2021-10-07 - SLPG Meeting

Date & Time

20:00 to 21:00 UTC Thursday 7th October 2021

Goals

- Announce publication of ECL v1.6
- Progress ECL enhancements - access to historical refsets

Location

Zoom meeting link (password: 764978)

Attendees

- Chair: Linda Bird
- Project Group: Roger Jane, Michael Lawley, Daniel Karlsson, Alejandro Lopez Osornio, Kai Kewley, Rob Hausam

Agenda and Meeting Notes

Description	Owner	Notes
Welcome and agenda	Linda Bird	Welcome to new attendees

ECL v1.6 - Concept Filters	Linda Bird	• ECL v1.6 published (including subexpressions in filters)
		  ◦ For example
		  ▪ < 64572001 |Asthma| {{ C moduleId = << 19481000087107 | Canada Health Infoway maintained module | }}
		• URI standard for xst unpublished content

ECL v1.7 - Access to historical refsets	All	Proposal
		
		To leverage the existing memberOf (^) function, which currently only brings back the 'referencedComponentId', rather than the whole 'member' (i.e. row) as you might expect a 'memberOf' function to do. If we defined an expanded form of the current 'memberOf' function, which explicitly shows that by default we're selecting attributeOrder = 0 from the member, then the following 2 ECL would be equivalent.
		
		1. ^ 447562003 |ICD-10 complex map refset|
		2. ^ [referencedComponentId] 447562003 |ICD-10 complex map refset|
		
		This would allow us to select other columns of the refset using the same memberOf function - e.g.
		
		• ^ [mapTarget] 447562003 |ICD-10 complex map refset|
		
		And we could then introduce a 'Member filter' with a similar attribute reference - e.g.
		
		• ^ [referencedComponentId] 447562003 |ICD-10 complex map refset| {{ M mapTarget = "L56.2" }}
		
		Additional Examples
		
		• Find all the inactive concepts associated by **any** historical association to a subtype of 195967001 | Asthma (disorder) |
		  ◦ ^ < 900000000000522004 | Historical association refset| {{ M targetComponentId = < 195967001 |Asthma| }}
		• Find all the SNOMED CT concepts that map **to** ICD-10 code "L56.2"
		  ◦ ^ 447562003 |ICD-10 complex map refset| {{ M mapTarget = "L56.2" }}
		  ◦ ^ [referencedComponentId] 447562003 |ICD-10 complex map refset| {{ M mapTarget = "L56.2" }}
		• Find the SNOMED CT concept that maps **to** ICD-10 code "L56.2" with a mapRule = 1 and mapGroup = 1
		  ◦ ^ [referencedComponentId] 447562003 |ICD-10 complex map refset| {{ M mapGroup = #1, mapPriority = #1, mapTarget = "L56.2" }}
		• Find the reason that concept 67415000 |Hay asthma| was inactivated (answer: 900000000000482003 |Duplicate component|)
		  ◦ ^ [valueId] 900000000000489007 |Concept inactivation indicator attribute value refset| {{ M referencedComponentId = 67415000 |Hay asthma| }}
		• Find the active concept that is the 'same as' inactivated concept 67415000 |Hay asthma|
		  ◦ ^ [targetComponentId] 900000000000527005 |SAME AS association refset| {{ M referencedComponentId = 67415000 |Hay asthma| }}
		
		Questions

1. Should we allow the column headers (ie fields) from the file name and/or the refset attribute concept to be used and/or the attributeOrder?
   ANSWER: OPTION 2 only
   - **OPTION 2**: ^ < 900000000000522004 | Historical association refset| {{ M targetComponentId = < 195967001 |Asthma| }}
   - ^ [referencedComponentId] < 900000000000522004 | Historical association refset| {{ M targetComponentId = < 195967001 |Asthma| }
2. Should we allow a 'or self' operator, to make it more efficient to create historical subsets - e.g. the following expression constraint would return all the inactive concepts that were replaced by an equivalent subtype of |Asthma| PLUS all the subtypes of |Asthma| as well. This would be useful for finding all the matches that you may need to look for in a health record.
   - **FULL:** < 195967001 |Asthma| OR ^ 900000000000527005 |SAME AS association refset| {{ M targetComponentId = < 195967001 |Asthma| }}
   - **SYNTACTIC SUGAR OPTIONS**
     a. << 195967001|asthma| {{ +HISTORY-SAME }}
        i. NEW editorial rules - SAME_AS
        ii. e.g. use cases requiring very high confidence
     b. << 195967001|asthma| {{ +HISTORY }}
        /* Adds inactive referencedComponents with matching targetComponentId for ALL historical association refsets */
        i. NEW editorial rules - SAME_AS, REPLACED_BY (and WAS_A), PARTIALLY_EQUIVALENT_TO (only for conjunction - includes the parts)
     c. << 195967001|asthma| {{ +HISTORY-ALL }}
        i. NEW editorial rules - HISTORY + ALTERNATIVE, POSSIBLY_EQUIVALENT_TO (includes disjunction with ambiguity), POSSIBLY_REPLACED_BY
        ii. e.g. to identify patients for review of notes
   - **QUESTIONS for EAG subgroup**
     - How are the changes to the historical association rules being managed?
     - Will existing historical associations be updated to match the new editorial policy?
       - This is under discussion
     - How will transitive dependencies be managed moving forward?
       - GOOD NEWS! Inactive concepts will always point towards an active target (except for a few exceptions in the distant past)
   - OTHER
     a. << 195967001|asthma| {{ +HISTORY (< 900000000000522004 | Historical association refset|) }}
     b. << 195967001|asthma| {{ +HISTORY (90000000527005 |SAME AS association reference set| 900000000000523009 |POSSIBLY EQUIVALENT TO association reference set|) }} /* uses only the identified historical association refsets */
     c. << 195967001|asthma| {{ +HISTORY (SAME_AS POSSIBLY_EQUIVALENT_TO)
     d. << 195967001|asthma| {{ +HISTORY (SAM-A POSS-ET ALT MOV-F MOV-T REF-T REP-B SIM-T WAS-A PAR-E POSS-RB)
   - Possible abbreviations
     - ALTERNATIVE = **ALT**
     - PARTIALLY_EQUIVALENT_TO = **PAR-ET** /* doesn't currently exist but has been proposed */
     - ~~MOVED_FROM = **MOV-F** /* not useful but doesn't hurt? */~~
     - ~~MOVED_TO = **MOV-T** /* not useful but doesn't hurt? */~~
     - POSSIBLY_EQUIVALENT_TO = **POS-ET**
     - POSSIBLY_REPLACED_BY = **POS-RB** /* doesn't currently exist but has been proposed */
     - ~~**REFERS_TO = REF-T** /* for inactivated descriptions */~~
     - REPLACED_BY = **REP-B**
     - SAME_AS = **SAM-A**
     - ~~**SIMILAR_TO = SIM-T** /* not used */~~
     - WAS_A = **WAS-A** /* not used moving forward? */
   - Consider use cases defined by MAG
3. Should we allow more than one column in a refset to be selected? What should the rules be about using this inside a subExpressionConstraint? e.g.
   - ^ [*] 447562003 |ICD-10 complex map refset|
   - ^ [referencedComponentId, mapTarget] 447562003 |ICD-10 complex map refset|
4. Should we allow filters to be nested inside other filters? e.g.
   - ^ 434532547 {{ M annotation = (* {{ D term = "chronic" }}) }}
   - ^ [annotation] 434532547 {{ M referencedComponent = (<< 4756478567 {{ C moduleId = 546576478 }}) }}

Reference set descriptor templates (for easy reference)

		• 900000000000521006 |Association type reference set|
		  ◦ 0 - referencedComponentId (900000000000532006 |Association source component|)
		  ◦ 1 - targetComponentId (900000000000533001 |Association target component|)
		• 900000000000523009 |POSSIBLY EQUIVALENT TO association reference set|
		  ◦ 0 - referencedComponentId (900000000000532006 |Association source component|)
		  ◦ 1 - targetComponentId (900000000000533001 |Association target component|)
		• 900000000000480006 |Attribute value type reference set| - e.g.
		  ◦ 0 - referencedComponentId (449608002 |Referenced component|)
		  ◦ 1 - valueId (900000000000491004 |Attribute value|)
		• 900000000000489007 |Concept inactivation indicator attribute value reference set|
		  ◦ 0 - referencedComponentId (449608002 |Referenced component|)
		  ◦ 1 - valueId (900000000000481005 |Concept inactivation value|)
		• 900000000000496009 |Simple map reference set|
		  ◦ 0 - referencedComponentId (900000000000500006 | Map source concept |)
		  ◦ 1 - mapTarget (900000000000499002 | Scheme value |)
		• Proposed updates to simple maps:
		  ◦ 900000000000496009 |Simple map from SNOMED CT reference set|
		  ▪ 0 - referencedComponentId (900000000000500006 | Map source |)
		  ▪ 1 - mapTarget (900000000000505001 | Map target |)
		  ◦ |Simple map to SNOMED CT reference set|
		  ▪ 0 - referencedComponentId (900000000000505001 | Map target |)
		  ▪ 1 - mapSource (900000000000500006 | Map source |)
		
		See discussion below "Querying Refset Attributes"

onlyThe items below are currently on hold		

Postcoordination Topics		• Discuss feedback on transformation implementation
		  ◦ Resources
		  ▪ Expression transformation service generates the classifiable form and the necessary normal form from a close to user form expression
		  ▪ Contribute your expression examples and write your feedback for consideration and discussion
		  ◦ Recap of SNOMED on FHIR discussions
		  ▪ What is the functionality scope of a terminology server that supports postcoordination? For example, does it include:
		  • Classifying multiple expressions in a single substrate? What are the use cases for this?
		  • Assigning (local) identifiers to expressions? What are the use cases for this?
		  • Autogenerating or assigning a term to an expression? What are the use cases for this?
		  ▪ Does a terminology server that supports postcoordination, include all the functions of an expression repository?
		  ▪ What is the relationship between a terminology server that supports postcoordination, and an expression repository?
		  ◦ Outstanding questions
		  ▪ What are the pros and cons of extending SCG to allow an expression as the focus of a postcoordinated expression?
		  • Note: This was raised in context of a NNF generated over a postcoordinated substrate, where the proximal parent is an expression
		  ◦ ▪ Example of using expressions in focus concept
		  • **(** 125605004 |Fracture of bone|:363698007 |finding site| = 84167007 |Foot bone| **)** : 272741003 |Laterality| = 7771000 |Left|
		  • 125605004 |Fracture of bone|:363698007 |finding site| = 84167007 |Foot bone| **,** 272741003 |Laterality| = 7771000 |Left|
		  ▪ What is the expected NNF when classifying an expression that is equivalent to a precoordinated concept? For example:
		  • **Expression that is equivalent to 111273006 |Acute respiratory disease|**
		  • 64572001 | Disease (disorder) | :
		  {263502005 |Clinical course (attribute)| = 424124008 |Sudden onset AND/OR short duration (qualifier value)|}
		  {363698007 |Finding site (attribute)| = 89187006 |Airway structure (body structure)|}
		  • Options:
		  1. 111273006 |Acute respiratory disease| :
		  {263502005 |Clinical course| = 424124008 |Sudden onset AND/OR short duration|}
		  {363698007 |Finding site| = 89187006 |Airway structure|}
		  2. 50043002 |Disorder of respiratory system (disorder)| +
		  2704003 |Acute disease (disorder)| :
		  {263502005 |Clinical course| = 424124008 |Sudden onset AND/OR short duration|}
		  {363698007 |Finding site| = 89187006 |Airway structure|}
		  3. Other?
		  ◦ Recap of internal discussions with Content Team
		  ▪ Inter-attribute dependencies
		  ▪ Grouping rules

Dynamic Templates		• Continue discussion on dynamic templates
		  ◦ Inter-attribute dependencies
		  ▪ **Acute/Chronic and Inflammation -** Adding a clinical course requires specializing the inflammation morphology
		  • E.g. |Pyelonephritis| : |Clinical course| = |Chronic|
		  should be
		  |Pyelonephritis| : |Clinical course| = |Chronic|, |Associated morphology| = |Chronic inflammation|
		  • E.g. |Pyelonephritis| : |Clinical course| = |Sudden onset AND/OR short duration|
		  should be
		  |Pyelonephritis| : {|Clinical course| = |Sudden onset AND/OR short duration||, |Associated morphology| = |Acute inflammation|
		  ▪ **Infectious Causative Agents -** Adding a |causative agent| = |Domain Bacteria| or |Virus| requires adding a |Pathological process| = |Infectious process|
		  • E.g. |Nephritis|: |Causative agent| = |Domain bacteria|
		  should be
		  |Nephritis|: |Causative agent| = |Domain bacteria|, |Pathological process| = |Infectious process|
		  ▪ **Congenital and Acquired -** Adding an |Occurrence| of |Congenital| to a focus concept with an abnormal morphology, requires adding a |Pathological process| of |Pathological development process|
		  • E.g. |Koilonychia|: |Occurrence| = |Congenital|
		  should be
		  |Koilonychia|: |Occurrence| = |Congenital|, |Pathological process| = |Pathological developmental process|
		  ▪ **Situations with Explicit Context**
		  1. if the procedure context = |Planned|, then the temporal context should be << |Current of specified time|
		  a. If the procedure context = |In progress|, then the temporal context should be << |Current|
		  b. If the procedure context = |Performed| or |Done|, then the temporal context should be << |Current or past (actual)|
		  ▪ Note: for this use case (of |Procedure with explicit context|) perhaps we just recommend (or require) that the full role group is spelled out.
		  ▪ Next steps
		  • Representation of the content rules
		  ◦ Who creates the complete list of rules and how?
		  ▪ What formalism?
		  ▪ Determine which are mandatory and which are optional
		  ◦ Implementation of content rules - e.g.
		  ▪ Guided data entry by pre-populating role groups in expression template based on definition of focus concepts (for design-time use, such as mapping)
		  ▪ Mandatory content rules could be added to transform process

Postcoordination Use Case Examples	All	Example 1 - Dentistry / Odontogram • Requires an expression template to create expressions. • Resulting expression still requires a transformation to make it classifiable Example 2 - Terminology binding • Uses a fixed expression template to combine codes entered into separate fields • The procedure+laterality example still requires a transformation to make it classifiable Example 3 - Mapping • Design-time activity • Map targets may not be able to be fully represented using concept model attributes • In many cases, an extension (with primitive concepts) should be recommended where there are gaps in the mapping • There may be some cases in which postcoordination is helpful (e.g. LOINC to SNOMED CT map) Example 4 - Natural Language Processing • Usually run-time activity. • May require manual confirmation of coding suggestions (unless low clinical risk, eg for suggesting relevant patient records for manual review)
Postcoordination Guidance	Linda Bird , Anne Randorff Højen , Kai Kewley	Practical Guide to Postcoordination

- **Proposal - Use syntax (i.e. braces) to distinguish refinement vs new role group**
- - There should be a syntactic distinction between refinement and constructive addition (ie adding a new role group). That is:
    1. 83152002 |Oophorectomy| : **405815000 |Procedure device| = 122456005 |Laser device|**
       - is classified as (i.e. the refinement is added to the role groups in the definition of the focus concept(s)):
         - 83152002 |Oophorectomy| :
           { 260686004 |Method| = 129304002 |Excision - action|,
           405813007 |Procedure site - Direct| = 15497006 |Ovarian structure|,
           405815000 |Procedure device| = 122456005 |Laser device| }
    2. 83152002 |Oophorectomy| : **{ 405815000 |Procedure device| = 122456005 |Laser device| }**
       - is classified as:
         - 83152002 |Oophorectomy| :
           { 260686004 |Method| = 129304002 |Excision - action|,
           405813007 |Procedure site - Direct| = 15497006 |Ovarian structure|},
           { 405815000 |Procedure device| = 122456005 |Laser device| }
  - However, for attributes which are **always** self-grouped - i.e. Priority, Due to, After, Before, During, Clinical course, Temporally related to, and all Observable entity attributes (see Relationship Group), these must always be put into their own role group:
  - 1. 125605004 | Fracture of bone |: 42752001 | Due to (attribute) | = 1912002 | Fall |
       - is classified as:
         - 125605004 | Fracture of bone |: { 42752001 | Due to (attribute) | = 1912002 | Fall | }
         - or
         - 125605004 | Fracture of bone |:
           { 363698007 |Finding site| = 272673000 |Bone structure|,
           116676008 |Associated morphology | = 72704001 |Fracture| }
           { 42752001 | Due to (attribute) | = 1912002 | Fall | }
- **Proposal: Expression forms needed for this** (see 3.4 Transforming Expressions)
  - Close to user form - e.g. 83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 | Laser device|
  - Canonical close to user form - e.g. 83152002:405815000=122456005
  - Classifiable form (SCG) - e.g. 83152002:{260686004=129304002,405813007=15497006,405815000=122456005}
    - PLUS Classifiable form (OWL) - e.g.
      - EquivalentClasses(:123063
        ObjectIntersectionOf (:71388002
        ObjectSomeValuesFrom(:609096000 ObjectIntersectionOf(ObjectSomeValuesFrom(:260686004 :129304002)
        ObjectSomeValuesFrom(:405813007 :15497006))))
  - Necessary normal form - e.g. 83152002+416376001:{260686004=129304002,405813007=15497006,405815000=122456005}
    - PLUS Necessary normal form (tables)
      - Relationships:
        - (123063 116680003 83152002) - 0
        - (123063 260686004 129304002) - 0
        - (123063 405813007 15497006) - 1
        - (123063 405815000 122456005) - 1
  - Primitive expressions - "<<<" (only useful in a mapping context) .... relies on the assigned identifier (which are necessarily semantically unique).

- **Proposed Transformation Rules - Refinements (in valid domain of focus concepts)**
  Close-to-user-form - IF the grouping of the refinement is **not** concept model valid THEN
  If there is a single (non-self-grouped) role group in the definition of the focus concept, then any ungrouped (but groupable) refinements are merged with this role group
  If there is more than one (non-self-grouped) role group in the definition then flag as ambiguous and require refinement
  NEED TO FIND a realistic clinical example where this may occur // Prevent failing cases from coming up // use template
  ALTERNATIVE: Refinement is applied to all (non-self-grouped) role groups in the definition
  Self-grouped attributes in the refinement are grouped on their own - i.e. Priority, Due to, After, Before, During, Clinical course, Temporally related to, and all Observable entity attributes (see Relationship Group)
  Self-grouped attributes in the definition of the focus concept(s) are left unchanged

- 1. **Single refinement**
  83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|
  - 83152002 |Oophorectomy| :
    { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure|,
    405815000 |Procedure device| = 122456005 |Laser device| }
- 2. **Two groupable refinements**
  83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|, 363700003 |Direct morphology| = 367643001 |Cyst |
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure|,
    405815000 |Procedure device| = 122456005 |Laser device| ,
    363700003 |Direct morphology| = 367643001 |Cyst | }
- 3. **One groupable refinement with one self-grouped refinement**
  83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|, 260870009 |Priority| = 394849002 |High priority|
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure|,
    405815000 |Procedure device| = 122456005 |Laser device| }
    { 260870009 |Priority (attribute)| = 394849002 |High priority| }
- 4. **Refinement attribute matches (or subsumed by) attribute in focus concept's definition**
  83152002 |Oophorectomy| : 260686004 |Method| = 277261002 |Excision biopsy (qualifier value)|
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
  - 260686004 |Method| = 277261002 |Excision biopsy (qualifier value)| ,
  - 405813007 |Procedure site - direct| = 15497006 |Ovarian structure| }
- 5. **Refinement explicitly in role group**
  83152002 |Oophorectomy| : { 260686004 |Method| = 281615006 | Exploration - action | , 405813007 |Procedure site - direct| = 367643001 |Cyst | }
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure| },
    { 260686004 |Method| = 281615006 | Exploration - action | ,
    405813007 |Procedure site - direct| = 367643001 |Cyst |}

- **Proposed Transformation Rules - Refinements (NOT in valid domain of focus concepts)**
  Close-to-user-form - IF the refinement's attribute is **not** valid for the domain of the focus concept THEN
  If there is a single role group in the definition of the focus concept, which has an attribute value in the domain of the refinement's attribute THEN nest the relevant attribute value with the refinement added to the attribute value
  (Note: It doesn't matter if the role group is self-grouped or not (see example 1 below)
  If there is more than one role group in the definition of the focus concept, which has an attribute value in the domain of the refinement's attribute THEN (non-self-grouped) role group in the definition then flag as ambiguous and require refinement

- 1. **Left aural temperature**
  - 415974002 |Aural temperature|: 272741003 |Laterality| = 7771000 |Left|
  - 415974002 |Aural temperature|: {704327008 |Direct site| = (42859004 |Ear drum|: 272741003 |Laterality| = 7771000 |Left|)}
- 2. **Malignant tumor of right ovary**
  - 363443007 |Malignant tumor of ovary|: 272741003 |Laterality| = 24028007 |Right|
  - 363443007 |Malignant tumor of ovary|:
  - { 116676008 |Associated morphology| = 367651003 |Malignant neoplasm of primary, secondary or uncertain origin|,
    363698007 |Finding site| = (15497008 |Ovarian structure| : 272741003 |Laterality| = 24028007 |Right|) }

- **Other Example - Emergency excision of appendix**
  1. 80146002 | Excision of appendix | :
     260870009 |Priority| = 25876001 |Emergency|

- **Other Example - Fracture of bone**

- 1. 125605004 |Fracture of bone|: 363698007 |finding site| = 84167007 |Foot bone|
- 2. 125605004 |Fracture of bone|: {363698007 |finding site| = 84167007 |Foot bone| }
- 3. 125605004 |Fracture of bone|: {116676008 |Associated morphology| = 72704001 |Fracture|, 363698007 |finding site| = 84167007 |Foot bone| }
- 4. 64572001 |Disease|: {116676008 |Associated morphology| = 72704001 |Fracture|, 363698007 |finding site| = 84167007 |Foot bone| }

The items below are currently on hold

Other Options for Future Progress		1. URIs for draft editions
		2. ECL extensions
		  a. Primitive/Defined filters concept filter
		  b. Concept+Description filters (e.g. effectiveTime, module, active)
		  c. Accessing Refset attributes (e.g. **historical association refsets**) **historical ECL**
		  d. OR use full syntax to be able to query any table (e.g. Relationship table) - ie expand ECL into something more verbose (e.g. SNOMED query language)
		3. Template extensions

URIs for Extended Editions		**ON HOLD -** How to refer to an 'extended edition' using a URI - e.g. "International Edition plus the following 2 nursing modules: 733983009 |IHTSDO Nursing Health Issues module|and 733984003 |IHTSDO Nursing Activities module| Use Case - Need to execute an ECL, that refers to "^ 733991000 | Nursing Health Issues Reference Set (foundation metadata concept) |" and/or "^ 733990004 | Nursing Activities Reference Set (foundation metadata concept) |", where the substrate includes the international edition, plus the modules that include these reference sets July 2020 International Edition URI: http://snomed.info/sct/900000000000207008/version/20200731 July 2020 International Edition + nursing modules URI ?? - For example: • http://snomed.info/sct/900000000000207008/version/20200731/module/733983009/time/20200131/module/733984003/time/20200131 • http://snomed.info/sct/900000000000207008/version/20200731/modules/733983009:733984003 • http://snomed.info/sct/900000000000207008:733983009:733984003/version/20200731:20190731:20200131 • Canonical order? Or order doesn't matter? • Constraints on what can go in the additional packages (only refsets and their metadata)

Querying Refset Attributes	Linda Bird	**ON HOLD -** 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	**ON HOLD - 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	**ON HOLD - 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
		  ▪ 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/90000000000207008/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/90000000000207008/version/20180131, **DIALECT** Y ○ << 64572001 |Disease| {{ acceptableTerm = "*heart*"}} **VERSION** http://snomed.info/sct/90000000000207008/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/90000000000207008/version/20180131, **DIALECT** W **Notes** ○ Allow nested where, version, language ○ Scope of variables is inner query

File	Modified

No files shared here yet.

Agenda and Meeting Notes

Description	Owner	Notes
Welcome and agenda	Linda Bird	

Postcoordination Transformations		• Discuss feedback on transformation implementation
		  ◦ Recap from last meeting
		  ▪ Agreed to always use "===" at the start of the classifiable form (to indicate that it is ready for classification.
		  ▪ Try the expression transformation service to see the current transformation result and retrieve the classifiable form and the necessary normal form
		  ▪ Contribute your expression examples and write your feedback for consideration and discussion
		  ▪ Discussed the NNF output for expressions that are equivalent to a precoordinated concept - Did we reach a decision?
		  ◦ Questions for discussion
		  ▪ What is the expected NNF when classifying an expression that is equivalent to a precoordinated concept? For example:
		  • **Expression that is equivalent to 111273006 |Acute respiratory disease|**
		  • 64572001 | Disease (disorder) | :
		  {
		  263502005 |Clinical course (attribute)| = 424124008 |Sudden onset AND/OR short duration (qualifier value)|
		  }{
		  363698007 |Finding site (attribute)| = 89187006 |Airway structure (body structure)|
		  }
		  • Options:
		  1. 111273006 |Acute respiratory disease| :
		  {
		  263502005 |Clinical course| = 424124008 |Sudden onset AND/OR short duration|
		  }{
		  363698007 |Finding site| = 89187006 |Airway structure|
		  }
		  2. 50043002 |Disorder of respiratory system (disorder)| +
		  2704003 |Acute disease (disorder)| :
		  {
		  263502005 |Clinical course| = 424124008 |Sudden onset AND/OR short duration|
		  }{
		  363698007 |Finding site| = 89187006 |Airway structure|
		  }
		  3. Other?
		  ▪ What is the functionality scope of a terminology server that supports postcoordination?
		  • What are the use cases for classifying multiple expressions into a single substrate?
		  • What are the use cases for assigning (local) identifiers to expressions?
		  • Are these functionalities supported by the terminology server and/or an expression repository?
		  • What is the relationship between the terminology server and the expression repository?
		  • What are the use cases in which a term needs to be autogenerated for an expression?

Dynamic Templates		• Continue discussion on dynamic templates
		  ◦ Inter-attribute dependencies
		  ▪ **Acute/Chronic and Inflammation -** Adding a clinical course requires specializing the inflammation morphology
		  • E.g. |Pyelonephritis| : |Clinical course| = |Chronic|
		  should be
		  |Pyelonephritis| : |Clinical course| = |Chronic|, |Associated morphology| = |Chronic inflammation|
		  • E.g. |Pyelonephritis| : |Clinical course| = |Sudden onset AND/OR short duration|
		  should be
		  |Pyelonephritis| : {|Clinical course| = |Sudden onset AND/OR short duration||, |Associated morphology| = |Acute inflammation|
		  ▪ **Infectious Causative Agents -** Adding a |causative agent| = |Domain Bacteria| or |Virus| requires adding a |Pathological process| = |Infectious process|
		  • E.g. |Nephritis|: |Causative agent| = |Domain bacteria|
		  should be
		  |Nephritis|: |Causative agent| = |Domain bacteria|, |Pathological process| = |Infectious process|
		  ▪ **Congenital and Acquired -** Adding an |Occurrence| of |Congenital| to a focus concept with an abnormal morphology, requires adding a |Pathological process| of |Pathological development process|
		  • E.g. |Koilonychia|: |Occurrence| = |Congenital|
		  should be
		  |Koilonychia|: |Occurrence| = |Congenital|, |Pathological process| = |Pathological developmental process|
		  ▪ **Situations with Explicit Context**
		  1. if the procedure context = |Planned|, then the temporal context should be << |Current of specified time|
		  a. If the procedure context = |In progress|, then the temporal context should be << |Current|
		  b. If the procedure context = |Performed| or |Done|, then the temporal context should be << |Current or past (actual)|
		  ▪ Note: for this use case (of |Procedure with explicit context|) perhaps we just recommend (or require) that the full role group is spelled out.
		  ▪ Next steps
		  • Representation of the content rules
		  ◦ Who creates the complete list of rules and how?
		  ▪ What formalism?
		  ▪ Determine which are mandatory and which are optional
		  ◦ Implementation of content rules - e.g.
		  ▪ Guided data entry by pre-populating role groups in expression template based on definition of focus concepts (for design-time use, such as mapping)
		  ▪ Mandatory content rules could be added to transform process

Postcoordination Use Case Examples	All	Example 1 - Dentistry / Odontogram • Requires an expression template to create expressions. • Resulting expression still requires a transformation to make it classifiable Example 2 - Terminology binding • Uses a fixed expression template to combine codes entered into separate fields • The procedure+laterality example still requires a transformation to make it classifiable Example 3 - Mapping • Design-time activity • Map targets may not be able to be fully represented using concept model attributes • In many cases, an extension (with primitive concepts) should be recommended where there are gaps in the mapping • There may be some cases in which postcoordination is helpful (e.g. LOINC to SNOMED CT map) Example 4 - Natural Language Processing • Usually run-time activity. • May require manual confirmation of coding suggestions (unless low clinical risk, eg for suggesting relevant patient records for manual review)
Postcoordination Guidance	Linda Bird , Anne Randorff Højen , Kai Kewley	Practical Guide to Postcoordination

- **Proposal - Use syntax (i.e. braces) to distinguish refinement vs new role group**
- - There should be a syntactic distinction between refinement and constructive addition (ie adding a new role group). That is:
    1. 83152002 |Oophorectomy| : **405815000 |Procedure device| = 122456005 |Laser device|**
       - is classified as (i.e. the refinement is added to the role groups in the definition of the focus concept(s)):
         - 83152002 |Oophorectomy| :
           { 260686004 |Method| = 129304002 |Excision - action|,
           405813007 |Procedure site - Direct| = 15497006 |Ovarian structure|,
           405815000 |Procedure device| = 122456005 |Laser device| }
    2. 83152002 |Oophorectomy| : **{ 405815000 |Procedure device| = 122456005 |Laser device| }**
       - is classified as:
         - 83152002 |Oophorectomy| :
           { 260686004 |Method| = 129304002 |Excision - action|,
           405813007 |Procedure site - Direct| = 15497006 |Ovarian structure|},
           { 405815000 |Procedure device| = 122456005 |Laser device| }
  - However, for attributes which are **always** self-grouped - i.e. Priority, Due to, After, Before, During, Clinical course, Temporally related to, and all Observable entity attributes (see Relationship Group), these must always be put into their own role group:
  - 1. 125605004 | Fracture of bone |: 42752001 | Due to (attribute) | = 1912002 | Fall |
       - is classified as:
         - 125605004 | Fracture of bone |: { 42752001 | Due to (attribute) | = 1912002 | Fall | }
         - or
         - 125605004 | Fracture of bone |:
           { 363698007 |Finding site| = 272673000 |Bone structure|,
           116676008 |Associated morphology | = 72704001 |Fracture| }
           { 42752001 | Due to (attribute) | = 1912002 | Fall | }
- **Proposal: Expression forms needed for this** (see 3.4 Transforming Expressions)
  - Close to user form - e.g. 83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|
  - Canonical close to user form - e.g. 83152002:405815000=122456005
  - Classifiable form (SCG) - e.g. 83152002:{260686004=129304002,405813007=15497006,405815000=122456005}
    - PLUS Classifiable form (OWL) - e.g.
      - EquivalentClasses(:123063
        ObjectIntersectionOf (:71388002
        ObjectSomeValuesFrom(:609096000 ObjectIntersectionOf(ObjectSomeValuesFrom (:260686004 :129304002)
        ObjectSomeValuesFrom(:405813007 :15497006))))
  - Necessary normal form - e.g. 83152002+416376001:{260686004=129304002,405813007=15497006,405815000=122456005}
    - PLUS Necessary normal form (tables)
      - Relationships:
        - (123063 116680003 83152002) - 0
        - (123063 260686004 129304002) - 0
        - (123063 405813007 15497006) - 1
        - (123063 405815000 122456005) - 1
  - Primitive expressions - "<<<" (only useful in a mapping context) .... relies on the assigned identifier (which are necessarily semantically unique).

- **Proposed Transformation Rules - Refinements (in valid domain of focus concepts)**
  Close-to-user-form - IF the grouping of the refinement is **not** concept model valid THEN
  If there is a single (non-self-grouped) role group in the definition of the focus concept, then any ungrouped (but groupable) refinements are merged with this role group
  If there is more than one (non-self-grouped) role group in the definition then flag as ambiguous and require refinement
  NEED TO FIND a realistic clinical example where this may occur // Prevent failing cases from coming up // use template
  ALTERNATIVE: Refinement is applied to all (non-self-grouped) role groups in the definition
  Self-grouped attributes in the refinement are grouped on their own - i.e. Priority, Due to, After, Before, During, Clinical course, Temporally related to, and all Observable entity attributes (see Relationship Group)
  Self-grouped attributes in the definition of the focus concept(s) are left unchanged

- 1. **Single refinement**
  83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|
  - 83152002 |Oophorectomy| :
    { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure|,
    405815000 |Procedure device| = 122456005 |Laser device| }
- 2. **Two groupable refinements**
  83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|, 363700003 |Direct morphology| = 367643001 |Cyst |
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure|,
    405815000 |Procedure device| = 122456005 |Laser device| ,
    363700003 |Direct morphology| = 367643001 |Cyst | }
- 3. **One groupable refinement with one self-grouped refinement**
  83152002 |Oophorectomy| : 405815000 |Procedure device| = 122456005 |Laser device|, 260870009 |Priority| = 394849002 |High priority|
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure|,
    405815000 |Procedure device| = 122456005 |Laser device| }
    { 260870009 |Priority (attribute)| = 394849002 |High priority| }
- 4. **Refinement attribute matches (or subsumed by) attribute in focus concept's definition**
  83152002 |Oophorectomy| : 260686004 |Method| = 277261002 |Excision biopsy (qualifier value)|
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
  - 260686004 |Method| = 277261002 |Excision biopsy (qualifier value)| ,
  - 405813007 |Procedure site - direct| = 15497006 |Ovarian structure| }
- 5. **Refinement explicitly in role group**
  83152002 |Oophorectomy| : { 260686004 |Method| = 281615006 | Exploration - action | , 405813007 |Procedure site - direct| = 367643001 |Cyst | }
  - 83152002 |Oophorectomy| :
  - { 260686004 |Method| = 129304002 |Excision - action| ,
    405813007 |Procedure site - direct| = 15497006 |Ovarian structure| },
    { 260686004 |Method| = 281615006 | Exploration - action | ,
    405813007 |Procedure site - direct| = 367643001 |Cyst |}

- **Proposed Transformation Rules - Refinements (NOT in valid domain of focus concepts)**
  Close-to-user-form - IF the refinement's attribute is **not** valid for the domain of the focus concept THEN
  If there is a single role group in the definition of the focus concept, which has an attribute value in the domain of the refinement's attribute THEN nest the relevant attribute value with the refinement added to the attribute value
  (Note: It doesn't matter if the role group is self-grouped or not (see example 1 below)
  If there is more than one role group in the definition of the focus concept, which has an attribute value in the domain of the refinement's attribute THEN (non-self-grouped) role group in the definition then flag as ambiguous and require refinement

- 1. **Left aural temperature**
  - 415974002 |Aural temperature|: 272741003 |Laterality| = 7771000 |Left|
  - 415974002 |Aural temperature|: {704327008 |Direct site| = (42859004 |Ear drum|: 272741003 |Laterality| = 7771000 |Left|)}
- 2. **Malignant tumor of right ovary**
  - 363443007 |Malignant tumor of ovary|: 272741003 |Laterality| = 24028007 |Right|
  - 363443007 |Malignant tumor of ovary|:
  - { 116676008 |Associated morphology| = 367651003 |Malignant neoplasm of primary, secondary or uncertain origin|,
    363698007 |Finding site| = (15497008 |Ovarian structure| : 272741003 |Laterality| = 24028007 |Right|) }

- **Other Example - Emergency excision of appendix**
  1. 80146002 | Excision of appendix | :
     260870009 |Priority| = 25876001 |Emergency|

- **Other Example - Fracture of bone**

- 1. 125605004 |Fracture of bone|: 363698007 |finding site| = 84167007 |Foot bone|
- 2. 125605004 |Fracture of bone|: {363698007 |finding site| = 84167007 |Foot bone| }
- 3. 125605004 |Fracture of bone|: {116676008 |Associated morphology| = 72704001 |Fracture|, 363698007 |finding site| = 84167007 |Foot bone| }
- 4. 64572001 |Disease|: {116676008 |Associated morphology| = 72704001 |Fracture|, 363698007 |finding site| = 84167007 |Foot bone| }

The items below are currently on hold

Other Options for Future Progress		1. ECL extensions 2. a. Primitive/Defined filters concept filter b. Concept+Description filters (e.g. effectiveTime, module, active) c. Accessing Refset attributes (e.g. **historical association refsets**) **historical ECL** d. OR use full syntax to be able to query any table (e.g. Relationship table) - ie expand ECL into something more verbose (e.g. SNOMED query language) 3. Template extensions
URIs for Extended Editions		**ON HOLD -** How to refer to an 'extended edition' using a URI - e.g. "International Edition plus the following 2 nursing modules: 733983009 |IHTSDO Nursing Health Issues module|and 733984003 |IHTSDO Nursing Activities module| Use Case - Need to execute an ECL, that refers to "^ 733991000 | Nursing Health Issues Reference Set (foundation metadata concept) |" and/or "^ 733990004 | Nursing Activities Reference Set (foundation metadata concept) |", where the substrate includes the international edition, plus the modules that include these reference sets July 2020 International Edition URI: http://snomed.info/sct/900000000000207008/version/20200731 July 2020 International Edition + nursing modules URI ?? - For example: • http://snomed.info/sct/900000000000207008/version/20200731/module/733983009/time/20200131/module/733984003/time/20200131 • http://snomed.info/sct/900000000000207008/version/20200731/modules/733983009:733984003 • http://snomed.info/sct/900000000000207008:733983009:733984003/version/20200731:20190731:20200131 • Canonical order? Or order doesn't matter? • Constraints on what can go in the additional packages (only refsets and their metadata)
Querying Refset Attributes	Linda Bird	**ON HOLD -** 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	**ON HOLD - 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	**ON HOLD - 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

File **Modified**

No files shared here yet.