

## **Glossary**

# Contents

**Glossary of Terms and Concepts.....3**

# Glossary of Terms and Concepts

---

## **access policy**

An access policy is a set of logical rules that determines how users can consume data stored in PHEMI Central. The access policy specifies what user authorizations are required to interact with data tagged with specified sensitivity, or visibility. Access policies can be applied to data sources and datasets.

## **authorizations**

User authorizations are configurable labels you can assign to PHEMI Central users. Authorization definitions are configured in PHEMI Central by the PHEMI Administrator in accordance with the organization's governance policies.

## **cell (field)**

A cell, or field, is the smallest unit of data storage in PHEMI Central. A cell is a single data item, which can range from a single byte up to gigabytes, plus the metadata associated with the data item. Any piece of raw data, regardless of size, is stored in a single cell. Elements of derived data (transformed from the raw data) are also each stored individually in cells. Any cell can be protected by applying data visibilities. For derived data, each derived item can be individually assigned a visibility (which may be different than that configured for the data source) by the DPF performing the processing.

## **code library**

A code library is a package of executable code that is included in a DPF archive. Whether the code is source or compiled depends on the coding language. Code libraries must be portable and self-contained; that is, all dependencies required for the DPF to function must be bundled inside the library, in the appropriate way, for whatever language is being used.

## **dataset**

A dataset is a view, or map, of an underlying set of data. Data items in a dataset can be selected from across multiple data sources. The dataset is a view, or map, to the underlying data. The actual content of the dataset (that is, the dataset's data) is generated when the dataset is executed or when it is queried against.

## **data category**

Data categories are a way to classify data into broader groupings. Examples of data categories are "Research Reports," "X-Rays," and "Prescriptions."

## **Data Processing Function, DPF**

A Data Processing Function, or DPF, is an executable piece of code that supplies the instructions for processing raw data (for example, a log message or medical report) to extract from heterogeneous data sources meaningful, context-specific information (such as a temperature reading or blood glucose measurement) that can be queried or exported for analysis. The code is executed by the PHEMI Central DPF Engine, which uses it to direct curation of the data. The input to a DPF is the raw binary data ingested into the system. The output of a DPF is a set of structured elements, each of which includes a type property (for example, INT or STRING) and can specify data visibilities (for example, SECRET or IDENTIFIABLE) on a per-field basis. The data elements output by a DPF are called derived data. The collection of derived data produced by a DPF is automatically indexed in PHEMI Central.

## **data source**

In PHEMI Central, a data source is a given collection of data, such that the information is governed by a single set of management and governance rules and policies. A data source configuration should be defined for any collection of data to be stored in the system and managed by the same retention, legal, and governance rules.

## **data visibilities**

See visibilities.

**derived data**

Derived data is data that has been parsed, extracted, or otherwise enriched or processed by running a DPF on stored raw data. The set of derived data items can be searched, queried, further processed, or exported from the system.

**digital asset**

A digital asset is any piece of data stored with metadata in the system. This may be raw data that has had metadata applied on collection, or it may be derived data that has been parsed, indexed, catalogued, and/or enriched with additional metadata.

**DPF archive**

A DPF archive is the set of code that makes up a DPF. A DPF archive is delivered as a ZIP file archive. It consists of two parts: a manifest file and a code library. To associate a DPF with a data source, the DPF archive is ``registered`` with the data source by uploading the archive during data source configuration.

**ingestion**

Ingestion is the process by which data is brought into in PHEMI Central. The sending system (the data source) submits the data to PHEMI Central, which listens for the data using a web service. Data can also be ingested manually, by using the PHEMI Central Management and Governance Console. The specific characteristics of data ingestion can be specified per data source as part of the data source configuration.

**JSON**

JSON stands for JavaScript Object Notation. JSON is a lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate. JSON is used in the body of several REST requests in the PHEMI RESTful API. PHEMI Central also includes a system DPF that can create derived data from JSON objects, providing the objects conform to PHEMI's JSON specification.

**key-value pairs**

A key-value pair is a set of two linked data items: a key which uniquely identifies some item of data, and the data itself. PHEMI Central uses key-value store to efficiently store, process, and retrieve data.

**M2M**

M2M is a way of referring to machine-to-machine interfaces, used in machine-to-machine communication.

**manifest file**

A manifest file is a JSON file that specifies the output of a DPF. With the code library, the manifest file makes up the DPF archive that is uploaded to register the DPF with a data source. The manifest file should include the properties of the DPF along with the details of each derived data item to be generated.

**metadata**

Metadata is information about a piece of data. In PHEMI Central, metadata is information about how a given piece of data is to be managed. When a piece of raw data is ingested into PHEMI Central, information from the connection (for example, the timestamp) together with policy information configured for the data source (for example, the data visibility) and some derived information (for example, a "time to live," as derived from the timestamp and the data retention policy) is used to create metadata properties that are stored with the data. Further, PHEMI Central also automatically indexes and catalogues all stored data, whether raw or derived; the indexes and catalogues can also be considered a kind of metadata.

**PII**

Personally Identifiable Information, or PII, is a legal concept used in US privacy law and information security to mean information that can be used on its own or with other information to identify, contact, or locate a single person or to identify an individual in context. When thinking about PII, it is important to distinguish legal requirements to remove attributes uniquely identify an individual from a general technical ability to identify individuals. Because of the versatility and power of modern re-identification algorithms, together with the amount

of information freely available from all sources, the absence of PII data does not guarantee that de-identified data cannot be used, perhaps in combination with other data, to identify individuals.

### **privacy-level visibilities**

Privacy-level visibilities are data visibilities that characterize the privacy level of a data item. PHEMI Central includes predefined privacy-level visibilities designed to apply to data domains where privacy is important.

- **IDENTIFIED.** The data contains Personally Identifying Information that potentially identifies an individual. Examples of information of this type include name, Social Insurance Number, and date of birth.
- **DE-IDENTIFIED.** The data contains IDENTIFIED information that has been masked or encrypted.
- **NON-IDENTIFIED.** The data is not identifying in and of itself. Examples of this type of information include weight or favorite food.

Although privacy-level visibilities are preconfigured, their descriptions can be modified by configuration.

### **raw data**

In PHEMI Central, raw data items are files, objects, records, images, and so on that are submitted for ingestion into the system. Raw data is stored exactly as received, along with the metadata generated for it on ingestion.

### **REST, RESTful API**

Representational Statement Transfer (REST) is an architectural style that uses HTTP requests and associated methods (POST, PUT, GET, and DELETE) to create, update, read, and delete data. A RESTful API is an application programming interface (API) based on REST.

### **visibilities**

All raw data and derived data stored in PHEMI Central can be tagged with labels that provide information about the data's sensitivity. This sensitivity is described in terms of the visibility the data should have to different system users. The visibility tags you define for your data should reflect the sensitivity of the data as identified by your organization.