2020-07-15 - SLPG Meeting

Date & Time

20:00 to 21:00 UTC Wednesday 15th July 2020

Location

Zoom meeting link (password: 764978)

Attendees

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

Goals

- To finalize draft collation/folding wording
- To agree on options for supporting case sensitivity/regex
- To develop examples to illustrate new term searching functionality

Apologies

Peter Jordan, Anne Randorff Højen

Agenda and Meeting Notes

Description	Owner	Notes
Welcome and agenda	Linda Bird	**NOTE**: Next meeting to be held on Wednesday 29th July
Concrete Values	Linda Bird	**Specifications** - **SCG v2.4** (with booleans) has been published - **ECL v1.4** (with booleans, childOrSelfOf and parentOrSelfOf) has been published - **STS v1.1** and **ETL v1.1** (with booleans) to be published (after MAG meeting this week) - **MRCM** (with updated rangeConstraint) - 5.3 MRCM Attribute Range Reference Set

Expression Constraint Language	Linda Bird	- **Recent Updates to WIP**
		  ○ WIP 5.3 Informative Comments - Updated to document new term filter rules
		  ○ WIP 5.5 Collation of Term Filters - Draft wording added for review and editing
		- **This Week's Questions**
		  ■ Confirm updated wording re collation recommendations.
		  ■ Should we support an option to allow case sensitive searches? If so, should this be supported by (a) an additional parameter, (b) regex searching?
		- **On Hold**
		  ○ Can/should we register ECL as a MIME type? – Waiting for volunteer time to complete registration form
		- **To Do -** Child or self (<<!) and Parent or self (>>!)
		  ○ New examples to be added

- **TERM SEARCH FILTERS** - Syntax currently being drafted
  - **Examples**
    - < 404684003 |Clinical finding (finding)| {{ term = "heart att"}}
    - < 404684003 |Clinical finding (finding)| {{ term != "heart att"}} – A concept for which there exists a description that does not match – E.g. Find all the descendants of |Fracture| that have a description that doesn't contain the word |Fracture|
    - < 404684003 |Clinical finding (finding)| MINUS * {{ term = "heart att"}} – A concept which does not have any descriptions matching the term
    - < 404684003 |Clinical finding (finding)| {{ term = match: "heart att" }} – match is word (separated by white space) prefix any order; Words in substrate are ....; Search term delimiters are any mws
    - < 404684003 |Clinical finding (finding)| {{ term = wild: "heart* *ack" }}
    - < 404684003 |Clinical finding (finding)| {{ term = ("heart" "att") }}
    - < 404684003 |Clinical finding (finding)| {{ term != ("heart" "att") }} – matches concepts with a description that doesn't match "heart" or "att"
    - < 404684003 |Clinical finding (finding)| {{ TERM = (MATCH:"heart" WILD:"*ack") }}
    - < 404684003 |Clinical finding (finding)| {{ term = "myo", term = wild:"*ack" }} — Exists one term that matches both "myo" and "*ack"
    - < 404684003 |Clinical finding (finding)| {{ term = "myo" }} {{ term = wild:"*ack" }} -– Exists one term that matches "myo", and exists a term that matches "*ack" (filters may match on either same term, or different terms)
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = se }}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, typeId = 900000000000013009 |synonym| }}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, typeId = (900000000000013009 |synonym| 900000000000003001 |fully specified name|)}}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, typeId != 900000000000550004 |Definition|}}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, type = syn }}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, type != def }}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, type = (syn fsn) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "hjärta", language = SE, type != (syn fsn) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "cardio", dialectId = 900000000000508004 |GB English| }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialectId = (999001261000000100 |National Health Service realm language reference set (clinical part)| 999000691000001104 |National Health Service realm language reference set (pharmacy part)|) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialect = en-gb }}
    - < 404684003 |Clinical finding (finding)| {{ dialect != en-gb }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialect = (en-nhs-clinical en-nhs-pharmacy) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialect = en-nhs-clinical (900000000000548007 |Preferred|) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialect = en-nhs-clinical (prefer) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialect = en-nhs-clinical (accept) }}
    - < 404684003 |Clinical finding (finding)| {{ term = "card", dialect = en-nhs-clinical (prefer accept), dialect = en-gb (prefer) }}
    - < 404684003 |Clinical finding| MINUS * {{ dialect = en-nhs-clinical}}
    - < 73211009 |diabetes| MINUS * {{ dialect = en-nz-patient }}
    - < 73211009 |diabetes| MINUS < 73211009 |diabetes| {{ dialect = en-nz-patient }}
    - < 73211009 |diabetes| {{ term = "type" }} MINUS < 73211009 |diabetes| {{ dialect = en-nz-patient }}
    - (< 404684003 |Clinical finding|:363698007|Finding site| = 80891009 |Heart structure|) {{ term = "card" }} MINUS < (404684003 |Clinical finding|:363698007|Finding site| = 80891009 |Heart structure|) {{ dialect = en-nz-patient }}
    - < 73211009 |Diabetes| {{ term = "type" }} OR < 49601007 |Disorder of cardiovascular system (disorder)| {{ dialect = en-nz-patient }}
  - **Previous Decisions**
    - Wild Term Filter - Everything inside the quotation marks is the search term (including leading and trailing spaces - Note: Match term is tokenized, but wild search is not
    - Acceptability will be an option directly attached to a dialect filter - for example:
      - * {{ term = "card", dialect = en-nhs-clinical (accept prefer), dialect = en-gb (prefer) }}
      - * {{ term = "card", dialect = en-nhs-clinical, dialect != en-nhs-clinical (accept), dialect = en-gb (900000000000548007 |Preferred|) }}
    - The default behaviour of a system implementing these ECL queries with term searching, is to use *a symmetric searching* at the *secondary level*. This means that the search is, by default, case insensitive, with some character normalization behaviour (as determined by the value of the language). Which characters are normalized in the search string and target term index should be determined using the CLDR (Collation Locale Data Repository) rules for the given language.

      - "This means that characters in the search that are unmarked will match a character in the target that is either marked or unmarked at the same level, but a character in the query that is marked will only match a character in the target that is marked in the same way. At the secondary level, an unaccented 'e' would be treated as unmarked, while the accented letters ' é', 'è' would (in English) be treated as marked. Thus a lowercase query character matches that character or the uppercase version of that character, and an unaccented query character matches that character or any accented version of that character even if *strength* is set to secondary." [http://www.unicode.org/reports/tr10/#Asymmetric_Search_Secondary]

- ○ **DONE - Send recommendation to MAG to consider the following**
  1. **Dialect Alias Refset**
     - Alternative 1 - Annotation Refset
       - Dialect_Alias refset : alias + languageRefset-conceptId - e.g. "en-GB", 900000000000050 8004
       Example row
         - ○ referencedComponentId = 999001261000000100
         - ○ dialectAlias = nhs-clinical
     - Alternative 2 - Add alias as a synonym to the language refset concept
       - Create a simple type refset that refers to the preferred alias for each language refset
  2. **Constructing a Language Refset from other Language Refset**
     - ■ Allowing an intensional definition for a language refset
     - ■ Includes order/precedence of language refsets being combined
- **Potential Use cases** - Note some of these will be **out of scope** for the simple ECL filters
  - ○ Find concepts with a term which matches "car" that is preferred in one language refset and not acceptable in another
  - ○ Find the concepts that ..... have a PT = X in language refset = Y
  - ○ Find the concepts that ..... have a Syn = X in language refset = Y
  - ○ Find the concepts that ... have one matching description in one language, and another matching description in another language
  - ○ Find the concepts that have a matching description that is in language refset X and not in language refset Y
  - ○ Find the concepts that .... have a matching description that is either preferred in one language refset and/or acceptable in another language refset
  - ○ Returning the set of concepts, for which there exists a description that matches the filter

  - ○ Intentionally define a reference set for chronic disease. Starting point was ECL with modelling; This misses concepts modelled using the pattern you would expect. So important in building out that reference set.
  - ○ Authors quality assuring names of concepts
  - ○ Checking translations, retranslating. Queries for a concept that has one word in Swedish, another word in English
  - ○ AU use case would have at most 3 or 4 words in match
  - ○ Consistency of implementation in different terminology services
  - ○ Authoring use cases currently supported by description templates
  - ○ A set of the "*ectomy"s and "*itis"s

Querying Refset Attributes	Linda Bird	Proposed syntax to support querying and return of alternative refset attributes (To be included in the SNOMED Query Language)
		
		- **Example use cases**
		  - ○ Execution of maps from international substance concepts to AMT substance concepts
		  - ○ Find the anatomical parts of a given anatomy structure concept (in |Anatomy structure and part association reference set)
		  - ○ Find potential replacement concepts for an inactive concept in record
		  - ○ Find the order of a given concept in an Ordered component reference set
		  - ○ Find a concept with a given order in an Ordered component reference set

- Potential syntax to consider (brainstorming ideas)
  - **SELECT ??**
    - SELECT 123 |referenced component|, 456 |target component|
      FROM 799 |Anatomy structure and part association refset|
      WHERE 123 |referenced component| = (< 888 |Upper abdomen structure| {{ term = "*heart*" }})
    - SELECT id, moduleId
      FROM concept
      WHERE id IN (< |Clinical finding|)
      AND definitionStatus = |primitive|
    - SELECT id, moduleId
      FROM concept, ECL("< |Clinical finding") CF
      WHERE concept.id = CF.sctid
      AND definitionStatus = |primitive|
    - SELECT ??? |id|, ??? |moduleId|
      FROM concept (< |Clinical finding| {{ term = "*heart*" }} {{ definitionStatus = |primitive| }})
    - **Question** - Can we assume some table joins - e.g. Concept.id = Description.conceptId etc ??
    - **Examples**
      - Try to recast relationships table as a Refset table + graph-based extension
      - Find primitive concepts in a hierarchy
  - **ROW ... ?**
    - ROWOF (|Anatomy structure and part association refset|) ? (|referenced component| , |target component|)
      - same as: ^ |Anatomy structure and part association refset|
    - ROWOF (|Anatomy structure and part association refset|) . |referenced component|
      - same as: ^ |Anatomy structure and part association refset|
    - ROWOF (|Anatomy structure and part association refset|) {{ |referenced component| = << |Upper abdomen structure|}} ? |targetComponentId|
    - ROWOF (< 900000000000496009|Simple map type reference set| {{ term = "*My hospital*"}}) {{ 449608002|Referenced component| = 80581009 |Upper abdomen structure|}} ? 900000000000505001 |Map target|
      - (ROW (< 900000000000496009|Simple map type reference set| {{ term = "*My hospital*"}}) : 449608002|Referenced component| = 80581009 |Upper abdomen structure|). 900000000000505001 |Map target|
  - **# ... ?**
    - # |Anatomy structure and part association refset| ? |referenced component\
    - # (|Anatomy struture and part association refset| {{|referenced component| = << |Upper abdomen structure|) ? |targetComponentid|
  - **? notation + Filter refinement**
    - |Anatomy structure and part association refset| ? |targetComponentId|
    - |Anatomy structure and part association refset| ? |referencedComponent| (Same as ^ |Anatomy structure and part association refset|)
      (|Anatomy structure and part association refset| {{ |referencedComponent| = << |Upper abdomen structure}})? |targetComponentId|
    - **(** |Anatomy structure and part association refset| {{ |targetComponentId| = << |Upper abdomen structure}} **)** ? |referencedComponent|
    - **(** |My ordered component refset|: |Referenced component| = |Upper abdomen structure **)** ? **|priority order|**
    - **? |My ordered component refset| {{ |Referenced component| = |Upper abdomen structure| }} . |priority order|**
    - **? |My ordered component refset| . |referenced component|**
      - **equivalent to ^ |My ordered component refset|**
    - **? (<|My ordered component refset|) {{ |Referenced component| = |Upper abdomen structure| }} . |priority order|**
    - **? (<|My ordered component refset| {{ term = "*map"}}) {{ |Referenced component| = |Upper abdomen structure| }} . |priority order|**
    - **REFSETROWS (<|My ordered component refset| {{ term = "*map"}}) {{ |Referenced component| = |Upper abdomen structure| }} SELECT |priority order|**
  - **Specify value to be returned**
    - ? 449608002 |Referenced component|?
      734139008 |Anatomy structure and part association refset|
    - ^ 734139008 |Anatomy structure and part association refset| (Same as previous)
    - ? 900000000000533001 |Association target component|?
      734139008 |Anatomy structure and part association refset|
    - ? 900000000000533001 |Association target component|?
      734139008 |Anatomy structure and part association refset| :
      449608002 |ReferencedComponent| = << |Upper abdomen structure|
    - ? 900000000000533001 |Association target component|?
      734139008 |Anatomy structure and part association refset|
      {{ 449608002 |referencedComponent| = << |Upper abdomen structure| }}
    - (? 900000000000533001 |Association target component|?
      734139008 |Anatomy structure and part association refset| :
      449608002 |ReferencedComponent| = (<< |Upper abdomen structure|) : |Finding site| = *)

Returning Attributes	Michael Lawley	**Proposal (by Michael) for discussion**
		
		• 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|
		
		  ○ Perhaps something like:
		    ■ ? R.type ? (<< 404684003 |Clinical finding|)
		  ○ This could be extended to, for example, return different values - e.g.
		    ■ ? |Simple map refset|.|maptarget| ? (^|Simple map refset| AND < |Fracture|)

Reverse Member Of	Michael Lawley	**Proposal for discussion**
		
		What refsets is a given concept (e.g. 421235005 |Structure of femur|) a member of?
		
		• Possible new notation for this:
		  ○ ^ . 421235005 |Structure of femur|
		  ○ ? X ? 421235005 |Structure of femur| = ^ X

Expression Templates	Peter G. Williams	**ON HOLD WAITING FROM IMPLEMENTATION FEEDBACK FROM INTERNAL TECH TEAM** - WIP version - https://confluence.ihtsdotools.org/display/WIPSTS/Template+Syntax+Specification

Expression Templates	Peter G. Williams	• **ON HOLD WAITING FROM IMPLEMENTATION FEEDBACK FROM INTERNAL TECH TEAM**
		• WIP version - https://confluence.ihtsdotools.org/display/WIPSTS/Template+Syntax+Specification
		  ■ Added a 'default' constraint to each replacement slot - e.g. default (72673000 |Bone structure (body structure)|)
		  ■ Enabling 'slot references' to be used within the value constraint of a replacement slot - e.g. [[+id (<< 123037004 |Body structure| MINUS << $findingSite2) @findingSite1]]
		  ■ Allowing repeating role groups to be referenced using an array - e.g. $rolegroup[1] or $rolegroup[!=SELF]
		  ■ Allow reference to 'SELF' in role group arrays
		  ■ Adding 'sameValue' and 'allOrNone' constraints to information slots - e.g. sameValue ($site), allOrNone ($occurrence)
		  ■ See changes in red here: 5.1. Normative Specification
		
		Examples:
		
		[[+id]]: [[1..*] @my_group sameValue(morphology)] { |Finding site| = [[+id (<<123037004 |Body structure (body structure)| MINUS << $site[! SELF]) @site]] , |Associated morphology| = [[+id @my_morphology]] }
		
		• Implementation feedback on draft updates to Expression Template Language syntax
		  ○ Use cases from the Quality Improvement Project:
		    ■ Multiple instances of the same role group, with some attributes the same and others different. Eg same morphology, potentially different finding sites.
		
		Note that QI Project is coming from a radically different use case. Instead of *filling* template slots, we're looking at existing content and asking "exactly *how* does this concept fail to comply to this template?"
		
		For discussion:
		
		[[0..1]] { [[0..1]] 246075003 |Causative agent| = [[+id (< 410607006 |Organism|) @Organism]] }
		
		Is it correct to say either one of the cardinality blocks is redundant? What are the implications of 1..1 on either side? This is less obvious for the self grouped case.
		
		Road Forward for SI
		
		1. Generate the parser from the ABNF and implement in the Template Service
		2. User Interface to a) allow users to specify template at runtime b) tabular (auto-completion) lookup STL
		3. Template Service to allow multiple templates to be specified for alignment check (aligns to none-off)
		4. Output must clearly indicate exactly what feature of concept caused misalignment, and what condition was not met.
		
		Additional note: QI project is no longer working in subhierarchies. Every 'set' of concepts is selected via ECL. In fact most reports should now move to this way of working since a subhierarchy is the trivial case. For a given template, we additionally specify the "domain" to which it should be applied via ECL. This is much more specific than using the focus concept which is usually the PPP eg Disease.
		
		FYI Michael Chu

Description Templates	Kai Kewley	• **ON HOLD**
		• Previous discussion (in Malaysia)
		  ■ Overview of current use
		  ■ Review of General rules for generating descriptions
		    • Removing tags, words
		    • Conditional removal of words
		    • Automatic case significance
		    • Generating PTs from target PTs
		    • Reordering terms
		  ■ Mechanism for sharing general rules - inheritance? include?
		  ■ Description Templates for translation
		  ■ Status of planned specification

Query Language - Summary from previous meetings	Linda Bird	**FUTURE WORK** **Examples: version and dialect** - << 64572001 |Disease| {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131 - << 64572001 |Disease| {{ synonym = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131 - << 64572001 |Disease| {{ FSN = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131 - << 64572001 |Disease| {{ FSN = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131, **DIALECT** W - << 64572001 |Disease| {{ preferredTerm = "*heart*"}} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131, **DIALECT** Y - << 64572001 |Disease| {{ acceptableTerm = "*heart*"}} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131, **DIALECT** Y - (* {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20180131, **DIALECT** Z) MINUS (* {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/900000000000207008/version/20170731, **DIALECT** W) - X MINUS Y **WHERE** X = * , Y = (* {{ term = "*heart*" }}) **VERSION** http://snomed.info/sct/900000000000207008/version/20180131, **DIALECT** W **Notes** - Allow nested where, version, language - Scope of variables is inner query
Confirm next meeting date /time	Linda Bird	Next meeting is scheduled for Wednesday 22nd April 2020 at 20:00 UTC.

File	Modified
ZIP Archive equiv.zip	2020-Jul-28 by Ed Cheetham