners depending on their roles in Table 1. In both cases, as outlined in Section 2.5, a Linux OS server is preferred. Deployment is also possible on Windows using tools like Docker Desktop [13].

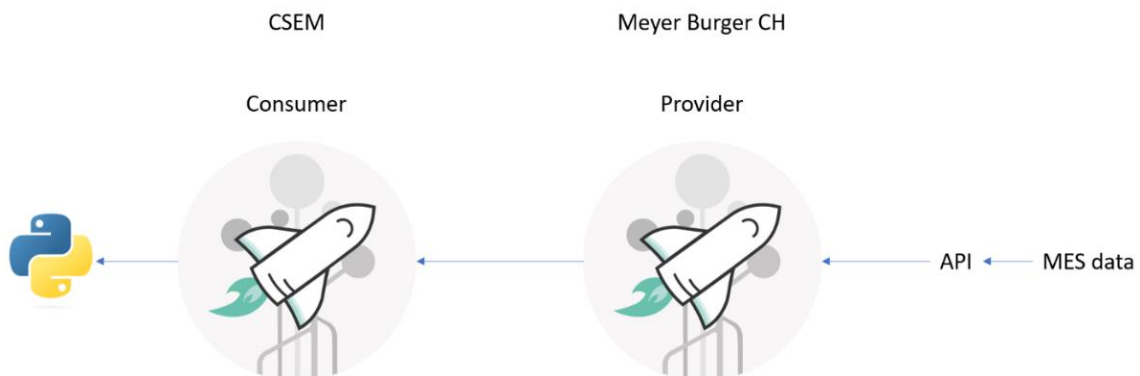


Figure 14: Typical exchange scenario between Meyer Burger and CSEM

Provider connector

To deploy a provider connector, partners require Docker for the connector deployment, an open port, or a domain to host the connector, and access to the desired data within their local network.

Consumer/client connector

To deploy a consumer connector, partners just need docker.

3.3.2 Providing code and guidance to partners

CSEM supported the project partners for deployment and provided the code base and direct help. A step-by-step guide for deployment is outlined below.

Gitlab Repository

CSEM has provided code on Gitlab to ease partners deployment and shared the code with all partners. CSEM's repository contains the following files:

- Generic docker-compose
- Prefilled configuration file (except password)
- Procedure and scripts to generate self-certified SSL certificates
- Procedure to deploy connector
- Procedure to test connector
- Python scripts to interact with connectors: add resource on a provider connector / use data from a provider connector
- Small API examples that can be used between data and connector.

Connector deployment (step by step, according to the deployment documentation README file)

- Partner clones the private repository
- Edit the configuration files
- Generate its SSL certificates if they didn't have one yet: private key, crt file, cert file, keystore file
- Send the cert file to CSEM
- CSEM registers the partner connector to PILATUS DAPS
- CSEM sends the truststore file that contains the certificates of each partner to all partners in the data space
- Partner add truststore and its keystore files to their connector.
- Partner launches their connector with the command: docker-compose up -d
- Partner can test to download data from CSEM data
- If needed, partner can test the sharing of data.

Additional security information:

- Connector private key stays on its server
- DAPS validates connector identities
- Connectors only accept requests from other connectors of the PILATUS data space.

3.3.3 Testing phase with partners: lessons learned.

As of this writing, all WP5 participating partners have tested the deployment of connectors as data consumers. However, deployment as data providers— i.e., serving data sources—has yet to be finalized and tested by some partners and is in progress at EURAC and Meyer Burger. Integrating partners into the data space and aiding in deployment proved to be more difficult and time-consuming than anticipated. Several security concerns and technical issues were prompted by partners acting as data providers because of the potential sensitivity of the data shared. Most of these issues, and the solutions/answers provided by CSEM, are listed below. Technical issues and answers/solutions were discussed during WP monthly meetings to ensure transparency.

General issues

Getting in contact with IT admins

CSEM encountered a recurring challenge in facilitating the deployment of connectors with partners: identifying the appropriate IT personnel within the partner organizations to execute deployment and conduct tests of the connectors proved to be uneasy. Given that the connectors utilize Docker and are most effectively administered on Linux servers, individuals with Linux administration skills are required

for data space deployment and administration. Regrettably, IT personnel from partners were not included during the project's scoping and definition phases, making it occasionally difficult to connect with partners' staff possessing the requisite Linux IT admin skills.

Preference to self-hosted cloud storage solutions for sharing sensitive data

Manufacturing partners showed a preference for their existing cloud storage solutions for data exchange (secure cloud storage exchange platform) and were reluctant to use an additional data exchange mean, mostly because of security standards. This issue could be partially lifted by explaining to partners the features of the data space connectors, as outlined in Sections 1.2 and 1.3 and by putting additional measures in place in addition to the standard IDS security measures (IP restriction, GUI Authentication, SSL encryption of GUI). There is a reluctance to a new technology that could be mitigated by having clear information and trainings on the topic. A good idea to limit the reluctancy for sensitive data is doing a comparative assessment of their security features and the dataspace security. That would allow to clarify whether more security features need to be added or motivate the implementation of dataspace.

Technical issues

GUI authentication.

The pre-implemented GUI tool from the IDSA Dataspace Connector has no authentication layer by default (but to download data, users must give the connectors passwords, so at worst only the data catalogue is exposed). CSEM provided the following answer: first the GUI must not be exposed to the internet (this is only a nice to have feature for debug and administration), moreover, the GUI can be deployed only on a machine with restricted access, or an authentication layer can be added using a proxy (like [Nginx](#)).

Web portal SSL encryption

Similarly, SSL encryption of the GUI web portal can be added using a proxy like Nginx.

IP access restriction

In the worst-case scenario, a third-party actor could illegitimately access data exposed to the connectors by getting a partner's IDs and certificates (even if the DAPS is here to ensure the unique authentication of all participants). To add an additional layer of security, IP access restriction to data provider connectors has been put in place.

Self-signed certificates vs. third-party signed certificates

An issue was raised about the potential security risks of self-signed certificates for DAPS authentication. CSEM provided the following answer: self-signed certificates have the same security level as certified (third-party signed) certificates. Moreover, only CSEM is registering the certificates that can be used in the PILATUS Data space, and the private key of the certificate stays with the partner and is not exchanged. Usage of third-party signed certificate is not advised in the particular case of the data space DAPS authentication.

3.4 Contribution to project objectives

The work in this deliverable and future work in WP5 contributes to objective 3, especially 3.c "Implementation of Manufacturing Execution System (MES) for product and process data collection and traceability" (first implementation of such a European data space for the PV industry).

3.5 Contribution to major project exploitable result

The work here contributes to key result "Data space and analytics apps for quality Optimisation".

4 Conclusion and future steps

The current deliverable from the PILATUS project establishes the groundwork for a data space tailored to the solar industry. It offers a thorough functional blueprint centred on IDSA dataspace connectors, a DAPS implementation from the Fraunhofer ISST, and additional enhanced security features (in particular IP restrictions). The deliverable has introduced a comprehensive vocabulary designed for data exchange within the solar industry, encompassing solar cells and modules production lines, defect classifications, and field performance and reliability assessments. Feedbacks from partners during the deployment and testing phase have been collected.

A significant takeaway from the testing phase is the necessity to involve IT administrators from the onset of the project. This ensures timely and accurate deployment on the partners' end and would minimize potential security-related challenges.

The now-running connectors will be used in WP4 and WP5 of the PILATUS project, especially in task T5.4, which focuses on data acquisition and causal analysis across the entire PV value chain, including quality tests from manufacturing lines (modules and cells) and field performance measurements of the modules at research facilities.

5 Risks and interconnections

5.1 Risks/problems encountered

Risk No.	What is the risk	Probability of risk occurrence ¹	Effect of risk ¹	Solutions to overcome the risk
WP5.1	The data space connector is connecting to MES databases and can be a gateway for external attacks or disrupt the production process	2	1	Apply additional security features to the data space connectors (Nginx for GUI, IP restrictions). Further security enhancement measures can be implemented if aligned with partners internal security needs.
WP5.4	Not all data-providing partners have deployed the connectors in data provider mode by the start of WP5.4	2	2	CSEM provides continuous support to partners for deployment.

¹⁾ Probability risk will occur: 1 = high, 2 = medium, 3 = Low

6 Deviations from Annex 1

None

7 References

- [1] 'IDS-RAM V4.2.0'. [Online]. Available: https://github.com/International-Data-Spaces-Association/IDS-RAM_4_0/
- [2] W. Stallings, *Network security essentials: applications and standards*, Sixth edition. Upper Saddle River: Pearson, 2016.
- [3] 'OneDrive Security: How safe are your files in 2023'. [Online]. Available: <https://www.cloudwards.net/onedrive-security/>
- [4] 'OWASP Top Ten list'. [Online]. Available: <https://owasp.org/www-project-top-ten/>
- [5] '2023 Data Breach Investigations Report', Verizon. [Online]. Available: <https://www.verizon.com/business/resources/T4f1/reports/2023-data-breach-investigations-report-dbir.pdf>
- [6] 'Microsoft Sharepoint Server: Security vulnerabilities'. [Online]. Available: https://www.cvedetails.com/vulnerability-list/vendor_id-26/product_id-11116/Microsoft-Sharepoint-Server.html
- [7] *Technical specification: solar photovoltaic energy systems : terms, definitions and symbols*, 3rd ed. Geneva, Switzerland: International Electrotechnical Commission, 2016.
- [8] M. Köntges, S. Kurtz, C. Packard, U. Jahn, K. A. Berger, and K. Kato, *Performance and reliability of photovoltaic systems: subtask 3.2: Review of failures of photovoltaic modules: IEA PVPS task 13: external final report IEA-PVPS*. Sankt Ursen: International Energy Agency, Photovoltaic Power Systems Programme, 2014.
- [9] 'Dynamic Attribute Provisioning Service (DAPS)', [Online]. Available: https://www.dataspaces.fraunhofer.de/de/software/identity_provider.html
- [10] IDSA, 'Data Connector Report', Feb. 2023. [Online]. Available: https://internationaldataspaces.org/wp-content/uploads/dlm_uploads/IDSA-Data-Connector-Report-February-2023-1.pdf
- [11] 'Docker Compose Overview', [Online]. Available: <https://docs.docker.com/compose/>
- [12] T. Bellebaum and C. Banse, 'Omejdn Configuration for the DAPS use case'. [Online]. Available: <https://github.com/International-Data-Spaces-Association/omejdn-daps>
- [13] 'Docker Desktop'. [Online]. Available: <https://www.docker.com/products/docker-desktop/>

8 Acknowledgement

The author(s) would like to thank the partners in the project for their valuable comments on previous drafts and for performing the review.

Project partners:

#	Partner short name	Partner Full Name
1	UNR	Uniresearch BV
2	MBG	Meyer Burger (Germany) GmbH
3	MBI	Meyer Burger (Industries) GmbH
4	FhG	Fraunhofer Gesellschaft zur Forderung der Angewandten Forschung EV
5	FZU	Fyzikalni Ustav AV CR V.V.I
6	EURAC	Accademia Europea di Bolzano
7	EXATEQ	Exateq GmbH
8	TNO	Nederlandse Organisatie Voor Toegepast Natuurwetenschappelijk Onderzoek TNO
9	NCR	Norwegian Crystals AS
10	ULIEGE	Universite de Liege
11	PADA	Finproject SpA
12	ISRA	ISRA VISION GmbH
13	CSEM	CSEM Centre Suisse d'Eletronique et de Microtechnique SA – Recherche et Developpement
14	MBCH	Meyer Burger AG
15	MBR	Meyer Burger Research AG
16	PASAN	PASAN SA
17	WCH	Wacker Chemie AG
18	EPFL	École Polytechnique Fédérale de Lausanne
19	CPT	Cambridge Photon Technology Limited

Disclaimer/ Acknowledgment



Copyright ©, all rights reserved. This document or any part thereof may not be made public or disclosed, copied or otherwise reproduced or used in any form or by any means, without prior permission in writing from the PILATUS Consortium. Neither the PILATUS Consortium nor any of its members, their officers, employees or agents shall be liable or responsible, in negligence or otherwise, for any loss, damage or expense whatever sustained by any person as a result of the use, in any manner or form, of any knowledge, information or data contained in this document, or due to any inaccuracy, omission or error therein contained.

All Intellectual Property Rights, know-how and information provided by and/or arising from this document, such as designs, documentation, as well as preparatory material in that regard, is and shall remain the exclusive property of the PILATUS Consortium and any of its members or its licensors. Nothing contained in this document shall give, or shall be construed as giving, any right, title, ownership, interest, license or any other right in or to any IP, know-how and information.

This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101084046. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.