2019-02-13 - SLPG Meeting

Date & Time

20:00 UTC Wednesday 13th February 2019

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

- Review actions from last meeting
- Consider implementation support requirement for ECL transitivity/role chaining.
- Review 2019 work items, including:
    - Proposed new ECL language features
    - Updates to URI standard
    - Enhancement to template language
    - Draft Query Language

Attendees

- Chair: Linda Bird
- Project Group: Ed Cheetham, Michael Lawley , Anne Randorff Højen, Rob Hausam, Harold Solbrig

Apologies

- Daniel Karlsson,

Agenda and Meeting Notes

Description	Owner	Notes
Welcome and apologies	Linda Bird	
Actions from last week	Linda Bird	• Actions from last week: ◦ Post authoratative ANTLR syntax used by SNOMED International ◦ Update ABNF with additional UTF characters - please refer to: ▪ https://github.com/shexSpec/grammar/blob/master/ShExDoc.g4 ▪ https://github.com/hsolbrig/FormalExpressionSpec/blob/master/grammar/ECL.g4 ▪ https://www.fileformat.info/info/unicode/char/d7/index.htm ◦ SLPG members to bring back a use case for precomputed transitive and role chained relationships
ECL - Transitivity, Reflexivity & Role chaining	Linda Bird Kai Kewley	Proposed extension to ECL to support transitive relationships and role chaining (to align with new enhanced DL axioms) • Example 1 **Direct relationship** • < |Body structure|: << 774081006 |Proper part of| = << 51185008 |Chest| **Transitive relationship** • < |Body structure|: << 774081006 |Proper part of|* = << 51185008 |Chest| • <<(< |Body structure|: << 774081006 |Proper part of| = << 51185008 |Chest|) • Proposal to consider: ◦ < |Disease|: |Causative agent|* = << |Staph aureus| ◦ < |Disease|: (|Due to| OR |Causative agent|)* = << |Staph aureus| ◦ < |Disease|: (|Due to|* OR |Causative agent|*) = << |Staph aureus| ◦ < |Disease|: ** = << |Staph aureus| Example 2 **Direct relationship** • < 71388002 ||: 363701004 |Direct substance| = 372687004 |Amoxicillin| **Role chained relationship (via 738774007 |is modification of|)** • < 71388002 ||: 363701004 |Direct substance|* = 372687004 |Amoxicillin| • <<(< 71388002 ||: 363701004 |Direct substance| = 372687004 |Amoxicillin|) What implementation support will be required? Should we provide easy access to those relationships that can be inferred by transitivity and role chains (note: These will be excluded from the inferred relationship file as they are redundant). If so, then what format should be used - for example, a TSV file with the following columns: ◦ sourceId ◦ destinationId ◦ typeId ◦ relationshipGroup Things without compelling use cases won't be prioritised.

ECL - Executing maps	Linda Bird	Proposed extension to ECL to support the execution of maps (focusing on the resolution of historical refsets)
		• The specific use-case here comes initially from Jeremy and relates to being able to work with inactive concepts via the historical association maps. For example, given an ECL expression that identifies a set of concepts 'c' to be used for retrieving patient records, you probably also want to retrieve records for sameAs (c) and replacedWith (c)
		  ◦ Example:
		  ▪ ~~(< 72704001 |Fracture| AND ^ 900000000000527005 |SAME AS association reference set|) . 900000000000533001 |Association target component|~~
		  ▪ (900000000000527005 |SAME AS association reference set| . 900000000000533001 |Association target component|): |Referenced component| = < |Fracture|
		  ▪ Michael's existing approach: mapsTo(|SAME AS|, < |Fracture|)Or mappedTo(…)mappedFrom (|SAME AS|, |inactive concept|)mappedFrom(|REPLACED BY|, |inactive concept|)
		  ▪ Alternative suggestion: Use the substrate to include historical snapshots.
		  • E.g. Historical and current concepts that are fractures and have an associated morphology, but not historical morphologies
		  • (....) AND (...)

ECL - Returning attributes	Michael Lawley	Proposal from Michael: • Currently ECL expressions can match (return) concepts that are either the source or the target of a relationship triple (target is accessed via the 'reverse' notation or 'dot notation', but not the relationship type (ie attribute name) itself. For example, I can write: << 404684003|Clinical finding| : 363698007|Finding site| = <<66019005|Limb structure| << 404684003|Clinical finding| . 363698007|Finding site| But I can't get all the attribute names that are used by << 404684003|Clinical finding|
Template Syntax	Linda Bird	New requirements • 2 slots must have the same value • 2 slots must have different values • The value of 1 slot must subsume the value of another • Default value for slots

URI Standard	Linda Bird / Michael Lawley	• Finalize and publish language and language instance URIs
		• Proposal from Michael:
		  ◦ Provisional releases contain content that should be treated as somehow separate and distinct from normal production releases. However, to ingest, manipulate, and process with standard toolchains (e.g., terminology servers), it still needs to be identified while still remaining distinct from production content.
		This proposal is that a parallel URI space (`http://snomed.info/xsct`) be set aside for such provisional releases. Mirroring the `http://snomed.info/sct` URI space, this would include:
		  ▪ `http://snomed.info/xsct` meaning the Provisional SNOMED CT code system
		  ▪ `http://snomed.info/xsct/{moduleId}` meaning a specific Edition of Provisional SNOMED CT, and
		  ▪ `http://snomed.info/xsct/{moduleId}/version/{effectiveTime}` meaning a specific Version of Provisional SNOMED CT
		The choice of "xsct" follows the use of the "x" prefix in the corresponding package and file naming conventions.
		  ◦ Use case: Need to load preview releases into tool chain. However, these are not for production use - this could be dangerous. Could separate these for development. Identify these experimental releases explicitly. By making it an international standard, it promotes the idea for others to make this a safe practice. Question: How does SnowStorm handle this, with loading preview data?

Query Language - Summary from previous meetings	Linda Bird	**Examples: version and language** ○ << 64572001 |Disease| {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131 ○ << 64572001 |Disease| {{ synonym = "*heart*" }} **VERSION** http://snomed.info/sct/90000000000207008/version/20180131 ○ << 64572001 |Disease| {{ FSN = "*heart*" }} **VERSION** http://snomed.info/sct/90000000000207008/version/20180131 ○ << 64572001 |Disease| {{ FSN = "*heart*" }} **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** W ○ << 64572001 |Disease| {{ preferredTerm = "*heart*"}} **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** Y ○ << 64572001 |Disease| {{ acceptableTerm = "*heart*"}} **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** Y ○ (* {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** Z) MINUS (* {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/90000000000207008/version/20170731, **LANGUAGE** W) ○ X MINUS Y **WHERE** X = * , Y = (* {{ term = "*heart*" }}) **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** W **Notes** ○ Allow nested where, version, language ○ Scope of variables is inner query
		Examples: where ○ X MINUS >! X **WHERE** X = (<< 1234 : 5678 = << 6547) ○ X MINUS >! X **WHERE** X = (<< 1234 : 5678 = << 6547) **VERSION** http://snomed.info/sct/90000000000207008/version/20180131 ○ X MINUS >! Y **WHERE** X = (<< 1234 : 5678 = << 6547), Y = (<< 1456) **VERSION** http://snomed.info/sct/90000000000207008/version/20180131 ○ X MINUS >! X **WHERE** X = (<< 1234 : 5678 = << 6547) **VERSION** http://snomed.info/sct/90000000000207008/version/20180131 , **LANGUAGE** 900000000000508004 |GB English| ○ X MINUS >! X **WHERE** X = (<< 1234 : 5678 = << 6547) **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** 999001881000000108|GB clinical extension LRS|, 900000000000508004 |GB English| ○ X minus >! X **WHERE** X = (< M **WHERE** M = (< 1234))) **VERSION** http://snomed.info/sct/90000000000207008/version/20180131, **LANGUAGE** 999001881000000108|GB clinical extension LRS|, 900000000000508004 |GB English| **Notes** ■ Allow nested variable definitions, but recommend that people don't due to readability ■ Scope of variables is the inner query ■ No recursion e.g X **WHERE** X = 1234 MINUS X • ie can't use a variable in its own definition • ie X is only known on the left of the corresponding **WHERE**, and not on the right of the **WHERE**

Keywords for Term-based searching:

- **D.term**
  - D.term = "*heart*"
  - D.term = wild:"*heart*"
  - D.term = regex:".*heart.*"
  - D.term = match:"hear att"
  - D.term = (sv) wild: "*heart*"
- **D.languageCode**
  - D.languageCode = "en"
  - D.languageCode = "es"
- **D.caseSignificanceId**
  - D.caseSignificanceId = 900000000000448009 |entire term case insensitive|
  - D.caseSignificanceId = 900000000000017005 |entire term case sensitive|
  - D.caseSignificanceId = 900000000000020002 |only initial character case insensitive|
- **D.caseSignificance**
  - D.caseSignificance = "insensitive"
  - D.caseSignificance = "sensitive"
  - D.caseSignificance = "initialCharInsensitive"
- **D.typeId**
  - D.typeId = 900000000000003001 |fully specified name|
  - D.typeId = 900000000000013009 |synonym|
  - D.typeId = 900000000000550004 |definition|
- **D.type**
  - D.type = "FSN"
  - D.type = "fullySpecifiedName"
  - D.type = "synonym"
  - D.type = "textDefinition"
- **D.acceptabilityId**
  - D.acceptabilityId = 900000000000549004 |acceptable|
  - D.acceptabilityId = 900000000000548007 |preferred|
- **D.acceptability**
  - D.acceptability = "acceptable"
  - D.acceptability = "preferred"

Additional Syntactic Sugar

- **FSN**
  - FSN = "*heart"
    - D.term = "*heart", D.type = "FSN"
    - D.term = "*heart", D.typeId = 900000000000003001 |fully specified name|
  - FSN = "*heart" **LANGUAGE** X
    - D.term = "*heart", D.type = "FSN", D.acceptability = * **LANGUAGE** X
    - D.term = "*heart", D.typeId = 900000000000003001 |fully specified name|, acceptabilityId = * **LANGUAGE** X
- **synonym**
  - synonym = "*heart"
    - D.term = "*heart", D.type = "synonym"
    - D.term = "*heart", D.typeId = 900000000000013009 |synonym|
  - synonym = "*heart" **LANGUAGE** X
    - D.term = "*heart", D.type = "synonym", D.acceptability = * **LANGUAGE** X
    - D.term = "*heart", D.typeId = 900000000000013009 |synonym|, (D.acceptabilityId = 900000000000549004 |acceptable| OR D.acceptabilityId = 900000000000548007 |preferred|) **LANGUAGE** X
- **synonymOrFSN**
  - synonymOrFSN = "*heart"
    - synonym = "*heart" OR FSN = "*heart"
    - D.term = "*heart", (D.type = "synonym" OR D.type = "fullySpecifiedName")
  - synonymOrFSN = "*heart" **LANGUAGE** X
    - synonym = "*heart" OR FSN = "*heart" **LANGUAGE** X
    - D.term = "*heart", (D.type = "synonym" OR D.type = "fullySpecifiedName"), D.acceptability = * **LANGUAGE** X
- **textDefinition**
  - textDefinition = "*heart"
    - D.term = "*heart", D.type = "definition"
    - D.term = "*heart", D.typeId = 900000000000550004 |definition|
  - textDefinition = "*heart" **LANGUAGE** X
    - D.term = "*heart", D.type = "definition", D.acceptability = * **LANGUAGE** X
    - D.term = "*heart", D.typeId = 900000000000550004 |definition|, D.acceptabilityId = * **LANGUAGE** X
- **Unacceptable Terms**
  - (D.term = "*heart") MINUS (D.term = "*heart", D.acceptability = * **LANGUAGE** X)

		Language preferences using multiple language reference sets
		• **LRSs that use the same Language tend to use 'Addition' - i.e. child LRS only includes additional acceptable terms, but can override the preferred term**
		  ◦ **E.g. Regional LRS that adds local dialect to a National LRS**
		  ◦ **E.g. Specialty-specific LRS**
		  ◦ **E.g. Irish LRS that adds local preferences to the en-GB LRS**
		  ▪ 99999900 |Irish language reference set| **PLUS** |GB English reference set|
		• **LRSs that define a translation to a different language tend to use 'Replacement' - i.e. child LRS replaces set of acceptable and preferred terms for any associated concept**
		  ◦ **E.g. Danish LRS that does a partial translation of the International Release**
		  ▪ 999999 |Danish language reference set| **ELSE** |GB English reference set|

Other topics	Linda Bird	• Any other topics?
Confirm next meeting date /time	Linda Bird	The next SLPG meeting will be held in 2 weeks at 20:00 UTC on **Wednesday 6th February.**

File	Modified

No files shared here yet.