2018-01-31 - SLPG Meeting

Date & Time

20:00 UTC Wednesday 31st January 2018

Teleconference Details

To join the meeting please go to https://snomed.zoom.us/j/471420169.

Further information can be found at SLPG meeting information

Goals

- Discuss recent comments - ECL requirements and string collation
- Updates to SNOMED URI Standard
- Progress SNOMED Query language

Attendees

- Chair: Linda Bird
- Project Group: Michael Lawley, Daniel Karlsson, Ed Cheetham, Anne Randorff Højen

Apologies

Agenda and Meeting Notes

Description	Owner	Notes
Welcome and apologies	Linda Bird	
Recent comments	Linda Bird	**Michael**: "How can I write an ECL expression to match attribute names - for example, list all the attribute names that are used by << 404684003 |Clinical finding|. • **Added to ECL future requirements page.** **Daniel**: "Query language and collation" When specifying the lexical search type for term matching there is a need to specify the collation used, and to specify the default collation for the language in which the terms are to be matched are represented. Examples based on mysql collation behavior: "AAO" matches "ÅÄÖ" in utf8_generic_ci and utf8_unicode_ci (and utf8_german2_ci) but not in utf8_swedish_ci collation. "Aåa" matches "aåa" in utf8_generic_ci and utf8_swedish_ci but not in utf8_bin collation (i.e. case insesitive vs. sensitive, sometimes you need case sensitivity when searching...). Similar behavior can be implemented e.g. by java.text.Collator in java or by the collection.find() or cursor.collation() method in MongoDB. • **Action from last week: Is there a standard or generic approach to defining Collation behaviour?** • **Using LET ... IN ...** ◦ (LET substrate = X IN *) MINUS (LET substrate = Y IN *) ◦ LET X = (<< 1234 : 5678 = << 6547) IN X MINUS >! X ▪ Find leaf nodes of X ◦ (LET substrate = X IN << 12345) MINUS (LET substrate = Y IN << 12345)
URI Standard	Linda Bird	• SLPG to review proposed updates to SNOMED CT URI Standard ◦ In particular, 2.9 URIs for Language Syntaxes and 2.10 URIs for Language Instances. ◦ Suggestions on 2.7 Comparing URIs for Equality of Reference would also be appreciated
Query Language	Linda Bird	Outstanding questions about relationship filters • Consider issues and potential resolutions. Questions needing resolution include:

- **Do we need inferred relationship filters?**
  - Answer: Tentative Yes
- **If yes, then what are some good use cases?**
  - Answer: QA - Find me all the source concepts of a relationship added in an extension module
  - Answer: QA - Find me all the source concepts of a relationship added at a particular effective time
  - **ACTION** (Daniel) - To investigate QA example use cases based on RVF
- **How should no brackets be interpreted?**
  1. What does this mean? "< 404684003 |Clinical finding| {{ C.definitionStatusId = 900000000000074008 |Primitive|}}"
     - 1a) (< 404684003 |Clinical finding|) {{ C.definitionStatusId = 900000000000074008 |Primitive|}}
     - 1b) < (404684003 |Clinical finding| {{ C.definitionStatusId = 900000000000074008 |Primitive|}})
     - Answer: 1a - **AGREED**
  2. What does this mean? "< 404684003 |Clinical finding| : 363698007 |Finding site| = << 80891009 |Heart structure| {{ C.definitionStatusId = 900000000000074008 |Primitive|}}"
     - 2a) (< 404684003 |Clinical finding| : 363698007 |Finding site| = << 80891009 |Heart structure|) {{ C.definitionStatusId = 900000000000074008 |Primitive|}}
     - 2b) < 404684003 |Clinical finding| : 363698007 |Finding site| = ((<< 80891009 |Heart structure|) {{ C.definitionStatusId = 900000000000074008 |Primitive|}})
     - 2c) < 404684003 |Clinical finding| : 363698007 |Finding site| = << ((80891009 |Heart structure|) {{ C.definitionStatusId = 900000000000074008 |Primitive|}})
     - Answer: 2a - **AGREED**
- **If we do support relationship filters, then where should they go and how should they be bracketed?**
  1. Directly after the attribute - for example:

        - 1a) < 404684003 |Clinical finding| : 363698007 |Finding site| {{ R.moduleId = 32506021000036107 |AU extension|}} = << 80891009 |Heart structure|
        - or
        - 1b) < 404684003 |Clinical finding| : **(** 363698007 |Finding site| {{ R.moduleId = 32506021000036107 |AU extension|}} **)** = << 80891009 |Heart structure|
    2. Directly after the refinement (with brackets) - for example:
        - 2a) < 404684003 |Clinical finding| : (**(** 363698007 |Finding site| = << 80891009 |Heart structure| **)** {{ R.moduleId = 32506021000036107 |AU extension|}})
        - or
        - 2b) < 404684003 |Clinical finding| : **(** 363698007 |Finding site| = << 80891009 |Heart structure| {{ R.moduleId = 32506021000036107 |AU extension|}} **)**
    3. Directly after the =/!= , for example:
        - 3a) < 404684003 |Clinical finding| : 363698007 |Finding site| = {{ R.moduleId = 32506021000036107 |AU extension|}} << 80891009 |Heart structure|
    4. Directly after the constraint operator (Interpretation - there exists a relationship in the chain) - for example:
        - 3a) < {{ R.moduleId = 32506021000036107 |AU extension|}} 404684003 |Clinical finding|
    5. After the concept being constrained - for example:
        - 4a) < 404684003 |Clinical finding| {{ R.moduleId = 32506021000036107 |AU extension|}}
        - or
        - 4b) **(** < 404684003 |Clinical finding| **)** {{ R.moduleId = 32506021000036107 |AU extension|}}
    - Answer(s): ? 2b and 4a (Note: 1b is more consistent with dotted notation)
- **What exactly do relationship filters mean?**
    - < 404684003 |Clinical finding| {{ R.moduleId = 32506021000036107 |AU extension|}}
    - *There exists* ***at least one*** |is a| relationship, that connects the given descendant to |Clinical finding|, meets the given filter criteria
    - ***All*** |is a| relationships, that connect the given descendant to |Clinical finding|, meet the given filter criteria
- **How do we indicate the execution order for relationship filters?**
    - Dotted notation - For example "ANY . << 246090004 |Associated finding| . 363698007 |Finding site|"
        1. Only returns those concepts that result from using a |Finding site| relationship in the AU core (**Note**: return the destinationConcept of these relationships)
            - 1a) ANY . << 246090004 |Associated finding| . 363698007 |Finding site| {{ R.moduleId != 'AU Core' }}
            - or
            - 1b) ANY . << 246090004 |Associated finding| . **(** 363698007 |Finding site| {{ R.moduleId != 'AU Core' }} **)**
            - Answer: ? 1b
        2. Only returns those concepts that result from using an |Associated finding| relationship (or type of |associated finding| relationship) in the AU core (**Note**: return the destinationConcept of these relationships)
            - 2a) ANY . << 246090004 |Associated finding| {{ R.moduleId != 'AU Core' }}. 363698007 |Finding site|
            - or
            - 2b) ANY . **(** << 246090004 |Associated finding| **)** {{ R.moduleId != 'AU Core' }}. 363698007 |Finding site|
            - or
            - 2c) ANY . **(** << 246090004 |Associated finding| {{ R.moduleId != 'AU Core' }} **)**. 363698007 |Finding site|
            - Answer: ? 2c
    - Cardinality
        1. Exactly one matching relationship in the AU core (but may be other matching relationships in other modules)
            - 1a) < 404684003 |Clinical finding| : [1..1] **(** 63698007 |Finding site| = << 80891009 |Heart structure| {{ R.moduleId = 32506021000036107 |AU extension|}} **)**
            - or
            - 1b) < 404684003 |Clinical finding| : [1..1] **(** 63698007 |Finding site| = << 80891009 |Heart structure| **)** {{ R.moduleId = 32506021000036107 |AU extension|}}
            - Answer: ? 1a
        2. Exactly one matching relationship, and this relationship is in the AU core
            - 2a) < 404684003 |Clinical finding| : **(** [1..1] 63698007 |Finding site| = << 80891009 |Heart structure| {{ R.moduleId = 32506021000036107 |AU extension|}} **)**
            - or
            - 2b) < 404684003 |Clinical finding| : **(** [1..1] 63698007 |Finding site| = << 80891009 |Heart structure| **)** {{ R.moduleId = 32506021000036107 |AU extension|}}
            - Answer: ? 2a

Outstanding questions about lexical filters

		○ **Which keyword filters must be qualified with a component/refset type? (e.g. "C", "D", "R", "M")** ■ For example, do we qualify "languageRefSetId", "preferredTerm", "fullySpecifiedName", "acceptableTerm"? ■ ○ **What options do we provide for constraining terms, preferred terms, fully specified names, acceptable terms and/or selecting language refsets?** ■ See slide deck
Confirm next meeting date /time	Linda Bird	Next meeting to be held at 20:00 UTC on **Wednesday 14th February 2018**

File	Modified

No files shared here yet.