nehta

SNOMED CT-AU Release 20131130
Australian Implementation Guidance

1 November 2013

Approved for external information

Document ID: NEHTA-1426:2013

National E-Health Transition Authority

National E-Health Transition Authority Ltd
Level 25
56 Pitt Street
Sydney, NSW, 2000
Australia
www.nehta.gov.au

Document information

Key information

Owner	National Clinical Terminology and Information Service, NEHTA
Date of next review	14 April 2014
Contact for enquiries	NEHTA Help Centre t: 1300 901 001 e: help@nehta.gov.au

Quality reviews

dv #	Reviewer(s)	Type of review	Purpose
002	Technical writer	Editorial	Reformatted document to align with current NEHTA standards; performed basic editorial review.
003	Terminology Analyst	Content	Updated content

Product version history

Product version	Date	Release comments
20080829	29 Aug 2008	Initial Release
20090416	16 Apr 2009	Revised to include updated definitions and scenarios for implementation.
20091130	30 Nov 2009	Updated for SNOMED CT-AU Release 1.0
20100531	31 May 2010	Updated for SNOMED CT-AU May 2010 Release
20100731	31 Jul 2010	Publication of Emergency department reference sets
20101130	30 Nov 2010	Updated for SNOMED CT-AU November 2010 Release
20110531	31 May 2011	Updated for SNOMED CT-AU May 2011 Release
20120531	31 May 2012	Updated for SNOMED CT-AU May 2012 Release
20121130	30 Nov 2012	Updated for SNOMED CT-AU November 2012 Release
20130531	31 May 2013	Updated for SNOMED CT-AU May 2013 Release

Table of contents

1 Introduction

1.1 Purpose

This document provides implementation guidance, for software developers, technical consumers and the general SNOMED CT®[1]-AU community of practice. It is intended to provide practical guidance for Australian implementations and is to be used in conjunction with the IHTSDO's *SNOMED CT Technical Implementation Guide* [1].

The document will be updated and revised based on the feedback from the community of use, such that it meets its objective of providing practical advice for implementers.

1.2 Intended audience

This document is intended for those in the SNOMED CT-AU community of practice who have a solid understanding of SNOMED CT, particularly for Technical Healthcare Vendors. This includes developers and testers who are responsible for producing, assuring or maintaining products that integrate with SNOMED CT-AU. The document assumes the reader has a basic understanding of software development, and database management.

1.3 Scope

This document is limited to providing detailed implementation information for Australian implementers that is not addressed within *SNOMED CT Technical Implementation Guide* [1]. It is limited to information about how to implement, rather than information on the development of release artefacts/products. For example the Reference Set Library and Development Approach for Reference Sets documents describe products that are part of the release whereas this document is intended for guidance on the options to implement certain aspects of SNOMED CT-AU.

It does not cover content already described in the *SNOMED CT Technical Implementation Guide* [1]. Where possible, similar chapters will be cross-referenced from within this document. All code samples provided are for demonstration purposes only, and do not represent the most efficient or robust approach. Implementers are advised to conduct their own performance tuning and ensure appropriate exception handling.

1.4 SQL examples

Where possible examples of SQL codes are provided to assist in demonstration of how features work. The schema for these code samples are aligned with that used in the *SNOMED CT-AU Reference Set Toolkit Guide* [2]. If the Reference Set Implementation Toolkit has been set up, most of the queries provided may be run, as provided, on the same database.

---

[1] This material includes SNOMED Clinical Terms® (SNOMED CT®) which is used by the permission of the International Health Terminology Standards Development Organisation (IHTSDO®). All rights reserved. SNOMED CT was originally created by The College of American Pathologists. IHTSDO®, SNOMED® and SNOMED CT® are registered trademarks of the IHTSDO.

It should also be noted that the SNOMED CT-AU Reference Set Implementation Toolkit uses a Snapshot release, which contains active and inactive content. However, most queries in this document don't differentiate between active and inactive content for brevity. Developers who wish to exclude inactive components from their queries should generally apply the 'active=1' clause to each table.

2 Australian dialect reference set

The *Australian dialect reference set* (ADRS) is a Language reference set. Language reference sets are used to express local language preferences and dialectic differences across descriptions from a parent language; for example British English preferences versus United States English preferences.

RF2 only provides two types of descriptions:

- Fully Specified Names – the true meaning of the concept.
- Synonyms – other names for a concept useful in a variety of settings.

Language reference sets in RF2 annotate RF2 Synonym descriptions with one of three mutually exclusive values to provide localisation:

Preferred	Indicates that Synonyms annotated with this value are the preferred way of describing the concept (also known as the 'Preferred Term'). A Language reference set is required to have exactly one active 'preferred' Synonym for each concept.
Acceptable	Indicates that Synonyms annotated with this value are not the preferred way of describing a concept, however are acceptable in the Language reference set's context. A Language reference set may refer to zero or more of a concept's descriptions with this value.
Not Acceptable	Indicates that the Synonym is not an acceptable way to describe the concept in this Language reference set's context. Language reference sets do not explicitly reference Synonyms with this value, rather "Not Acceptable" is implied by the absence of a reference to Synonyms that are unacceptable.

2.1 Technical summary

The ADRS is a component reference set as described in Section 5.6.2.8[2] of the *SNOMED CT Technical Implementation Guide* [1]. As with all reference sets the referencedComponentId is the component being referenced. For language reference sets this field contains an id from the descriptions file. The last field, acceptabilityId[3], indicates the preference with the current range of values being:

- 900000000000548007 |*Preferred*|
- 900000000000549004 |*Acceptable*|

Using both the descriptions file and ADRS it is possible to determine which descriptions are considered preferred. Each active concept has exactly one preferred description.

[2] http://www.ihtsdo.org/fileadmin/user_upload/doc/en_us/tig.html?t=trg2rfs_spec_lang
[3] http://www.ihtsdo.org/fileadmin/user_upload/doc/en_gb/eg.html?t=glrfn_f2_AcceptabilityId

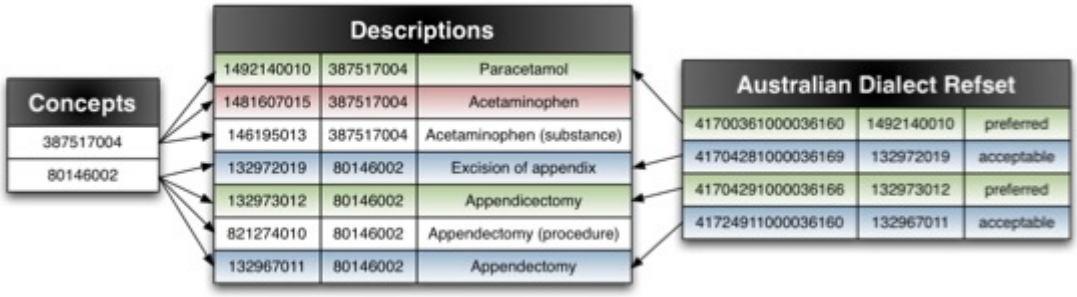

Figure 1: Relationship between the Concepts, Descriptions and Language Reference set

Figure 1 above shows an example taken from SNOMED CT-AU, showing how the *Australian dialect reference set* expresses that:

- 'Paracetamol' is preferred (green) and 'Acetaminophen' is not acceptable (red).
- 'Appendicectomy' is preferred (green), while 'Appendectomy' is acceptable (blue), as is 'Excision of appendix'.

That is, the Language reference set can be joined with the content of the Descriptions file to determine the preferred and acceptable Synonyms for a concept or concepts.

Note: Fully Specified Names are not referenced – they are always acceptable and preference is irrelevant.

Note: Columns have been omitted from the example for brevity.

2.1.1 SQL examples

Consider the concept 387517004. All available descriptions can be retrieved using the following query:

```
select descriptions.conceptId,descriptions.id,term
from descriptions
where descriptions.conceptId = 387517004
```

The descriptions that are referenced in the ADRS can be identified by extending the query as:

```
select descriptions.conceptId,descriptions.id,term, valueid
from descriptions
inner join description_refset AS ADRS
on descriptions.id = ADRS.referenceddescriptionid
where descriptions.conceptId = 387517004
```

And ultimately, the current preferred term for the concept 387517004 can be retrieved with:

```
select descriptions.conceptId,descriptions.id,term, valueid
from descriptions
inner join description_refset AS ADRS
on descriptions.id = ADRS.referenceddescriptionid
where descriptions.conceptId = 387517004
and ADRS.valueid = 900000000000548007 -- ConceptId for 'Preferred'
and ADRS.active = 1
```

Note: It is necessary to apply the “ADRS.active = 1” criterion to identify the current preferred term, as this can change over time.

To improve readability, developers may prefer to create a custom function or method that accepts a conceptId as a parameter and returns the relevant preferred term. For example:

```
delimiter //
DROP FUNCTION IF EXISTS get_ADRS_PreferredTerm
//
CREATE FUNCTION get_ADRS_PreferredTerm(candidate bigint(20)) RETURNS
varchar(300)

BEGIN
   RETURN (select term
                from descriptions
                inner join description_refset AS ADRS
                on descriptions.id = ADRS.referenceddescriptionid
                where descriptions.conceptId = candidate
                and ADRS.valueid = 900000000000548007
                and ADRS.active = 1);
END
//
```

This function can then be called whenever a preferred term is required.

```
select get_ADRS_PreferredTerm(64459004);
```

2.2 Implementation suggestions

2.2.1 Performance

Retrieving the preferred terms for concepts is likely to be the most frequently used terminology process within any implementation. Some performance issues may be observed if the nested query illustrated above is repeatedly called. To mitigate this, implementers may choose to store denormalised data structures. For the ADRS this involves creating a precomputed table of concept IDs and their associated Preferred descriptions.

Alternatively, index optimisation alone may provide sufficient performance gains.

3 Term searching

Searching for appropriate SNOMED CT-AU concepts by entering some text and selecting a concept from search results is the most common method for giving end users access to the broad range of content in SNOMED CT-AU. To ensure a positive user experience, it is important to provide an effective search capacity. Section 7.6.1 of the *SNOMED CT Technical Implementation Guide* [1] provides some options for producing effective searches. This guide provides some additional approaches.

3.1 Indexing

Basic indexing of the term column in the descriptions file has limitations. Most users will expect to be able to retrieve search results without typing in the whole phrase they are looking for, which may even mean partial words. Such search criteria may require the use of wild cards.

For example: A user searching for “fracture”, may just type in “frac”.

Using the query:

```
select term from descriptions
where term like 'frac%';
```

produces the expected results and with acceptable performance. However, such “starts with” searches will not identify terms where ‘frac’ appears anywhere but the start of the description. It is possible to prefix the search string with a wild card such that:

```
select term from descriptions
where term like '%frac%';
```

This query will yield results that contain ‘frac’ anywhere within the string, however since basic SQL indexing typically relies on the characters at the front of a string – the index is useless and performance will suffer.

There are a number of alternatives that are both simple to use and provide good results. These include:

Native Full Text Indexing: Most relational database management systems now provide native Full Text indexing. The features and syntax vary between systems, so the relevant RDBMS user documentation should be consulted. Such indexes require little extra setup and are easy to use.

Specialised Software Library*:* There are a number of software libraries available that can also be used to produce powerful indexes and searching capability. Configuration will depend on the library chosen. Lucene[4] is an example of such a library, and versions are available for most major development languages.

[4] http://lucene.apache.org/

3.2 Restrict to scope to reference sets

An unrestricted search upon all of SNOMED CT-AU will yield what appear to be duplicates to an end user and risk of selecting a concept that is inappropriate for the context.

For example, a search for “ulcer” might retrieve the following concepts:

- 56208002 |*Ulcer*|
- 429040005 |*Ulcer*|

Each of these concepts represents a different idea. The first refers to a morphological abnormality, that is, an ulcer. The second is the actual disorder as might be recorded in a clinical encounter. Refer to the *SNOMED CT User Guide* [3] for further details.

One way to restrict the scope is limit the search to concepts from a reference set of appropriate scope. For example, if searching for a value for a diagnosis field - only *Clinical finding* concepts might be appropriate, and the *Clinical finding foundation reference set* may be applied. Similarly, certain data elements in NCTIS information models have value domains that are restricted to particular reference sets. Implementers must ensure that users only populate these fields using the relevant concepts.

3.2.1 SQL examples

Consider a basic search for the term “ulcer” as described above.

Unconstrained, the search may be executed as:

```
Select conceptId, term from descriptions
where term like 'ulcer';
```

As noted this would yield two results. To restrict the search to *Clinical findings*, use the reference set 32570071000036102|*Clinical finding foundation reference set* |, as follows.

```
Select term from descriptions
where term like 'ulcer'
and conceptId in (select referencedConceptId from concept_refset
                  where refsetId = 32570071000036102);
```

Only the disorder concept 429040005 |*Ulcer*| is now returned. Note that even concepts within the same hierarchy may have identical synonyms. Problems relating to this may be addressed by restricting results to preferred terms, as described in Section 3.3 below.

Developers may find it useful to create a function to determine if a given concept ID is a member of a certain reference set.

3.3 Use all Synonyms but limit to Preferred Terms

The availability of synonyms in SNOMED CT-AU provides a way for users to search for concepts using a variety of terms. However synonyms are not necessarily unique, even within a single hierarchy, so care must be taken to ensure users

select the concept that unambiguously matches their intentions. A search for "raised blood pressure" may return two concepts with that synonym, both from the *Clinical findings* hierarchy:

- 24184005 |*Raised blood pressure*|
- 38341003 |*Raised blood pressure*|

Additionally, a search may yield matches on several similar synonyms for the same concept. A search of disorders containing the word "hypertension" might return the following terms:

- Hypertension
- HTN – Hypertension
- systemic arterial hypertension
- HT – Hypertension
- BP+ - Hypertension

These terms are all synonyms for the same concept (38341003 |*Hypertensive disorder*|), but from a user perspective, this can raise confusion or doubt relating to which term they should choose. Different results might suggest different concepts.

One approach is to use all the available synonyms to execute the search and identify the associated concepts (using the concept ID), then just display the preferred terms for those concepts.

3.3.1 SQL examples

The simplest way to produce a list showing only individual concepts and preferred terms is to use a function such as get_ADRS_PreferredTerm(conceptId), as described in Section 2.1.1 above.

Using the example above of searching for "raised blood pressure", unconstrained queries might look like:

```
select conceptId, term from concepts
inner join descriptions on concepts.id = descriptions.conceptid
where term = 'raised blood pressure'
and concepts.active = 1;
```

This returns the following:

- 24184005 |*Raised blood pressure*|
- 38341003 |*Raised blood pressure*|

We can change the query so that the result set provides the preferred terms, by using the created function (bolded):

```
select conceptId, get_ADRS_PreferredTerm(conceptId) from concepts
inner join descriptions on concepts.id = descriptions.conceptid
where term = 'raised blood pressure'
and concepts.active = 1;
```

This revised query will now return the following results:

- 24184005 |*Finding of increased blood pressure*|
- 38341003 |*Hypertension*|

Note that the same concepts are returned, but they can now be distinguished by an end user (who would generally not be exposed to the identifiers).

The same approach can also address the issue of multiple synonyms for the same concept being returned, by using the “distinct” keyword.

```
select distinct conceptId, get_ADRS_PreferredTerm(conceptId)
from concepts
inner join descriptions on concepts.id = descriptions.conceptid
where term like '%hypertension%'
and concepts.active = 1;
```

This query returns 201 unique concepts, based on 517 matching terms[5].

This approach leverages the synonyms in SNOMED CT-AU so that users can find concepts using phrases that aren’t necessarily the preferred terms. Although limiting the search results to just preferred terms should reduce the risks of over choice, including misunderstanding, some users may not be comfortable choosing a term that looks markedly different from their input. Developers need to consider their customers and provide a solution that is appropriate.

[5] Based on SNOMED CT-AU November 2013 data. Different releases may produce in different results.

4 Subsumption queries

The concepts in SNOMED CT-AU are organised in hierarchy of increasing specificity, where concepts at the top are broad classes, and those at the bottom representing more specific concepts. Specificity increases through the tree, with concepts joined by 116680003 |*Is a*| relationships being more specific.

It is straightforward to identify if immediate subtype/supertype relationship using the relationships table, as these are the IS A relationships in the file. However, part of the authoring process of SNOMED CT-AU involves classifying the terminology, which includes determining the minimum set of IS A relationships required. As a consequence this set of relationships can change between releases due to the introduction of new content, so it is often more useful to look at all subtype concepts (rather than just direct descendants).

Section 7.7 of the *SNOMED CT Technical Implementation Guide* [1] (TIG) describes many aspects of using these relationships to determine if one concept is a subtype of another. Of specific interest is the use of a transitive closure table, as described in Section 7.7.5.2 Transitive Closure Implementation[6] which also includes code for producing a transitive closure table (though the table names will need to be modified if used in the Reference Set Toolkit database). The transitive closure table provides a fast and easy way for determining if two concepts have a subtype/supertype relationship.

[6] http://www.ihtsdo.org/fileadmin/user_upload/doc/en_us/tig.html?t=tsg2_test_optimizeConcept_transitiveClosure_impl

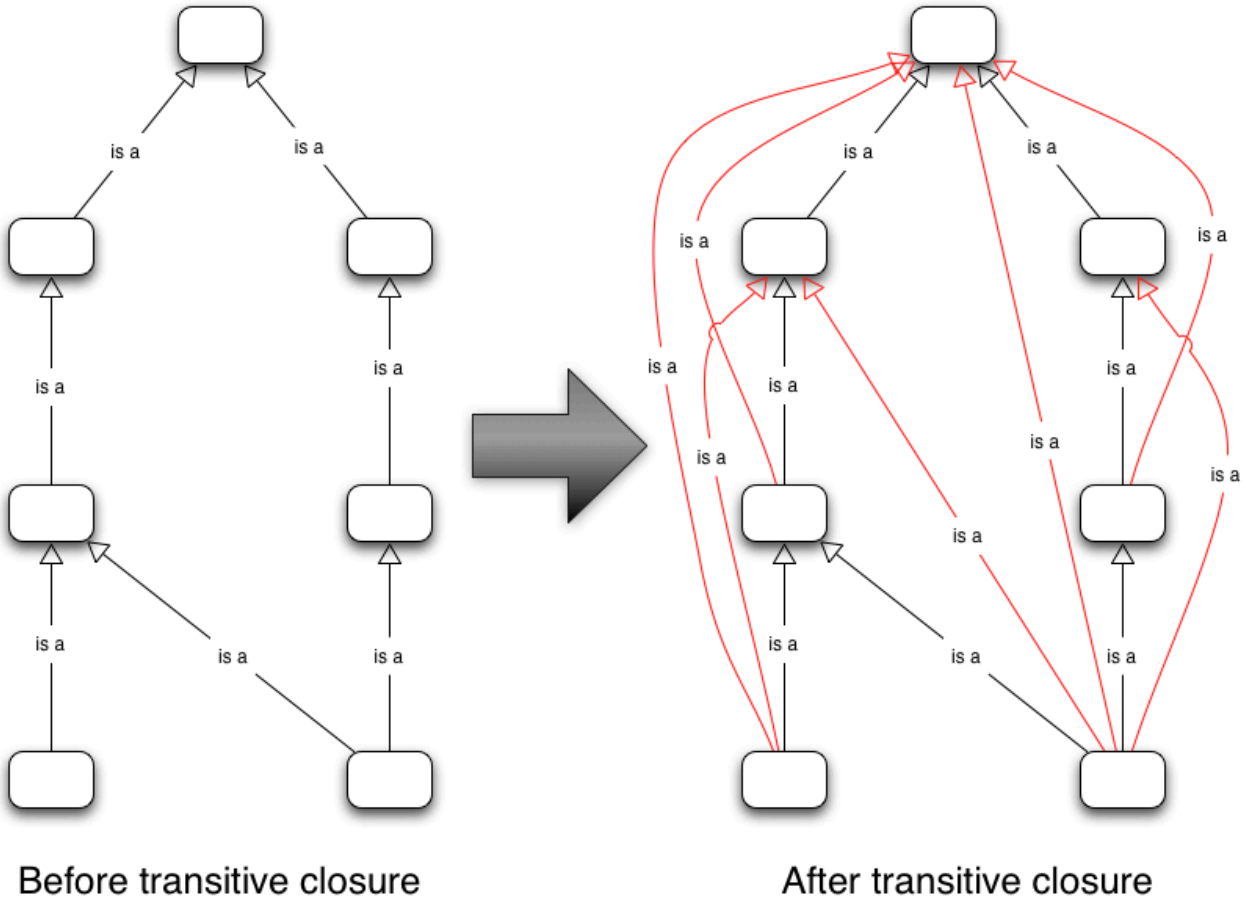

Figure 2 Example of a transitive closure

4.1 SQL examples

Using the concept 442183006 |*Psychogenic dyskinesia*| as an example, the immediate ancestors (parents) of this concept can be identified directly from the relationship file using:

```
select sourceId,destinationId from relationships
where typeId = 116680003
and sourceId = 442183006
```

This query shows the two IS A relationships *Psychogenic dyskinesia* has.

Table 1: Distributed IS A relationships for Psychogenic dyskinesia

sourceId	**destinationId**
Psychogenic dyskinesia	*Dyskinesia*
Psychogenic dyskinesia	*Psychophysiologic disorder*

Using the transitive closure table, a similar query such as the following will return a larger set of 17 IS A relationships to **all** the ancestors of the concept.

```
select sourceId,destinationId from Transitive_Closure
where sourceId = 442183006
```

The query results are tabulated below.

Table 2: Transitive closure IS A relationships for Psychogenic dyskinesia

sourceId	destinationId
Psychogenic dyskinesia	*Dyskinesia*
Psychogenic dyskinesia	*Psychosomatic factor in physical condition*
Psychogenic dyskinesia	*Mental state finding*
Psychogenic dyskinesia	*Psychophysiologic disorder*
Psychogenic dyskinesia	*Motor dysfunction*
Psychogenic dyskinesia	*Neurological finding*
Psychogenic dyskinesia	*Perception AND/OR perception disturbance*
Psychogenic dyskinesia	*Motor nervous system finding*
Psychogenic dyskinesia	*Psychological finding*
Psychogenic dyskinesia	*Functional finding*
Psychogenic dyskinesia	*Finding by site*
Psychogenic dyskinesia	*SNOMED CT Concept*
Psychogenic dyskinesia	*Psychological finding of perception*
Psychogenic dyskinesia	*Clinical history and observation findings*
Psychogenic dyskinesia	*Finding of movement*
Psychogenic dyskinesia	*Mental state, behaviour and/or psychosocial function finding*
Psychogenic dyskinesia	*Clinical finding*

5 Custom reference sets

Many implementers may need to create custom or refined reference sets. In creating these there are two broad approaches: Inclusion and Exclusion.

5.1 Inclusion-based customisation

A custom reference set based on Inclusion principles leverages the published SNOMED CT-AU reference sets, to create larger or smaller implementation reference sets. Larger reference sets may be produced by combining smaller ones, or smaller sets created by specifying extra criteria to apply to an existing reference set.

5.1.1 SQL examples

SNOMED CT-AU includes the 32570351000036105 |*Musculoskeletal finding reference set*|, which contains over ten thousand concepts. Yet a specific implementation for podiatry may only require concepts relating to the foot structure (56459004 |*Foot structure*|). The property 363698007 |*Finding site*| may be used to focus the restriction.

```
select referencedconceptid from concept_refset AS MSrefset
where MSrefset.refsetId = 32570351000036105
and MSrefset.active = 1
and MSrefset.referencedconceptid in
   (select sourceId from relationships
   where active = 1
   and typeId = 363698007
   and destinationId = 56459004);
```

The result is a much smaller set of less than 30 concepts. Note the query above is specifically retrieving only concepts where the finding site is 56459004 |*Foot structure*|. It would be more accurate to use a subsumption query (as described in Section 4 above) to include all concepts where the finding site is a type of foot structure. That is, replace “destinationId = 56459004” with a subsumption clause like “is_KindOf(destinationId, 56459004)”. This query would produce around 800 concepts – all Musculoskeletal finding relating to foot structures.

5.2 Exclusion-based customisation

A custom reference set may also be created by excluding certain content. This would be where the requirements of a reference set are specified and concepts that are also in another reference set are excluded. An example might be where a reference set of all congenital deformities is required. The result may include a number of veterinary concepts, however these can be excluded using the *Australian non-human reference set*.

Note: The 75901000036104 |*Australian non-human reference set*| is a subset of SNOMED CT-AU content that is considered irrelevant for use when authoring human medical records. The scope for all the reference sets available in SNOMED CT-AU is human records, and NCTIS published reference sets include the non-human concepts. To ensure exclusion of non-human content, a custom reference set should be a subset of

existing SNOMED CT-AU reference sets. However, custom reference sets may be created and have the non-human content excluded afterwards.

5.2.1 SQL examples

Using the example above where a reference set of congenital deformities is needed, the requirements might be specified as all concepts that are types of both 66091009 |*Congenital disease*| and 417893002 |*Deformity*| are included. Assuming a subsumption function such as “is_KindOf (candidate,supertype)”, such a query may be specified as:

```
select id from concepts
where is_KindOf(id,417893002)
and is_KindOf (id,66091009)
```

This query produces around 260 concepts[7] – including some that do not relate to humans. To exclude these, the query can be amended to restrict concepts that are also in the *Australian non-human reference set*:

```
select id from concepts
where is_KindOf(id,417893002)
and is_KindOf(id,66091009)
   and id not in (select referencedconceptid from concept_refset
                  where refsetId = 75901000036104);
```

The results from this query now only include 254 concepts. That is, six non-human concepts have been excluded:

- 92990000 |*Congenital bent tail*|
- 93094004 |*Congenital hooked tail*|
- 93231003 |*Congenital hyperextension of paw*|
- 93233000 |*Congenital hyperflexion of paw*|
- 93416005 |*Congenitally short snout*|
- 370482006 |*Schistosomus reflexus*|

[7] Based on SNOMED CT-AU November 2013 data. Different releases may produce in different results.

6 References

1. IHTSDO. *SNOMED CT Technical Implementation Guide*. Copenhagen: IHTSDO; 2013. July 2013 release. Available from: http://www.snomed.org/doc.

2. NEHTA. *Reference Set Toolkit Guide: SNOMED CT-AU 20131130*. Sydney: NEHTA; 2013. 20131130 Release. Available from: https://nehta.org.au/aht/index.php.

3. IHTSDO. *SNOMED CT User Guide*. Copenhagen: IHTSDO; 2013. July 2013 release. Available from: http://www.snomed.org/doc.