

SNOMED CT, post coordination and ValueSet \$expand

Purpose

This page is intended to capture the majority position on the issue of the behaviour of the \$expand operation relative to SNOMED CT editions and post-coordination - a topic that has been greatly discussed.

The intent of the page is to capture this position as a distillation of the discussion for use in refining the FHIR specification in HL7 processes.

The problem

When expanding a ValueSet, the current default is to include post-coordinated content (unless the parameter "excludePostCoordinated" is included in the request and set to true). Clarification resulted in determining that, as currently specified, the correct response to such a request is to return all pre-coordinated and all possible post-coordinated concepts matching the ValueSet definition - that is all possible calculable post-coordinated content given the pre-coordinated content in the edition, post-coordinated grammar, and concept model.

However this is very difficult for servers to calculate (both technically difficult to implement and computationally costly) and rarely what a client is likely to want. Most current server implementations also currently do not calculate and respond with post-coordinated content due to the difficulty and relative lack of utility or client demand, even when the client does not specify "excludePostCoordinated". As a result they incorrectly respond with an expansion based on pre-coordinated content only without declaring it a SUBSETTED response.

Were servers to be updated to implement the specification as it currently stands, it is unlikely that any would implement such that they respond with all possible post-coordinated content matching the ValueSet. Therefore most, if not all, servers would return a SUBSETTED response, or "too costly" to a request where the client omits "excludePostCoordinated" (currently the most common case).

While a "too costly" response in this very common case is not useful to the client, a SUBSETTED response is useful. However as this would almost always be the case it does not allow the client to distinguish whether the expansion was generated from a fragment of SNOMED CT pre-coordinated content, or whether it was generated from all the pre-coordinated content but does not include all possible post-coordinated expressions.

All of these issues lead to varied behaviour of servers and a potential lack of portability of client code to different servers, as responses may vary.

Recommendations

That an expansion of a ValueSet where excludePostCoordinated is omitted or false is not a SUBSETTED response if it contains

1. all matching pre-coordinated content from the version of SNOMED CT it is expanded against
2. all the explicitly listed post-coordinated expressions in the ValueSet

That is, a ValueSet expansion result against SNOMED CT only responds with a SUBSETTED response if the response was generated from a fragment/subset of the pre-coordinated content of the SNOMED CT version it was expanded against, and/or post-coordinated expressions explicitly listed in the ValueSet were not included.

Unless the client specifies excludePostCoordinated, it can assume that the response may contain some (but unlikely all) of the post-coordinated expressions that are syntactically and semantically valid given the SNOMED CT pre-coordinated content from the version being expanded, post coordination grammar and the concept model.

An additional, related, recommendation is that for code systems with similarly unbounded or very large possible combinations given post-coordination grammars (e.g. UCUM and MIME) libraries of well known post-coordinated codes be defined, centralised and version managed; and that similarly expansion responses not be SUBSETTED provided they respond with all matching pre-coordinated content from the expanded version of the code system and matching expressions from the expression library.

Rationale

The recommendations are intended to

- define consistent, predictable behaviour that server implementers can create and clients can depend upon
- enable the client to determine from the server response whether all pre-coordinated (statically published) content from the code system version being expanded and all expressions explicitly included in the ValueSet are included or not

Richer responses to describe what the nature of the SUBSETTED response is were discussed however felt infeasible due to the wide variety of possible inclusions and exclusions of calculated post-coordination that could occur.

The behaviour of \$validate-code and code:in should remain unchanged as defaults match most likely expected behaviour - post-coordinated expressions tested against \$validate-code or code:in should return true if they meet the ValueSet definition. Unlike the case of \$expand, these cases involve the client specifying a finite set of post-coordinated expressions to the server and are hence considered tractable, unlike the case of \$expand that requires calculating an unbounded or extremely large set of expressions most of which are unlikely to be used.