2.5. Generating Necessary Normal Form Relationships from the OWL Refsets

The logic definitions are represented by the OWL axiom refset that is a replacement of the RF2 stated relationship file. As a result, the nature of the inferred relationship file in the distribution normal form (DNF) has changed, because the new DL features are not representable in the current relationships file. The inferred relationship file will maintain the same format and structure, but it is no longer equivalent to the stated form (containing all necessary and sufficient conditions). In fact, it is a collection of all the necessary conditions of precoordinated concepts and represents a subset of the full semantics.

Necessary Normal Form

The Necessary Normal Form (NNF) is a replacement for the Distribution Normal Form for inferred relationships. The NNF is a precalculated distribution form for practical purposes, for example, to support the continuity of existing implementations based on relational databases and queries by the expression constraint language.

The NNF consists of the full set of necessary relationships of precoordinated concepts after removal of redundant relationships within a given concept definition. Within the scope of a SNOMED CT terminology, necessary relationships are defined only for precoordinated concepts (aka OWL's named classes). Let C be a precoordinated concept and D be either a precoordinated concept or a complex expression. If an axiom is in the form of SubClassOf(C D) or EquivalentClasses(C D), then all of the derivable and necessary relationships of D are necessary relationships of C.

The NNF does not include class disjointness, transitive properties, reflexive properties and sufficient conditions represented as General Concept Inclusions (GCIs) in the OWL axiom refset.

Inferred relationships for concrete values, e.g. decimal, integer, string, or dateTime, should be included in a separate relationship file.

Rules for Determining Redundant Relationships

Rule 1 - Class and Role inclusions

Given two relationships, A and B, A with r = C and B with s = D, within the same role group, A is redundant if:

- r is the same as or a supertype of s, and
- C is the same as or a supertype of D

Note, "crossover relationships", where r is a supertype of s, and C is instead a subtype of D do not result in a redundant relationship.

Example for Class inclusion

Stated relationships



Inferred relationships before the removal of redundant relationships



Inferred relationships after the reduction



For | Fracture of radius|, the relationship | Finding site| = | Bone structure of radius and/or ulna| is inherited from | Fracture of forearm |, which is a redundant relationship because | Bone structure of radius is a subtype of | Bone structure of radius and/or ulna|. The relationship | Associated morphology| = | Traumatic abnormality| is inherited from | Injury of radius |, which is a redundant relationship because | Fracture (morphologic abnormality) | is a subtype of | Traumatic abnormality |.



referenced	
tld	owlExpression

Inferred Relationships in Necessary Normal Form

	(stated relationships)	sour celd	desti natio nId	relation shipGro up	type Id
125605004	EquivalentClasses(:125605004 ObjectIntersectionOf(:64572001 ObjectSomeValuesFrom(:609096000 ObjectIntersectionOf(ObjectSomeValuesFrom(:116676008 :72704001) ObjectSomeValuesFrom(:363698007 : 272673000)))))	125 605 004	2840 03005	0	116 680 003
		125 605 004	7270 4001	1	116 676 008
		125 605 004	2726 73000	1	363 698 007
12676007	EquivalentClasses(:12676007 ObjectIntersectionOf(:64572001 ObjectSomeValuesFrom(:609096000 ObjectIntersectionOf(ObjectSomeValuesFrom(:116676008 :72704001) ObjectSomeValuesFrom(:363698007 : 62413002)))))	126 760 07	6596 6004	0	116 680 003
		126 760 07	4293 53004	0	116 680 003
		126 760 07	7270 4001	1	116 676 008
		126 760 07	6241 3002	1	363 698 007
62413002	SubClassOf(:62413002 :299701004)	624 130 02	2997 01004	0	116 680 003

Example for Role inclusion

Stated relationships



Inferred relationships before the removal of redundant relationship



Inferred relationships after reduction



For concept | Kidney biopsy|, the relationship | Procedure site| = | Kidney structure| is inherited from | Procedure on kidney|, which is a redundant relationship to | Procedure site - Direct| = | Kidney structure| because | Procedure site - Direct| is a subtype of | Procedure site|. Because | Kidney structure| is a subtype of | Urinary system structure| and | Retroperitoneal compartment structure|, the inherited relationships for | Procedure site - Direct| are also redundant.

referenced Component Id owlExpression (stated relationshi		Inferred Relationships in Necessary Normal Form			
	owiExpression (stated relationships)	sour celd	desti natio nId	relation shipGro up	type Id
118851004	EquivalentClasses(:118851004 ObjectIntersectionOf(:71388002 ObjectSomeValuesFrom(:609096000 ObjectSomeValuesFrom(:363704007 :64033007))))	118 851 004	7138 8002	0	116 680 003
		118 851 004	6403 3007	1	363 704 007
7246002	EquivalentClasses(:7246002 ObjectIntersectionOf(:71388002 ObjectSomeValuesFrom(:609096000 ObjectIntersectionOf(ObjectSomeValuesFrom(:260686004 :129314006) ObjectSomeValuesFrom(:405813007 : 64033007)))))	724 6002	1188 51004	0	116 680 003
		724 6002	3629 95002	0	116 680 003
		724 6002	4302 12007	0	116 680 003
		724 6002	1293 14006	1	260 686 004
		724 6002	6403 3007	1	405 813 007
405813007	SubObjectPropertyOf(:405813007 :363704007)	405 813 007	3637 04007	0	116 680 003

Table 2.5-2: Example in OWL axiom refset and RF2 relationship file (NNF)

Rule 2 - Property chains including transitive properties

Given attribute r, s and t with a property chain SubObjectPropertyOf(ObjectPropertyChain(t s) r), and two relationships A and B, A with r = C and B with u = D, within the same role group, A is redundant if:

- Attribute u is the same as or a subtype of t, and
- D has a relationship to C via attribute s

Note the following:

- C does not need to subsume D
- Attribute t does not need to be the same as or a subtype of r
- Transitive properties are defined by a property chain in the form of

SubObjectPropertyOf(ObjectPropertyChain(r r) r) and thus it is a special case of the above.

Example for property chain:

Stated relationships of | Product containing precisely amoxicillin (as amoxicillin sodium) 1 gram/1 vial powder for conventional release solution for injection (clinical drug)| (the other model detail has been omitted):



Stated relationships of | Amoxicillin sodium (substance)| :



Inferred relationships before the removal of the redundant relationship



Inferred relationships after the reduction



For | Product containing precisely amoxicillin (as amoxicillin sodium) 1 gram/1 vial powder for conventional release solution for injection (clinical drug) , the relationship | Has active ingredient| = | Amoxicillin| is inherited from | Product containing amoxicillin|. If rule 1 for class inclusion was applied, th e relationships would not be considered redundant because | Amoxicillin sodium (substance)| is not a subconcept of | Amoxicillin (substance)|. Since | Amoxicillin sodium| | Is modification of| | Amoxicillin| and property chain of " | Has active ingredient| o | Is modification of| " is a subproperty of | Has active ingredient|, rule 2 actually compares the anonymous concepts for subsumption, i.e. | Has active ingredient| = | Amoxicillin| with | Has active ingredient| = | Amoxicillin sodium|. Therefore, the inherited relationship is redundant and can be removed from the NNF. Their relationships and property chain can be demonstrated in the following diagram.



Table 2.5-3: Example in OWL axiom refset and RF2 relationship file (NNF)

referencedCom	ferencedCom	Inferred Relationships in Necessary				
ponentId	mentId	Normal Form				
	(stated relationships)	source Id	destina tionId	relationshi pGroup	typeld	

27658006	EquivalentClasses(:27658006 ObjectIntersectionOf(:763158003 ObjectSomeValuesFrom(:609096000 ObjectSomeValuesFrom(:127489000 :372687004))))	27658 006	907040 04	0	11668 0003
		27658 006	372687 004	1	12748 9000
323739006	EquivalentClasses(:323739006 ObjectIntersectionOf(:763158003 ObjectSomeValuesFrom(:609096000 ObjectSomeValuesFrom(:762949000 :427483001))))	32373 9006	276580 06	0	11668 0003
		32373 9006	427483 001	1	76294 9000
127489000	SubObjectPropertyOf(ObjectPropertyChain(:127489000:738774007):127489000))	N/A	N/A		N/A
427483001	SubClassOf(:427483001 ObjectIntersectionOf(:440327007 ObjectSomeValuesFrom(:738774007 : 372687004)))	42748 3001	373298 001	0	11668 0003
		42748 3001	372687 004	0	73877 4007
		42748 3001	768681 000	0	72654 2003

Technical Implementation for Calculating the NNF

This fairly complex process uses the stated form and the output of the reasoner to calculate the necessary normal form which is represented in the relationship RF2 file.

The most straightforward way to produce the necessary normal form would be to use the Snomed OWL Toolkit or the Classification Service REST API which is language agnostic.

High Level Process

Classification

- 1. Read the Stated Form from RF2 files.
 - a. The following files are required: Concept, OWL Ontology Reference Set, OWL Axiom Reference Set.
 - b. If the edition has any active stated relationships then the Stated Relationship and MRCM Attribute Domain Reference Set files are also required.
- 2. Use the OWL API to infer the class hierarchy
 - a. Build the Ontology object using:
 - i. Axioms from the OWL Axiom Reference Set, making a note of any Transitive property and Property Chain axioms.
 ii. Axioms created by converting Stated Relationships to OWL Axioms using the MRCM Attribute Domain Reference Set for
 - list of attributes which should not be grouped in the given domain.
 - b. Use a reasoner to pre-compute the class hierarchy.

Necessary Normal Form Calculation

Calculating the necessary normal form happens in two passes of the hierarchy.org.snomed.otf.owltoolkit.normalform

- 1. Walk the class hierarchy in a top-down, breadth first, order.
 - a. For each class visited gather the stated attributes of this class and each inferred parent.
 - b. Compare the attributes and remove those which are found to be redundant because they are less specific in terms of depth in the hierarchy.
 - c. During this first pass build a hierarchy for property chains and transitive properties.
- 2. Walk the class hierarchy again in the same order reducing the attributes of each class further.
 - a. Compare the attributes and remove those which are found to be redundant because they are less specific in terms of depth in one of the alternate hierarchies.

For fine level detail the best source of information is the Java class org.snomed.otf.owltoolkit.normalform.RelationshipNormalFormGenerator which performs the Necessary Normal Form calculation.

Assignment for Role Group Number

It is important to clearly indicate if an attribute is grouped or not because | Role group (attribute)| has impact to semantics and classification results. | Role group| is represented by an integer in the field of relationshipGroup in the relationship file. In contrast, | Role group| is represented by 609096000 | Role group (attribute)| as an object property in the OWL axiom refset. After the stated relationship file is replaced by the OWL expression refset, role group numbers need to be generated for inferred relationships. The following rules should be followed in the inferred relationship file to provide consistent representation aligned with the concept model diagram and the OWL axiom refset.

- 1. All 116680003 | Is a relationships should be assigned in role group 0;
- 2. Attribute that is not grouped, not a value of | Role group (attribute)| or grouped=0 in MRCM, should be assigned in role group 0;
- 3. Attribute that is grouped, value of | Role group (attribute)| or grouped=1 in MRCM, should be assigned a role group number that is not 0. Each | Role group (attribute)| in the OWL axiom should be presented by a unique role group number. Note, role group merging is not covered here.

609096000 | Role group (attribute)| is explicitly represented for self-grouped attributes where there is only a single attribute in a role group in an OWL axiom. However, these self-grouped attributes and values are not explicitly represented in the current relationship files. This representation has caused confusion if an attribute in role group 0 is grouped or not. The following example demonstrates the changes to assignment of role group number after the implementation of the complete OWL axiom refset.

An example for the current diagram representation for attribute in role group 0 in the stated relationship file and concept model diagram

sourceld	destinationId	relationshipGroup	typeld
90708001	64033007	0	363698007



After the complete OWL axiom refset is implemented, | Role group| in the OWL axiom refset and concept model diagram should be represented as following.



Representation of | Role group in the NNF relationship file and concept model diagram

sourceld	destinationId	relationshipGroup	typeld
90708001	734045002	0	116680003
90708001	443820000	0	116680003

