2021-03-25 - SLPG Meeting

Date & Time

20:00 to 21:00 UTC Thursday 25th March 2021

Goals

- Consider the processes required for data entry, display, query and exchange of a postcoordinated expression

Location

Zoom meeting link (password: 764978)

Attendees

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

Apologies

- Peter Jordan

Agenda and Meeting Notes

Description	Owner	Notes
Welcome and agenda	Linda Bird	

Postcoordination Expression Processes	All	Postcoordinated expression processes
		• Generic
		• Use case specific
		  ◦ Mapping (from an interface term)
		  ◦ Terminology binding (using an expression template)
		  ◦ Natural Language processing

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/90000000000207008/version/20180131, **DIALECT** Z) MINUS (* {{ term = "*heart*" }} **VERSION** http://snomed.info/sct/90000000000207008/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
File pyelo.svg	2021-Mar-26 by Ed Cheetham
JPEG File pyelo.jpg	2021-Mar-26 by Ed Cheetham
JPEG File nephritis.jpg	2021-Mar-26 by Ed Cheetham
JPEG File koilo.jpg	2021-Mar-26 by Ed Cheetham
PNG File koilo.png	2021-Mar-26 by Ed Cheetham
JPEG File CTHip.jpg	2021-Mar-27 by Ed Cheetham

Download All