参考資料２

日米欧で作成中のドラフト

SC118DI07171

Guidelines for mapping core data models

Document Metadata

Property	Value
Date	2015-10-19
Status	For acceptance
Version	0.08
Authors	Alexandru Droscariu – PwC EU Services
Reviewed by	Nikolaos Loutas – PwC EU Services Susanne Wigard, European Commission
Approved by	

This document was prepared for the ISA Programme by:

PwC EU Services

Disclaimer:

Table of Contents

1 INTRODUCTION

The current document complements the guidelines for creating mappings between the Core Vocabularies and other data models initiatives, which are documented in the "Handbook for using the Core Vocabularies"[1] (henceforth referred to as "the handbook").

The objective is to support the effort of developing data model mappings. To do so, we provide a few examples of cases that are likely to happen when creating mappings, and which we encountered while mapping the Core Vocabularies with other data models, such as the Swedish Company data model, NIEM 3.0, KoSIT – XÖV, OSLO, and etc. We explain how we selected the mapping relations in these special cases. During the development of the mappings, we followed the method described in the handbook and we documented the mappings in a specially crafted spreadsheet. The mappings are published on the Core Data Model Mapping Directory (CDMMD)[2].

The current document is structured as follows. It first provides an introduction into the work of mapping data models explaining the needs behind. It continues with an overview of the structure of conceptual data models, the mapping relations, and the basic rules to apply the mapping relations. The document provides a description of the steps to apply mapping relations, and it closes providing several special cases from the experience of mapping the Core Vocabularies with the Swedish Company data model. We sprinkled the document with examples to highlight the reasoning behind applying the mapping relations.

[1] https://joinup.ec.europa.eu/site/core_vocabularies/Core_Vocabularies_user_handbook/Handbook-for-using-the-Core-Vocabularies_v0.50.pdf

[2] http://mapping.semic.eu

2 KEY FACTS ABOUT MAPPINGS

The handbook provides two reasons for creating mappings between data models:

- For documenting purposes, when the mappings at the conceptual level offer an entry point to domain experts to understand the mapped data models;
- For reconciliation of data sources, when mappings are used to extract basic information from conformant data models.

The syntax level mapping is a necessary input for the implementation of procedures for reconciliation of data sources. The conceptual level mapping supports the syntax level mapping. Therefore, conceptual level mapping is a prerequisite for syntax level mapping.

Mappings should be created between elements of two distinct data models only when the links are inherent in the meaning of the linked elements[3].

The mapping relations between the elements of two data models are independent of the syntax used to describe/represent the two data models.

Data reconciliation can be done without loss of meaning only between elements that have an exact match relationship. For the elements that have a broad or narrow relationship, the reconciliation can be done from the more specific element to the more generic element. Reconciling elements that have a close relationship involves the risk of losing meaning.

2.1 Mapping conceptual data models

Conceptual level mappings are developed between conceptual data models. The structure of a conceptual data model is composed of classes which describe real or abstract things such as buildings, people, organisations or services performed by a legal entity. Classes are composed of properties and/or associations. The difference between a property and an association is the type of values that may be assigned. A property (e.g. PersonFamilyName is a property of the Person class of the ISA Core Vocabularies) takes a singular value of a specific data type. An association (e.g. PersonAddress is an association of the Person class of the ISA Core Vocabularies) takes an instance of another class. The conceptual model does not specify how properties and associations are technically represented. For example, an association may refer to another object with some kind of identifier (e.g. a URI in RDF) or may embed the whole object (e.g. nesting in XML).

When developing conceptual level mappings, we refer to classes, properties and associations, generally, as elements. To decide the mapping relation to map two elements, consider the following aspects:

- Map classes to classes, properties to properties, and associations to associations.

[3] http://www.w3.org/TR/skos-reference/#mapping

- Consider the meaning of the two elements, regardless of:
    - The data structure or the properties used to describe the two elements.
    - The way each of the two elements are identified (e.g. “BE” VS “http://data.europa.eu/languages/BE” for the language code).
    - The way the two elements encode literals (e.g. “1990-06-09” VS “09/06/1990” for the birthdate).

2.2 Mapping relations

The mapping relations used for the conceptual level mappings are borrowed from SKOS[4]: exact, close, related, narrow, and broad match.

The handbook offers a comprehensive description of the mapping relations and their application. This chapter provides a summary explaining the meaning of the mapping relations, and how they should be applied.

The mapping relations are defined in terms of the set of subjects covered by the elements[5]. The figure below, extracted from the handbook, provides a graphical representation of the mapping relations.

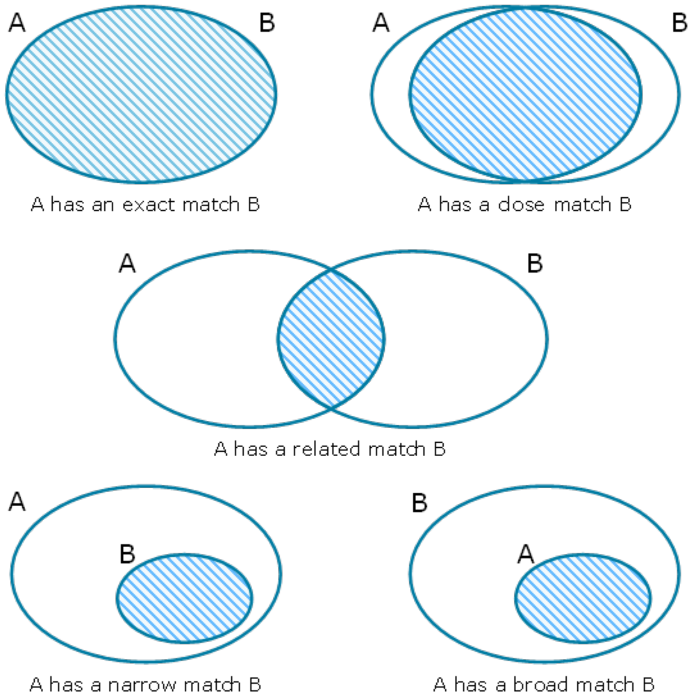

Figure 1 - Mapping relations: defined in terms of the set of subjects covered by the elements

The following examples will be further explained in chapter 3.

[4] http://www.w3.org/TR/skos-reference

[5] https://joinup.ec.europa.eu/site/core_vocabularies/Core_Vocabularies_user_handbook/Handbook-for-using-the-Core-Vocabularies_v0.50.pdf

2.2.1 Exact match

We say that “A has an exact match B” if the set of subjects of A is equal to the set of subjects of B. The definitions of A and B are equivalent.

Facts that are expressed for elements (classes, properties, or relationships) with an exact match relationship can be converted in both directions between data models without loss of meaning.

For example:

Core Vocabularies element	Mapping relation	Swedish Company data model
LegalEntity	Has exact match	**Juridisk person**

2.2.2 Close match

We say that “A has a close match B” if the set of subjects of A is mostly equal to the set of subjects of B. The number of subjects of A not included in B, and vice-versa, is negligible.

Elements with a close match relationship can be converted in both directions between data models with a minimal loss of meaning for some individuals.

For example:

Core Vocabularies element	Mapping relation	Swedish Company data model
AddressLocatorName	Has close match	**belägenhetsadress gårdsadressområde**
Description: The locator name is a proper noun applied to the real world entity identified by the locator. The locator name could be the name of the property or complex, of the building or part of the building, or it could be the name of a room inside a building. [INSPIRE] The key difference between a locator designator and a locator name is that the latter is a proper name and is unlikely to include digits.		Description: "Farm Address area" can be used to specify the location or to preserve historical names. "Farm Address area" should be used where there are names of farms, collections of buildings or a single house that is listed on a general map and where these names are used in the address today.

2.2.3 Related match

We say that “A has a related match B” if there is a meaningful intersection between the subjects of A and the subjects of B.

Elements with a related match relationship can only be converted with considerable error. Conversion is not advised. Such mappings can still be valuable to make semantic conflicts between data models better visible.

For example:

Core Vocabularies element	Mapping relation	Swedish Company data model
AgentPlaysRole	Has related match	**Relation till person**
Description: A public service in which the agent plays a role.		Relation till person (relationship to person = Role) Description: Relationships between persons are necessary to define the person. Here is for example information on who are the representatives of legal entities. It is also through this information that the address of a CEO can be linked to the limited company he/she represents. For example: a natural person authorized to sign for the legal entity.

2.2.4 Narrow match

We say that “A has a narrow match B” if the set of subjects of A is a superset of the set of subjects of B. The definition of A generalizes the definition of B.

Facts that are expressed for elements (classes, properties, or relationships) with a narrow/broad match relationship can only be converted into only the direction of the more general data model element.

For example:

Core Vocabularies element	Mapping relation	Swedish Company data model
LegalEntityLocation	Has narrow match	**Säte**
Description: Asserting the Location relationship implies only that the legal entity has some connection to a location in time or space. It does not		Description: The city in Sweden where the Board (of the legal entity) will meet and where the general meeting is to be held.

imply that the legal entity is necessarily at that location at the time when the assertion is made.		

2.2.5 Broad match

We say that "A has a broad match B" if B has a narrow match A.

For example:

Core Vocabularies element	Mapping relation	Swedish Company data model
LegalEntityRegisteredAddress	Has broad match	**Postadress**
Description: In almost all jurisdictions, legal entities must register a postal address. This may or may not be the actual address at which the legal entity does its business - it is commonly the address of their lawyer or accountant - but it is the address to which formal communications can be sent.		Description: The postal address of a Legal entity, natural person or workplace.

3 STEPS TO CREATE MAPPINGS

3.1 Conceptual level mappings

To formally define the mapping relations, we need to distinguish between an element and its subjects, i.e., the real-world things that are represented by the instances of the element.

An element of a data model is a class, a property, or an association. Classes are realised as instances (individuals) representing a real-world physical or conceptual thing. Classes are used to express facts about the generic characteristics of an individual instance. Properties or associations are used to construct facts about individuals.

To map data model A to data model B, one should proceed following the steps below:

For each element in A, find an element in B for which, potentially, there can be a relationship. Understand the definitions of the two elements to be mapped, and use your best intuition to find how the two sets of facts described by each of the two elements intersect:

1. If the two sets are equal, there is an **exact match** relation between the two elements;
2. If the two sets are not equal but they intersect,
    - 2.1. Analyse the hierarchy between the two sets, and decide if broad or narrow match can be used:
        - 2.1.1. If the set of facts described by the element in A includes the set of facts described by the element in B, then the element in A has a **narrow match** the element in B;
        - 2.1.2. Similarly, if the set of facts described by the element in A is included in the set of facts described by the element in B, then the element in A has a **broad match** the element in B;
    - 2.2. If it the set of facts described by the element in A does not include or is not included in the set of facts described by the element in B, decide if the intersection between the two sets is significant:
        - 2.2.1. If the intersection is significant, the element in A **has close match** the element in B;
        - 2.2.2. If the intersection is not significant, the element in A **has related match** the element in B;
3. If the two sets do not intersect, the element in A **has no match** in B.
4. Provide an explanation every time when there is a special case of matching between the two elements, to support the development of syntax mappings and the implementation of data reconciliation procedures.

The following chapters provide some example to showcase the steps above for each of the mapping relations.

3.1.1 Example for "Has exact match"

We choose the element **Address** (an address representing a location) from the Core Vocabularies and the element **Postadress** (contact option that unambiguously defines an actual area for mail delivery, usually linked with an indication of the addressee or recipient, and in some cases a particular address source) from the Swedish Company data model.

We then apply the mapping algorithm:

1. The two elements describe the same set of facts, and therefore **Address** *has exact match* **Postadress**.

3.1.2 Example for “Has close match”

We choose the element **AddressLocatorName** (the locator name is a proper noun applied to the real world entity identified by the locator. The locator name could be the name of the property or complex, of the building or part of the building, or it could be the name of a room inside a building. [INSPIRE] The key difference between a locator designator and a locator name is that the latter is a proper name and is unlikely to include digits.) in the Core Vocabularies and the element **belägenhetsadress gårdsadressområde** ("Farm Address area" can be used to specify the location or to preserve historical names. "Farm Address area" should be used where there are names of farms, collections of buildings or a single house that is listed on a general map and where these names are used in the address today) in the Swedish Company data model.

We then apply the mapping algorithm:

1. The two elements describe different concepts: **AddressLocatorName** could be the name of a building, room inside of a building, while **belägenhetsadress gårdsadressområde** describes farms or collections of buildings. Therefore, the set of facts described by the two elements are not equal.
2. The two elements describe similar facts: names of properties, buildings, collections of buildings. Therefore, *there is an intersection*.
   2.1. There is no hierarchy between the two sets of facts, as **AddressLocatorName** can be the name of a room inside of a building, while the **belägenhetsadress gårdsadressområde** refers only to the entire buildings.
   2.2. The two sets of facts described by **AddressLocatorName** and **belägenhetsadress gårdsadressområde** can be considered as significant. Therefore, **AddressLocatorName** *has close match* **belägenhetsadress gårdsadressområde**.

3.1.3 Example for “Has related match”

We choose the element **AgentPlaysRole** (a public service in which the agent plays a role.) from the Core Vocabularies and the element **Relation till person** (translated as relationship to person, or role, and described as: relationships between persons are necessary to define the person. The “Relation till person” provides for example information on who are the representatives of legal entities. It is also through this information that the address of a CEO can be linked to the limited company he/she represents. For example: a natural person authorized to sign for the legal entity) from the Swedish Company data model.

From the definitions of the two elements, we understand that (1) the **AgentPlaysRole** describes the relationship between *an agent* which can be either a natural person, or an organisation and *a public service* for which this agent plays a role, and (2) the **Relation till person** describes the relationship between two *persons* (natural persons or organisations).

We then apply the mapping algorithm:

1. It is obvious that the two elements describe sets of facts which are not equal.
2. There is an *intersection* between the two sets of facts, as both elements can describe roles played by persons.
   2.1. There is no hierarchy between the two sets of facts, as **AgentPlaysRole** describes a public service in which the agent (organisation or natural person) plays a role, while **Relation till person** refers to the relationship between two persons (physical or juridical).
   2.2. Decide if the intersection between the two sets of facts described by the two elements is significant:
      2.2.1. While the **AgentPlaysRole** describes a public role played by a person or an organisation the, **Relation till person** describes the relationship (both public or private) between two persons. Therefore, the intersection is not significant.
      2.2.2. Therefore, **AgentPlaysRole** *has related match* **Relation till person**.

3.1.4 Example for "Has narrow match"

We choose **LegalEntityLocation** (asserting the Location relationship implies only that the legal entity has some connection to a location in time or space. It does not imply that the legal entity is necessarily at that location at the time when the assertion is made) from the Core Vocabularies and **Säte** (the city in Sweden where the Board (of the legal entity) will meet and where the general meeting is to be held) from the Swedish Company data model.

We then apply the mapping algorithm:

1. Obviously the two sets of facts described by **LegalEntityLocation** and **Säte** are not equal.
2. Given the fact that **LegalEntityLocation** can describe cities, we can say that the two sets of facts intersect.
   2.1. While the **LegalEntityLocation** describes a location which can be a city, the **Säte** always describes a city as a location. Therefore, there is a hierarchy between the two elements.
      2.1.1. The set of facts described by **Säte** is always a set of cities where general meetings of companies are to be held. The set of facts described by **LegalEntityLocation** are locations related somehow with the legal entity. Therefore, the set of facts described by the **LegalEntityLocation** can include the set of facts described by **Säte** (a **Säte** is always a **LegalEntityLocation**, but not every **LegalEntityLocation** is a **Säte**). **LegalEntityLocation** *has narrow match* **Säte**.

3.1.5 Example for "Broad match"

We choose **LegalEntityRegisteredAddress** (in almost all jurisdictions, legal entities must register a postal address. This may or may not be the actual address at which the legal entity does its business - it is commonly the address of their lawyer or accountant - but it is the address to which formal communications can be sent) from the Core Vocabularies and **Postadress** (the postal address of a Legal entity, natural person or workplace) from the Swedish Company data model.

We then apply the mapping algorithm:

1. The set of facts described by **LegalEntityRegisteredAddress** and **Postadress** are not equal as **Postadress** can describe as well the address of natural persons or workplaces, while **LegalEntityRegisteredAddress** describes the postal address of legal entities.
2. Obviously, the set of facts described by the two concepts intersect.
   2.1. The set of facts described by the **LegalEntityRegisteredAddress** always describes the postal address of legal entities, while **Postadress** describes the postal address not only for legal entities, but also for natural persons. There is a hierarchy.
      2.1.1. The set of facts described by **LegalEntityRegisteredAddress** can't include the set of facts described by **Postadress**.
      2.1.2. The set of facts described by **LegalEntityRegisteredAddress** can be included in the set of facts described by **Postadress** (every **LegalEntityRegisteredAddress** is a **Postadress**, but not every **Postadress** is a **LegalEntityRegisteredAddress**). Therefore, **LegalEntityRegisteredAddress** *has broad match* **Postadress.**

3.2 Syntax level mappings

The syntax level mappings enable the exchange of information between IT systems that model semantically similar concepts, represented either with the same (e.g. XML), or with different data formats (e.g. XML and RDF). The representation of the syntax level mappings depends on the data formats of the source and target systems of the data transfer procedure. For example, XSL will be the choice to express syntax level mappings between a source representing its data as XML and a target representing its data as XML or RDF.

The existence of conceptual level mappings is a prerequisite to create syntax level mappings. When developing syntax level mappings, one should look not only at the mappings relations but at the descriptions provided together with the mapping relations. There can be information that further supports the development of syntax level mappings and the implementation of business logic to allow data manipulation at run time. For example, all the values of the attributes of the **Postadress** concept in the Swedish Company data model could be concatenated so that they can be transferred to the **Address.FullAddress** of the Core Location Vocabulary, despite the fact that at the concept level mapping the Address.FullAddress has no match within the Swedish Company data model. Another example is the transfer of data from **Identitetsbeteckning för person.id-beteckning** to **Person.PersonDateOfBirth** of the Core Person Vocabulary. It would be possible to develop a rule at the syntax level mapping which tests for the value of **id-beteckning typ** for an instance of **Identitetsbeteckning för person**, and when this value would be equal to *personnummer*, extract the first 6 to 8 characters from the value of

Identitetsbeteckning för person.id-beteckning and transfer the result into the **Person.PersonDateOfBirth**.

The following steps should be performed to develop syntax level mappings:

1. Choose the representation of the syntax mapping based on the representations of the of the source and destination data models;
2. Follow the mappings at the conceptual level and implement transformation rules which will receive as input the elements in the source data model and will produce as outputs the mapped elements in the destination data model. The transformation rules should implement the business logic described in the spreadsheet listing the mapping relations.

There are various tools[6] available for the development of transformations between many standard formats, including XML, RDF, JSON, etc.

XSLT[7] is a language to process and transform XML documents into other formats such as RDF, JSON, etc.

Openrefine[8] offers powerful functionalities for developing data reconciliation rules for multiple data formats.

The Core Data Model Mapping Directory (CDMMD)[9] publishes conceptual level mappings between ISA Core Vocabularies and other data models such as NIEM 3.0[10], Stelselcatalogus[11], KoSIT - XÖV[12], Swedish Company data model[13], etc.

3.3 Special cases

While creating mappings between elements of the ISA Core Vocabularies[14] and data models from national initiatives, a few special situations have been encountered that require a less intuitive approach for choosing the appropriate relationship. We noticed in several cases that a high degree of subjectivity can be introduced by the one who chooses the relationships. As higher the degree of subjectivity, as higher to potential for less consistent, core data models alignment.

---

[6] http://www.w3.org/wiki/ConverterToRdf

[7] http://www.w3.org/TR/xslt

[8] http://openrefine.org/

[9] http://mapping.semic.eu/

[10] https://www.niem.gov/technical/Pages/version-3.aspx

[11] www.stelselcatalogus.nl

[12] http://www.xoev.de/de/produkte

[13] http://uppgiftskrav.bolagsverket.se/gu

[14] https://joinup.ec.europa.eu/asset/core_vocabularies/description

The special cases which are described in the following tables, were encountered while mapping the Core Vocabularies with the Swedish Company data model[15]. They are however generic enough and it is very likely to come across them in other contexts as well.

In each of the special situations, together with the selected mapping relation, a description is provided to explain the reasoning behind the chosen solution. The mapping relations and the description should provide enough information to support the development of syntax level mappings, and the implementation of data reconciliation procedures.

Several special cases have been identified:

1. **Composite mapping.** This special case deals with elements from one data model (destination) for which a formula can be applied, and the result would match exactly an element in the source data model. For example, the period of time element in the Core Vocabularies matches exactly with the calculated difference between start date and end date from the Swedish Company data model.
2. **Partial mapping.** This case deals with elements of one data model (source) which can be composed by a combination of elements within the destination data model. For example, the full address element from the Core Vocabularies is a combination of multiple elements from the Swedish Company data model (more details, in the following chapters).
3. **Conditional mapping.** This special case deals with the elements of one data model (source) which match elements of the destination data model only when certain conditions are met. For example, the identifier of a legal organisation in the Core Vocabularies matches the identifier of a “person” in the Swedish Company data model, only for the instances of “person” which are legal organisations. The Swedish Company data model uses the “person” class to denote physical persons and juridical persons. The difference is made by the value of the property “type of person” which can take one of two values.
4. **Partial and conditional mapping.** This case is a combination of both situations described above. For example, the birth date in the Core Vocabularies matches exactly the first 6-8 digits extracted from the value of the identifier of a person in the Swedish Company data model, only when the type of person equals with “physical person”; otherwise, the type of person would identify a legal organisation.

The examples described in the next sections are structured as follows:

1. The column “Concepts of ISA Core Vocabularies”: lists the concepts which potentially match one or a combination of concepts of the Swedish Company data model;

[15] http://uppgiftskrav.bolagsverket.se/gu/ConceptModel/288841c9-624f-47bb-a007-9dbd21f2675b.htm

2. The column "Potentially match concepts of the Swedish Company data model": lists the concepts of the Swedish Company data model that potentially match the concepts of the ISA Core Vocabularies;
3. The column "Explanation": describes how the concepts could be matched, and what the chosen relation was. A similar description is made available in the spreadsheet documenting the mappings. This description should support the development of the syntax level mappings and the implementation of data reconciliation procedures;
4. The column "The selected mapping relation": lists the conceptual mapping relation that was finally chosen and documented in the spreadsheet.

3.3.1 Composite mapping

Composite mapping describes the context in which several elements in the destination data model can be combined according to a set of business rules (a formula can be applied) and the result matches exactly one element in the source data model.

The existence of such context, does imply that each distinct element in the destination data model will match another element in the source data model.

Concepts of ISA Core Vocabularies	Potentially match concepts of the Swedish Company data model	Explanation	The selected mapping relation
PeriodOfTime *Is defined as "An interval of time that is named or defined by its start and end dates".*	**Namn på fysisk person namn. namn giltigt från** *Defined as "name valid from"* **Namn på fysisk person namn. namn giltigt till** *Defined as "name valid to"*	The set of facts described by the **PeriodOfTime** potentially matches the set of fats described by the calculated difference between, generally, a start date and an end date. The start and end date are properties for several of the existing concepts in the Swedish Company data model such as **namn giltigt från** and **namn giltigt till**.	**PeriodOfTime** Has no match

3.3.2 Partial mapping

Partial mapping describes the context in which one element of a data model (source) matches (no necessarily exactly) a combination of several elements in the destination data model.

Concepts of ISA Core Vocabularies	Potentially match concepts of the Swedish Company data model	Explanation	The selected mapping relation
Address. AddressPOBox	**Postadress. utdelningsadress2**	A certain sequence of characters as part of the value of **utdelningsadress2** determines the post box value, according to SS 61 34 01. The set of facts described by the **AddressPOBox** and the set of facts described by the **utdelningsadress2** will never intersect.	**Address. AddressPOBox** Has no match
Address. AddressFullAddress	**Postadress**	The concept **AddressFullAdress** matches a combination of attributes of the **Postadress** (such as **postort, postnummer,** and **land)** concept. But the **AddressFullAddress** does not match close enough one single attribute of **Postadress**. Therefore, the set of facts described by the **AddressFullAddress** does not intersect with the set of facts described by any of the attributes of **Postadress**.	**Address. AddressFullAddress** Has no match

3.3.3 Conditional mapping

Conditional mapping describes the context in which the meaning of a certain element (let's call it EL1) in one data model depends on the value of another element (EL2) in the same data model. In such context, EL1 matches exactly one element in the mapped data model only when EL2 has certain values.

Concepts of ISA Core Vocabularies	Potentially match concepts of the Swedish Company data model	Explanation	The selected mapping relation
Person. PersonIdentifier *Is defined as "A formally-issued*	**Identitetsbeteckning för person** *Is defined as "a unique identifier that identifies a*	**PersonIdentifier** matches exactly **Identitetsbeteckning för person** *only if* **id-beteckning typ**, which is a property of the **Identitetsbeteckning för person** has the value	**PersonIdentifier** Has broad match **Identitetsbeteckning för person**

identifier for the person"	*particular person"*	*Personnummer*. Therefore, the set of facts described by the **PersonIdentifier** is equal to the set of facts described by the **Identitetsbeteckning för person** only in certain conditions. Thus, we can say that the concept **Identitetsbeteckning för person** is more generic than **PersonIdentifier**.	
LegalEntity. LegalEntityLegalIdentifier *Is defined as "The identifier given to the legal entity by the authority with which it is registered"*	**Identitetsbeteckning för person**	Similarly with the case above, the two concepts match, only if **id-beteckning typ** has one of the values *organisationsnummer*, *samordningsnummer*, or *GD-nummer*. Therefore, **Identitetsbeteckning för person** is a generalisation of **LegalEntityLegalIdentifier.**	**LegalEntityLegalIdentifier** Has broad match **Identitetsbeteckning för person**

3.3.4 Partial and conditional mapping

Partial and conditional mapping describes a context in which the meaning of a certain element (EL1) in one data model depends on the value of another element (EL2) in the same data model, but EL1 matches partially one element in the destination data model. EL1 matches partially one element in the destination data model, means that a business rule can be applied on the value of EL1 and the result matches the element in the destination data model.

Concepts of ISA Core Vocabularies	Potentially match concepts of the Swedish Company data model	Explanation	The selected mapping relation
Person. PersonDateOfBirth *Defined as "The date on which the person was born"*	**Identitetsbeteckning för person. id-beteckning**	**PersonDateOfBirth** matches **id-beteckning** only if **id-beteckning typ** has the value: *personnummer*. When **id-beteckning typ** has the value: *personnummer*, then the first 6 or 8 digits of the value of **id-beteckning** represent the date of birth, which will be encoded with the format: *yymmdd-nnnn*. Therefore, the set of facts described by the	**Person. PersonDateOfBirth** Has no match

		PersonDateOfBirth never intersect with the set of facts described by the **id-beteckning**.	

3.3.5 Mapping relations in special cases

In all the examples of special cases described in the previous chapters, we observed that the mapping relations do not always express the fact that by applying additional logic to the mapping relations, we could further describe that a transfer of data is possible between two IT systems implementing the two mapped data models. For example, in the case of composite matching, we conclude that the **PeriodOfTime** has no match in the Swedish company data model as the set of facts described by the **PeriodOfTime** does not intersect with a set of facts described by any concept in the Swedish Company data model. However, the difference between the existing properties **namn giltigt från** and **namn giltigt till** is an exact match of the **PeriodOfTime**. It could be interesting to express this kind of match. It could be interesting to further enrich the set of mapping relations so that logic could be used to express the relation between concepts.