Project for open data framework on smart manufacturing platforms
supported by the Ministry of Economy, Trade and Industry (METI) Japan

Connected Industries Open Framework for Manufacturing

Basic Specification and System Requirement

Ver.1.2

February 15, 2019

Industrial Value Chain Initiative

Content

1. Introduction

1.1 Purpose of this document

This document specifies the basic requirements for implementing a system for data distribution within the connected industries open framework (CIOF) for manufacturing. This document defines the basic requirements for the development of the system and subsystems that make up and constitute the system. As a whole, this document aims to specify the requirements that are available for the system implementation realizing a connected industries open framework for manufacturing.

The functions and requirements of the connected industries open framework for manufacturing shown in this document are necessary for the parties who conduct data distribution, but further specifications such as security measures are required before full-scale commercialization. Therefore, this document is designed with prototype development for concept demonstration in mind.

The content to be developed in the project supplementary budget for FY2018 for promoting data sharing corresponds to the prototype implementation specification separately specified.

1.2 Audience

The intended audience of this document is the developer of the subsystem of the connected industries open framework for manufacturing, and the integrator that integrates them and implements the open cooperative framework in individual business scenes.

In addition, companies that provide edge-side platforms, companies providing IT side platforms, standardization bodies that discuss collaboration among these platforms, and technicians, managers, and planners are also target audiences.

1.3 References

1. Study on the establishment of a data-related system for realizing Connected Industries, Ministry of Economy, Trade and Industry Information Economics Section (October, 2018) (In Japanese)

2. Development Specification of the Connected Industries Open Framework, Version 0.1 Draft 10, (January 8, 2018) (In Japanese)

3. Practice of Open & Close strategy for "Connected Industries" enabled by the manufacturing open framework – Interim report. (March 30th, 2018) (In Japanese)

4. Same as above: Appendix 1: Use Cases and Common Dictionaries (March 30, 2018) (In Japanese)

5. Same as above: Appendix 2: Use Case Demonstration (March 30, 2018) (In Japanese)

6. FY2018 Grant application document of the Industrial data sharing promotion project (June 2018) (In Japanese)

1.4 Abbreviations

ADD	**Actual Data Dictionary**
ASD	**Actual Service Dictionary**
DCC	**Data Component Category**
DCM	**Data Component Model**
DCR	**Data Component Record**
DRM	**Data Relation Model**
DPD	**Data Property Definition**
DPV	**Data Property Value**
DTM	**Dictionary Translation Map**
EAU	**Edge Application Unit**
ECC	**Event and Condition Category**
ECE	**Event and Condition Expression**
ECI	**Event and Condition Instance**
ECM	**Event and Condition Model**
ECS	**Event and Condition State**
ECR	**Event and Condition Record**

ECU	**Edge Control Unit**
EDU	**Edge Device Unit**
HCM	**Hyper Connection Manager**
HCS	**Hyper Connection Server**
HCT	**Hyper Connection Terminal**
HDS	**Hyper Dictionary Server**
PCC	**Process Component Category**
PCM	**Process Component Model**
PFD	**Process Flow Definition**
PCI	**Process Component Instance**
PCE	**Process Component Execution**
PTM	**Property Translation Map**
SCD	**Specific Category Dictionary**
SDD	**Specific Data Dictionary**
SSD	**Specific Service Dictionary**
TAP	**Trade Account Party**
TCP	**Trade Contract Profile**
TDP	**Trade Data Profile**
TSP	**Trade Service Profile**

2. Terminology

ADD：Actual Data Dictionary

A local dictionary that has a list of the DCMs to be used for each site or terminal (HCT) or the controller (ECU) under the terminal.

ASD：Actual Service Dictionary

A local dictionary that has a process model (PCM) and an event model (ECM) to be used for each site or terminal (HCT) or a controller (ECU) under the terminal.

DCC：Data Component Category

A category of data component model (DCM). Categories defined here are registered and used in the specific category dictionary (SCD).

DCM：Data Component Model

A unit of data to be registered in a dictionary. DCM corresponds to the Entity in the E-R model, class in the UML class diagram, and the table, or view in the RDB schema.

DCR：Data Component Record

Specific data content registered along with DCM. This corresponds to data, which we call in general, where data is a record or a collection of records in the RDB.

DPD：Data Property Definition

This is the definition of the items constituting the DCM, and the value of the data is set in this unit. It corresponds to a field in the RDB.

DPV：Data Property Value

This is a unit for expressing each value constituting a record. It corresponds to one piece of data of key & value type. The value can be either a data type or null.

DTM：Dictionary Translation Map

Relation of elements between an Actual Data Dictionary and a Specific Data Dictionary. It corresponds to each conversion source and conversion destination of the DCM. When the conversion source and the conversion destination are one or more concatenated DCMs,

there is a many-to-many relationship.

EAU：Edge Application Unit

A device that provides applications utilizing data within each site. All on-premise type business applications not executed in the cloud fall under this category.

ECC：Event and Condition Category

It shows the category of event component model (ECM). All ECMs belong to one category of an event component (ECC).

ECM：Event and Condition Model

Definition unit of an event in the cyberworld. It may trigger a PCM. When defined by the processing of a PCM, the ECM may be defined in association with an event in the physical world if it is defined in the state of a specific value of DCM.

ECI：Event and Condition Instance

An event implemented on a specific process or component. It is an instance of an event component model (ECM). An ECI can be defined as a state corresponding to an individual entity in the actual situation.

ECR：Event and Condition Record

A record that an event actually occurred in the event component state (ECS). This forms a historical data with a timestamp.

ECS：Event and Condition State

A state of event component model (ECM) corresponding to an individual entity in the implemented state. At each point in real time, it has the number of actual records (ECR) of this ECS.

ECE：Event and Condition Expression

Conditions in which an event occurs, constraint conditions, etc. are described by numerical expressions. Normally, a mathematical expression is described using elements such as external values (DPV) obtained from EDU.

ECU：Edge Control Unit

Equipment that manages data to be used and that controls data processing within each site according to each purpose. It acquires or provides data from other sites via a HCT.

EDU：Edge Device Unit

Devices consisting of sensors that acquire data from the physical world, or equipment attached to an actuator that applies data to the physical world, and that becomes the starting and ending points of data.

HCM：Hyper Connection Manager

Software that provides a UI for setting contents of data distribution, usage, dictionary use, conversion method, etc. when an ECU and an EAU cooperate among different sites.

HCS：Hyper Connection Server

Server that is located on the internet communicating with the cooperative terminals under its control. It enables cooperation among multiple sites through communication with other servers.

HCT: Hyper Connection Terminal

Communication terminal for internet access located in the private network of each site and which has a local IP address. The external internet communicates only with the preset HCS.

HDS：Hyper Dictionary Server

Server that manages a specific data dictionary, an actual data dictionary, and a dictionary translation table and responds to registration, modification, search, etc. While managing the actual data dictionary for each site, it supports modification of the Specific Data Dictionary.

PCC：Process Component Category

Definition that indicates the category of the process model (PCM). All PCMs belong to one of the Process Component Categories (PCCs).

PCM：Process Component Model

Unit of processing in the cyberworld. Based on the contents of a specific DCM, it performs a calculation according to a predetermined procedure and operates the contents of a specific DCM.

PCI：Process Component Instance

A unit in which a predefined software process (a PCM) is implemented on concrete hardware, such as a computer, and becomes executable. Software licenses are usually managed in this unit.

PCE： Process Component Execution

A unit executed by a process component model (PCM) on software implementation (PCI). The software execution log is recorded in this unit. License management in this unit is necessary.

PFD：Process Flow Definition

A more specific procedure of a process, defined step by step. Describe the flow of operation in words by means of a statement. Data generation, input, modification,

reference, etc. are performed in this unit

PTM：Property Translation Map

Relationship that indicates the correspondence between data property definitions (DPD) in two DCMs in different dictionaries.

SCD: Specific Category Dictionary

A dictionary that contains data category (DCC), process category (PCC), and event category (ECC). Multiple category dictionaries can be defined, but conversion among dictionaries cannot be defined.

SDD：Specific Data Dictionary

Dictionary that defines information of all DCM mutually agreed between multiple sites. It contains one or more DCMs. This in not a only one specific dictionary. A plurality of specific data dictionaries may exist.

SSD：Specific Service Dictionary

Dictionary that defines information of process models (PCM) and event models (ECM) that can be referenced from multiple sites. Actual data dictionaries of individual sites can be created based on contents held here.

TAP：Trade Account Party

The owner of each site that provides data or uses data when carrying out data distribution. It corresponds to a party making a transaction contract

TCP：Trade Contract Profile

Profile to define the form and method of data distribution and contract contents that are settled between two sites. Data storage method, attribution of rights, billing method, prohibited items, etc. are also included.

TDP：Trade Data Profile

The profile of the data component model (DCM) used in messages at the time of sending or receiving messages is described in a Specific Data Dictionary or an Actual Data Dictionary. In TDP, the DCM of the requested data also shows its structure, for example in a case where it is represented by the jointed DCMs in the Specific Data Dictionary.

TSP：Trade Service Profile

Profile of definition for the contents of the process component model (PCM) and related events (ECM) on the side providing and using data. Also shown are ECUs, EAUs, EDUs, and others actually to be executed. The DCM defined in a TSP can be queried by a HCM.

3. System overview

3.1 System features

The aim of the connected industries open framework (CIOF) for manufacturing is to provide a platform for edge areas to further digitize production sites in the manufacturing industry. It is a mechanism that makes it possible to easily distribute data through relatively simple procedures. The following three points are cited as features of CIOF.

Feature 1: Secure and traceable communication

Data on a manufacturing site are highly confidential and have a high value as intellectual property; thus taking such data outside the factory is accompanied by great risks. In the internet world, there are many security problems and it is a harsh fact that factories need to exercise strict management of data. CIOF is based on peer-to-peer communications, and, in principle, do not place data on the internet. The sending and receiving parties are specified in a contract beforehand, and at the time of transmission, all data are encrypted to ensure robustness.

Feature 2: Open and distributed dictionary

Each factory has different terminology and different ways of doing things, it is impossible to integrate and unify these different manufacturing crafts among different enterprises. As with the existing EDI (electronic data interchange), even if terms and specifications are unified beforehand, it ultimately is beginning from scratch. CIOF does not unify dictionaries, it operates multiple dictionaries simultaneously in parallel. In addition, in an open position, by making it easier to search for frequently used terms in these dictionaries, we have something that is user-friendly, based on the law of natural selection.

Feature 3: Self-improvement mechanism

A characteristic of manufacturing sites is that they have autonomous growth through kaizen. In CIOF, in order to solve problems at the bottom of the site, it is possible for each autonomous decentralized platform on each edge side to expand afterwards. A server for collaboration is a mechanism in which a plurality of servers collaborate, and it is possible to

cooperate with a framework using this approach

3.2 Basic System Configuration

Figure 1 shows the configuration of the system. Each edge side platform has a cooperation terminal (HCT) and a cooperation manager (HCM) inside it and manages cooperation. On the internet, a cooperation server (HCS) mediates data between HCTs, and a dictionary server (HDS) manages dictionaries necessary for cooperation. It is the control unit (ECU), the device unit (EDU), and the edge application (EAU) that actually provide or use data. Note that only the ECU and the EAU can communicate directly with the HCT

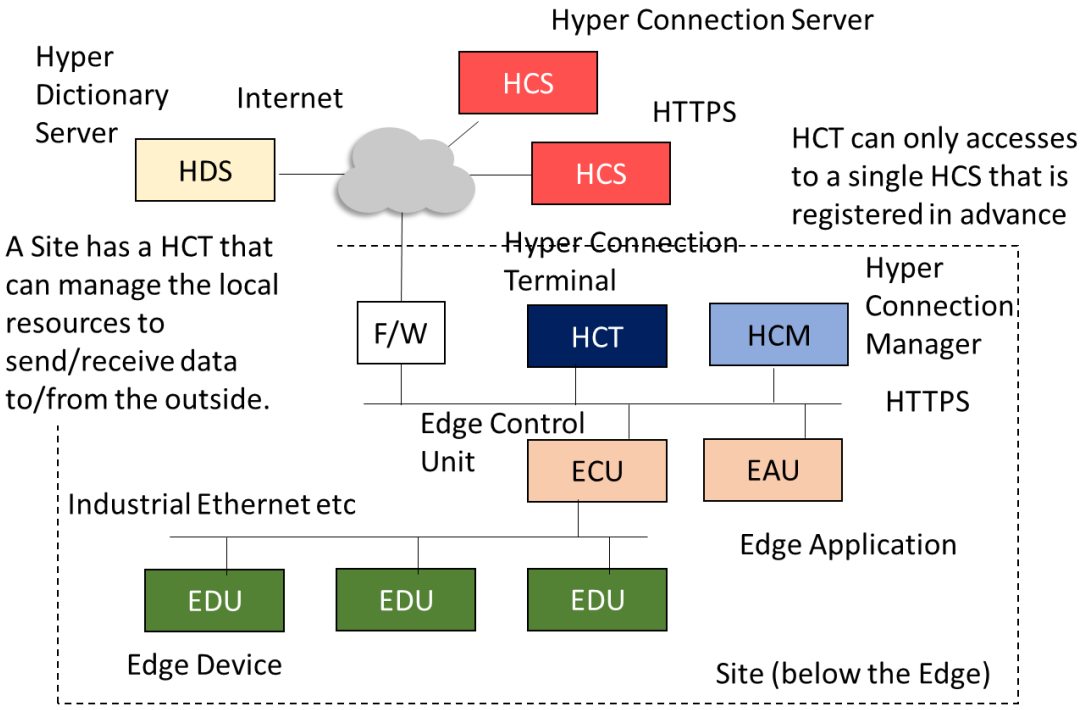

Figure 1 : System Architecture

Table 1 Level and description of system components

Layer	Name	Description
the Internet Layer	Hyper Connection Server (HCS)	HCS connects with HCS of 0 or more. If the destination HCT is not in control of itself, data are communicated between HCSs.
	Hyper Dictionary Server (HDS)	The HDS manages the data model (DCM) for each site together with the specific data dictionary, and provides it together with the conversion table.

Terminal layer	Hyper Connection Terminal (HCT)	The HCT always connects with one HCS. As a window of cooperation, we will bridge the internet layer and the edge layer.
	Hyper Connection Manager (HCM)	The HCM provides a GUI to the user who provides or uses the data and communicates with the HCT. In addition, necessary dictionaries are connected to the HDS to register and manage.
Edge layer	Edge Control Unit (ECU)	The ECU connects to one HCT. It has one or more EDUs under management and collectively manages the use and provision of those data
	Edge Application Unit (EAU)	The EAU connects with one HCT. Although it is equivalent to the ECU in terms of HCT, it does not have an EDU under management and has a process of actually using or providing data.
Device layer	Edge Device Unit (EDU)	The EDU is connected to one ECU. As a sensor or actuator, contact physical objects in physical world and actually use data or acquire data from there.

In practical terms, the device unit (EDU) corresponds to a machine tool, a controller, a robot, or the like constituting a production line, and the control unit (ECU) cooperates with them and communicates data obtained there with the outside Module, or module that receives data from the outside and controls groups of corresponding EDUs.

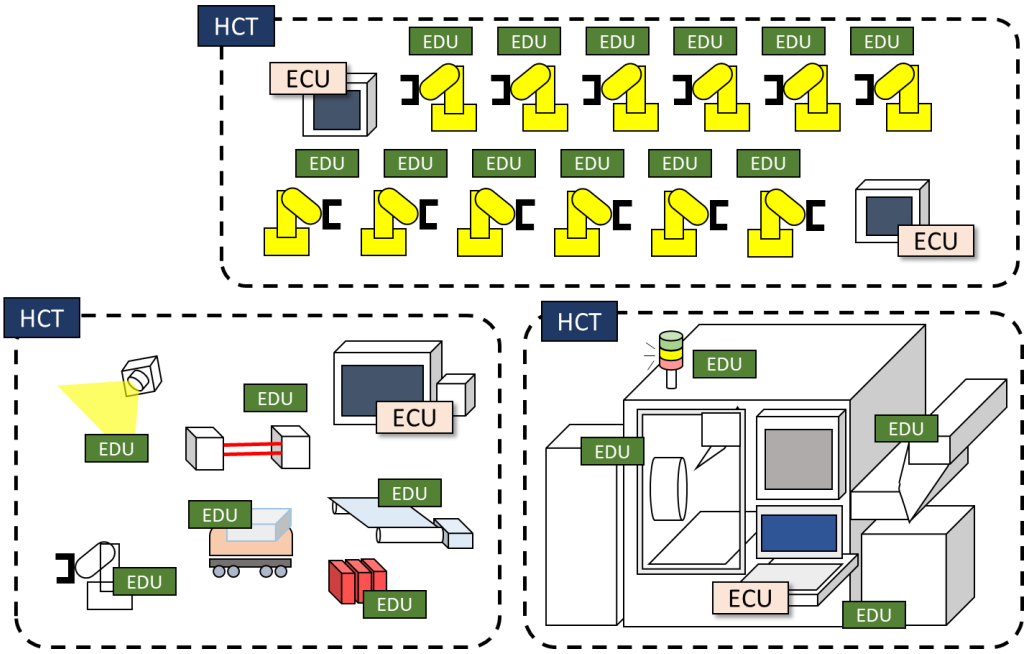

Figure 2 : Examples of device units

3.3 Data distribution format and granularity

In CIOF, the edge site platform encrypts and decrypts the contents of transmission in order to securely distribute data to other platforms. In addition, it is only HCS that can handle data contents on the Internet, and secures confidentiality as they are authenticated beforehand

In HCT, conversion is performed when data items and terms are different between actual data dictionary used at each site and specific data dictionary. On the dictionary server (HDS), a specific data dictionary (SDD) and an actual data dictionary (ADD) are registered in advance and a conversion table (DTM) is registered in advance. The HCT acquires these contents and converts according to the profile of the transaction contents.

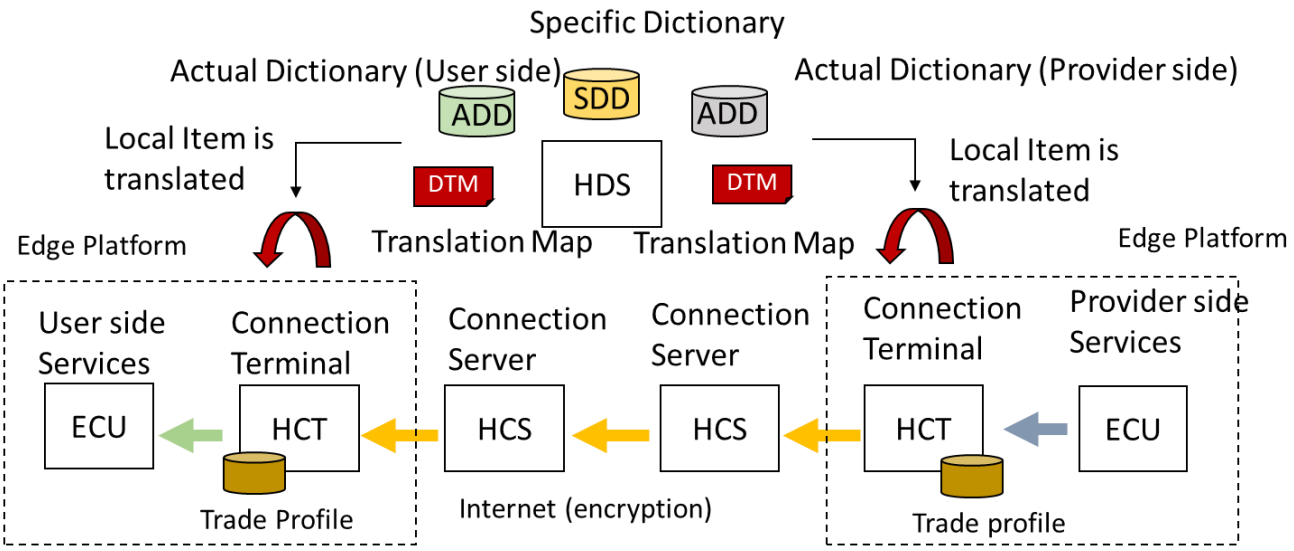

Figure 3 : Format of data distribution

The data distribution targeted by CIOF is targeted to areas that do not require a comparatively real-time nature between platforms. As shown in Fig. 4, within the edges of the production site, control that requires a real-time property of 1 second or less is conducted, but data distribution of such granularity and accuracy is not subject to CIOF. In addition, in the fundamental information system (ERP), data on daily, weekly, and monthly data with a larger granularity is often used, but such data are assumed to use conventional methods.

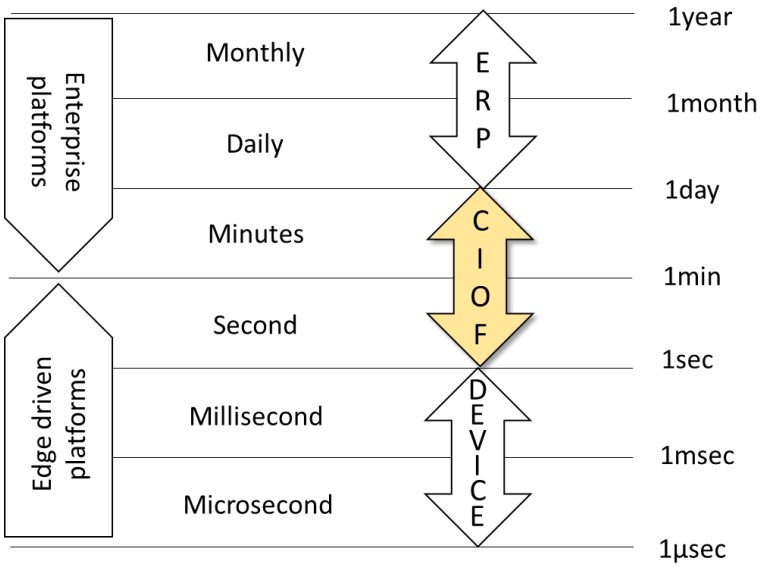

Figure 4 : Granularity of system cooperation

3.4 Security and Intellectual Property Management

From the viewpoint of intellectual property management on system security and data, the following measures are taken.

1) HCT and HCM are installed in the firewall of each site and are not installed directly so as to be accessible from the outside. Communication is always made from the HCT and HCM side by specifying a specific external server authenticated beforehand and by using PUSH type or PULL type starting from the site.

2) Content of data communication (DCR) is encrypted by the transmitting HCT and decrypted by the receiving HCT. The data communication content (DCR) can be relayed only by a previously authenticated HCS. The HCS can temporarily spool the relayed data communication contents (DCR), but this should not be persistent.

3) Transmission of keys for individual authentication of terminals and servers, and transmission of keys for encryption and decryption at the time of data communication are performed in the analog (physical) world including a manual procedure by the person in charge.

4) Profile information for data transactions is shared between the data-providing side and the data-using side, but it is assumed that the disclosure level can be set according to the situation pertaining in both sites (or example, EDU configuration, etc.) as required .

5) Requests and data contents actually made between the data provider and the data user are set with a globally unique ID so that they can be used in the transaction confirmation phase.

6) The specific dictionary (SDD and SSD) handled by the dictionary server (HDS) is commonly defined and agreed information between the data provider and the data user, and if it permits disclosure to all framework participants is not limited. As described above, for a dictionary and a dictionary conversion map in a common area, it is possible to set up a business operator that can access them as necessary.

3.5 Scope of development and prerequisites

For system development for the connected industries open framework for manufacturing, the following scope and preconditions shall be established.

1) The system to be developed is hardware and/or software handling discrete

digital data, and it does not cover analog signals, control signals, or stream-type continuous data, etc.

2) Data that need to be encoded according to specific standard specifications such as image data, audio data, CAD / CAM data, etc. are handled as files and are not handled internally. Each file is treated as data together with other attribute values given explicitly.

3) Define metadata to define the function and behavior of the system, and also the meaning and usage of the target data communicated by the system. Metadata includes dictionary data, dictionary conversion data, profile data, and the like.

4) The authentication server, authentication protocol, and other functions necessary for secure communication are technologies that are essential and widely used, and do not include newly developed elements. This approach applies to cyber-security concerns.

5) Historical data on communication transactions is necessary to realize the traceability function of data. The traceability function is one of the important pillars for cooperative systems, but it is outside the scope of this development.

6) The dictionary data and the dictionary conversion data are subjected to maintenance in a bottom-up manner, in a distributed and cooperative manner, and can be added or updated at any time. It is necessary to manage dictionary updates and maintain consistency in each one of the two hierarchies: the actual dictionary and the specific dictionary.

7) There will be more than one dictionary server (HDS) in the future, and these servers will manage and update the dictionary in cooperation with each other. However, in this fiscal year, there will be only one server and no distributed processing will be done.

8) HCS, HCT, HDS will publish necessary and sufficient source code for function execution to GitHub under an MIT license policy, and encourage development as open source.

4. Method of data transaction

4.1 Classification of use cases

Use cases in data transactions can be divided into the following four phases. In fact, data is transmitted from the data provider to the data user in the transaction implementation phase, but in addition to this phase, the contents of such transactions are determined between the two, and the individual edges of both sides also put emphasis on the transaction contract phase that connects according to the actual situation of the unit.

System integration phase

In the system integration phase, installation and registration of HCS and HCT, registration of ECU, EDU, EAU, editing of the actual data dictionary (ADD), registration of the specific data dictionary (SDD), registration of service (SSD), registration of event persons, registration of ECUs and EAU, etc. require authentication. Also, the dictionaries are registered and updated by an authenticated user.

Trade contract setup phase

The transaction contract phase refers to the necessary arrangements for individual companies or sites to provide and use data via HCT. Processing of the transaction contract phase is mainly a collaboration manager (HCM) and interacts with the person in charge. Registration and inquiry of the specific data dictionary (SDD), the actual data dictionary (ADD), registration of the dictionary conversion map (DTM), registration of the transaction profile, and the like.

System execution and control phase

In the system execution and control phase, data is actually sent and received between terminals (HCT). When distributing data, this is done according to the profile that we made beforehand in the transaction contract phase. There are use cases of PULL type and PUSH type depending on the relationship between the data provider and the data user. Send request, send data record (DCR), inquire dictionary conversion map, perform conversion.

Monitoring and Evaluation Phase

In the monitoring and validation phase, after the data have passed from the provider to the user, the fact is recorded, and a record of how the user has used the sent data is accumulated, and if necessary, an inquiry is sent. This corresponds to a confirmation of transaction record, settlement of transaction consideration, inquiry of software implementation (PCE), inquiry of event (ECR).

4.2 Classification of transaction contract type

PULL transaction use case

The data user specifies the type, structure and timing of data to be used, and requests the data provider to provide it. We will modify its contents with the data provider and provide data when agreeing to terms. In response to a request from a single data user, data may be provided a multiple times for each event.

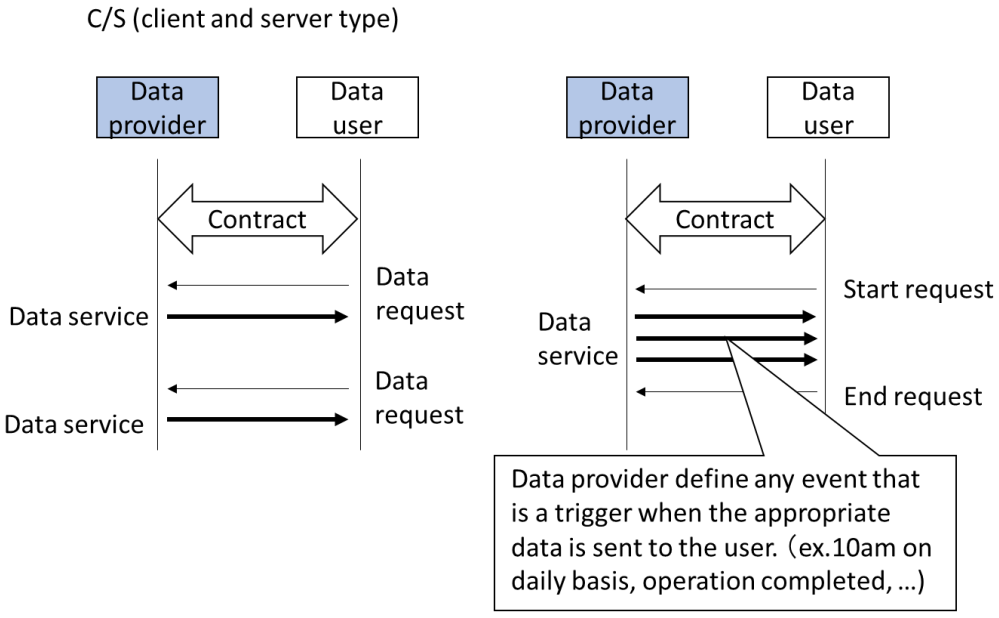

Figure 5 : PULL transaction contract

PUSH transaction use case

The data provider specifies the type and structure of data that can be provided and the timing, and the person requesting the data distribution requests the data to be provided. An approved data user receives the data. A plurality of data users may receive the data for one data provider's data provision

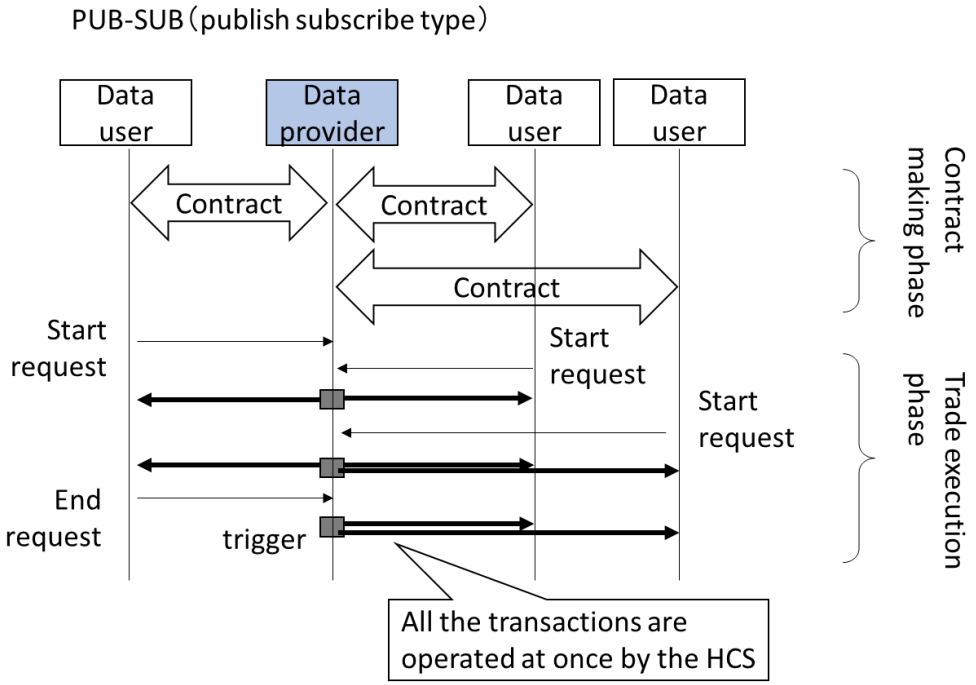

Figure 6 : PUSH transaction contract

4.3 Message structure in execution phase

Figure 7 shows the structure of a transmission message in the transaction execution phase. The message consists of a message body consisting of a list of data records (DCR) and a message header containing a transaction profile.

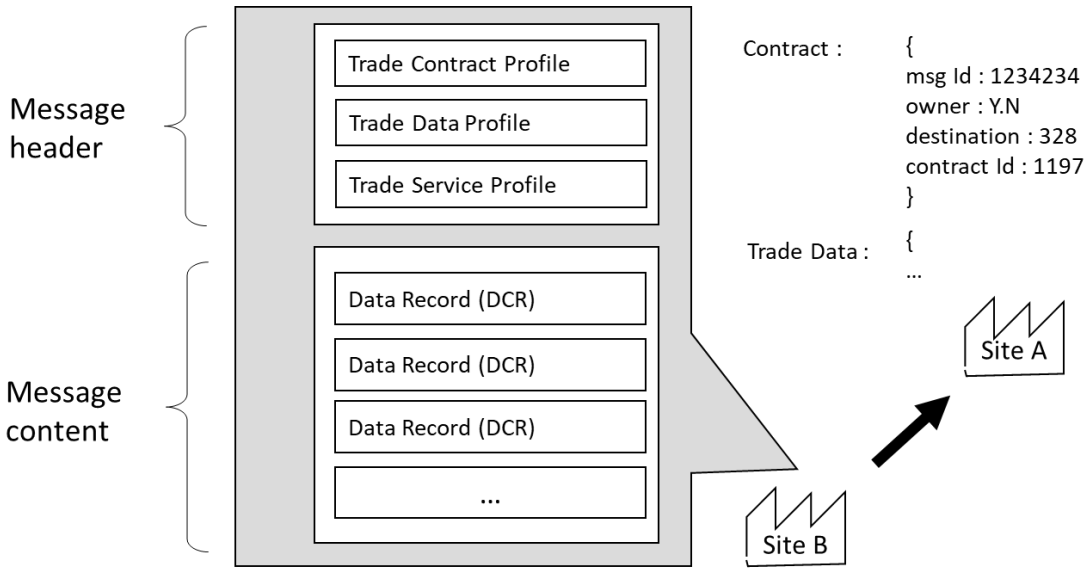

Figure 7: Transaction message structure

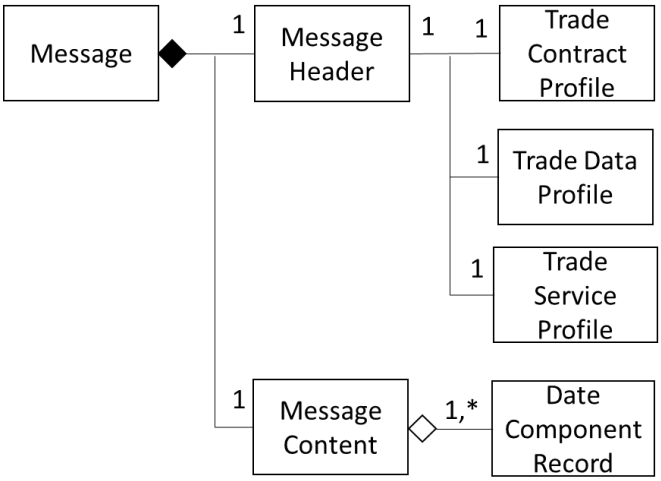

Figure 8: Object model of message (for sending)

If the message is for a request, it cannot contain a data record, and as shown in Fig. 9, only the character string-type request parameter is transmitted as the message body. In the transaction profile in the message header, only the transaction profile ID is specified, and the contents are inquired of each HCT.

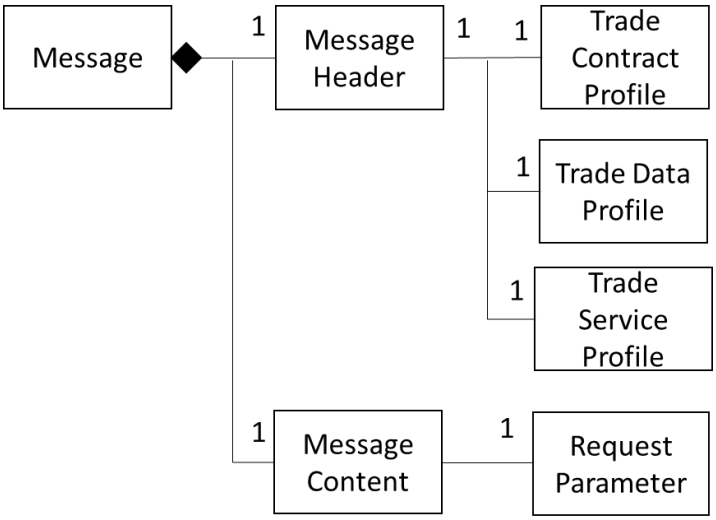

Figure 9: Object model of message (for requesting)

4.4 Data type

The data type specifies the type in the computer of the value of the entity of data (DCR and DPV). It is defined for each data item. It has Character String, Number, Numeric, DateTime, Boolean, Bulk, Reference, URI, and List. Bulk refers to the substance of data files such as images and sounds.

Table 2: Data type

Data type	Description
String	A string encoded in UTF 8. Since other data types also become character strings after conversion, this is almighty coding.
Number	Used when choices, order, etc. can be represented by numbers. Corresponds to ordinal scale, nominal scale, etc.
Numeric	Represents absolute value, relative value, etc., all other quantities such as integers and floating-point real numbers. It does not distinguish precision, etc. on mounting
DateTime	Represents the date and time that can be represented on the calendar. Basically it is treated as a point in time, not a period of time.
Boolean	Boolean value takes a value of either True or False or, in some cases, it is undefined. It is used for condition judgment.
Bulk	This data type is used when sending the entity of a data file such as images or audio as data
Reference	This corresponds to an external reference key and is used when defining a connection with another DCM. In fact, it is the data type of the referenced primary key.
ULI	Specify in plain text the location of relevant data, such as an external URL or an in-house server.
List	This corresponds to a case where the content is a list of objects that can be represented by the same data type. The data type is selected from the above. You cannot set the list in List.

5. Object model

5.1 Data component model (DCM)

Data provided or used at each site can be shown as below using the object model (class diagram of UML).

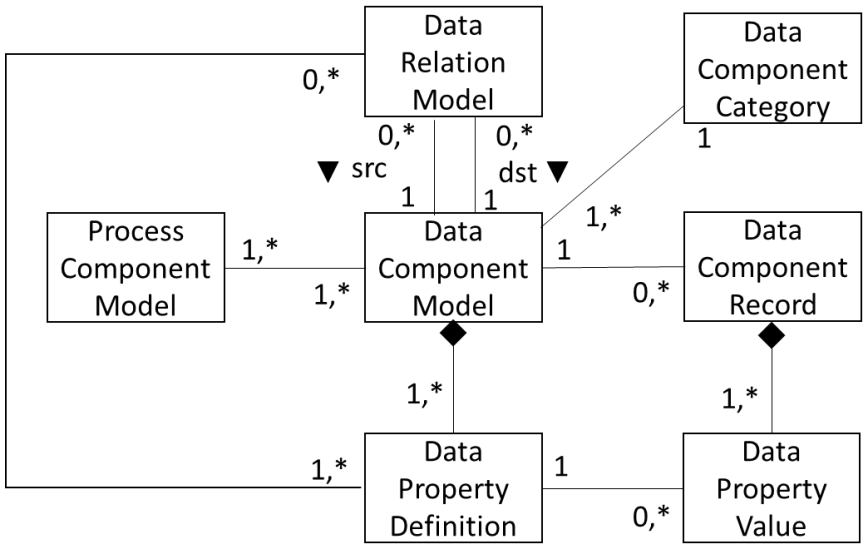

Figure 10 Data Component Model (DCM)

Data Component Model (DCM)

The data model (DCM) is a unit of data to be registered in the dictionary. Entity in the E-R map, class in the UML class diagram, table, or view in the RDB schema.

Data Component Record (DCR)

Data component record (DCR) is concrete data content registered in accordance with DCM. It is a record in RDB. In general, what we call data is a collection of records or records in many cases.

The relationship between the data model (DCM) and the data record (DCR) is shown in the following figure. DCM is merely type information and definition information, and the

actual data entity is set to the DCR.

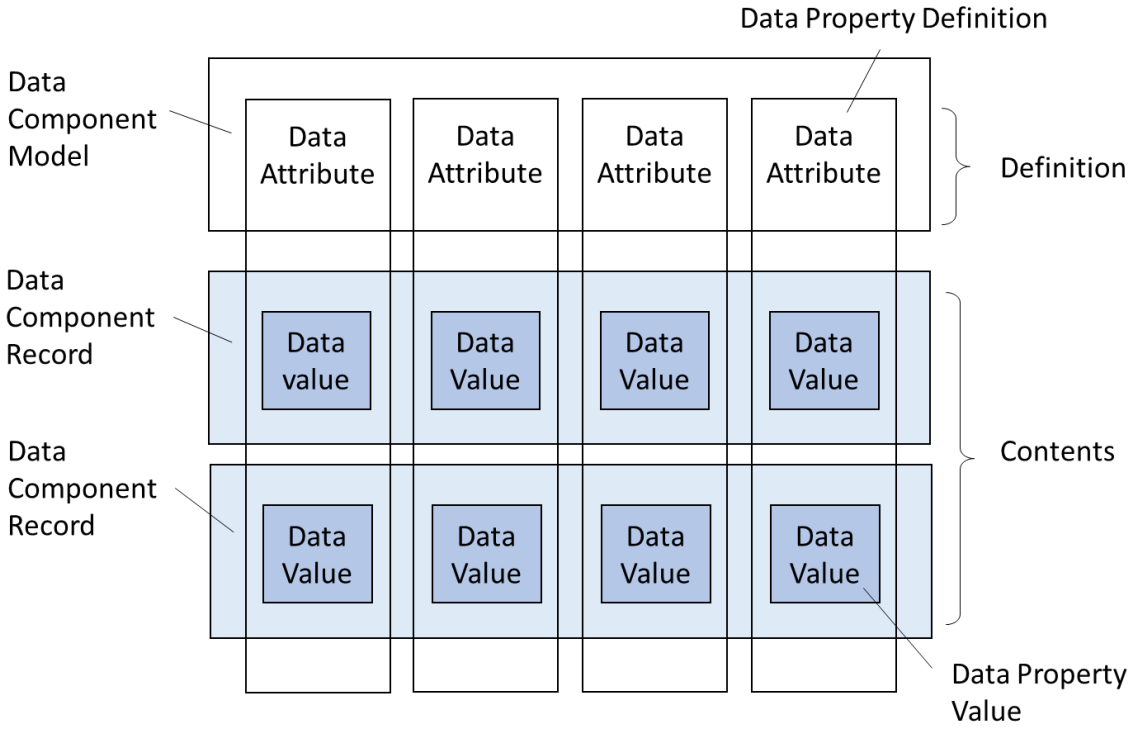

Figure 11 Models and contents of data component

Data Property Definition (DPD)

The data item (DPD) is an item constituting the data model (DCM), and the value of the data is set in this unit. Corresponds to a field in the RDB

Data Property Value (DPV)

This is a unit of each value constituting a record. It corresponds to one piece of data of key and value type. The value can be either a data type or null.

An example of an entry sheet for describing a data model (DCM) is shown below.

Table 3: Input sheet of Data Component Model (DCM)

DataID	1	DicID	2	DicType		Author	4
Name						Date	
CategoryID		Category	3				

No	Item name	Description	DataType	Null	Key	Default	Sample
	5		6	7	8	9	10

No	Link name	Link DCM	Join Key	Items
	11		12	13

No	Type	Process name	PCM	Remarks
		14		

No	Type	Information Name	InfoID	Remarks
		15		

1. The data ID is a unique symbol for specifying the target DCM. Also, the name of the data is indicated in the data name. In addition, in the explanation part, the outline description is described.

2. The dictionary ID is a symbol for identifying a dictionary. For the dictionary classification, either the specific data dictionary or the actual data dictionary is set.

3. The category ID and the category name are categories for classifying the data model. When specifying a pre-registered category, the ID is specified, and when there is no applicable category, the ID is indicated as a blank and an arbitrary category name is indicated.

4. Set the date on which the definition sheet was created and the entrant.

5. Specify the items (DPD) constituting the data model (DCM) in units of rows. Number shall be the serial number from 1. Item names must not overlap. In the explanation, a brief description of the item is given.

6. Either a character string, a numeric value, a number, a date date and time, a boolean value, an item

reference, a file reference, or a bulk is set as the data type.

7. If mandatory is checked, the content will always be set

8. Primary key means that records (DCR) can be uniquely selected by combinations of checks there.

9. The default value indicates a value that is automatically set when the content is not set.

10. An example of the actual value (DPV) is shown in the sample.

11. When concatenating another DCM to the target data, specify the ID of the DCM of the concatenated data ID. For the concatenated name, set the data name to be concatenated. To define multiple concatenations for the same data, change the name.

12. For the concatenated key, set the item name corresponding to the primary key of the data to be concatenated. The additional items show examples of items used in the target data from the items in the consolidated data.

13. In the additional item, a new item name added by concatenation is listed.

14. Specify the process to handle data. If the classification is generated, it indicates the name and ID of the process that generates the data. In the classification, there are generation, modification, use (corresponding to input and reference).

15. When the data correspond to predesigned information such as form, slip, screen, panel, board, etc., the name is indicated as an information name. The information ID is an ID for identifying the information, but it is not indispensable here.

5.2 Relation among data records

- The primary key is one item or an internal primary key item to be automatically generated. Also cover cases with multiple primary keys
- The number of data records of the primary DCM of the conversion source is the same for the DCM of the conversion destination. Actually, a conversion destination data record is generated corresponding to each conversion source data record, and a corresponding primary key is set.
- If the data type of the primary key is different and the value cannot be converted, give priority to the data type of the conversion source. Change the data type of the conversion destination. (Interactively specified at conversion table creation)
- If it is a required item but the corresponding item is not in the cooperation source, fill the value according to the default value setting

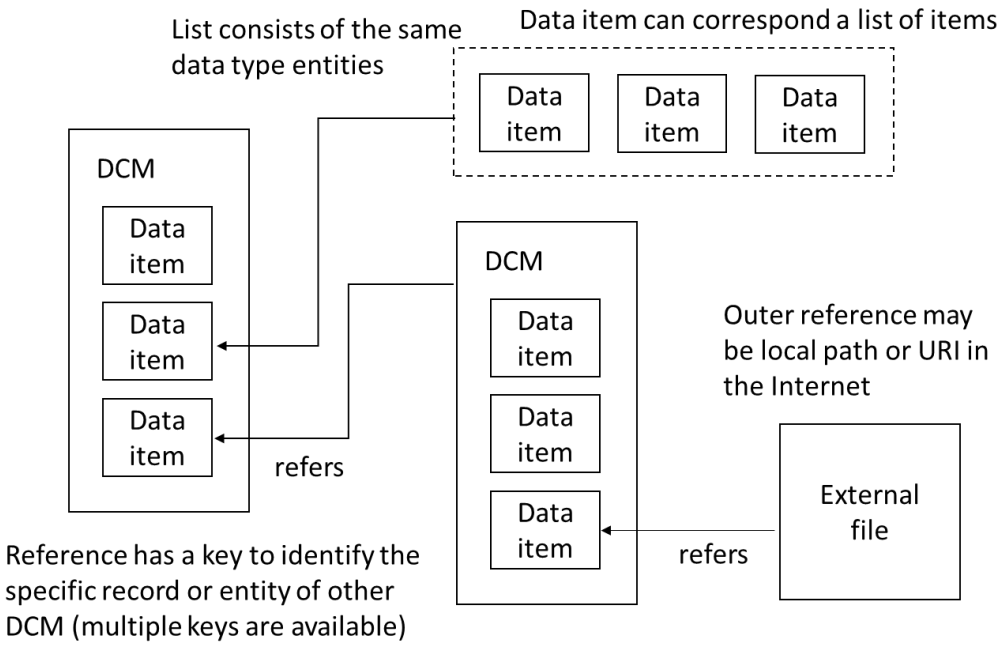

Figure 12 Relation of linked DCMs

5.3 Process component model (PCM)

The process of providing or utilizing a data model (DCM) (PCM) can be registered in the dictionary as part of the service model. Also, the process implementation (PCI) placed on the actual edge unit is registered as a profile in the HCT that manages each site.

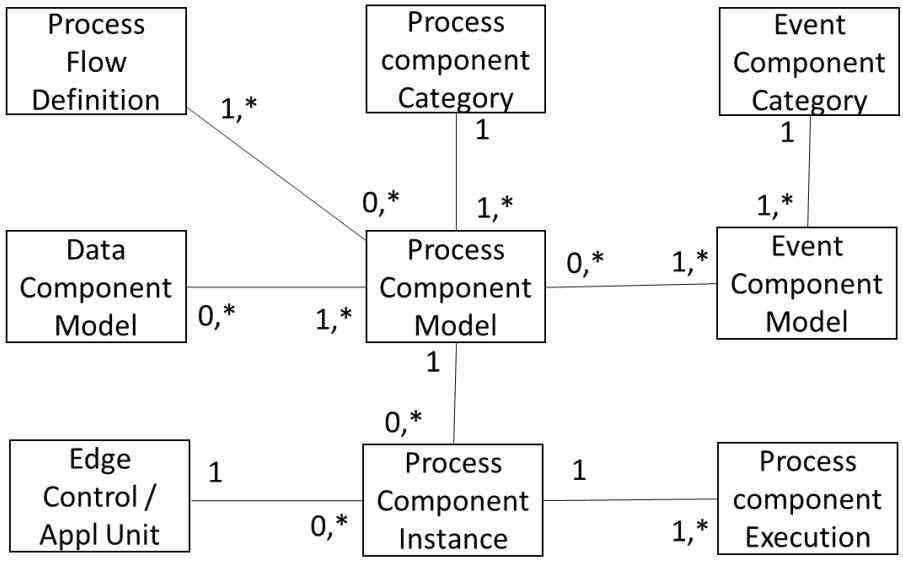

Figure 13 Process Component Model

Process Component Model (PCM)

Unit of processing in the cyber world. Based on the contents of a specific DCM, calculate according to a predetermined procedure and operate the contents of a specific DCM.

Process Component Category (PCC)

Indicates the category of process model (PCM). All PCMs belong to one category (PCC)

Process Component Instance (PCI)

A unit in which a predefined software process (PCM) is implemented on concrete hardware, such as a computer, and becomes executable. Licenses and other such items are managed in this unit.

Process Component Execution (PCE)

A unit (PCM) executed on software implementation (PCI). The software execution log is recorded in this unit. License management on this unit as necessary

Below is an example of an entry sheet for describing a process model (PCM).

Table 4: Input sheet of Process Component Model (PCM)

Process ID	1	DicID	2	Author	4
Process name				Date	
CategoryID		Category	3		

No	Data ID	Data name	Crt	Cng	Inpt	Ref	Remarks
	5			6			

No	Procedure	Target data
	7	

No	Type	Precondition	Target data
		8	

No	Type	Postcondition	Target data
		9	

Hardware requirements
10

Other requirements
11

1. The process ID is a unique symbol for specifying the target PCM. In addition, the process name indicates the name representing the process. Also, in the explanation part, the outline description is described.

2. The dictionary ID is a symbol for identifying a dictionary. The dictionary classification here is assumed to be an actual data dictionary

3. The category ID and the category name are categories for classifying processes, and when specifying pre-registered ones, IDs are designated, and when there is no corresponding category, the ID is indicated as a blank and indicates an arbitrary category name.

4. Set the date on which the definition sheet was created and the registrant

5. Describe the ID and name of the relevant data model (DCM) in carrying out the process

6. This shows the relationship between data and process. In the case of output, check what is generated, either generation (record newly added), setting (setting item value newly), change (item value changed). In the case of input, it specifies input (when starting the process) and reference (when data are introduced from the process).

7. Explain the operation procedure in the process. Branching, repeating, etc. should be avoided as much as possible

8. Prerequisites for running the process. Trigger is a trigger event (ECM). Specify "trigger" or "premise" for the classification

9. Indicates what is happening as a process execution result. When the classification is normal, it shows the expected state. In the case of exception, it indicates an error or desirable situation

10. If there is assumed hardware, or if there is a specific requirement, describe it

11. It shows other requirements and special notes.

5.4 Event component model (ECM)

The event component model (ECM) has internal events obtained as a result of implementing a process component model (PCM), external triggering of PCM, or external event serving as a starting condition. There are state events not directly related to the PCM, events physically occurring in the real world, and events periodically occurring corresponding to time. A state event is defined for a conditional expression composed of the value (DPV) of a specific item (DPD) of the data model (DCM), and is caused by a change in those values.

These event models (ECM) are stored as a specific service dictionary (SSD) and are queried by HCT or HCM from each site. Also, if it cannot be disclosed to the outside, it is defined as an actual service dictionary (ASD) and stored in the dictionary server

Event and Condition Model (ECM)

Definition unit of event in the cyber world. It will trigger a PCM. When defined by the PCM process, the ECM may be defined in association with an event in the physical world if it is defined in the state of a specific value of the DCM

Event and Condition Instance (ECI)

Corresponding event model (ECM), an event state corresponding to an individual entity in the implemented state. At each point in real time, it has the same number of states as this ECS.

Event and Condition Record (ECR)

Recorded that an event actually occurred in the event state (ECS)

Event and Condition State (ECS)

Corresponding event model (ECM), an event state corresponding to an individual entity in the implemented state. At each time point in real time, it has as many states as this ECS

Event and Condition Expression (ECE)

Conditions in which an event occurs, constraint conditions, etc. are described by numerical expressions. Normally, formulas are described using elements such as external values (DPV) obtained from EDU and the like as elements.

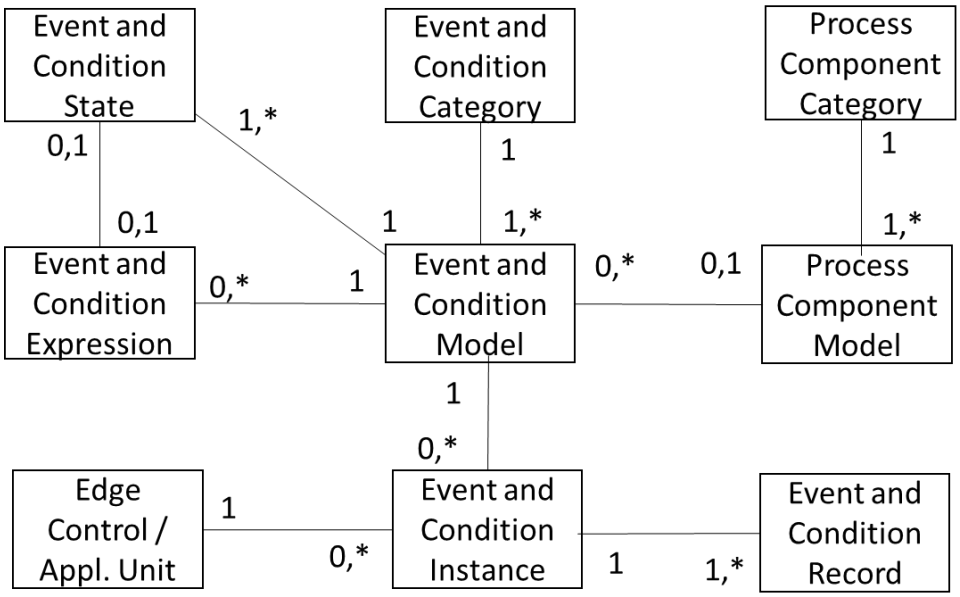

Figure 14: Event component Model (ECM)

An example of an entry sheet for describing an event model (ECM) is shown in Table 5 below.

Table 5: Input sheet of Event Component Model (ECM)

Event ID	1	DicID	2	Author	4
Event name				Date	
CategoryID		Category	3		

☐ Process Event 5

Process ID		Process name	
No	Type	Condition	Target data

☐ State Event 6

Data ID		Data name		
No	State ID	Item name	Expression or constraint	remarks

☐ Physical Event 7

No	description	target objects

☐ Calender Event 8

Repetition type				Regular cycle				
interval	0	min	select	Day		Hour		Minute
Num of cycle				W				week
Expire date				M				Day

Other constraints 9

1. The event ID is a unique symbol for specifying the target ECM. Also, the event name indicates the name representing the process. Also, in the explanation part, the outline description is described.

2. The dictionary ID is a symbol for identifying a dictionary. The dictionary classification is assumed here as an actual data dictionary.

3. The category ID and the category name are categories for classifying processes, and when specifying pre-registered ones, IDs are designated, and when there is no corresponding category, the ID is indicated as a blank and indicates an arbitrary category name.

4. Set the date on which the definition sheet was created and the registrant.

5. In case of an event occurring in connection with the execution of the process, check it and describe the

process ID and process name. For the classification, set the event category defined in the process model (PCM) such as start, end, interruption, abnormality

fic DCM, and describe the state name and condition, etc of each. Events will occur at the timing of that state.

7. Check if it is an event corresponding to an external event or user's operation and describe its contents.

8. Check for regular events. The unit of the repetition number is selected from one of minutes, hours, days, weeks, months, and years.

9. For events that conform to conditions other than the above as regular events.

6. Trade Profile

6.1 Structure of Trading Profile

The profile is information indicating the content and method of the data transaction. The profile is stored in the HCT of each site, and it is taken out every time a data transaction is carried out. Both the data-providing side and the data[-using side have profiles related to one data transaction separately and are uniquely associated with the data transaction contract ID. Profiles are created in the data transaction contract phase. The contents of each correspond, but they are different, you cannot directly know the contents of the profile of the other party in that form.

Trade Contract Profile (TCP)

The form and method of data distribution and contract contents are settled between two sites, iIncluding: data storage method, attribution of rights, billing method, prohibited items, etc.

Trade Data Profile (TDP)

Data descriptions (DCM) used in messages at the time of sending or receiving messages are described in sa pecific data dictionary or an actual data dictionary. The DCM of the requested data also shows its structure, for example in the case where it is represented by concatenation of a plurality of DCMs in the specific data dictionary.

Trade Service Profile (TSP)

Defines the contents of process (PCM) and related events (ECM) on the side providing and using data. Also shown are ECUs, EAUs, EDUs, etc. actually to be executed. DCM defined in a TSP can be queried by the HCM

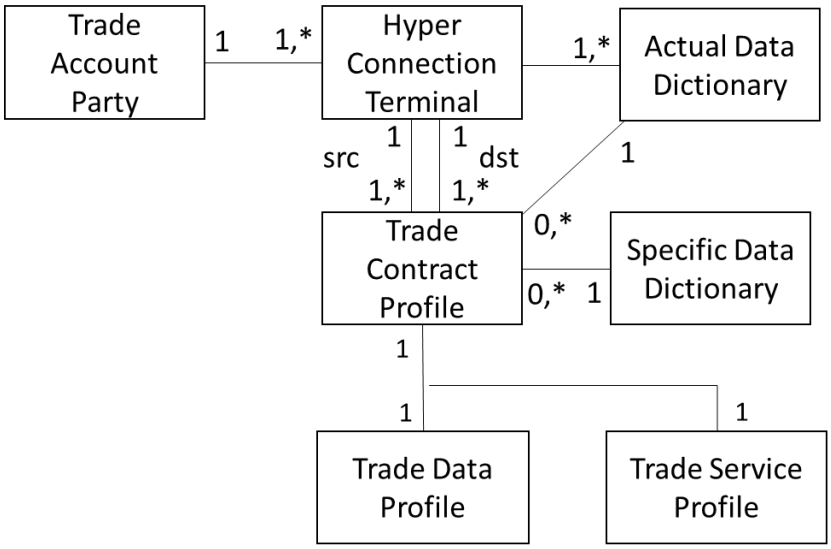

Figure15: Object model of Trade Profile

In a data transaction, a data provider and a data user have one common transaction contract profile (TCP), a different transaction data profile, and a transaction service profile on each side. As shown in Fig. 16, the transaction data profile has a dictionary conversion map (DTM) and an item conversion map (PTM) between two data dictionaries. One of the target data dictionaries is a specific data dictionary (SDD), and the other is the actual data dictionary (ADD) of the site on its own side.

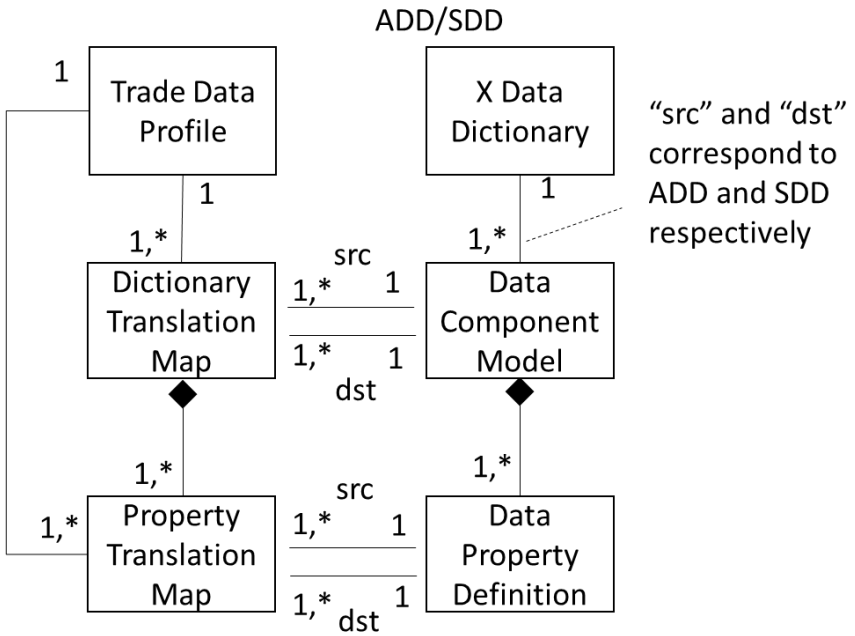

Figure 16: Object model of Trade data Profile

As shown in Fig. 17, the transaction service profile has a process implementation (PCI) in which a process model (PCM) is developed in a specific edge unit and an event implementation (ECI) in which an event model (ECM) is embodied , These are related to actual edge units (ECU, EDU, EAU) existing on each edge side as shown in Figs. 18 and 19

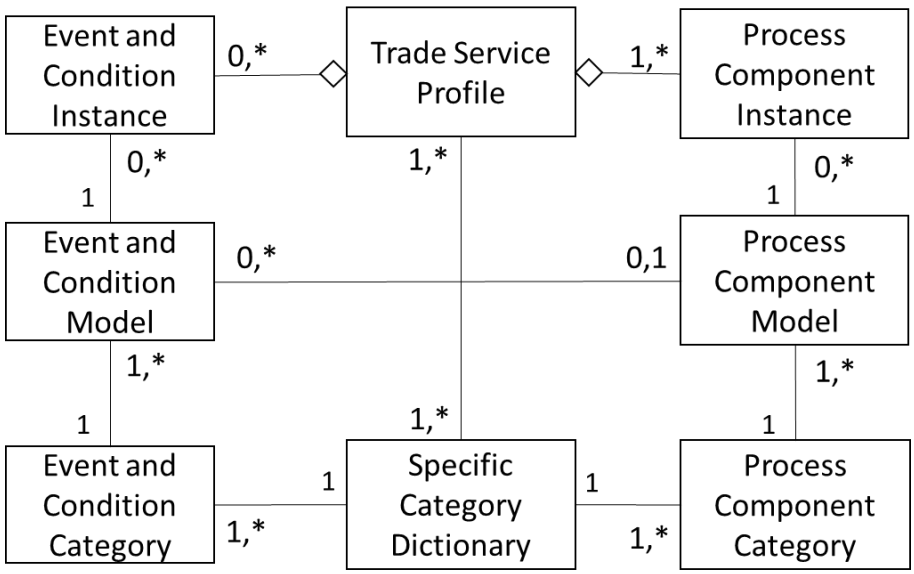

Figure 17: Object model of Trade Service Profile

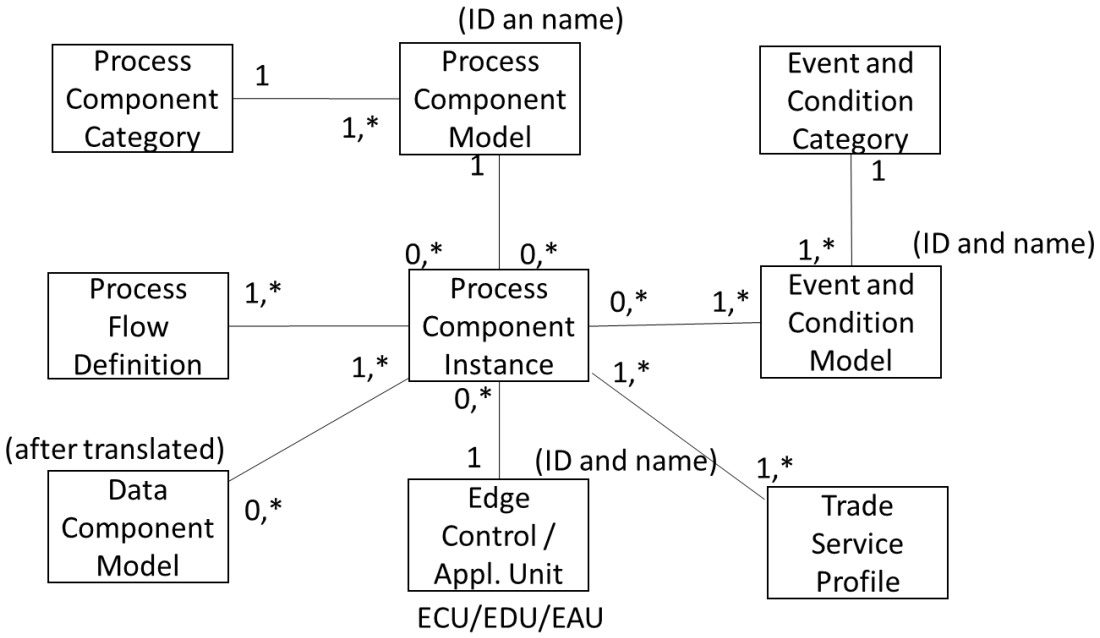

Figure 18: Object model of Process Component Instance

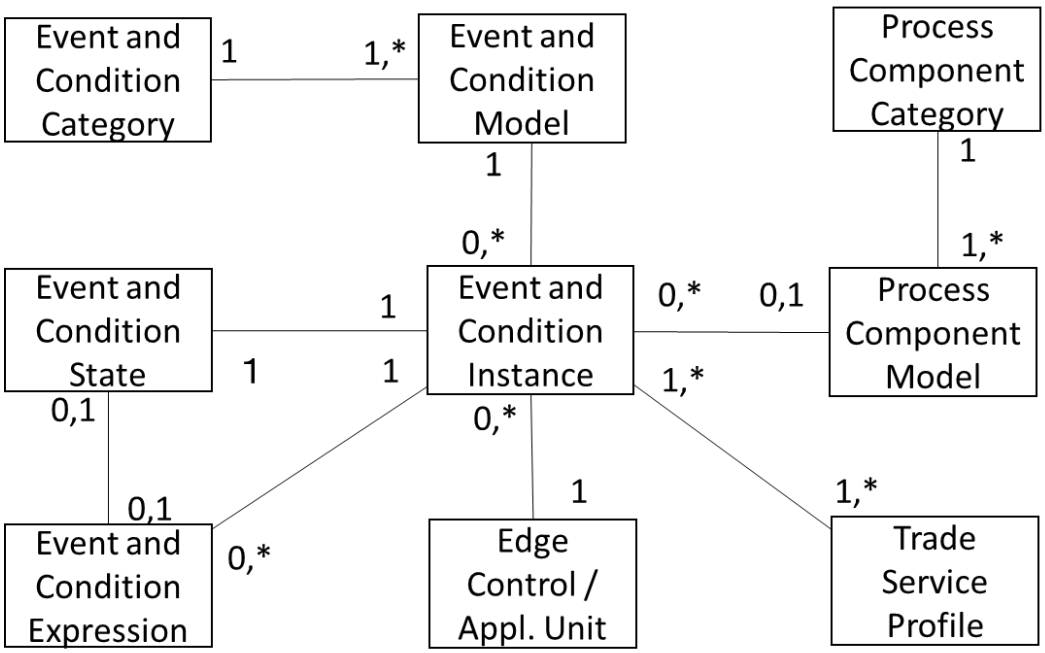

Figure 19: Object model of Event and Condition Instance

6.2 Trade Contract Profile

Registration is recorded in the cooperation manager (HCM) in the same way as registering dictionaries. These definitions are based on data model (DCM), process model (PCM), and event model (ECM). An example of the definition sheet of the transaction contract profile (TCP) is shown below.

Table 6: Input Sheet of Trade Contract Profile

Firm ID	1	Firm name	3		
Contract ID		Role	Provider	Date	4
Contract	2			Author	

Customer ID	5	Customer			
Dic ID	6	Dic name			
Trigger	7 Provider	Start Event	8	Event Type	
Pre-cond	9			Start date	
Post-cond				End date	

To HCT	10	ECU/EAU		Unit type	
Fom HCT		ECU/EAU		Unit type	

TDP	11	profile name	
From TSP		profile name	
To TSP		profile name	

☐Request Parameter

No	Property type	description
	12	

■terms and conditions

Exp. Store	13
Exp. Use	
Reusable	
Dupplication	

■Contract and Accounting

14

1. Set the ID and name of the business operator (herself or himself) who conducts the transaction.

2. Set the ID and transaction name of this transaction.

3. For role sharing, either the provider or the user is set up.

4. Set the date on which the profile was created and the name of the person in charge.

Set the business entity ID and the name of the counterparty side as a business partner.

6. Specify the ID of the specific data dictionary and the dictionary name to be used in the transaction.

7. The start trigger is "provider" in the case of the data-provider side (PUSH), and "user" in the case of the data-user side (PULL). It does not matter whether you are a provider or a user. Even if it is PULL in the transaction contract phase, it may become PUSH in the implementation phase. Here we show the form of the implementation phase.

8. The start event indicates an event that determines the timing at which the data provider transmits data. In the case of a PULL type, it is blank.

9. The transaction start requirement and the transaction end requirement, designate the necessary requirements and enter the date of actual start and end.

10. The business partner terminal designates the ID of the partner's HCT, and the execution unit indicates the unit (the data-providing process, the hardware on which the utilization process is executed) on which the process is executed. Unit classification is either ECU or EAU or EDU. The business partner server (HCS) can be retrieved from the business partner terminal. Also, if the execution unit is not connected directly from the terminal, it is assumed that the terminal knows the unit to be relayed.

11. The data profile specifies the ID and the name of the corresponding data profile, the providing service designates the ID and the name of the service profile of the data-providing side, and the usage service includes the service profile of the side using the data Specify ID and name.

12. If there is a request parameter, check it, and specify interpretation of the request parameter in the case of a PULL-type transaction during the transaction execution phase. In accordance with this content, the contents of the data provided by the data-providing unit are corrected.

13. For restrictions on data transactions, restrictions on storage, usage restrictions, restrictions on modifications, disclosure limits are specified here as text.

14. Specify other contract conditions, billing methods, etc. in text form.

6.3 Definition of Trade Data Profile

Below is an example of the entry format for entering the transaction data profile (TDP).

Table 7: Input sheet of Trade Data Profile

Firm ID	1	Firm name	3		
Profile ID		Customer ID		Date	4
Profile name	2			Author	

dictionary ID	5	Dictionary	

Data ID	6			Type	7
Data name					
Category ID		Category	8		

Pre-cond	9	Start date	
Post-cond		End date	

Exp. Store	10
Exp. Use	
Reusable	
Dupplication	

☐Linked DCMs

No	Data ID	Data name	Link key	Join type
1	11		12	13
2				

1. Set the ID and name of the business operator (self) who conducts the transaction.
2. Set the data profile ID and profile name.
3. In the transaction ID, set the ID of the corresponding transaction contract profile.
4. Set the date on which the profile was created and the name of the person in charge.
5. In the actual data dictionary, set the ID and name of the actual data dictionary for which these data are defined. Once the actual data dictionary is determined, the category dictionary is uniquely determined.
6. For the data ID, designate the ID and name of the data that are the subjects of the transaction.
7. For the classification, either "use" or "offer" is set. This indicates whether the process is the side that

uses data or the side that provides data.

8. Categories are specified by ID and name of the corresponding categories When there is no item corresponding to the category dictionary, ID is unnecessary.

9. The transaction start requirement and the transaction end requirement designate the necessary requirements and enter the date of actual start and end.

10. Regarding restrictions on data transactions, restrictions on storage, usage restrictions, restrictions on modifications, disclosure limits shall be specified here as text.

11. When the target data are linked with other data (DCM), they are shown here.

12. If the connection destination is blank, the primary data are targeted. If the connection destination has a connection (secondary connection), specify the number of consolidated data.

13. The concatenation method specifies the number of the concatenation method in the dictionary of each data model (DCM).

Table 8: Input Sheet for Profile of Translation Map

DCM of SDD		SDD		Profile ID	
data name	1				2

☐Linked DCMs

No	target DCM	target data name	link key	join type
1	3		4	5
2				

property translation map

No	link	actual data item	key	specific data item	remarks
1	6			7	8
2					
3					

1. The common data ID indicates the DCM in the common data dictionary corresponding to the data (element of the actual data dictionary). It also indicates the ID of the specific data dictionary to which the common data ID belongs.

2. For the profile ID, set the ID of the transaction data profile.

3. If there is a connection in common data, check fir the existence of consolidated data and specify the ID and name of the data (DCM).

4. For the connection destination, specify the ID of the DCM to which the DCM is connected. If blank, primary data are targeted. In a case where the link destination has a link (secondary link), the number of the line of consolidated data is specified.

5. In the concatenation method, specify the number of the connection method in the dictionary of each data model (DCM).

6. The definition of the map (PTM) between data items (DPD) is defined by showing the respective items for DCM on the specific data dictionary side and the DCM on the actual data dictionary side. When the concatenated part is blank, it becomes the primary DCM, and otherwise, the number of the connected data is set.

7. Associate one item of the conversion map with the other item. If there is a connection, specify he number of consolidation in common data. If there are special notes on conversion, it is stated in the remarks.

6.4 Definition of Trade Service Profile

Below is an example of the entry format when entering the transaction service profile (TSP). The transaction service profile identifies the processes and events handling the target data.

Table 9: Input Sheet for Trade Service Profile

Firm ID	1	Firm name	3		
Profile ID		Customer ID		Date	4
Profile name	2			Author	

Dic ID	5	Dictionary	
Data ID	6	Data name	

Process ID	7	Instance ID	8	Type	9
Process					
Category ID	10	Category			
Unit D	11	Unit name			
Terminal ID	12	Terminal			

Pre-cond	13	Start date	
Post-cond		End date	

■trace ability of event recording

No	event ID	Event name	type	Record
	14			

■conditions of data usage

No	Data ID	Data name	type	constraint
	15			

■authrized data users

No	Unit ID	Unit name	access level	permission
	16			

1. Set the ID and name of the business operator (herself or himself) who conducts the transaction.
2. Set the ID of the service profile and profile name.
3. In the transaction ID, set the ID of the corresponding transaction contract profile.

4. Set the date on which the profile was created and the name of the person in charge.

5. In the actual service dictionary, the ID and the name of the actual service dictionary for which this process is defined are set. If the actual service dictionary is decided, the category dictionary is uniquely determined. Processes and events to be set belong to the same service dictionary.

6. In the data ID, specify the ID and name of the data targeted for the transaction.

7. In the process ID, specify the ID and name of the process.

8. An arrangement ID is set as an ID indicating the state in which the process is installed in the unit.

9. "Provide" or "Use" is set for the classification. In the case of a provision process, events describing the timing of providing data are described (actually, they are OR relationships). In the case of the utilization process, in order to record the result of using the data, describe plural kinds of events as a result thereof. (Actually, it becomes OR.)

10. Category indicates the category of the data by ID and name. When there is no item corresponding to the category dictionary, ID is unnecessary.

11. The unit indicates the ID and name of the unit that carries out the process. A unit refers to an edge unit that executes a process, and it is a controller (ECU), an application (EAU), or a device (EDU) registered in the HCT in advance.

12. The terminal shows the ID and name of the terminal (HCT) to which the unit carrying out the process belongs.

13. Specify necessary requirements and start date and date of completion, as the start requirement and end requirement.

14. For events, events related to the relevant process are indicated, and in the case of a process on the user side in particular, an event for recording the usage record is defined. For the classification, specify process events, state events, etc. described in the event model.

15. In the constraints of related data, data generated / referenced / input / set / changed by the process is shown, and constraints on their handling are described. For data to be traded, the classification indicates whether the process is in a position to use the data or in a position to provide the data.

16. In the case of data secondary usage, this indicates the units that are permitted to handle data, and sets the disclosure level and access restriction content in text format for each.

7. Distributed Dictionaries

7.1 Concept of distributed dictionary

A dictionary gave the definition of the meaning of a word in other words originally, and a definition of the relationship between a set of vocabularies and another set of vocabularies. In this document, a set of data models (DCM) is defined as a dictionary. In other words, the dictionary expresses the meaning of a DCM using natural language. However, the same object that actually exists may be represented by a plurality of different DCMs. For example, the terms and usage used at production sites are often different for each company and factory. In such a case, those DCMs having the same meaning are subordinate to a multitude of dictionaries.

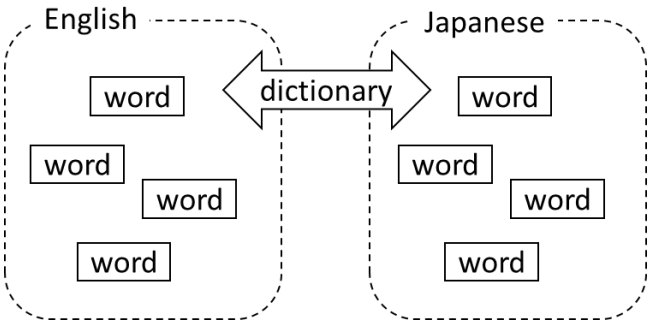

Figure 20: General concept of dictionary

In a case where the same object has a plurality of different expressions according to the dictionary, a dictionary conversion map (DTM) is used to convert one expression to another expression. The dictionary conversion map can take different actual dictionaries and define them, but in this specification, after defining a specific dictionary in a neutral form, all the actual dictionaries are separated from the specific dictionary to make it a conversion.

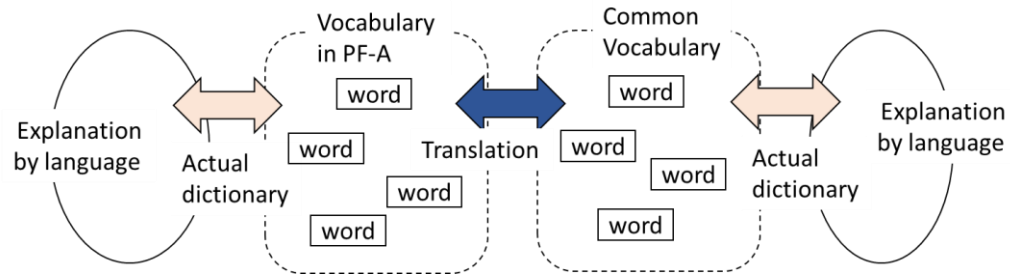

Figure 21: Translation map between dictionaries

The specific data dictionary (SDD) and the actual data dictionary (ADD) defined as the data dictionary have the same format. The data dictionary has DCM as an element. Therefore, for convenience, the specific data dictionary and the actual data dictionary can coexist in one data table and can be managed by uniquely identifying it with an identification code or the like.

The service dictionary has information on processes providing or utilizing data, and information on events that execute those processes. A service dictionary defined as a specific dictionary is treated as a reference model, and a process model (PCM) and an event model (ECM) corresponding to actual services existing on individual sites are registered in the actual dictionary

Both the specific data dictionary and the actual data dictionary has a unique ID. Also, the data dictionary has the revision number and the ID of the previous version. The specific service dictionary also has a unique ID and there can be multiple existences. The specific service dictionary also has the revision number and the ID of the previous version.

Categories are defined for data model (DCM), process model (PCM) and event model (ECM). These categories are registered in advance in the category dictionary, and they are selected from the category dictionary for use when defining each model. The category dictionary is only a specific dictionary, and no actual dictionary exists.

The categories of the data model (DCM), the process model (PCM), and the event model (ECM) are category dictionaries (CDD) uniquely designated by each data dictionary (SDD or ADD) or service dictionary (SSD or ASD) that has been registered.

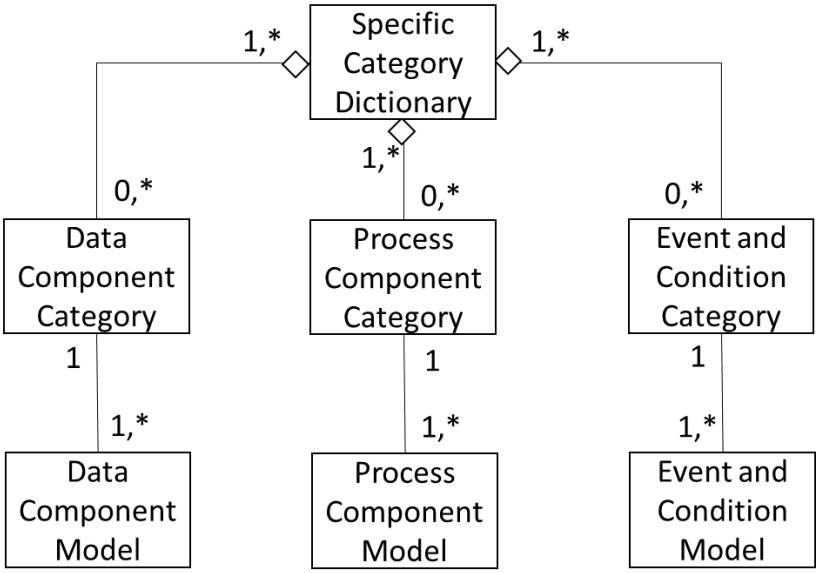

Figure 22: Object model of Specific Category Dictionary

Each item (DPD) of DCM in the data dictionary has a default value setting.

Specific Data Dictionary (SDD)

Definition information of all DCM mutually agreed among multiple sites. It contains one or more DCMs. A plurality of specific data dictionaries may exist.

Actual Data Dictionary (ADD)

This has a list of all the DCMs to be used for each site or terminal (HCT) or the controller (ECU) under the terminal.

Specific Category Dictionary (SCD)

Data category (DCC), process category (PCC), event category (ECC) summarized. The same category may be defined as different DCC, PCC, ECC depending on the dictionary difference.

7.2 Translation of DCM among dictionaries

A translation map sets the difference between the specific dictionary and the actual dictionary. In the case of specific categories, it corresponds to the definition of the difference from the previous version in the dictionary revision. With the information of the dictionary conversion, the map associates the definition before conversion (for example, source DCM) with the definition after conversion (destination DCM).

Dictionary Translation Map (DTM)

As for an actual data dictionary, specific data dictionary, etc., when there are definitions of two or more different DCMs for the same subject, they show their relationship, and associate one or more concatenated data items of the DCM to be converted to one DCM of the conversion destination.

Property Translation Map (PTM)

This indicates the correspondence among DCM's items (DPD) in a dictionary conversion.

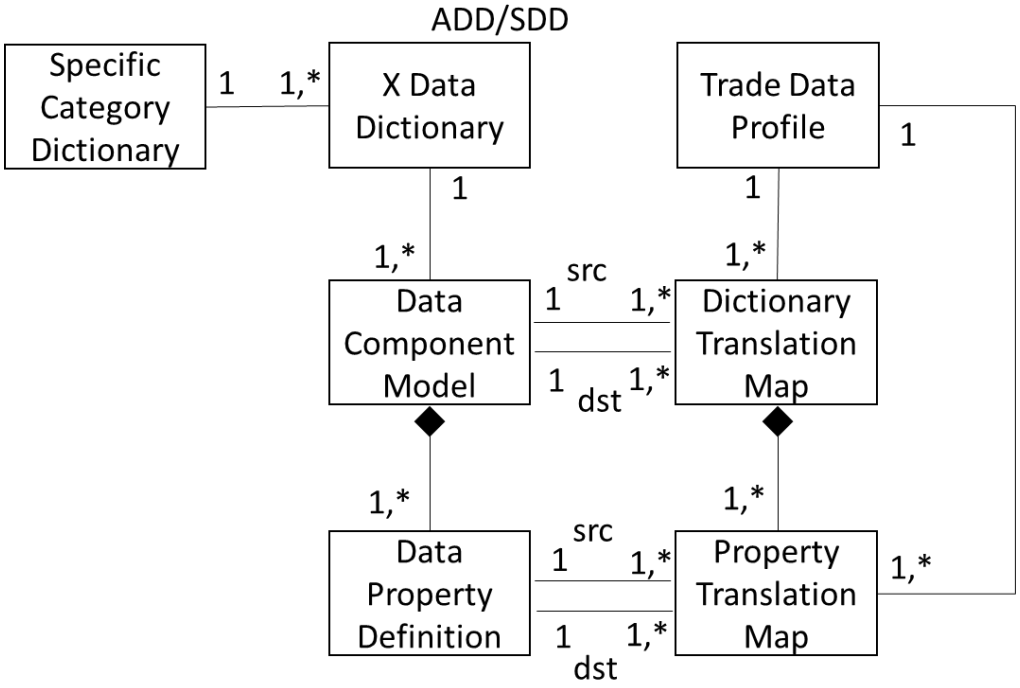

Figure 23: Object model of Dictionary Translation Map

7.3 Mapping mechanism

Normally, mapping is done at the desk in the transaction contract phase, or interactively by the cooperation manager (HCM). The basic concept of mapping is shown below. Here, the dictionary conversion map is defined based on the post-conversion (destination DCM). The dictionary ID of the conversion destination and the dictionary ID of the conversion source are confirmed in advance.

As shown in Fig. 24, both DCMs to be mapped are one to one. In this case, as a result of the mapping, an item conversion map (PTM) is defined, and the data item (DPD)

possessed by both DCMs is set therein. On the other hand, when there is a plurality of corresponding DCMs, as shown in Fig. 25, DCM (src DCM 2 in the drawing), to be linked with the primary DCM (src DCM 1 in the figure) it is brought out as well as the data item (DPD), in this case, it is also necessary to record the DCM connection information.

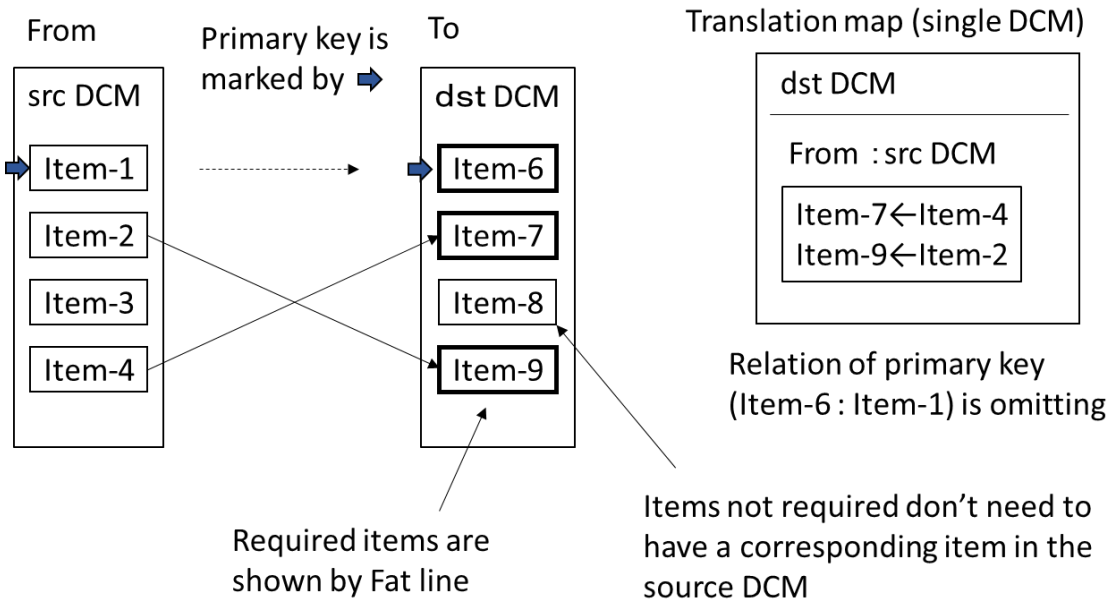

Figure 24: Mapping for one to one DCMs

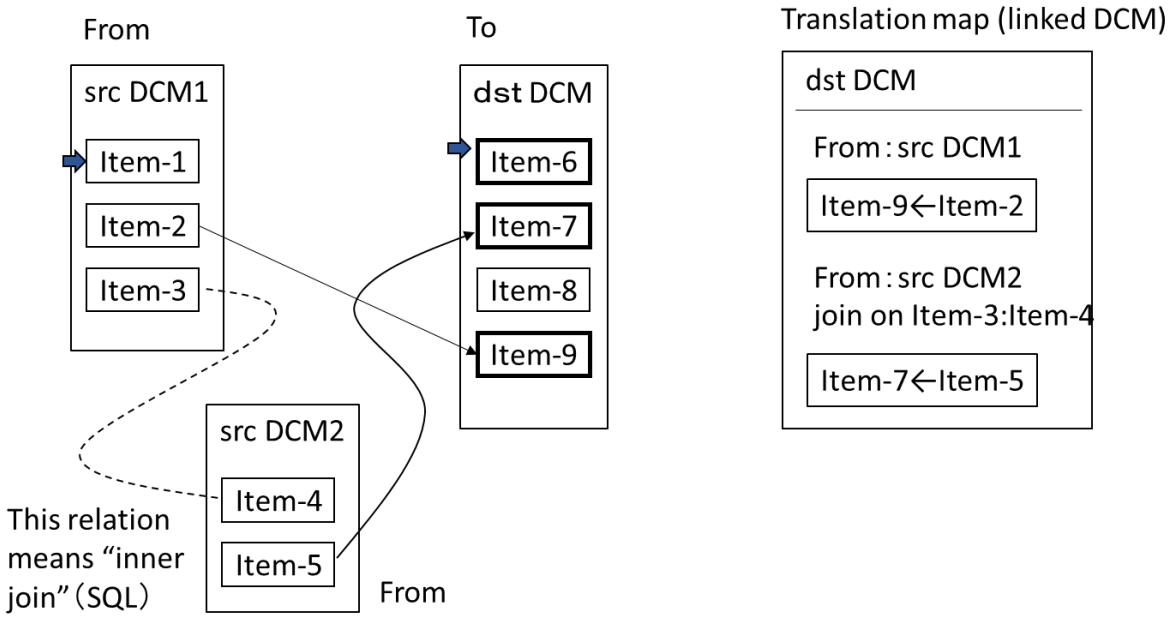

Figure 25: Mapping for one to many DCMs

7.4 Example of translation dictionary

Here, a specific example for interactively executing the dictionary conversion map in the PULL type use case will be described. First, as Step 1, mapping is performed between the actual data dictionary possessed by the data-using side and the specific data dictionary.

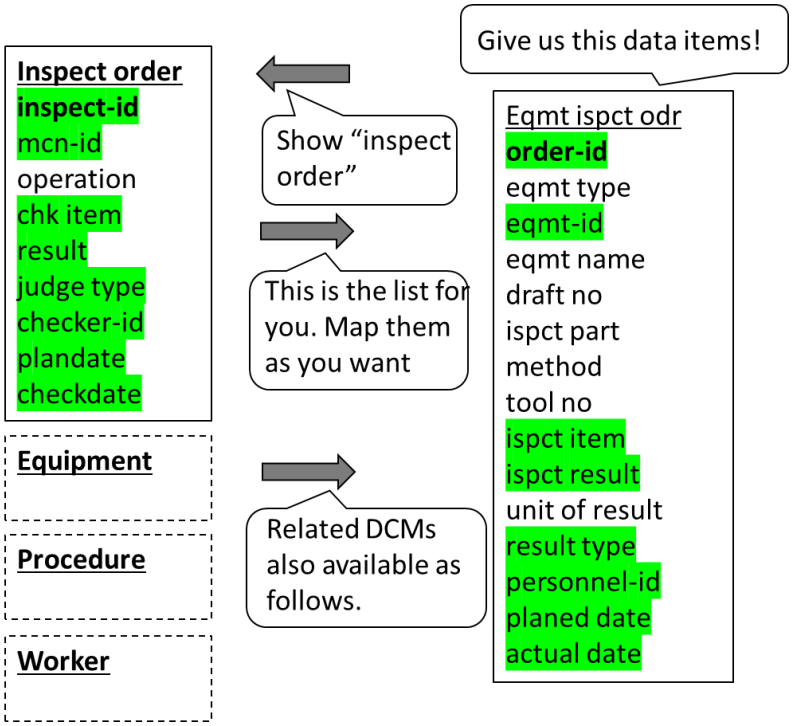

Figure 26: PULL type contract (Step 1)

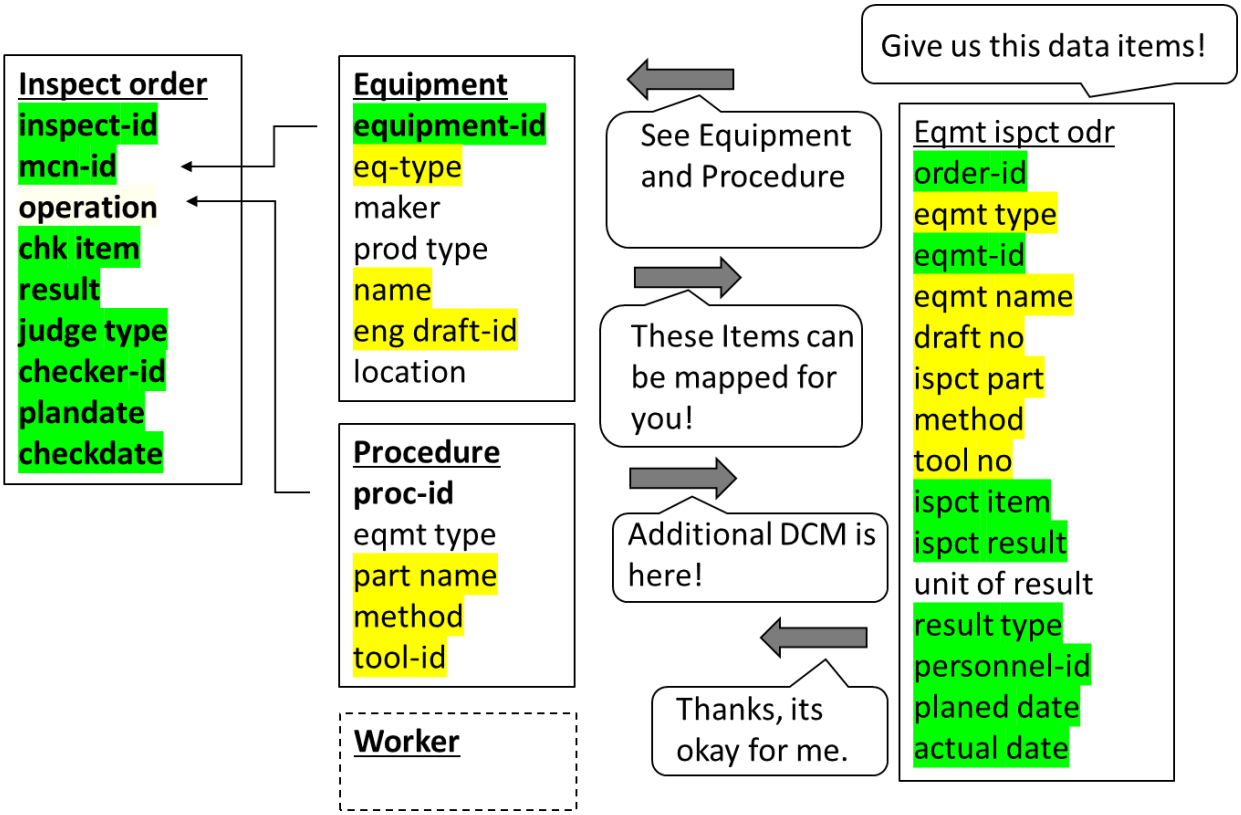

Figure 27: PULL type contract (Step 2)

In this example, since the primary DCM on the specific data dictionary side was the inspection instruction, and the data items that it had were not sufficient for the DCM called for in the initial equipment inspection instruction, the DCM that is the connected DCM and the inspection procedure brought the items into correspondence. As a result of these mappings, a dictionary conversion map (DTM) and an item conversion map (PTM) were obtained as shown in Fig. 28.

Dictionary translation map(DTM)

	src	dst	Join
DTM-01	Eqmt ispct odr	Inspect order	--
DTM-02	Eqmt ispct odr	Equipment	Join-01
DTM-03	Eqmt ispct odr	Procedure	Join-02

Property translation map（PTM）

	src DCM	src DPD	dst DCM	dst DPD
PTM-01	Eqmt ispct odr	Order-id	Inspect order	Inspect-id
PTM-02	Eqmt ispct odr	Eqmt type	Equipment	Eq-type
PTM-03	Eqmt ispct odr	Eqmt-id	Inspect order	Mcn-id
PTM-04	Eqmt ispct odr	Eqmt name	Equipment	name
PTM-05	Eqmt ispct odr	Draft no	Equipment	Eng draft-id
PTM-06	Eqmt ispct odr	Ispct part	Procedure	part name
PTM-07	Eqmt ispct odr	method	Procedure	method
PTM-08	Eqmt ispct odr	Tool no	Procedure	Tool-id
PTM-09	Eqmt ispct odr	Ispct item	Inspect order	Chk item
PTM-10	Eqmt ispct odr	Ispct result	Inspect order	Result
PTM-11	Eqmt ispct odr	Result type	Inspect order	Judge type
PTM-12	Eqmt ispct odr	Personnel-id	Inspect order	Checker-id
PTM-13	Eqmt ispct odr	Actual date	Inspect order	Checkdate

Figure 28: Translation map for PULL-type contract (user side)

Subsequently, in order to ascertain whether the requested contents expressed as a specific data dictionary can be provided by the data-providing side, the request is made in Step 3 and, as a result, from the providing side to Step 4 by the DCM. This has been shown to be possible; however, it can be seen that the contents of the original request are not all satisfied with only this DCM item.

Step 5-1 (Figure 31) shows a case where the agreement is made on the contents, but if that is not enough, as shown in Step 5-2 (Figure 32), the data-providing side further applies the concatenated DCM and converts the increasing data items. Ultimately, in Step 6, the data-providing side has completed the conversion map of the data-providing side, including the data item possessed by the inspection daily report, which is the connected DCM.

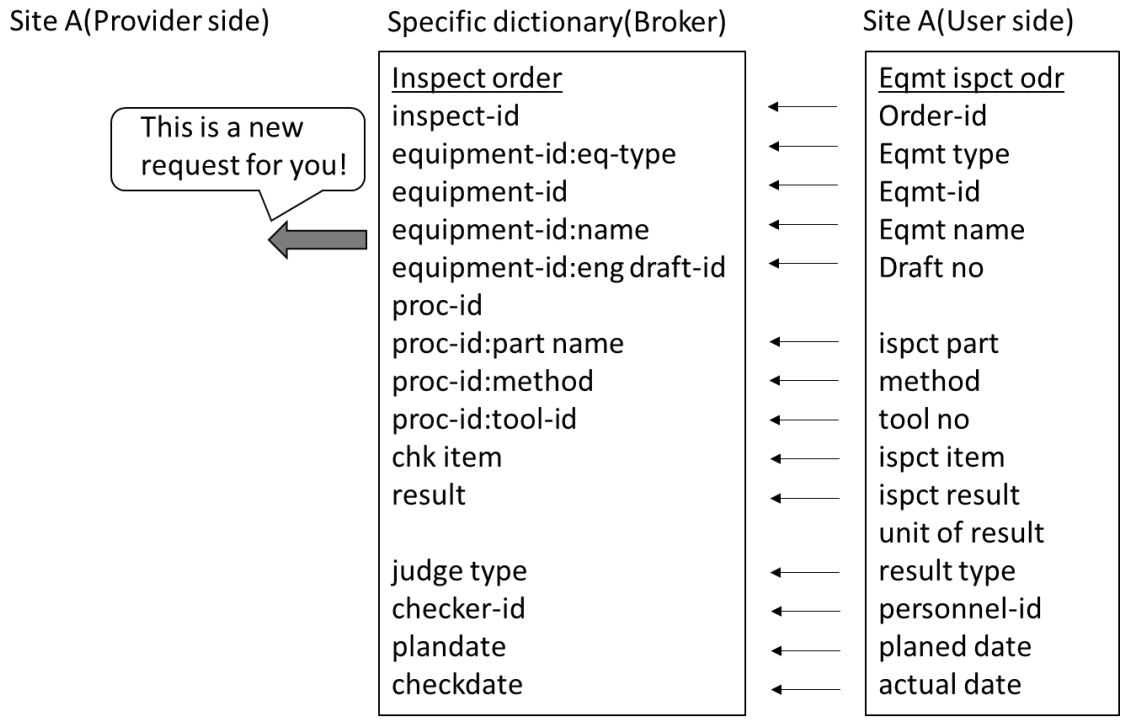

Figure 29: PULL-type contract at provider side (Step 3)

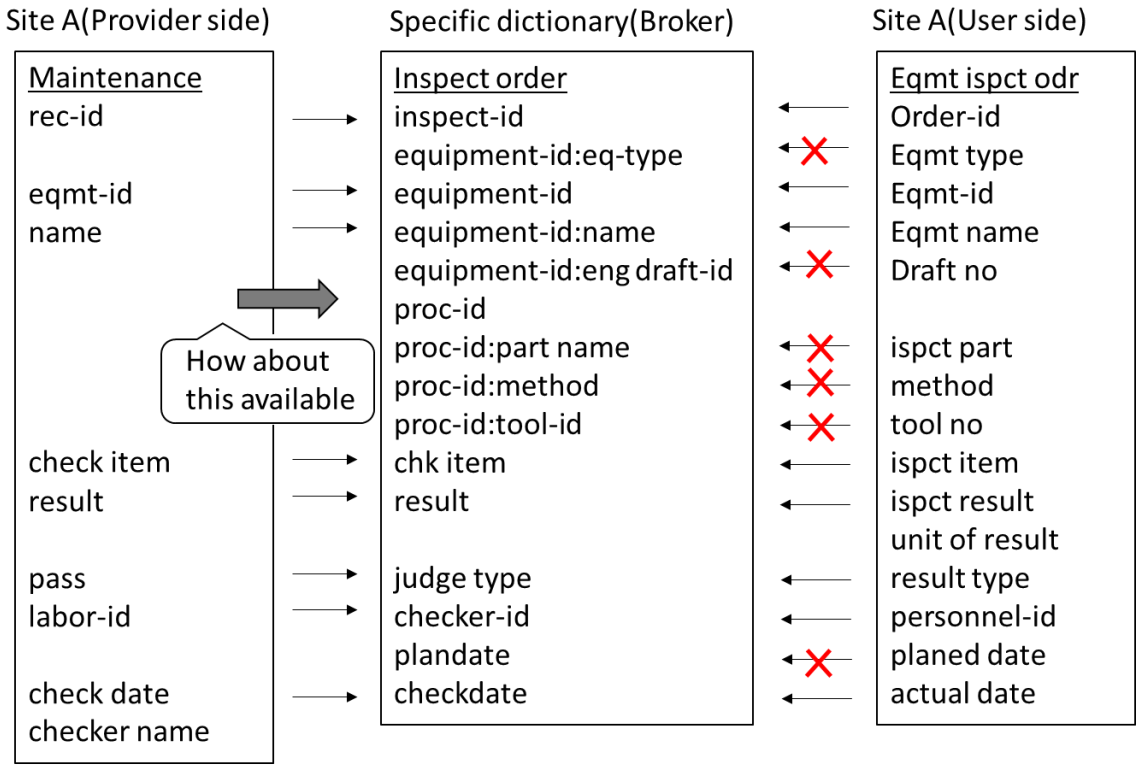

Figure 30: PULL-type contract at provider side (Step 4)

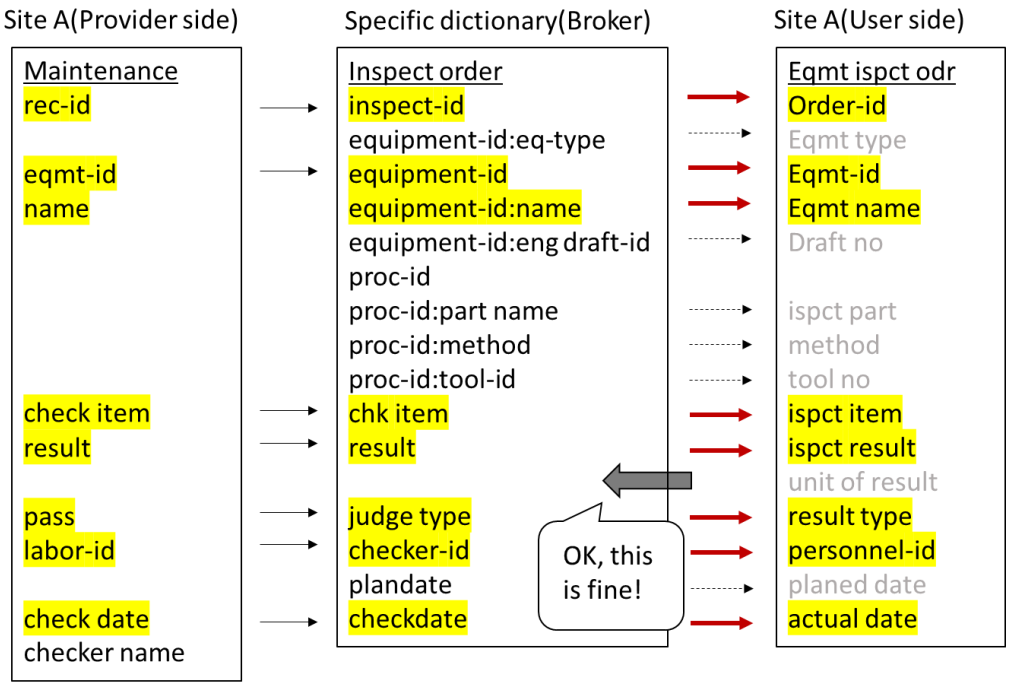

Figure 31: PULL-type contract at user side (Step 5-1)

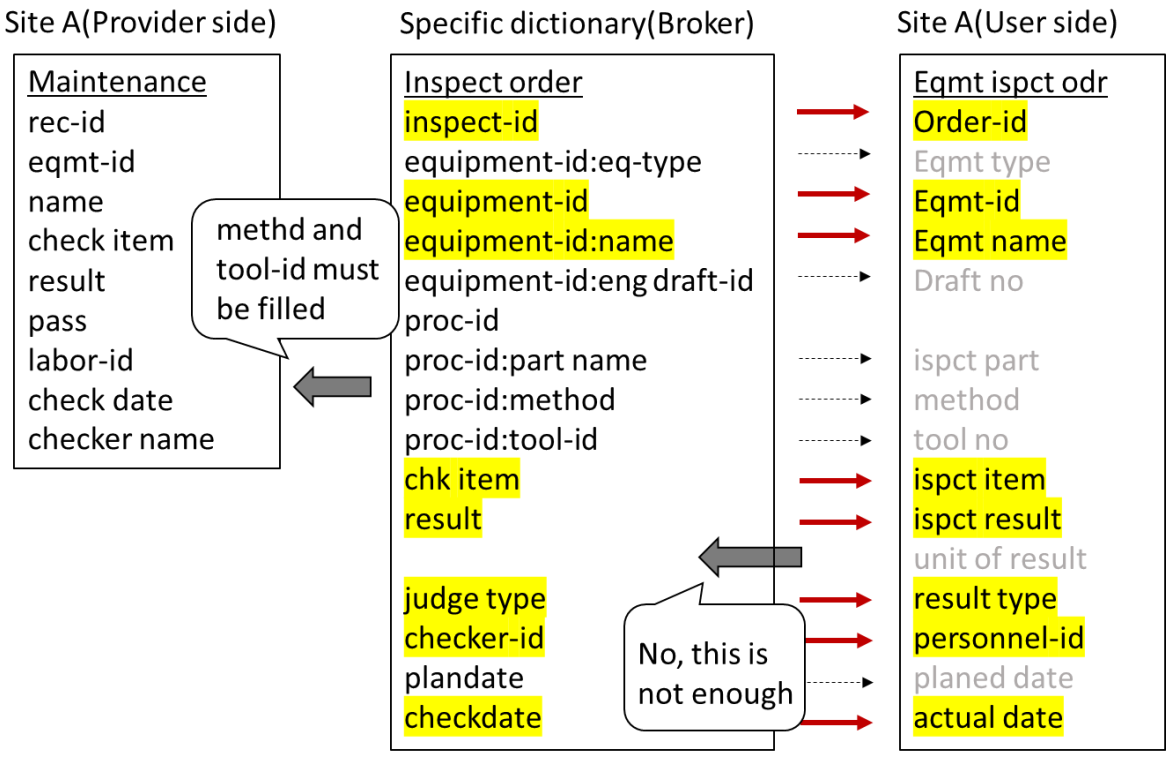

Figure 32: PULL-type contract at user side (Step 5-2)

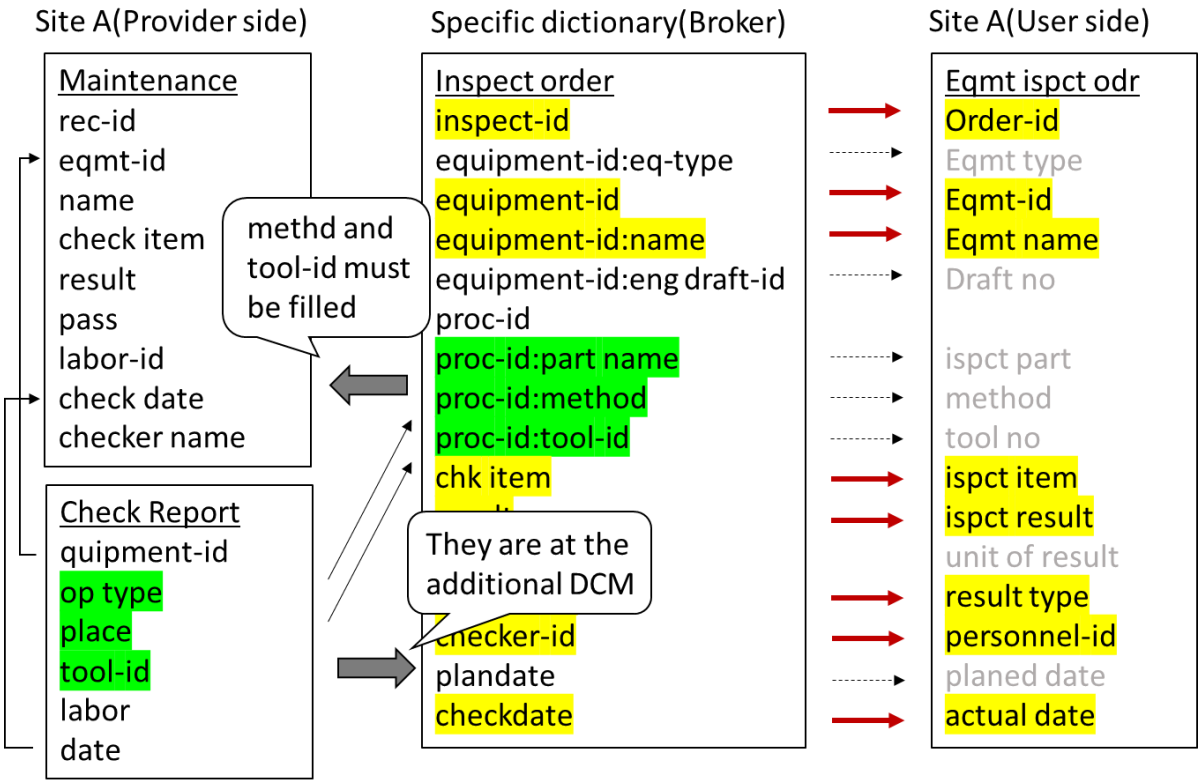

Figure 33: PULL-type contract at provider side (Step 6)

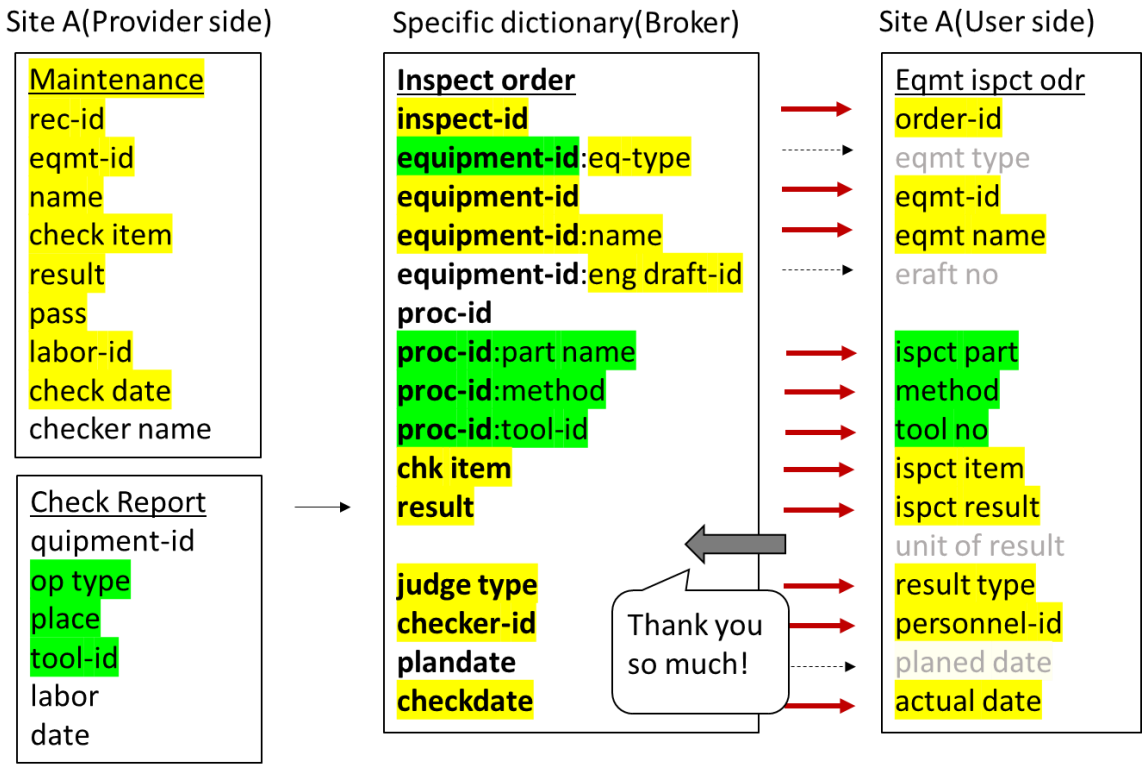

Figure 34: PULL-type contract at user side (Step 7)

Preconditions for mapping processing in dictionary conversion are summarized below.

1. The DCM can be qualified with the category (PCC) of the service (PCM) that uses it. Even in the same DCM, when a service qualifier is attached, it can be treated as a separate item, and items can be added, deleted, default values can be changed, and so on.

2. Conditions (where clause in a SQL statement) cannot be attached to a DCM that a user can request. For example, when requesting the operation history, it is not possible to designate time designation or condition designation each time. However, excluding cases where the negotiation is made independently with the provider depending on the value set for the parameter set at the time of request

3. If there is a required item at the time of mapping and the item cannot be provided, set a default value. If it is impossible with the default value, the transaction is not established.

4. After the mapping is decided, the actual data record is reconfigured each time according to the conversion table. Since reconfiguration is performed on the HCT, in the case of concatenated DCM, there are multiple DCMs to be transmitted per transaction. In this case, the number of records of the concatenated DCM can be grouped for each primary key for the number of records of the underlying DCM.

5. In the case of a PULL-type operation, the data requester can make the specific data dictionary the same as the actual data dictionary. In the case of a PUSH-type operation, the data-providing side can make the specific data dictionary the same as the actual data dictionary. In this case, conversion of the dictionary becomes a one-time event.

7.6 Data preparation inside the edge

Likewise, the coordination terminal (HCT) and the coordination unit (ECU) within the edge need to deconstruct and reconstruct data corresponding to the concatenated structure for the component or data model (DCM) under its control. If one request consists of multiple DCMs, inquiries are made for each, and the data (DCR) are integrated.

Figure 35 shows the inquiry from the data-providing side HCT to the ECU. Furthermore, Fig. 36 is a schematic diagram showing a case where they are distributed and arranged in

a plurality of units. The integrated DCM can be branched on a concatenated DCM basis.

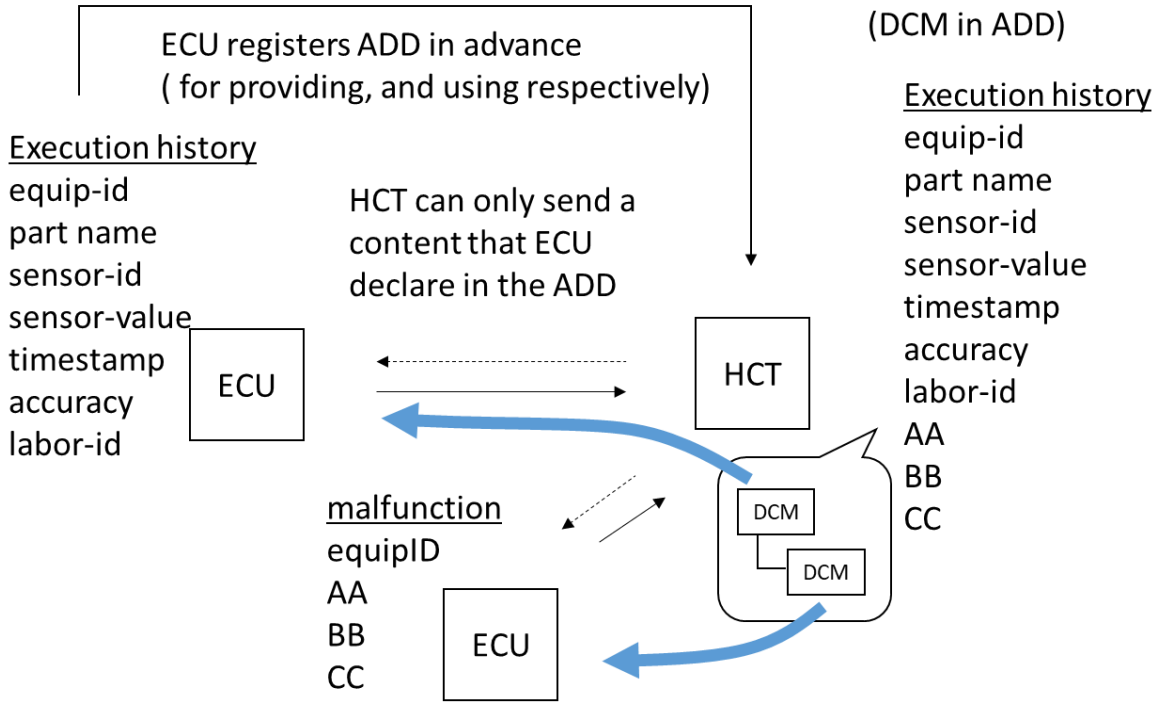

Figure 35: Data providing by ECU

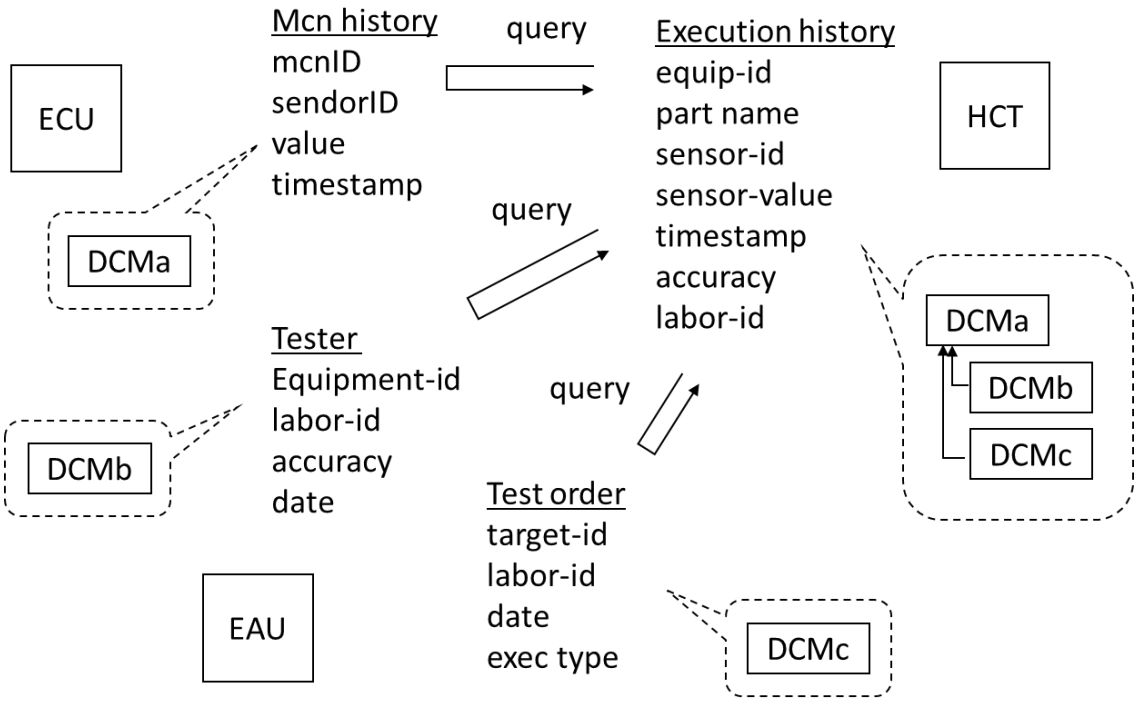

Figure 36: Multiple queries to EDUs

8. System use case

8.1 System Integration Phase

In the integration phase, make the following registration:

Table 10 Data registration on servers (1)

Registered content	Management server
Trade Account Party (TAP)	HCS
Hyper Connection Terminal (HCT)	HCS
Edge Unit (ECU, EAU, EDU)	HCT
Data Component Model (DCM)	HDS
Process Component Model (PCM)	HDS
Event Component Model (ECM)	HDS

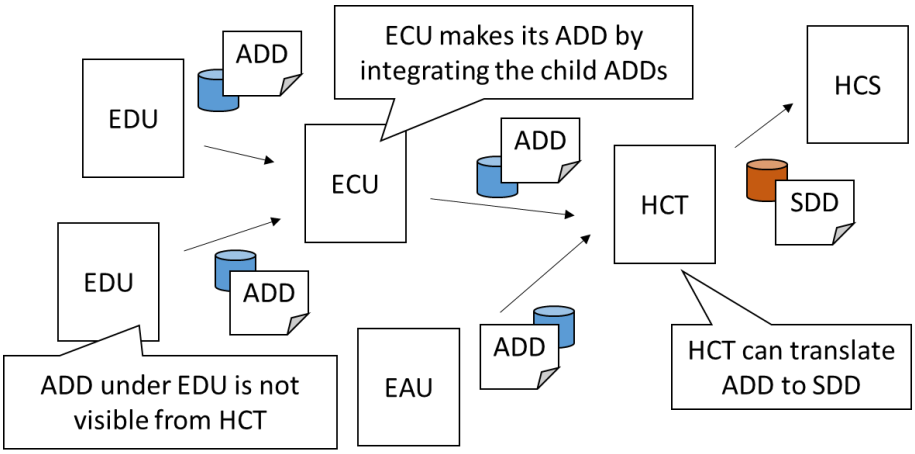

Figure 37 Register Dictionary of EDU

Restrictions include the following:

1. ECU and EAU must register themselves with HCT and receive certification
2. The ECU and the EAU have to register the services to use data themselves as a PCI in the HCT.
3. The ECU has to register the lower EDU to be managed in the HCT.
4. The ECU must register the service for which the EDU uses data to the HCT as a PCI (process implementation).
5. The ECU and the EAU must select or register one ADD to be used with the HCT.
6. ECU and EAU must select or register DCM to be used with HCT to HDS.
7. The ECU and the EAU shall correct if there is a shortage of items in the DCM used with the HCT.

8.2 Trade Contract Setup Phase (Common)

In the transaction contract phase, we make the following registration:

Table 12 Data registration on servers (2)

Registered content	Management server
Additional Actual Data Dictionary (ADD)	HDS
Additional Specific Data Dictionary (SDD)	HDS
Dictionary Translation Map (DTM)	HDS
Trade Data Profile (TDP)	HCT
Trade Service Profile (TSP)	HCT

In addition, regarding individual transactions, there are the following contents:

Registered content	Management server
Trade contract request	HCT
Confirm trade contract request	HCT
Sample data for mapping	HCT

The outline of the transmission of sample data is shown in the following figure:

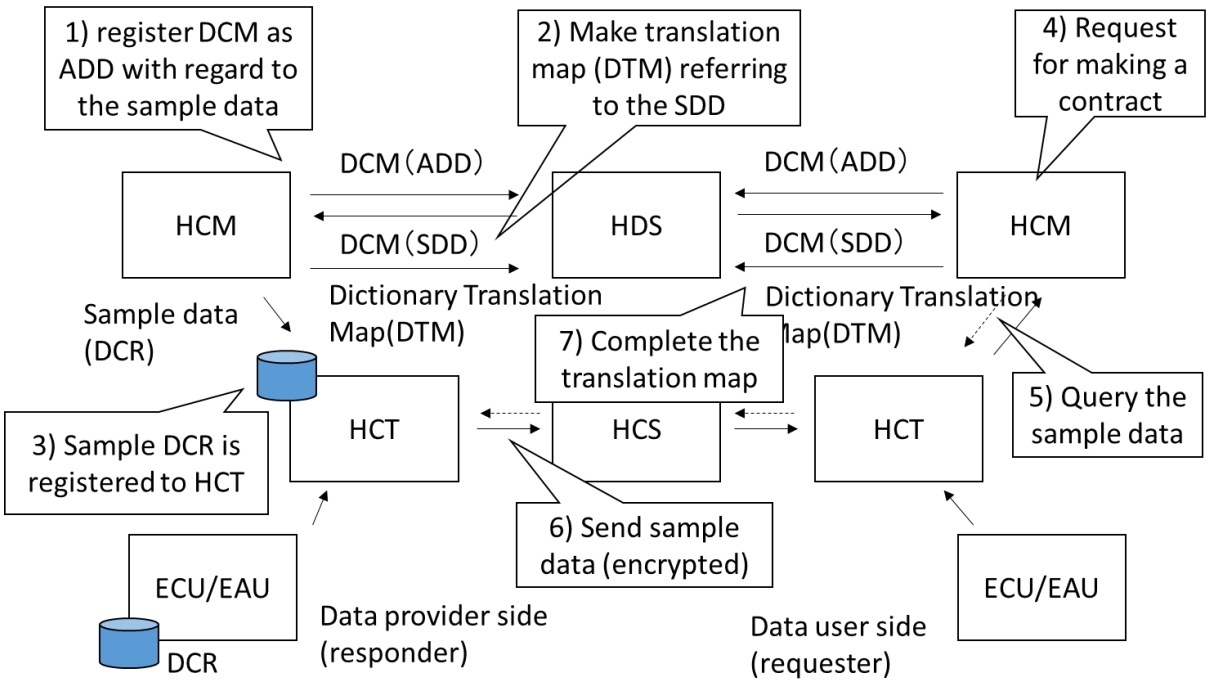

Figure 38 Send sample data

8.3 Trade Contract Setup Phase (PUSH type)

In the transaction contract phase, the flow of the PUSH type designating the data contents provided by the providing side is as follows.

Step 1 (Data provider’s processes)

In the push type, first, the data providing side sets available data and formats and services that can be provided.

1. Identify the DCM that can provide data with the EDU on the data provider side and the ECU.

2. Create conversion maps of specific data dictionary and actual data dictionary.

3. Register the created conversion map in the dictionary server

4. Register the target DCM as a profile together with the process category (PCC).

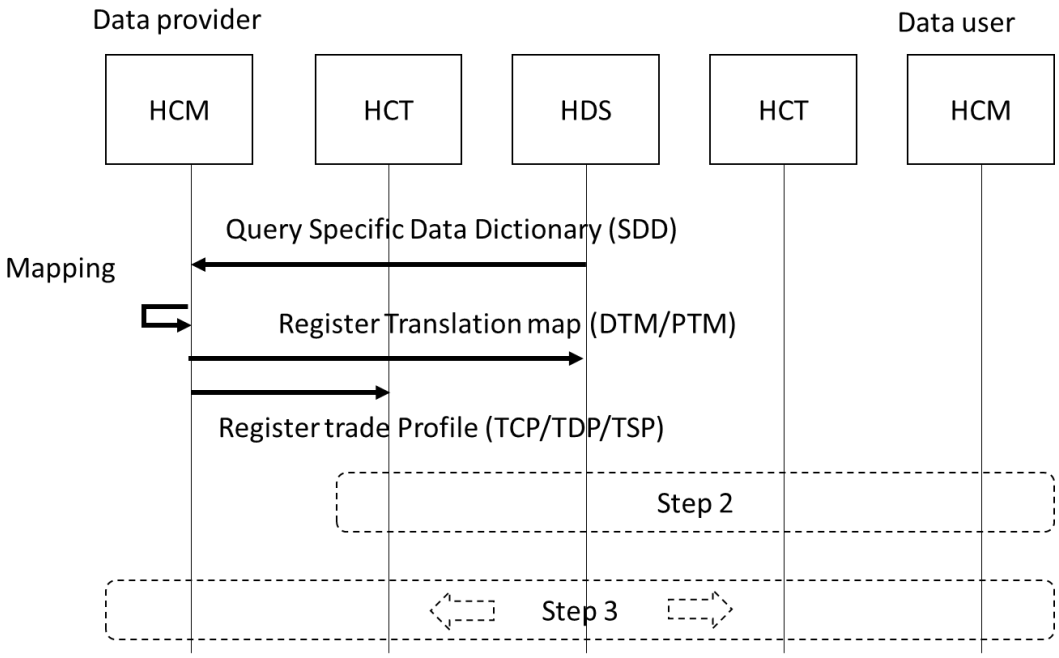

Figure 39 PUSH-type contract (Step 1)

Step 2 (Data user's processes)

The data user's side finds available data registered as a push type and applies for use to the provider.

1. Search available data (DCM) and process category (PCC).
2. Set request and acquire sample data from data provider.
3. Map available DCMs to actual data dictionary (ADD)
4. Register the conversion map and temporarily register the profile together

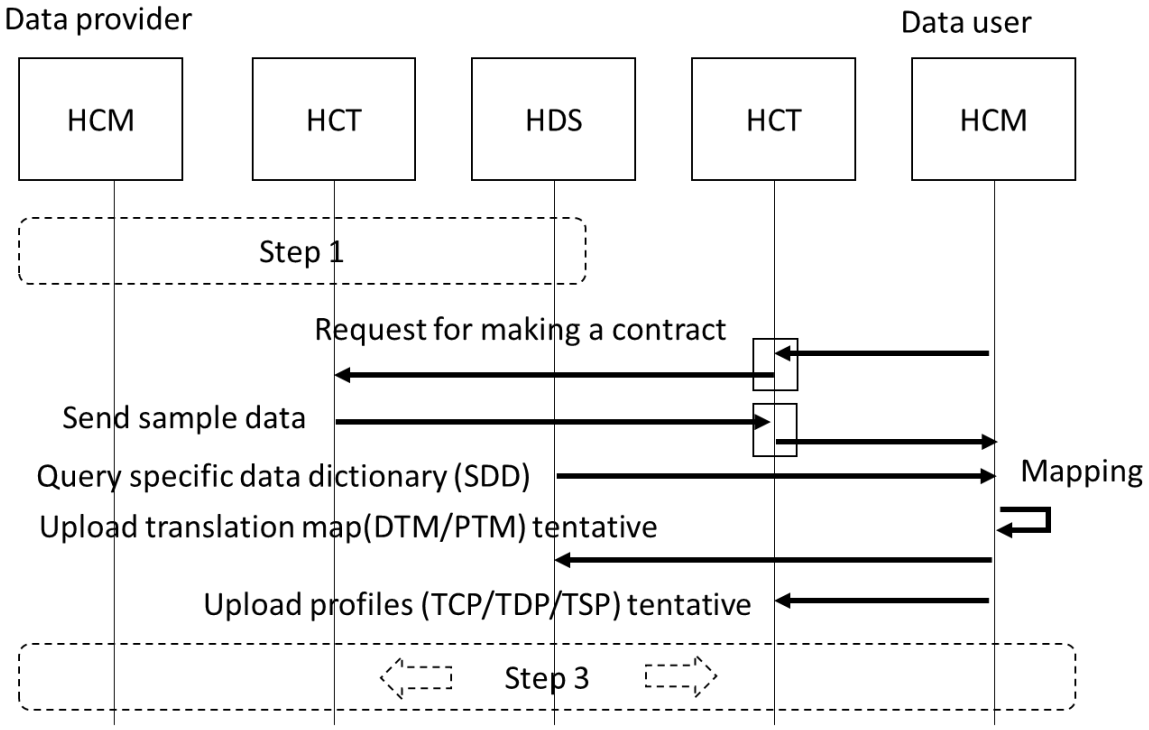

Figure 40 PUSH-type contract (Step 2)

Step 3 (Final confirmation processes)

Finally, the provider accepts the request from the use type, and completes the profile for the transaction.

1. Set usage conditions and send a transaction start request to the provider
2. The provider side judges the content and reply whether to start trading
3. On the data user side, confirm the conversion map and confirm the profile
4. Concurrently the data provider also confirms the profile.

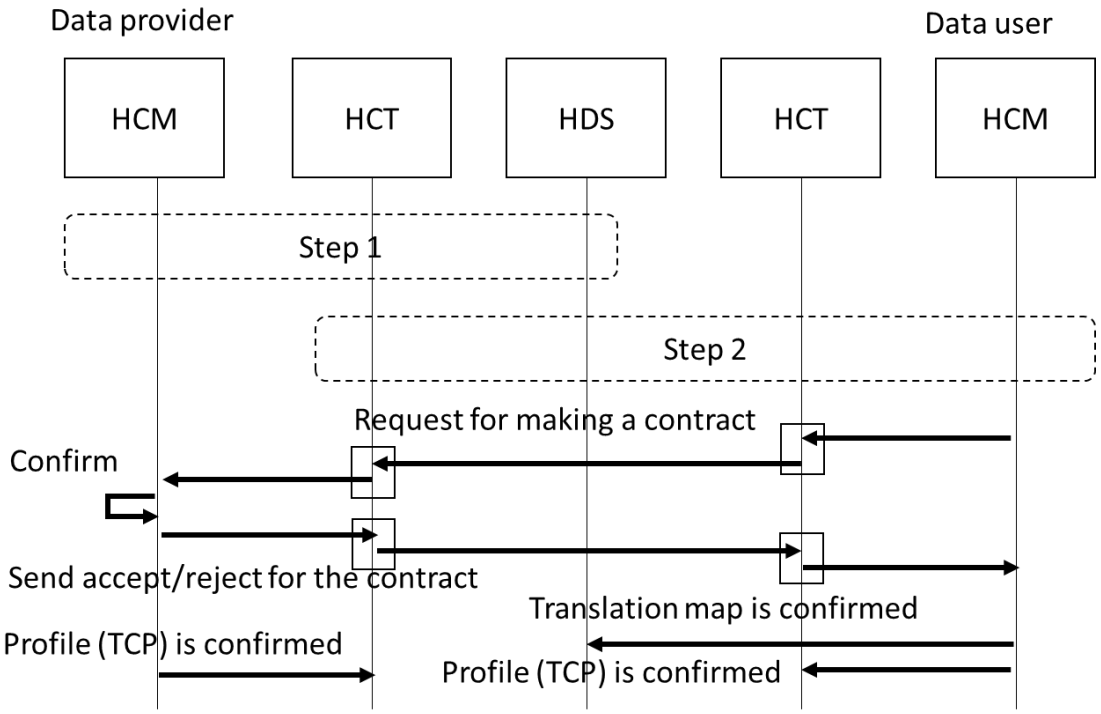

Figure 41 PUSH-type contract (Step 3)

8.4 Trade Contract Setup Phase (PULL type)

In the transaction contract phase, the flow of the PULL type that specifies the data contents the user wishes to use is as follows.

Step 1 (Data user's processes)

In the pull-type transaction contract use case, the data user side designates the data (DCM) and its properties (DPD) required and makes a request to the provider side. The flow is as follows.

1. Select the data you want to use from the actual data dictionary and search the specific data dictionary for the corresponding DCM.

2. Associate DCM with specific data dictionary and actual data dictionary and perform mapping.

3. Register the mapping result in the dictionary server as DTM / PTM.

4. Define and register request information as a profile

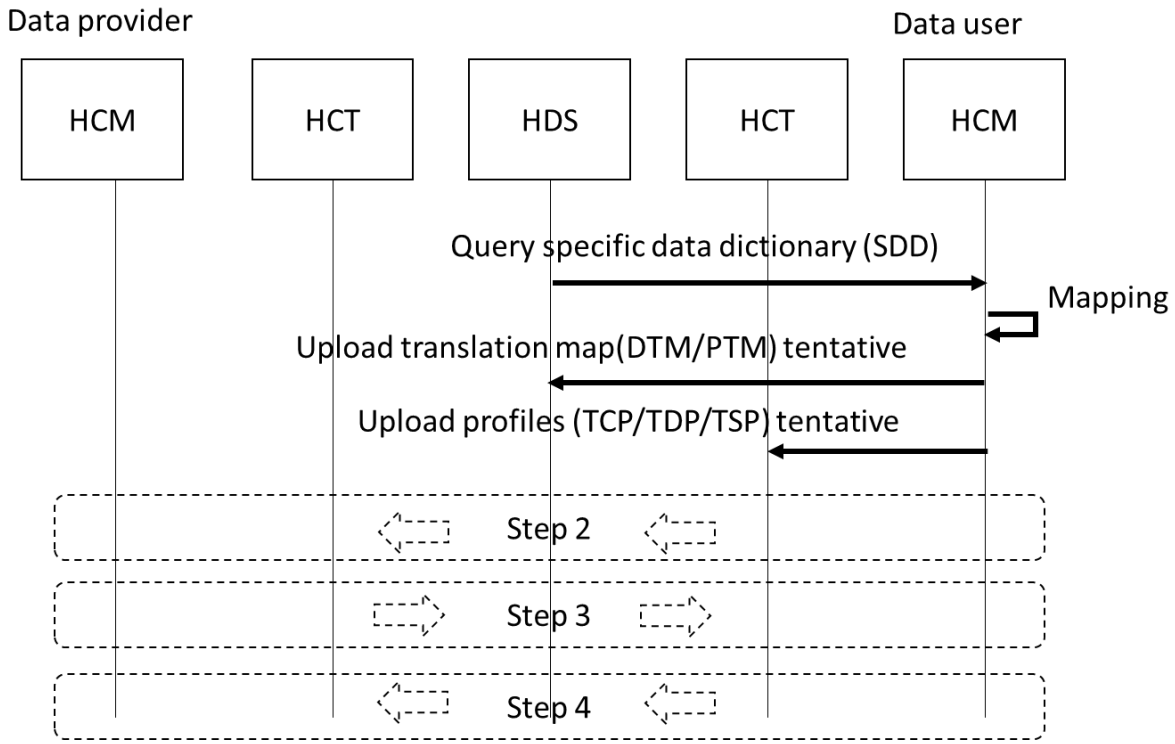

Figure 42: PULL-type contract (Step 1)

Step 2 (Data provider's processes)

When requesting with the contents set by the data user side and accepting it, temporarily register the profile with the following process.

1. Send a transaction start request from the data user side to the data provider side. At the time of request, add TSP (trade service profile).

2. The data provider who received the request introduces the specific data dictionary and performs mapping with the actual data dictionary.

3. The data provider decides whether or not data can be provided by the DCM using the actual data dictionary. If yes, provisionally registers a possible unit list as a conversion map.

4. Temporarily register the transaction profile.

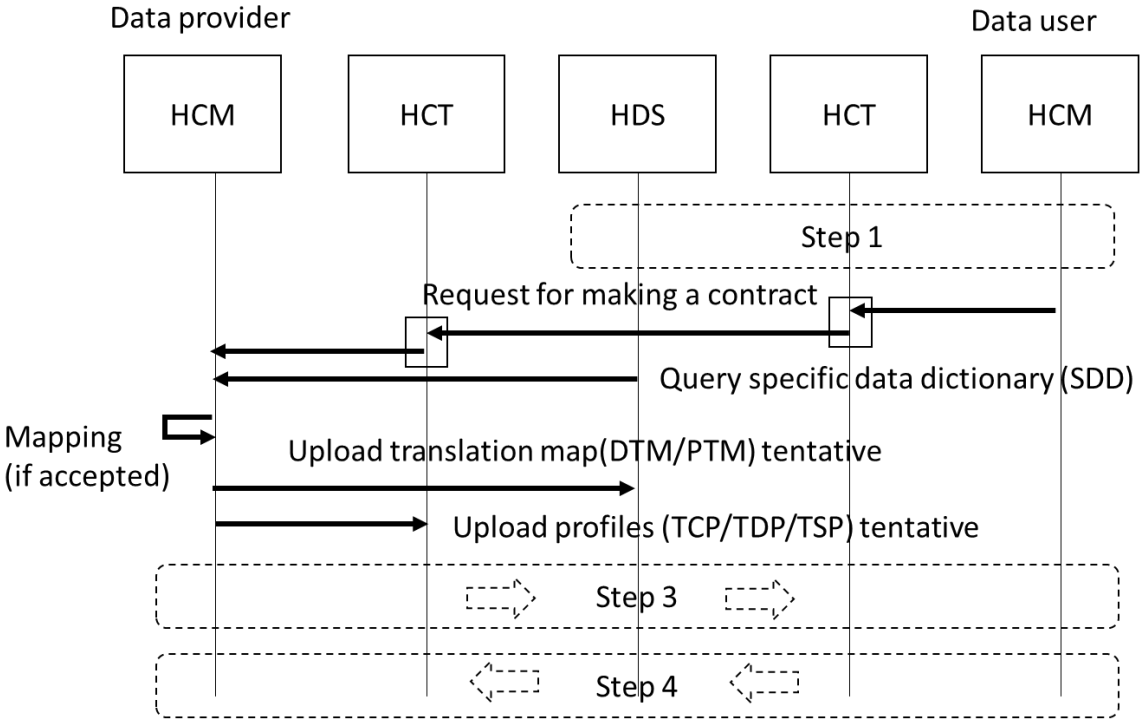

Figure 43: PULL-type contract (Step 2)

Step 3 (Data user's processes)

Upon acceptance of the request, the user side confirms the provisionally registered profile based on the content of the available data.

1. Send sample data to the user side together with acceptance notice.

2. The user side refers to the specific data dictionary and performs mapping with the actual data dictionary.

3. For the data user side, if the necessary item is not good, it is requested once again and it ends if it is not good.

4. If there is no problem, register the conversion map in the dictionary server.

5. Determine detailed transaction conditions and request parameter settings and register them in the profile.

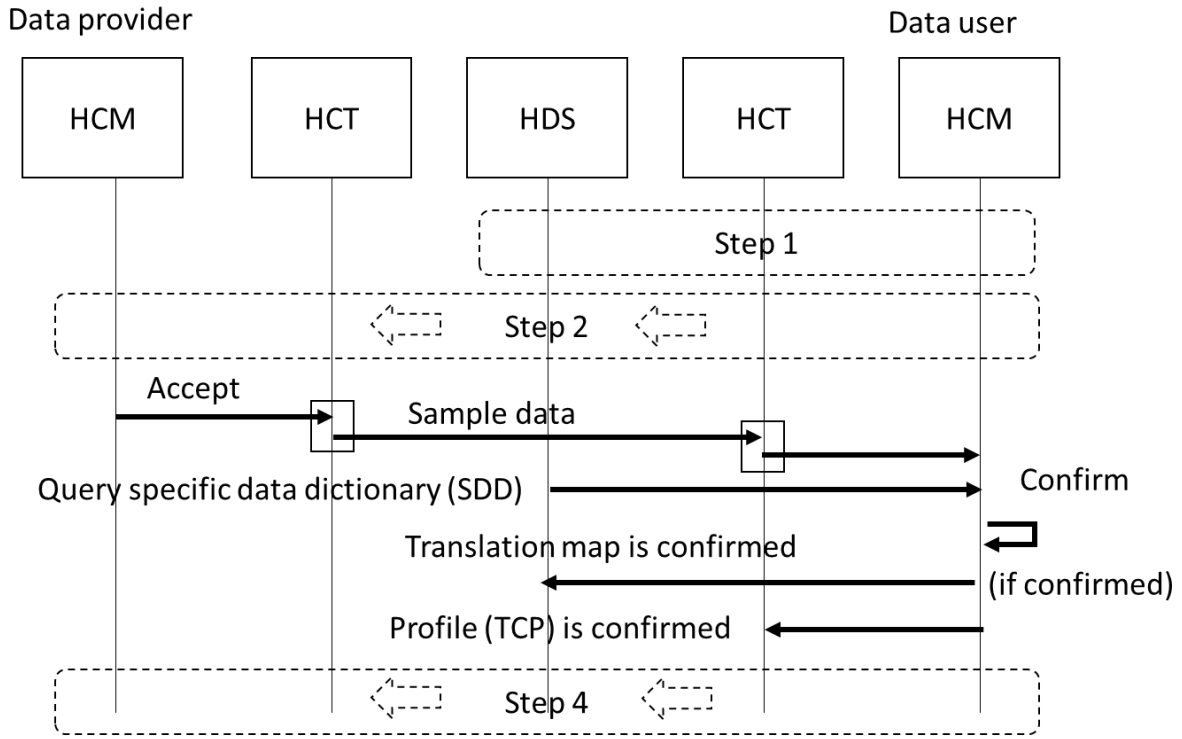

Figure 44: PULL-type contract (Step 3)

Step 4 (Final confirmation processes)

Upon receiving the final confirmation from the data user side, the data provider side also confirms the provisionally set profile.

1. Set the final transaction contract content and make final confirmation to the data provider side as to whether the transaction can be started.

2. The data provider confirms the content of the final transaction content, such as the charging method, and confirms the conversion map.

3. Finally switch the transaction profile from provisional setting to fixed.

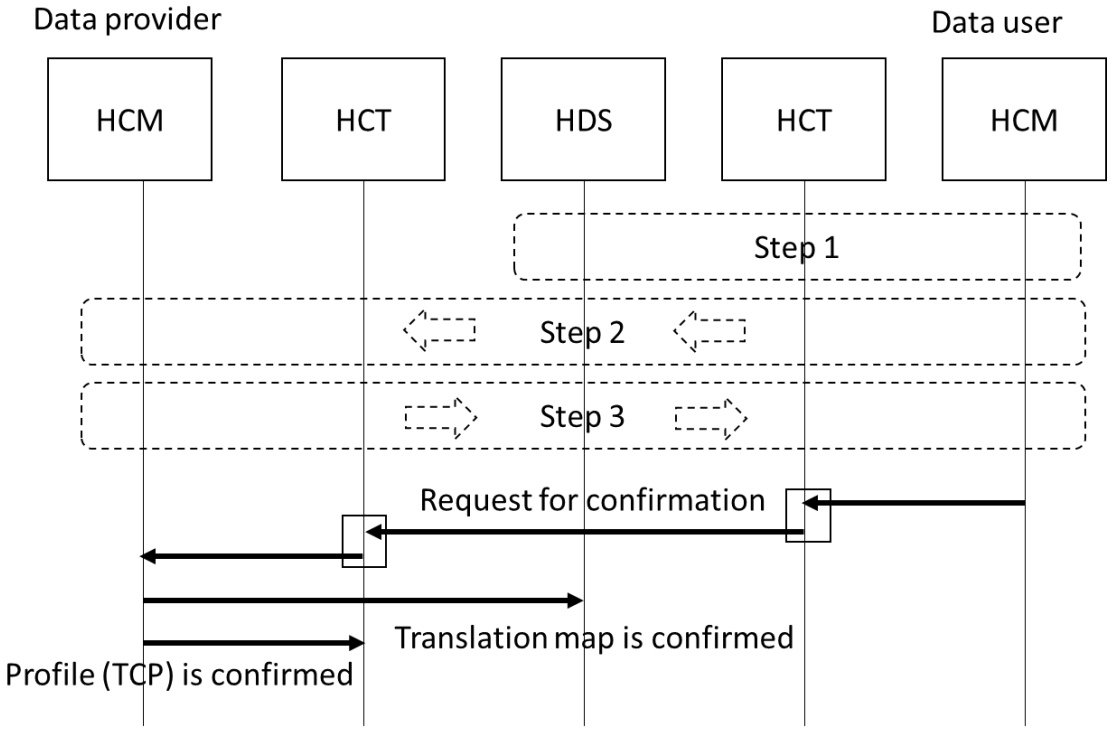

Figure 45: PULL-type contract (Step 4)

8.5 System execution and control phase

Figure. 46 shows the flow of cooperation among subsystems with respect to a PULL-type use case in the transaction execution phase. HCT (Cooperative Terminal) makes an inquiry to the HDS (Dictionary Server) for a dictionary and dictionary conversion map. It is the HCT that actually does the conversion.

Since the HCT is located on the edge side and is located inside the firewall, it always serves as a starting point for communication to the internet side. Therefore, for example, a delay due to asynchronism occurs between the HCS on the receiving side of the request information and the HCT.

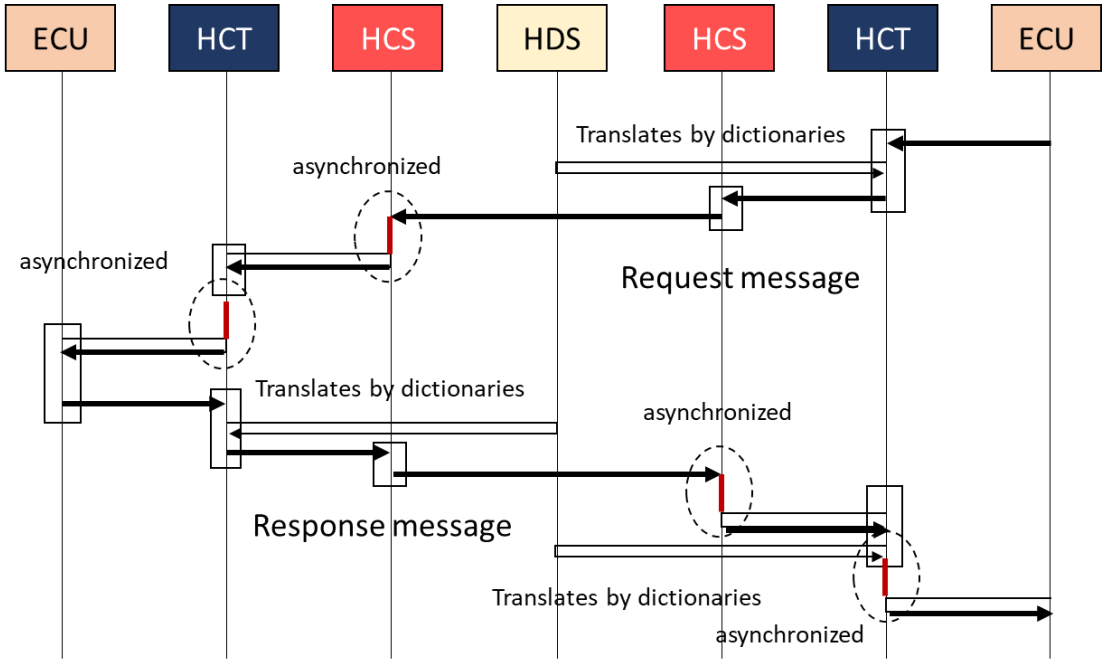

Figure 46: Data translation at execution and control phase (PULL use case)

Set Record data for linked DCMs

When the cooperating terminal (HCT) on the data-providing side combines multiple DCMs at the time of data transmission to make the configuration for the requested DCM, for each DCM other than the primary DCM, for each record of the primary DCM, it is necessary to make inquiries.

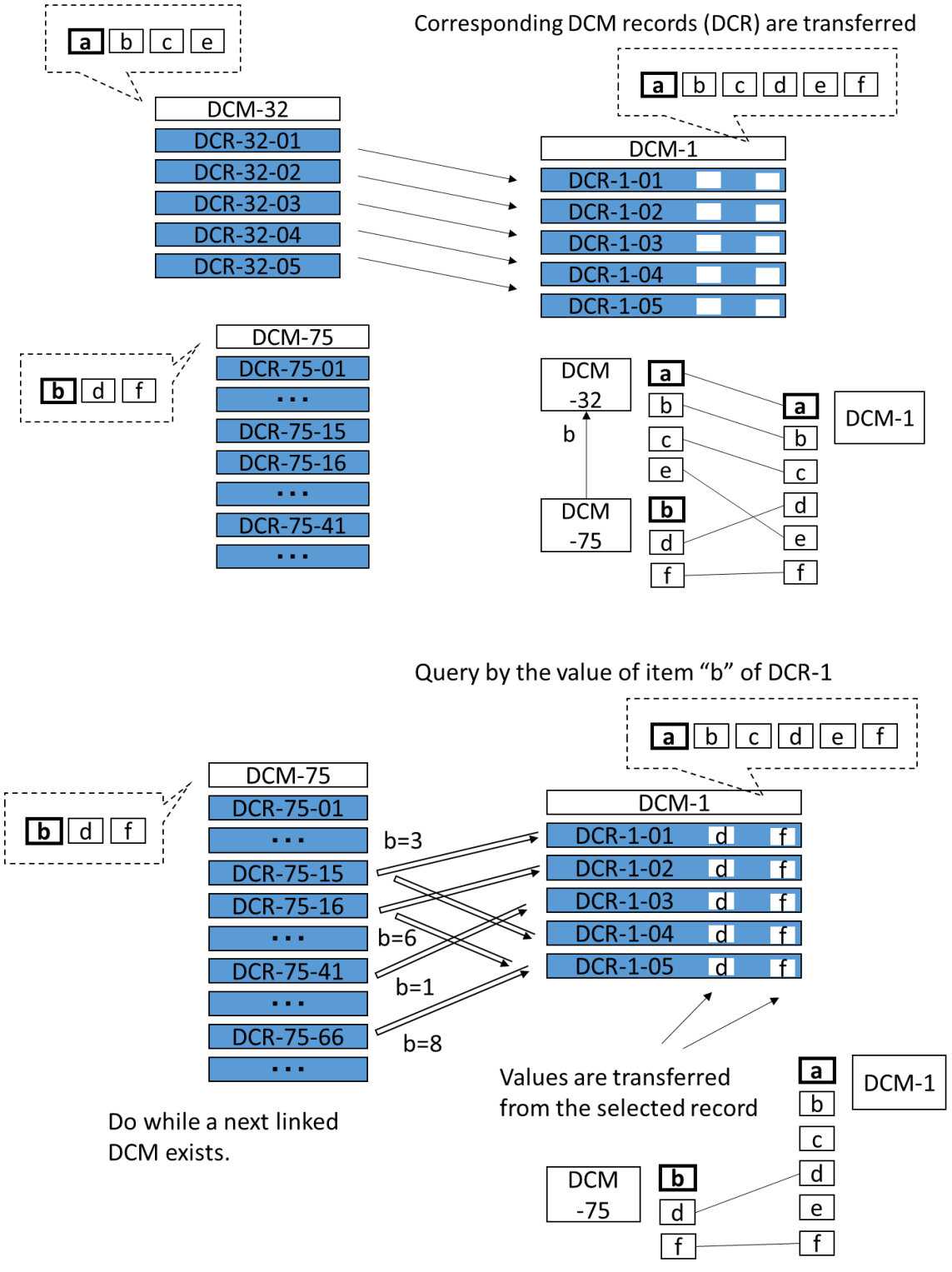

Figure 47: Link to multiple DCMs

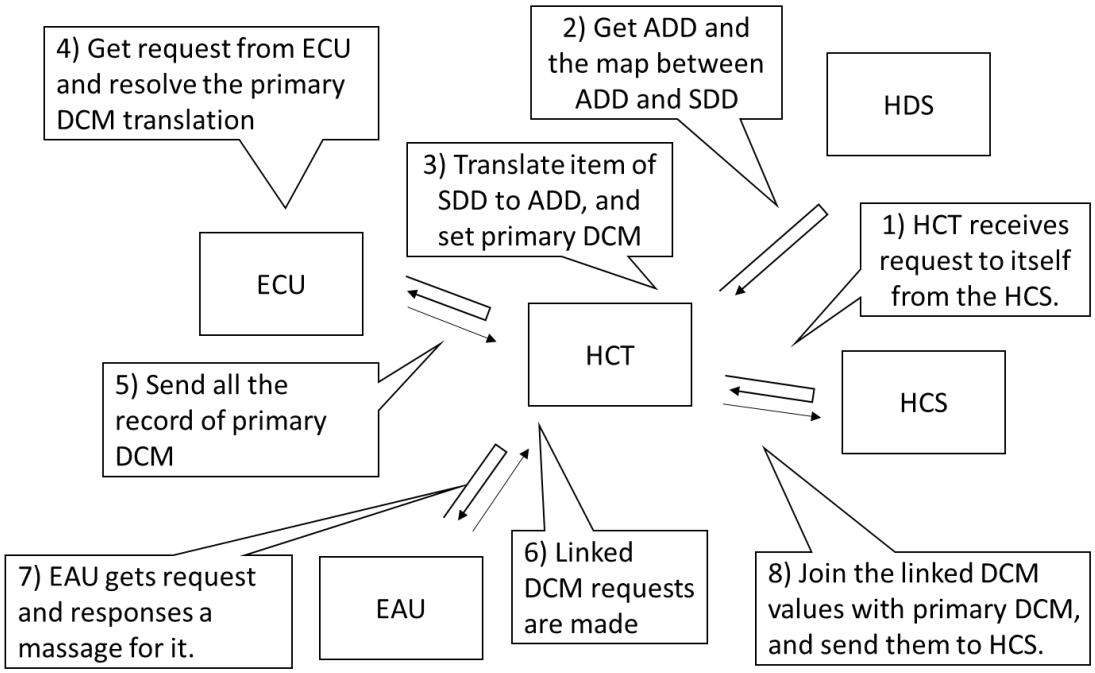

Figure 48: Process for a HCT request

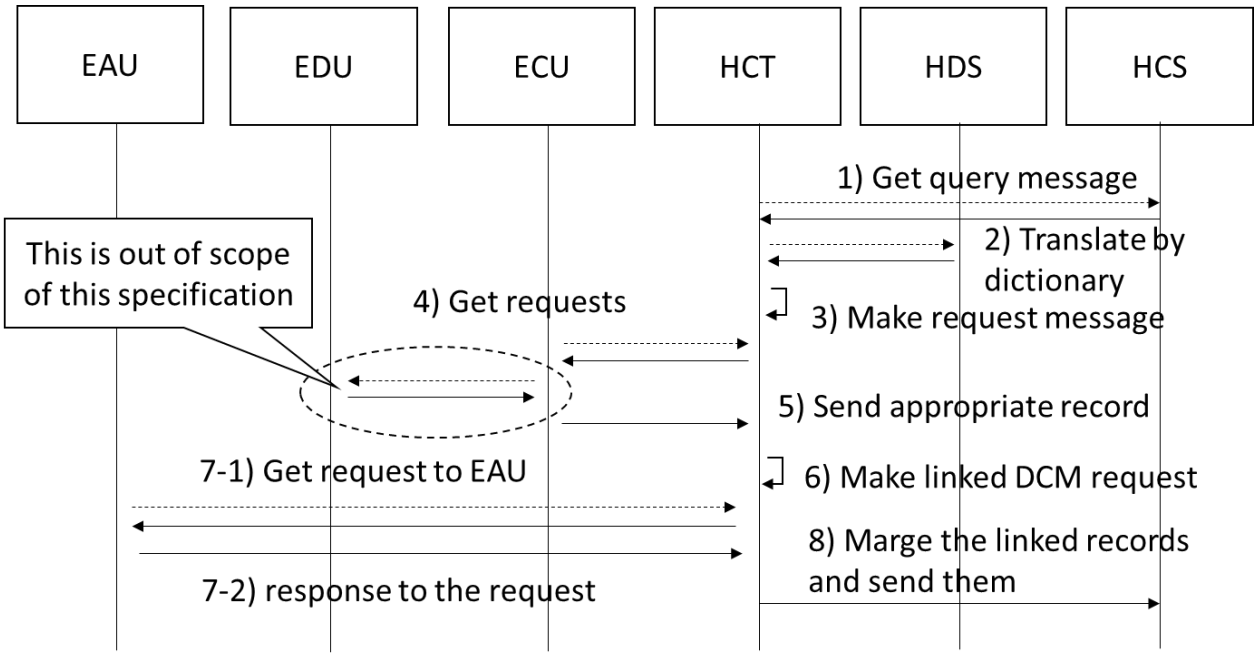

Figure 49: Query of the primary DCM (Step 4-5)

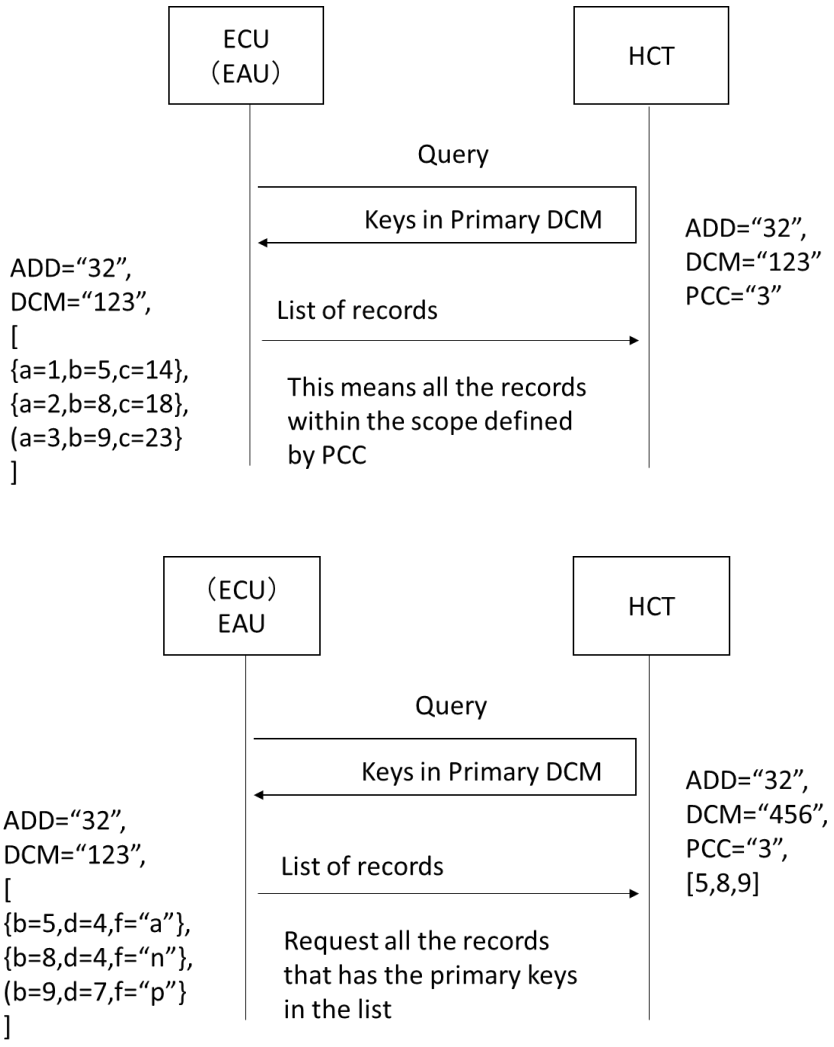

Figure 50: Linked DCM query (Step 7)

Some ECUs cannot perform queries for concatenated DCM (queries specifying a list of primary keys). In that case, the ECU corresponds only to the query of the main DCM, and preliminarily asserts that it is impossible to correspond to the query of the concatenated DCMs.

Procedure of profile usage

1. The data provider confirms the ID of the corresponding contract agreement profile when sending and receiving on receiving.

2. Retrieve the ID of the transaction data profile from the transaction contract profile ID. (Refer to HCT)

3. Acquire contents from actual data dictionary (ADD), specific data dictionary (SDD), and the ID of DCM before or after conversion belonging to each side.

4. Acquire the dictionary conversion map (DTM) based on the transaction data profile ID and acquire the item conversion map (PTM) belonging to the lower order.

8.6 Monitoring and Evaluation phase

Restriction items (transaction confirmation phase)

1. The ECU and the EAU shall report the service that the lower EDU herself or himself has managed as a PCE.

2. The ECU and the EAU shall report the statistical data of the event occurring on the user or the lower EDU to be managed.

3. The credit possessed by the business entity (TAP: Trade Account Party) having a trading account must correspond to PCE and ECR, or PCI, which are charged according to the transaction profile.

4. The historical data held by HCT must be block chained and stored in HCS in a dispersed manner to ensure that it has not been tampered

5. In the case that tampering is proved or the consistency cannot be confirmed, immediately invalidate the new transaction contract and transaction execution and notify the parties implementing the transaction.

For history management, refer to the following contents:

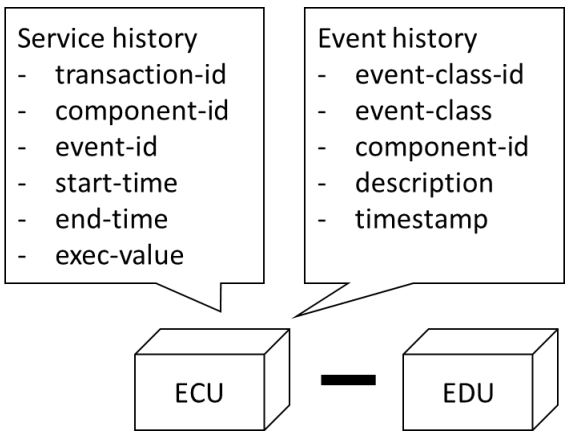

Figure 51: History data management