From Modelling to Designing:
A Streaming Network for Multi-Tenant Digital Twin Platforms

Based on the Reference Architecture for Industry 4.0.
Designed for Cross-Domain Applications.

Christoph Schranz, Mathias Schmoigl, Felix Strohmeier
Internet of Things
Salzburg Research
Salzburg, Austria
[prename].[surname]@salzburgresearch.at

***Abstract*—Modern business models rely increasingly on the interoperability of various Cyber-Physical Systems (CPS) and software systems. Different tenants, like producers, operators, suppliers and maintainers, are interested in different aspects of the system and therefore require different data of an asset. As those tenants demand different subsets of data of a CPS, complex entangled data flows emerge that are difficult to depict efficiently using traditional peer-to-peer data streaming. Even though multiple generic streaming platforms exist, the actual problem of entangled data flows is often neglected. The purpose of this paper is to reduce the complexity of modern multi-tenant, cross-enterprise streaming networks. Methodically, the Reference Architecture Model for Industry 4.0 (RAMI 4.0) is exploited to conceptualize a streaming network architecture that enables the scalable sharing of data between multiple tenants independently of their domain. Based on this concept, a Digital Twin Platform will be designed which will help to realize smart city visions.**

Keywords-CPS; Digital Twin; RAMI4.0; Industry 4.0; multi-tenant; multi-stakeholder; data streaming; streaming networks; smart-city;

I. INTRODUCTION

The Reference Architecture Model for Industry 4.0 (RAMI 4.0) was introduced by the German organization “Plattform Industrie 4.0” and is illustrated in Fig. 1. The model spans a three-dimensional room that helps to categorize “Industry 4.0” – components.

With the official publication of its norm DIN SPEC 91345:2016-04 [2] in April 2016, the model has gained momentum in manufacturing, where it supports structuring components and getting a common understanding of a complex architecture. Additionally, RAMI 4.0 comes with the administration shell, which is a conceptual layer that abstracts physical components in order to interact digitally with others [3]. However, a detailed consideration of the administration shell itself is out of scope for this paper, but will be part of further investigations.

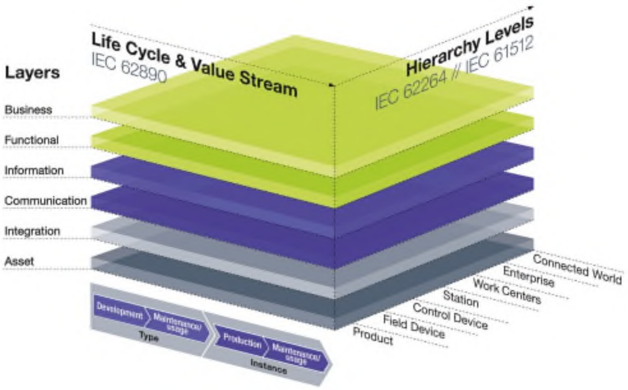

Figure 1. Reference Architecture Model for Industry 4.0 (RAMI 4.0). [1]

The manufacturing sector did not only establish a reference architecture, but also a de-facto-standard for the communication of devices. The Open Platform Communication – Unified Architecture (OPC-UA) is a protocol that is currently supported by a majority of “Industry 4.0” devices. In Fig. 2, OPC-UA and other protocols are mapped onto the ISO/OSI-Layers. In this case they are orthogonal to the Life-Cycle/Value Stream and hierarchy levels of RAMI 4.0 to visualize potential protocol lacks in the industrial domain.

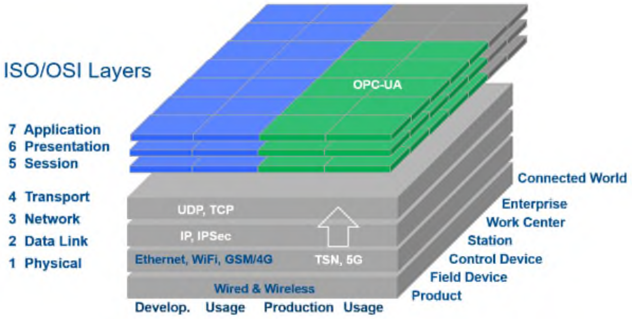

Figure 2. OPC-UA mapped onto OSI Layers and RAMI 4.0. [1]

The blocks marked in blue depict ISO/OSI Layers in the Development and Usage Life-Cycle, where an object only exists digitally, e.g., as a type design. The blocks in grey on the upper right, however, are associated with material instances in an “Enterprise” or “Connected World” hierarchy

ISBN: 978-1-61208-753-5

level. It indicates that OPC-UA fits for a “Work-Center” and levels below, but if data has to be shared across companies, it meets its limitations.

Moreover, the traditional concept of a stream of data from producers to consumers using a set of pipelines lacks when it comes to multi-tenancy, where tenants are interested in different compositions of subsets of multiple CPS. As an example, a single manufacturer of production machines delivers them to several customers. The manufacturer is interested in his/her machine’s data. Additionally, the customers - including operators, maintainers and logistic partners - would like to utilize a subset of the machine’s data to enhance their own production and maintenance process. Considering this scenario, each machine sends subsets of data to a tenant. However, as soon as additional machines are allowed, the 1:N relation between data producer and consumer expands to a M:N relation, because one machine sends its data to multiple tenants and one tenant can consume data from multiple producers.

As illustrated in Fig. 3, Arquimedes Canedo models data producers and tenants as nodes and subgraphs:

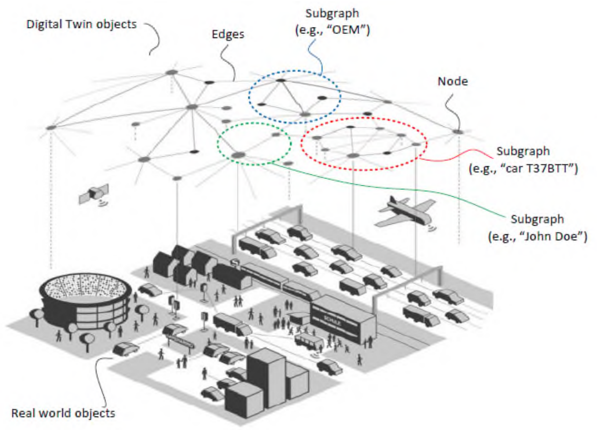

Figure 3. A. Canedo composes Nodes to Subgraphs in a Smart City. [4]

He describes his own scenario as follows:

„[…] real world objects such as cars, people, buildings, airplanes, highways, houses, transportation systems are represented digitally as Digital Twins. A real-world object is not represented by a single node, but by a subgraph of nodes and edges. For example, a car „T37BTT" is represented by multiple nodes and edges in a subgraph.“ – A. Canedo [4]

Hereby, the important term “Digital Twin” was mentioned, which can be regarded as an abstraction of a material or immaterial thing, which serves multiple purposes [5]. Therefore, a Digital Twin that refers to a real instance is often used as a synonym for CPS.

The representation of a real-world object also depends on its purpose. Different tenants that cope with the same “Subgraph” car during its usage, require different data to perform their job properly:

- The producer is interested in all kinds of feedback data to improve the production quality.
- The operator/driver is interested in data that enables or enhances the service “mobility”.
- The infrastructure provider could be merely interested in the car’s observation of the environment.
- The maintainer of the car is interested in data that supports his/her job, e.g., model numbers, operating distance/hours, detected anomalies, exhaust gas compositions, etc.
- The supplier of a component is interested in whether an updated version works as expected or not.
- The merchant is interested in parameters that determine the current value of the new or pre-owned car.
- Governments of countries where the car is used need to know, if it complies with national legal regulations.
- The automobile insurance may like to adjust its fee dynamically based on the individual driving style.

Hence, a single tenant requires only a subset of a CPS’s data. As soon as the number of assets grows, the requirements on the data flows get more complex, as, e.g., a producer would like to receive data from all his/her assets, the operator of all assets in the plant, and so forth.

In addition to the number of interconnected producers and consumers, an inconsistent protocol, data format and data schema also increases the complexity of a multi-tenant communication network. Moreover, tenants have to be distinct about their privacy, safety and security policies of their assets. Finally, if different platforms for managing data streams are used, the same number of credentials has to be managed as well.

These considerations demonstrate that for multi-tenant and multi-asset scenarios (as they usually do exist in smart factory and smart city visions), peer-to-peer data streams have to evolve to streaming networks to handle the complexity of entangled interests. This paper helps to get a common understanding of cross-domain data streams. It shows how the RAMI 4.0 architecture can be used as a basis for the identification, communication and meta-data management of physical devices and data-streams, which involve implementation considerations of a Digital Twin platform that will overcome domain boundaries.

Therefore, in Section II, a use case is introduced that serves to comprehensibly explain the modelling and designing in the subsequent Sections III respectively IV. Finally, Section V contains our conclusion and shows an outlook on further work.

II. USE CASE INTRODUCTION

In this section, an example use case is introduced to make the subsequent descriptions more comprehensive. We will start with the persona of Sue, who is a manager of a car rental company:

> *Sue is a manager of an Icelandic car rental company of connected cars. Iceland is known for continuously changing road conditions and therefore she is worried about the safety of her customers while driving. Slippages on icy roads or flooded pathways may lead to*

> *car crashes or other damages; however, if drivers are warned by nearby cars and sensor stations in real-time this risk would be reduced.*

As a first step, Sue's company wants to implement a communication between cars to be able to warn the driver from nearby cold temperatures measured by her own car and other cars of her car fleet. Unfortunately, the density of cars owned by her company is too small to make useful statements. Therefore, she wants to buy temperature data from another car fleet and a weather service provider in order to increase the geospatial data density and therefore the safety of her customers. She also knows that her data can be of value for other car rental companies and others.

Briefly, Sue is interested in data exchange with other temperature data providers. For such data exchanges a digital online platform needs to be developed, which allows sharing data between its users. To provide her data on such platform, Sue has to go through the following workflow that can be generalized and adapted for similar use cases:

1. Register her company and users on the platform.
2. Register the connected cars with their available sensors.
3. Share selected temperature data of her cars securely and anonymized to others.
4. Request data from another car rental company and a local weather service provider.
5. Send and receive data securely to and from the connected cars.

III. MODELLING ACCORDING TO RAMI 4.0

In this section, the introduced scenario that can be associated with the transportation sector is modelled according the RAMI 4.. Although this model was originally designed for the industrial manufacturing domain, it is of special interest to demonstrate how such complete reference model can be applied across sectors, also because traditional manufacturing companies are increasingly interconnected with their customers, suppliers, logistics and other business partners, and cars can be considered as moving *assets.*

A. Modelling Hierarchy Levels

The first important consideration that has to be addressed is the logical unique identification of tenants, which relates to CPSs, client applications, meta-data management and data streaming. As RAMI 4.0 already defines hierarchy levels for contexts, these are utilized to construct unique namespace prefixes for tenants. RAMI 4.0 uses the following seven hierarchy levels:

Connected World → Enterprise → Work Center → Station → Control Device → Field Device → Product

1) Abstraction of Real-World CPSs

In order to map the first two levels, the already familiar and legally clarified domain categorization is utilized. The top-level and second-level domain are mapped to the "Connected World", respectively "Enterprise Level". Hence, the CPS identifiers in our example start with:

at.superrent

Synopsis: [top-level domain].[second-level domain]

As the mapping of "Work Center" and "Station" on real-world contexts is rather ambiguous [6], these levels will be investigated later and the modelling is continued with a bottom up approach where it is of interest which level of RAMI 4.0 can be associated as a CPS.

The term "product" refers to a tangible thing that has no direct digital interface [6] and therefore cannot communicate its own state by itself. Hence, a "product" represents either passive "thing" or a sub-component of an associated CPS like a smart asset, which could be depicted in the meta-data management of our designated Digital Twin platform.

A "field device" is a cyber-physical device that in general does not have a direct connection to the internet. Therefore, it does not abstract the physical world in the internet as a gateway. This step is rather examined by the "control device" level in the RAMI 4.0 [6]. As a result, the "control device" is the lowest level of RAMI 4.0 associated with a CPS. This implies that a CPS like a connected car must be connected both to the internet, as well as to underlying devices that sense or actuate the environment. In our use case, a car represent as CPS and the identifier can be expanded to:

at.superrent.*.*.car1

Synopsis: [top-level domain].[second-level domain].[…].[…].[CPS]
where '[…]' was not discussed yet.

The level "station" can be regarded as a set of CPSs and passive things that ensemble for a specific process or business service. In our case, the car fleet 1 constitutes a mobility service, which consists of multiple connected cars:

at.superrent.*.carfleet1.car1

Synopsis: [top-level domain].[second-level domain[…].[station].[CPS]

The final level of abstraction is the "work center", which organizes multiple stations, i.e., business services within a single company. As the organizational structures of companies vary significantly in their depth and labelling, this level must allow a broad spectrum of hierarchy depths. This is accomplished by allowing an arbitrary number of groups, which are separated with dashes in the global namespace.

Each level considered real world CPS can be identified in alignment of the RAMI 4.0 hierarchy levels as follows:

at.superrent.is-icecars.carfleet1.car1

Synopsis:
[top-level domain].[second-level domain].
[group('-' group)].[station].[CPS]*

2) *Namespaces for Real-World Instances*

As a further result, the derived namespaces are utilized to identify stations and CPSs globally, which were previously only unique within a specific context. The identifications are used as prefixes for the following types of instances:

a) Topic Identification in Data Streaming

The Internet of Things trend that appeared several years ago enabled CPSs to easily measure and send various system properties in near real-time with a suitable sample rate. This kind of data transfer is commonly known today as (real-time) data streaming.

Data streaming functionality requires an appropriate platform, on which a data stream has certain characteristics. For example, the unique identification of stream topics (similar to the concept of a table in a database; it refers to one single stream) is a fundament of every data-streaming platform. To guarantee, that data on a specific topic can be associated with its station of origin, the namespace for a station is used as a prefix for topics. The suffix of an associated topic is either "internal" or "external" and specifies its type.

Data Streaming topic identification, example: (Note that data are related to a topic.)

at.superrent.is-icecars.carfleet1.internal
at.superrent.is-icecars.carfleet1.external
Synopsis:
[top-level dom.].[second-level dom.].
[group('-' group)].[station].[internal | external]*

b) Instance Identification

The Management of Data about Instances varies significantly across and sometimes also within organizations. Our Digital Twin platform provides a unique namespace to identify basic instances of a semantic standard that distinguishes the CPS from its instances like "Sensors", "Actuators", "Observations" and "Datastreams". This CPS should have the capability to be augmented with arbitrary properties like core-data as well as meta-data.

B. *Modelling Layers*

As the identification along the RAMI 4.0 hierarchy levels has been described above, the next step is to map basic services of our Digital Twin platform onto the RAMI 4.0 Layers (z-axis).

A Digital Twin can be regarded as an abstraction of the real-world, which serves for multiple purposes [5]. Hence, this description of the term implies that a Digital Twin is neither part of an asset, nor of a business model. It can rather be associated with the functional-, information- and communication layer of RAMI 4.0, to which a data producer or consumer in the control device is connected to. Therefore, the primary goal of any Digital Twin platform is to organize data and data-flows in a way that facilitates decision-making. This process is often based on visualizations and analysis of an organized and cleaned dataset. Therefore, functionalities are mapped onto the functional layer in Fig. 4 should be provided by our Digital Twin platform.

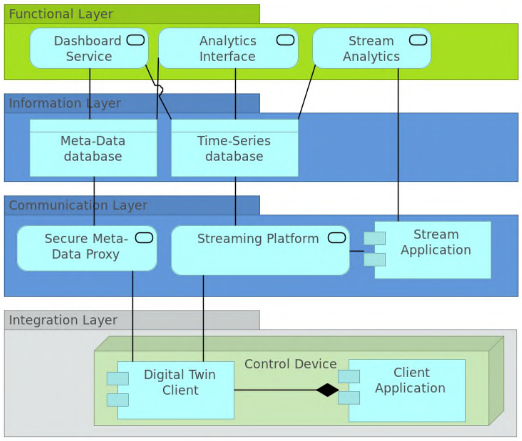

Figure 4. Mapping of Digital Twin services on RAMI 4.0 layers.

A next step is to separate the information from the communication layer, whereby the information-layer is used to store time-series and meta-data. The communication level, in contrast, should be context-free.

For reasons of security, proxies, firewalls or gateways are needed as a part of an extended security mechanism.

C. *Modelling Life Cycle and Value Stream*

The investigation of life cycle phases of a CPS helps to understand the discrepancies between PLCDM and time-series data better. There are phases in which a product is designed and exists non-materially like its early development. In comparison, a "smart product" in its usage phase produces usage data that varies significantly in schema, update frequency, validity period and so forth. Nonetheless, the rather static data from earlier life cycles can play an important role in subsequent phases, e.g., increased failure occurrences of some models, batch numbers, etc. Conversely, producers of an asset might be interested in some usage data in order to increase the product quality rapidly to a higher level. In conclusion, it is of importance to connect data of different lifecycle phases.

RAMI 4.0 splits the lifecycle axis into four phases, where the first two are immaterial and the last two material:

TYPE: DEVELOPMENT → TYPE:
MAINTENANCE/USAGE → INSTANCE: PRODUCTION
→ INSTANCE: MAINTENANCE/USAGE

Type-related data are usually very purpose-specific and differs even in basic aspects like its schema and complexity. There already exist very sophisticated and implemented software concepts that handles theses phases, like CAD, Software-in-the-loop, Model-in-the-loop and Hardware-in-the-loop [7].

Consequently, the effort for establishing a general semantic for this data would be over-proportionally high. Therefore, domain- and company-specific data semantics likely will not be changed in the near future, which implies, that integration methods have to be developed to reconcile different data appearances.

In contrast to that, there are multiple semantic standards that can be applied on instance-related data, as it is sensed or actuated in most cases and therefore follow the similar patterns.

However, the linkage between data of the *instance:production-phase* and the *instance:maintenance/usage-phase*, as the taxonomy above shows, remains a non-trivial part of a Digital Twin platform. A solution could be again the development of a meta-description for external references.

IV. DESIGNING A MULTI-TENANT DIGITAL TWIN PLATFORM ARCHITECTURE

Based on the model described in Section 3, this section will now show how the Digital Twin Platform was designed for the cross-domain use-case including multiple tenants.

A. *High Level Component Architecture*

As the identification of CPS and tenants has been described in section III.A.2), the different interaction types of CPSs and users with the platform has to be considered. Different interfaces have to be provided for CPSs and users. While the communication of CPS narrows down to data streaming and meta-data usage, a user interaction is much broader, as it involves managing of organizations, platform users, meta-data for CPS and observations over an interface. In Fig. 5, multiple components for a Digital Twin platform are illustrated.

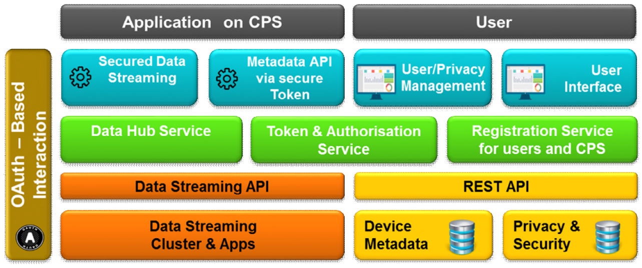

Figure 5. High Level Component Architecture.

The security component “OAuth-based Interaction” is present in each layer, which indicates the usage of security concepts in each service.

Components with blue background represent interfaces, implemented as plain APIs for CPSs, or graphically for users. The Data Hub service enables the scalable sharing of data between tenants, which are discussed in the next section. The Token & Authorization service will manage the access control on various topics for CPS clients. The Registration service connects the user interface with the backend that stores organizations, users and meta-data of CPS.

Finally, the orange components illustrate the secured internal data streaming API, as well as the actual cluster including with its deployed applications for data streaming and sharing.

B. *Multi-Tenant Dataflow*

Based on the high-level component diagram, the previously described use case is depicted in Fig. 6. On the very left, multiple CPSs are listed, grouped by their tenant. Referring to the RAMI 4.0, each CPS would be a “control device” in the hierarchy level and a tenant like the Car Fleet 1 represents a “station”.

The second column gives an overview of services that provide security mechanisms and meta-data management through methods of indirection like proxies and advanced API.

In the third column, topics of the data-streaming cluster are listed. Each tenant is connected to exactly two topics that start with the tenant identifier and end with “.internal” respectively “.external”. The unique identification of tenants within the platform is aligned on the RAMI 4.0 hierarchy levels.

The colorized arrows in Fig. 6 illustrate the dataflow between tenants, whereby the color represents the tenant of origin. The dataflow is kept clear, as each tenant can publish data only into its own “internal” topic and consume data both from “internal“ and “external” topics.

As Fig. 6 implies, the distribution of data to other tenants is done by streaming applications in the stream hub on the right side, where each tenant is connected to one application that parses data sharing contracts into a distribution logic, which is then deployed by the stream hub service. The data sharing contracts will include temporal, as well as geospatial filtering criteria, in order to have more control over data flow to external tenants.

V. CONCLUSION AND FURTHER WORK

In the presented work, the RAMI 4.0 was used as a starting point from which a cross-domain model was derived that handles several requirements to abstract real-world instances. The scenario of an Icelandic Car Fleet company was utilized to better illustrate data flows and to demonstrate the usage in a non-manufacturing domain. This approach lead to an architecture that facilitates managing even entangled data flows creating a data-streaming network.

In next steps, the conceptual architecture has to be sharped in regards to:

- Authentication and authorization mechanisms specialized for CPS
- Consideration of RAMI 4.0’s administration shell for meta-data management

Additionally, an initial prototype of such a Digital Twin Platform will be implemented to validate and enhance the concept.

ACKNOWLEDGMENT

We would like to thank our project partners from the Digital Transfer Centre Salzburg (“DTZ”, https://www.dtz-salzburg.at). DTZ is a collaboration by Fachhochschule Salzburg and Salzburg Research, funded by the regional government of Salzburg under the WISS2025 Knowledge Initiative.

REFERENCES

[1] S. W. Lin et al., IIConsortium and Plattform Industrie 4.0. [Online] Architecture Alignment and Interoperability, 05.12.2017. [retrieved: September 2019] https://www.iiconsortium.org/pdf/JTG2_Whitepaper_final_20171205.pdf.

[2] DIN Deutsches Institut für Normung e. V., DIN SPEC 91345:2016-04 Reference Architecture Model Industrie 4.0 [Online] Beuth, 04.2016. [retrieved: September 2019] https://www.beuth.de/en/technical-rule/din-spec-91345/250940128.

[3] Plattform Industrie 4.0, www.plattform-i40.de. [Online] Structure of the Administration Shell. [retrieved: September 2019] https://www.plattform-i40.de/PI40/Redaktion/EN/Downloads/Publikation/structure-of-the-administration-shell.pdf.

[4] A. Canedo, Industrial IoT Lifecycle via Digital Twins. International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS). 2016.

[5] S. Grösser, wirtschaftslexikon.gabler.de. [Online] Springer Gabler, 19.02.2018. [retrieved: September 2019] https://wirtschaftslexikon.gabler.de/definition/digitaler-zwilling-54371/version-277410.

[6] R. Heidel, M. Hoffmeister, M. Hankel, and U. Döbrich, Basiswissen RAMI 4.0. Referenzarchitecturmodell mit Industrie 4.0-Komponente. s.l. : Beuth Verlag, 2017.

[7] S. Lescarret and S. Saliou, Acsystème. [Online] Model Based Design, 01.2015. [retrieved: September 2019] http://www.acsysteme.com/en/model-based-design-1.

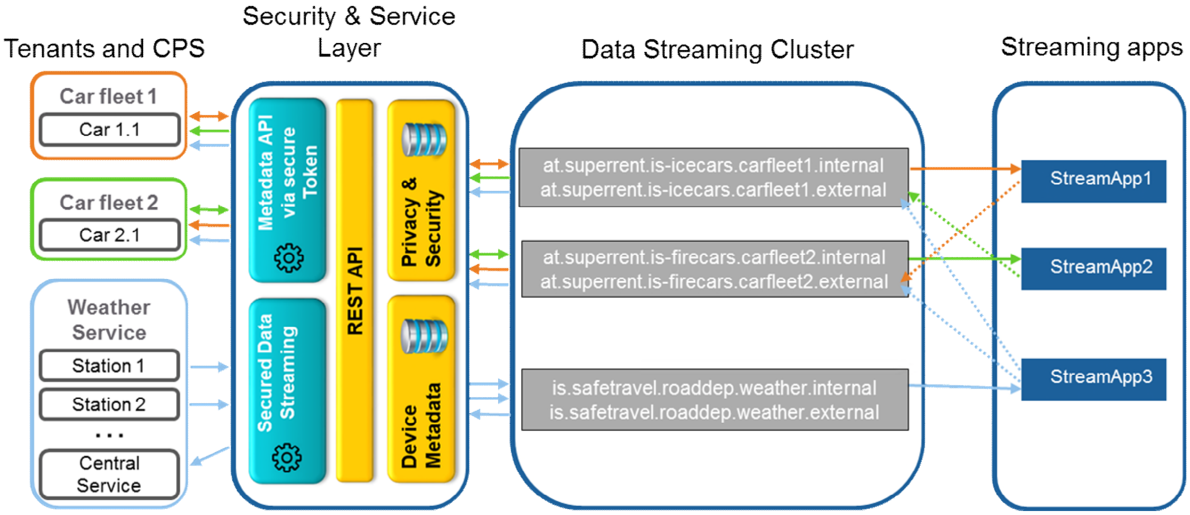

Figure 6. An exemplary Dataflow between multiple tenants.