Chapter 6

Catalog and Entity Management Service for Internet of Things-Based Smart Environments

Umair ul Hassan, Adegboyega Ojo, and Edward Curry

Keywords Entity management · Knowledge graphs · Data catalog · Dataspace · Internet of Things · Smart environments

6.1 Introduction

A fundamental requirement for intelligent decision-making within a smart environment is the availability of information about entities and their schemas across multiple data sources and intelligent systems. This chapter first discusses how this requirement is addressed with the help of catalogs in dataspace; it then details how entity data can be more effectively managed within a dataspace (and for its users) with the use of an entity management service. Dataspace provide a data co-existence approach to overcome problems in current data integration systems in a pay-as-you-go manner. The idea is to bootstrap the integration with automated integration, followed by incremental improvement of entity consolidation and related data quality. The catalog and entity management services are core services needed to support the incremental data management approach of dataspace. We provide an analysis of existing data catalogs that can provide different forms of search, query, and browse functionality over datasets and their descriptions. In order to cover the entity requirements, the catalog service is complemented with an entity management service that is concerned with the management of information about entities.

The chapter is organised as follows. Section 6.2 introduces the important role of entity data. Section 6.3 lists the key requirements and challenges of implementing a catalog and entity service in a dataspace. Section 6.4 examines existing catalogs as described in the literature. Section 6.5 details the implementation of a catalog in the dataspace, with Sect. 6.6 detailing the entity management service. Section 6.7 details the access control service, while Sect. 6.8 describes how a data source joins the dataspace. Finally, Sect. 6.9 summarises the chapter.

6.2 Working with Entity Data

Within a smart environment, analytical and operational activities of intelligent systems revolve around entities of interest. For example, within intelligent energy systems, energy consuming entities (i.e. electrical devices, lights, heating units) are the main entities of interest whereas products and customers are the primary entities for intelligent marketing systems. Typically, in a smart environment, the information about core entities is spread across data silos, including inventory systems and customer relationship systems. Consolidation of this information is known to be among the top priorities of data managers [154]. However, successful integration of information requires overcoming the heterogeneity of data that exists at various levels of detail [155]. Consider the example of a marketing analyst who is preparing a report on a set of company products. For this purpose, the analyst has some data available in a spreadsheet on their local computer that needs to be consolidated with data available in the company's billing system. The first challenge in such consolidation exists at the information representation level due to different data formats and semantics of data models used for describing the products. Once both datasets have been converted to a common format and schema, the analyst will need to perform four actions: (1) discover mapping relationship between attributes of product schemas in the spreadsheet and billing system; (2) determine equivalence relationships among products stored in both data sources; (3) merge the values of mapped attributes for equivalent products to generate a consolidated dataset; and (4) clean the resultant dataset for redundant or conflicting attribute values.

There are several process-oriented methodologies and technical tools available to minimise the manual effort required to achieve the analyst's data integration and data quality workflow. However, a fundamental requirement of integration is the availability of exact information about entities and their schemas across multiple data sources. This chapter first discusses how this requirement is addressed with the help of a catalog in a dataspace, it then details how entity data can be more effectively managed within a dataspace (and for its users) with the use of an entity management service.

6.3 Catalog and Entity Service Requirements for Real-time Linked Dataspaces

Driven by the adoption of the Internet of Things (IoT), smart environments are enabling data-driven intelligent systems that are transforming our everyday world, from the digitisation of traditional infrastructure (smart energy, water, and mobility), the revolution of industrial sectors (smart autonomous cyber-physical systems, autonomous vehicles, and Industry 4.0), to changes in how our society operates (smart government and cities). To support the interconnection of intelligent systems in the data ecosystem that surrounds a smart environment, there is a need to enable the sharing of data among intelligent systems.

6.3.1 *Real-time Linked Dataspaces*

A data platform can provide a clear framework to support the sharing of data among a group of intelligent systems within a smart environment [1] (see Chap. 2). In this book, we advocate the use of the dataspace paradigm within the design of data platforms to enable data ecosystems for intelligent systems.

A dataspace is an emerging approach to data management that recognises that in large-scale integration scenarios, involving thousands of data sources, it is difficult and expensive to obtain an upfront unifying schema across all sources [2]. Within dataspace, datasets *co-exist* but are not necessarily fully integrated or homogeneous in their schematics and semantics. Instead, data is integrated on an “*as-needed*” basis with the labour-intensive aspects of data integration postponed until they are required. Dataspace reduce the initial effort required to set up data integration by relying on automatic matching and mapping generation techniques. This results in a loosely integrated set of data sources. When tighter semantic integration is required, it can be achieved in an incremental “*pay-as-you-go*” fashion by detailed mappings among the required data sources.

We have created the Real-time Linked Dataspace (RLD) (see Chap. 4) as a data platform for intelligent systems within smart environments. The RLD combines the pay-as-you-go paradigm of dataspace with linked data, knowledge graphs, and real-time stream and event processing capabilities to support large-scale distributed heterogeneous collection of streams, events, and data sources [4].

6.3.2 *Requirements*

To further support data integration and quality, an RLD must hold information about its participant data sources irrespective of whether they contain primarily static datasets or produce streams of highly dynamic data [19, 156]. Among the primary support services of a dataspace, the catalog service is responsible for managing detailed descriptions of all the data sources that form a dataspace [78]. At a basic level, the descriptions must contain information such as the owner, creation date, type of the data, and semantic information about the data source. At a more detailed level, the catalog must also describe the schema of a data source, the query endpoints, the accuracy of data, access licenses, and privacy requirements. Besides descriptions of individual data sources, the dataspace should also maintain descriptions of relationships between data sources in appropriate forms such as bipartite mappings, dependency graphs, or textual descriptions. The catalog must be able to accommodate a large number of data sources; support varying levels of descriptions about data sources and their relationships, and should make descriptions available in both human and machine-readable formats.

The catalog service plays a crucial role in providing information services for participants in the dataspace, including search, browse, and query services. The

catalog should also maintain, wherever possible, a basic entity management service in the form of an inventory of the core entities of interest that includes details on their identifier, type, creation date, core attributes, and associated data source. The catalog can then support simple queries that can be used to answer questions about the presence or absence of an entity in a data source or determine which source contains information on a particular entity. Furthermore, assigning canonical identifiers to entities supports data integration and enrichment as part of stream processing algorithms.

The following primary requirements for a catalog and Entity Management Service (EMS) are needed to support the incremental data management approach of dataspace:

- **Data Source Registry and Metadata:** The requirement to provide a registry for both static and dynamic data sources as well as their descriptions.
- **Entity Registry and Metadata:** The requirement to provide a registry for entities and their descriptions.
- **Machine-Readable Metadata:** The requirement to store and provide metadata about data sources and entities in machine-readable formats using open standards such as JavaScript Object Notation (JSON) and Resource Description Framework (RDF).
- **HTTP-Based Access:** The requirement to allow HTTP access to data source and entity descriptions.
- **Schema Mappings:** The capability to define mappings between schema elements.
- **Entity Mappings:** The capability to define mappings between entities.
- **Semantic Linkage:** The capability to define semantic relationships and linkages among schema elements and entities.

In addition to the above primary requirements, the following secondary requirements are important for the successful and sustained use of the catalog and an EMS:

- **Search and Browse Interface:** The requirement to provide a user interface over the catalog and EMS, which allows searching and browsing over all the elements stored.
- **Authentication and Authorisation:** The requirement to verify the credentials of users and applications accessing the catalog and EMS which can limit access to sources/entities based on access policies or rules.
- **Data Protection and Licensing:** The requirement to fulfill the privacy and confidentiality requirements of data owners and providing licensing information on the use of data.
- **Keyword Queries:** The requirement to support keyword-based queries over all the data stored in the catalog and EMS.
- **Provenance Tracking:** The requirement of tracking lineage of changes made to the catalog and EMS by users and applications.

6.4 Analysis of Existing Data Catalogs

This section provides a short analysis of some existing software and platforms that can be used for implementing data catalogs. The objective of this analysis is to provide a high-level overview of these software packages and their coverage of primary and secondary requirements identified in the previous section. This analysis focuses on a selected list of open-source software while readers are directed towards relevant industry reports to assess proprietary software [157]. In terms of data management, most commercial data catalogs have been developed over existing Master Data Management (MDM) solutions of software vendors. This shift from MDM to data catalogs is primarily driven by the concept of data lakes, which is an industry term used to refer to a loose collection of heterogeneous data assets in enterprises.

Table 6.1 lists the open-source software included in the analysis. QuiltData allows the creation and sharing of data packages using Python. The Comprehensive Knowledge Archive Network (CKAN) is primarily designed for implementing data portals for organisations that publish and share data. CKAN is widely used by public sector organisations and governments to publish open datasets. Dataverse is a web-based platform for data preservation and citation developed at Harvard University. It is primarily used to create a citable reference to a dataset that can be used in publications.

Similarly, the DSpace platform is designed to serve as a repository of digital assets, including multimedia, documents, and datasets. Another software for sharing and preserving research outputs is Zenodo, developed and maintained by CERN. By comparison, the Kylo project from Teradata is designed from a data integration perspective that includes a metadata registry for data sources. The difference between Kylo and other software is evident by the fact that Kylo has been developed by an industry leader where other software primarily originate from academia.

A quick analysis of Table 6.2 reveals that most of the open-source software is limited to addressing the requirements of maintaining a data registry and providing machine-readable access to metadata through HTTP. Most of the catalogs do not address the requirement to manage entity information and the need to provide mappings between schemas and entities. All of the catalogs, except CKAN and Kylo, provide a registry of datasets that are stored internally by the software. On the other hand, both CKAN and Kylo also register external data sources, thus addressing

Table 6.1 Existing open-source software for data catalogs

Open-source software	URL	License
QuiltData	https://github.com/quiltdata/quilt	Apache
CKAN	https://github.com/ckan/ckan	AGPL
Dataverse	https://github.com/IQSS/dataverse	Apache
DSpace	https://github.com/DSpace/DSpace	BSD
Zenodo	https://github.com/zenodo/zenodo	GPL
Kylo (Teradata)	https://github.com/Teradata/kylo	Apache

Table 6.2 Overview of requirements coverage of open-source data catalogs

	Primary					Secondary						
	Data source registry and metadata	Entity registry and metadata	Machine-readable metadata	HTTP-based access	Schema mappings	Entity mappings	Semantic linkage	Search and browse interface	Authentication and authorisation	Data protection and licensing	Keyword queries	Provenance tracking
Requirements												
QuitData	+	–	+	++	–	–	–	+	++	–	++	+
CKAN	++	–	++	++	–	–	–	++	++	+	++	+
Dataverse	+	–	+	++	–	–	–	++	++	+	++	+
DSpace	+	–	+	++	–	–	–	++	++	+	+	+
Zenodo	+	–	+	++	–	–	–	++	++	+	++	+
Kylo	++	–	+	++	+	–	–	++	+	+	+	+

++ requirement is well covered
+ requirement is partially covered
– requirement is not covered

a key requirement of a catalog in a dataspace. In terms of machine-readable data, all catalogs provide access to the metadata in JSON format, and CKAN provides data in RDF format.

In terms of secondary requirements, all data catalogs address the requirements with almost full coverage. However, data protection and provenance tracking requirements are only partially addressed by all software. Data protection and licensing requirements are mainly addressed by associating licenses with datasets or data sources in the catalog. Provenance tracking is only limited to the changes made to the metadata instead of the dataset or data source.

6.5 Catalog Service

Based on the coverage of the primary and secondary requirements identified, CKAN was chosen as the base software to create the catalog service for the RLD. The catalog extends the CKAN portal with the additional functionality necessary to cover the primary requirements for the RLD. The catalog service provides a registry of the following:

- **Datasets:** A dataset contains contextual information about a building or a thing within a smart environment, real-time sensors data, enterprise data (e.g. customer data, enterprise resource planning systems), or open data such as weather forecast data.
- **Entities:** An entity defines a concrete instance of a concept within the smart environment (e.g. a sensor or a water outlet). The catalog tracks critical entities in the smart environment and links them with the datasets and streams that contain further information about the entities. Metadata about an entity includes the identifier, entity type, and associated datasets.
- **Users and Groups:** Individual users can include data managers, data analysts, and business users who might belong to one or more groups divided along organisational structures or projects.
- **Applications:** Applications are the descriptions of software and services that utilise the dataspace and its services. For example, mobile applications, public displays, data services, analytic tools, web applications, and interactive dashboards.

6.5.1 *Pay-As-You-Go Service Levels*

Dataspace support services follow a tiered approach to data management that reduces the initial cost and barriers to joining the dataspace. When tighter integration into the dataspace is required, it can be achieved incrementally by following the service tiers defined. The incremental nature of the support services is a core enabler

of the pay-as-you-go paradigm in dataspace. The tiers of service provision provided by the catalog in the RLD follows the 5 star pay-as-you-go model (detailed in Chap. 4). The level of service provided by the catalog increases as follows:

- 1 Star* **Registry:** A simple registry of datasets and streams, only pointing to the interfaces available for access.
- 2 Stars* **Metadata:** Describing datasets and streams in terms of schema and entities in a non-machine-readable format (e.g. PDF document).
- 3 Stars* **Machine-readable:** Machine-readable metadata and simple equivalence mappings between dataset schemas to facilitate queries across the dataspace.
- 4 Stars* **Relationships:** Relations among schemas and concepts across the dataspace.
- 5 Stars* **Semantic Mapping:** Semantic mappings and relationships among domains of different datasets; thus, supporting reasoning and schema agnostic queries.

The main requirement not covered by CKAN was the need for more advanced support for entity management within the RLD (e.g. entity registry, schema and entity mapping, and semantic linkage). In order to cover these entity requirements, the catalog service in the RLD is complemented with an entity management service that is concerned with the management of information about entities.

6.6 Entity Management Service

Managing information about the critical entities in a smart environment is an essential requirement for intelligent decision-making applications that rely on accurate entity information. Similar to MDM, there have been efforts to develop web-scale authoritative sources of information about entities, for example, Freebase [158] and DBpedia [159]. These efforts followed a decentralised model of data creation and management, where the objective was to create a knowledge base. A similar authoritative source of entity information within a dataspace would significantly improve the experience of working with entity data.

Fundamental to the RLD approach is to treat entities as first-class citizens (as illustrated in Fig. 6.1) in the dataspace, which is achieved by using entity-centric knowledge graphs and support from the EMS. The EMS is concerned with the maintenance of information about entities within the smart environment and together with the catalog service acts as the canonical source of entity (meta)data. The EMS facilitates sharing and reusing of entity data within the RLD using (1) a knowledge-graph entity representation framework for structuring entity data and (2) standard ontology languages for defining the semantics of data [160]. Ontologies, also referred to as vocabularies, provide a shared understanding of concepts and entities within a domain of knowledge which supports automated processing for data using reasoning algorithms.

Fig. 6.1 Example of managed entities in the Entity Management Service [4]

The relationship of entities across data sources and intelligent systems in a smart environment can quickly become complicated due to the barriers of sharing knowledge among intelligent systems. This is a significant challenge within traditional data integration approaches and the use of linked data and knowledge graph techniques that leverage open protocols and W3C that can support the crossing of knowledge boundaries when sharing data among intelligent systems.

The EMS leverages the principles of linked data from Tim Berners-Lee (see Chap. 2) [41] and adapts them to the management of entities. Thus, the EMS has the following “Linked Entity” principles:

- **Naming:** Each managed entity within the EMS is identified using a Uniform Resource Identifier (URI). Managed entities can be a person, a building, a device, an organisation, an event or even concepts such as risk exposure or energy and water consumption.
- **Access:** Each managed entity within the EMS can be accessed via an HTTP-based URI which can be used to retrieve detailed entity data.
- **Format:** When an entity URI is looked up (i.e. dereferenced) to retrieve entity data, useful information about the entity is provided using open-standard formats such as RDF or JSON-LD.

- **Contextualisation:** Entity data includes URIs to other entities so that more information can be discovered on-the-fly. Referencing other entities, through URIs, thus creates a knowledge graph that could be traversed by automated software to discover and link information with the dataspace.

6.6.1 *Pay-As-You-Go Service Levels*

Similar to the tiered approach used by the catalog, the level of active entity management follows the 5 star pay-as-you-go model of the RLD. The entity management service has the following levels of incremental support:

- 1 Star **No Service:** Entities are not managed.
- 2 Stars **Documented:** Entity descriptions (e.g. schema and identifiers) are documented in a non-machine-readable format (e.g. PDF document).
- 3 Stars **Source-level:** Machine-readable entity at the source-level.
- 4 Stars **Multi-source Mapping:** Canonical identifiers for entities in the dataspace and mapping across sources.
- 5 Stars **Entity Knowledge Graphs:** Entities are semantically linked to other related entities, data, and streams across the dataspace to create a knowledge graph.

6.6.2 *Entity Example*

The EMS follows the incremental dataspace philosophy; in practice, you only connect data sources related to an entity on an as-needed basis. The approach encourages that entities should be as minimal as possible to achieve the desired results. Figure 6.2 describes a minimal data model for entities in one of the smart water pilots.

The key entities of the data model and the sources they originate from are:

- **Sensor:** Measures the flow of water and generates a stream of data to calculate the water consumption levels of the area covered by the sensor (from the Internet of Things Platform).
- **Observation:** The sensor output including the units and rate of measurement (from the Internet of Things Platform).
- **Outlet:** Information on the actual physical water outlet is necessary for analysis and decision-making. It is possible that a single sensor might be installed for a set of outlets. In such cases, a cumulative assessment of water consumption is needed (outlet description crowdsourced using human task service).
- **Location:** Information on the associated spatial locations serviced by the water pipe (from Building Management System).
- **User Group:** Each sensor is associated with a set of users who have permission to access the data (from Enterprise Access Control). The access control service leverages this information.

Fig. 6.2 Minimal data model for entities in an intelligent water management system [4]

6.7 Access Control Service

The access control service ensures secure access to the data sources defined in the catalog. Access is managed by defining access roles for applications/users to the data source/entity that are declared in the catalog. The access control service is an intermediary between the applications/users and the dataspace by using the catalog as a reference to verify access for applications/users to the actual data sources. The advantage of this approach is to keep the applications/user’s profiles centrally managed by the catalog under the governance of the dataspace managers. Within the pilot deployments, we defined three types of roles for access control: (1) dataspace managers, (2) application developers/data scientists, and (3) end-users. To simplify the process of securely querying data sources, the access control service offers a secure query service to applications. As illustrated in Fig. 6.3, the workflow of an application using the secure query capability of the access control service and the roles of the users are as follows¹:

¹Demo Video available at this link: <https://www.youtube.com/watch?v=KukUd5VCheY>

Fig. 6.3 Query workflow using the access control service [4]

1. The user connects to the application (App1).
2. The application maps the user ID to its profile and access as the secure query service via an identification token (API Key).
3. The query service verifies the application ID and its API Key and checks that it has the right to access the data source (e.g. dataset or an entity).
4. Authorisation results are sent back to the query service.
5. If the user is authorised, the query service gets the data from the source.
6. Results from the data source are sent back to the query service.
7. The query service sends the data to the application.
8. The application returns the data to the user (e.g. via a UI or a file).

6.7.1 Pay-As-You-Go Service Levels

In terms of tiered levels of support for the access control service, this is defined by the capability to increasingly limit access to more fine-grained levels within a data source. The access control service has the following levels of service:

- 1 Star **No Service:** The access control service does not manage the source.
- 2 Stars **Coarse-grained:** Access is limited to the user at the dataset level.
- 3 Stars **Fine-grained:** Access is limited to users at the entity level with the use of the secure query service.

- 4 Stars **Data Anonymisation:** Access to sanitised data for privacy protection. (Not supported in pilots).
- 5 Stars **Usage Control:** Usage of the data is controlled as it moves around the dataspace. (Note: This functionality is not currently implemented).

6.8 Joining the Real-time Linked Dataspace

The RLD is composed of multiple data sources, including real-time sensor streams, historical databases, large text files, and spreadsheets. The RLD adopts a pattern in which the publisher of the data is responsible for paying the cost of joining the dataspace. This is a pragmatic decision as it allows the dataspace to grow and enhance gradually. For a data source to become a part of the dataspace, it must be discoverable and must conform to at least the first star rating of the RLD. The registration process entails detailing the metadata of the source, which helps users of the catalog in locating and using the data source. A seven-step approach has been defined for including a data source into the RLD (see Fig. 6.4). The seven steps are Register, Extract/Access, Transform, Load, Enrich, Map, and Monitor (RETLEMM). Some of the steps are optional and depend on the capability of the data source to meet requirements around machine-readable data and query & search capabilities and interfaces. The RETLEMM steps are:

- **Register:** A new data source joining the dataspace would require it to be registered in the dataspace catalog. The registration means that the catalog contains an entry describing the data source at a minimum regarding type, access, and format. Completion of this step will give the data source a rating of one star and the data source is considered part of the dataspace since it can be accessed and

Fig. 6.4 RETLEMM process for a data source to join the RLD

used. Further optional information about the data source can include the physical address of files, a query interface/endpoint, additional metadata, and entity data.

- **Extract/Access:** The second step of the process is to allow access to the data from the data source in a machine-readable format; this will rate the source as a minimum of 2 stars. How data is accessed depends on the data source; for simple sources with limited capability (e.g. Excel), the data may need to be extracted. For more sophisticated data sources (e.g. database), the data may be accessible via a query interface. To demonstrate all the process steps in this example (see Fig. 6.4), it is assumed that information is extracted in the form of CSV files. The use of an open format will move it to 3 stars.
- **Transform:** Given the CSV representation, the next step is to convert the data into an appropriate format for publishing. A simple semi-automated process for transforming the CSV files to RDF files is possible using tools such as Microsoft Excel and OpenRefine. A similar process can be used to perform an on-the-fly transformation of the results of a database query (see Adapters below). This step moves the data towards 4 stars.
- **Load:** Once the data has been converted and represented in the RDF format, the next step is to store it in an appropriate data store. For this step, any general-purpose RDF store may be used. However, it is necessary for the RDF store to have the necessary publishing, querying, and search functionalities to support applications. This step is not necessary where the data source has a queryable interface and results can be transformed into on-the-fly RDF. The data is now 4 stars.
- **Enrich:** The above steps are enough to support analytical and decision support applications. Nevertheless, it is desirable to enhance the metadata with additional information such as links to related entities in other datasets. This optional step adds contextual information to achieve the overall entity-centric vision of the RLD. The data will move towards 5 stars.
- **Map:** Similar to the enrich step, the schema and entities of a data source may be mapped to other data sources and entities in the catalog. This facilitates integration and deduplication of classes and entities. Also, it allows the automated processing of data collected from multiple datasets using advanced reasoning and schema agnostic query tools. The data will now be 5 stars.
- **Monitoring:** It is not unusual for a data source to change or update its definitions and attributes. These changes can introduce data quality issues and errors which can affect the performance of the dataspace. The RLD utilises a simple monitoring process to check for changes in data sources in terms of availability and data quality.

When a data source joins the dataspace, the RETLEMM process can be performed manually by the data source owners with the help of the dataspace support services. However, the Extract, Transform, and Load (ETL) steps can be automated to speed up the process. Automation is desirable for large-scale historical data and real-time metering data. In the following, we discuss the two alternatives for automation:

- **Adapters:** Adapters can be considered a non-materialised view of a data source. They encode the ETL process in the form of mappings between the source data

format and the target data format. In the case of a historical database, the data resides in the source, and the ETL is performed on-the-fly every time queries are posted on a non-materialised view. In the case of a real-time data stream, the ETL is performed on-the-fly as data is generated by the streaming source.

- **Scheduled Jobs:** This form of the ETL process is performed either once for a large static database or periodically for a large dynamic database. It is a common activity among existing data warehouse implementations.

6.9 Summary

This chapter underlines the need for a catalog service for successful implementation of Real-time Linked Dataspaces. Specifically, it is established that the catalog should not only serve as a registry of data sources in a dataspace but also provide an entity management service. Based on a set of requirements identified for the catalog, a short analysis of existing open-source software is provided to assess their coverage of requirements. The design of the catalog and entity management service for the Real-time Linked Dataspace is detailed including aspects such as tiered services levels, entity modelling, access control, and the process for a data source to join the dataspace using the catalog and entity management service.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.