2.1 URIs for Editions and Versions

Background

A SNOMED CT edition logically consists of the complete set of members of one or more modules.[1] Since the module dependency reference set (MDRS) tracks the explicit dependencies between a version of a module and all the versioned modules it depends on, a module identifier[2] is a natural identifier for an edition. When combined with a timestamp corresponding to a sourceEffectiveTime appearing in the MDRS, the module identifier can unambiguously identify a version of an edition.

Form

The URIs that identify unversioned editions (i.e. editions) and versioned editions (i.e. versions) take the following respective forms:

> http://snomed.info/sct/{sctid}

> http://snomed.info/sct/{sctid}/version/{timestamp}

Note, while it would be possible to extend this pattern to support multiple root modules, each with their own sourceEffectiveTime, this would introduce non-trivial complexities. For example, the modules they each depend upon may themselves overlap but have different versions (targetEffectiveTime) in which case the implied content would be inconsistent.

Examples

The following table shows some examples of URIs for editions and versions.

Table 2.1-1: Examples

Resource	URI
SNOMED CT International Edition	http://snomed.info/sct/900000000000207008
SNOMED CT International Edition, 20130731	http://snomed.info/sct/900000000000207008/version/20130731
SNOMED CT-AU	http://snomed.info/sct/32506021000036107
SNOMED CT-AU, 31 May 2013	http://snomed.info/sct/32506021000036107/version/20130531
SNOMED CT-AU, 30 Nov 2012	http://snomed.info/sct/32506021000036107/version/20121130
SNOMED CT-SE	http://snomed.info/sct/45991000052106

For a more extensive list of SNOMED CT edition URI examples, please refer to 4.4.2 Edition URI Examples

---

Footnotes

RefNotes
1 While there may be additional files associated with a release, it is only the module content which affects the computable meaning of a concept (i.e. the inferable relationships and subsumption between post coordinated expressions).
2 This is the identifier of the module concept, as would be used in the module dependency reference set.