

Expression Constraint Language - Specification and Guide

Expression Constraint Language

Exported on 11/22/2023

Table of Contents

1	1. Introduction.....	8
1.0.1	Background	8
1.0.2	Purpose.....	8
1.0.3	Scope	8
1.0.4	History	9
1.0.5	Audience.....	10
1.0.6	Document Overview	10
2	2. Use Cases	12
2.1	2.1 Terminology Binding.....	12
2.2	2.2 Intensional Reference Set Definitions	12
2.3	2.3 SNOMED CT Content Queries	13
2.4	2.4 SNOMED CT Concept Model.....	13
3	3. Requirements	14
3.1	3.1 General SNOMED CT Language Requirements	14
3.2	3.2 Expression Constraint and Query Requirements	14
3.3	3.3 Concept Model Requirements	17
4	4. Logical Model.....	18
4.1	4.1 Details	19
5	5. Syntax Specification.....	21
5.1	5.1 Brief Syntax (Normative).....	21
5.2	5.2 Long Syntax (Informative)	26
5.3	5.3 Informative Comments	32
5.4	5.4 Order of Operation	56
5.4.1	Unary Operators.....	56
5.4.2	Binary Operators	56
5.4.3	Filter Constraints.....	58
5.4.4	History Supplements	59
5.5	5.5 Character Collation for Term Filters.....	60
6	6. Examples.....	64
6.1	6.1 Simple Expression Constraints.....	64

6.1.1	Self	64
6.1.2	Descendant of	64
6.1.3	Descendant or Self of	65
6.1.4	Child of.....	66
6.1.5	Child or Self of	66
6.1.6	Ancestor of.....	67
6.1.7	Ancestor or Self of	67
6.1.8	Parent of	67
6.1.9	Parent or Self of.....	68
6.1.10	Member of.....	69
6.1.11	Any	70
6.1.12	Alternate Identifier	71
6.2	6.2 Refinements.....	72
6.2.1	Attributes.....	72
6.2.2	Attribute Groups.....	74
6.2.3	Attribute Constraint Operators	75
6.2.4	Concrete Values	76
6.2.5	Reverse Attributes.....	78
6.2.6	Dotted Attributes	79
6.2.7	Any Attribute Name and Value	81
6.3	6.3 Cardinality	82
6.3.1	Attribute cardinality.....	82
6.3.1.1	Overview.....	82
6.3.1.2	Unconstrained Cardinalities.....	83
6.3.1.3	Default Cardinalities	84
6.3.1.4	Non-redundant Attributes.....	84
6.3.1.5	Attribute Cardinality in Groups	85
6.3.2	Attribute Group Cardinality	85
6.3.2.1	Unconstrained Cardinalities.....	86
6.3.2.2	Default Cardinalities	87
6.3.2.3	Non-redundant Attribute Groups.....	88
6.3.2.4	Attribute and Attribute Group Cardinalities	88
6.3.3	Reverse Cardinalities	89
6.4	6.4 Conjunction and Disjunction	90
6.4.1	Compound Expression Constraints	90

6.4.2	Attribute Conjunction and Disjunction	93
6.4.3	Attribute Group Conjunction and Disjunction.....	95
6.4.4	Attribute Value Conjunction and Disjunction	96
6.5	6.5 Exclusion and Not Equals	97
6.5.1	Exclusion of Simple Expressions	97
6.5.2	Exclusion of Attribute Values.....	97
6.5.3	Not Equal to Attribute Value.....	98
6.6	6.6 Constraint Comments.....	99
6.6.1	Comments	99
6.7	6.7 Nested Expression Constraints.....	100
6.7.1	Constraint Operators	100
6.7.2	MemberOf Function	101
6.7.3	Compound Expression Constraints	102
6.7.4	Dotted Attributes	103
6.7.5	Refinement	104
6.7.6	Attribute Values.....	105
6.8	6.8 Description Filters	105
6.8.1	Overview	105
6.8.2	Term Filter	105
6.8.3	Language Filter.....	107
6.8.4	Description Type Filter.....	108
6.8.5	Dialect Filter	109
6.8.6	Acceptability Filter	110
6.8.7	Filters with Negation.....	111
6.8.8	Module Filter	112
6.8.9	Effective Time Filter	113
6.8.10	Active Filter.....	115
6.8.11	Description Id Filter.....	115
6.9	6.9 Concept Filters	116
6.9.1	Overview	116
6.9.2	Module Filter	118
6.9.3	Effective Time Filter	118
6.9.4	Active Filter.....	120
6.10	6.10 Member Filters.....	121

6.10.1	Overview	121
6.10.2	Member Field Filters	121
6.10.3	Module Filter	123
6.10.4	Effective Time Filter	123
6.10.5	Active Filter.....	124
6.11	6.11 History Supplements	124
6.11.1	Background	124
6.11.2	History Supplements	125
6.11.2.1	Overview	125
6.11.2.2	Template	125
6.11.2.3	Profiles.....	126
6.11.3	Use Case Examples	128
6.11.3.1	Use Case 1.....	128
6.11.3.2	Use Case 2.....	129
6.12	6.12 Top and Bottom	129
6.12.1	Top of set	130
6.12.2	Bottom of set.....	130
6.12.3	Use Case Examples	131
6.12.3.1	Use Cases for Bottom.....	131
7	7. Implementation Considerations	133
7.1	7.1 Authoring.....	133
7.1.1	Language-Based Authoring	133
7.1.2	Form-Based Authoring.....	134
7.2	7.2 Parsing	135
7.3	7.3 Validating.....	136
7.4	7.4 Executing	136
7.5	7.5 Storing	136
7.6	7.6 Displaying	137
7.7	7.7 Exchanging	137
8	Appendix A – Examples Of Valid Expressions	139
8.1	A.1 Simple Expression Constraints - Valid Expressions.....	139
8.2	A.2 Refinements - Valid Expressions	142
8.3	A.3 Cardinality - Valid Expressions	149

8.4	A.4 Conjunction and Disjunction - Valid Expressions.....	155
8.5	A.5 Exclusion and Not Equals - Valid Expressions	159
8.6	A.6 Nested Expression Constraints - Valid Expressions.....	163
9	Appendix B – Examples Of Invalid Expressions	166
9.1	B.1 Simple Expression Constraints - Invalid Expressions	166
9.2	B.2 Refinements - Invalid Expressions	168
9.3	B.3 Cardinality - Invalid Expressions	177
9.4	B.4 Conjunction and Disjunction - Invalid Expressions.....	183
9.5	B.5 Exclusion and Not Equals - Invalid Expressions	186
9.6	B.6 Nested Expression Constraints - Invalid Expressions	190
10	Appendix C - Dialect Aliases.....	193
11	Appendix D - ECL Quick Reference	196
11.1	Syntax Overview.....	196
11.2	Examples	198
12	Appendix E - Reference Set Fields	206
13	References	210
14	Previous Versions	211
15	Recent Updates	212
15.1	The most recently updated pages in this document are listed below	212



Leading healthcare
terminology, worldwide

The *Expression Constraint Language* is a formal syntax for representing SNOMED CT expression constraints. Expression constraints are computable rules used to define a bounded sets of clinical meanings represented by either precoordinated or postcoordinated expressions. Expression constraints can be used to restrict the valid values for a data element in an EHR, as the intensional definition of a concept-based reference set, as a machine processable query that identifies a set of matching expressions, or as a constraint that restricts the range of an attribute defined in the SNOMED CT concept model.

This document defines and describes the current version of the Expression Constraint Language - ECL v2.2.

Web browsable version: <http://snomed.org/ecl>

SNOMED CT Document Library: <http://snomed.org/doc>

© Copyright 2023 International Health Terminology Standards Development Organisation, all rights reserved.

This document is a publication of International Health Terminology Standards Development Organisation, trading as SNOMED International. SNOMED International owns and maintains SNOMED CT®.

Any modification of this document (including without limitation the removal or modification of this notice) is prohibited without the express written permission of SNOMED International. This document may be subject to updates. Always use the latest version of this document published by SNOMED International. This can be viewed online and downloaded by following the links on the front page or cover of this document.

SNOMED®, SNOMED CT® and IHTSDO® are registered trademarks of International Health Terminology Standards Development Organisation. SNOMED CT® licensing information is available at <http://snomed.org/licensing>. For more information about SNOMED International and SNOMED International Membership, please refer to <http://www.snomed.org>¹ or contact us at info@snomed.org².

¹ <http://www.ihtsdo.org/>

² <mailto:info@ihtsdo.org>

1 1. Introduction

1.0.1 Background

SNOMED CT is a clinical terminology with global scope covering a wide range of clinical specialties and requirements. The use of SNOMED CT expressions in Electronic Health Records (EHRs) provides a standardized way to represent clinical meanings captured by clinicians and enables the automatic interpretation of these meanings. SNOMED CT expressions are a structured combination of one or more concept identifiers used to represent a clinical idea in a logical manner. The [SNOMED CT Compositional Grammar](http://snomed.org/scg)³ provides a lightweight syntax for the representation of SNOMED CT expressions.

In contrast, a *SNOMED CT Expression Constraint* is a computable rule that can be used to define a *bounded set* of clinical meanings represented by either precoordinated or postcoordinated expressions. Expression constraints can be used as formal constraints on the content of a particular data element in an EHR, as the intensional definition of a concept-based reference set, as a machine processable query that identifies a set of matching precoordinated or postcoordinated expressions, or as a constraint that restricts the range of an attribute defined in the SNOMED CT concept model.

1.0.2 Purpose

The purpose of this document is to define and describe a formal language for representing SNOMED CT Expression Constraints. A SNOMED CT Expression Constraint is a computable rule that defines a bounded set of clinical meanings represented by either precoordinated or postcoordinated expressions. Two equivalent syntaxes are presented – a brief syntax, which is designed to be as compact as possible for interoperable communication between systems, and a long syntax, which introduces textual alternatives to the symbols from the brief syntax. This document also provides examples and guidance to assist in the implementation of this language.

1.0.3 Scope

This document presents the specification of an Expression Constraint Language, which can be used to represent SNOMED CT Expression Constraints. It includes a logical model of the language, two syntaxes, a set of example expression constraints and a summary of implementation considerations.

The Expression Constraint Language specified in this document is part of a consistent set of computer processable languages designed to support a variety of use cases involving the use of SNOMED CT. Other SNOMED CT computable languages include:

- [Compositional Grammar](http://snomed.org/scg)⁴: designed to represent SNOMED CT expressions; and
- [Template Syntax](http://snomed.org/sts)⁵: which allow slots to be added to expressions, expression constraints or queries that can be filled with specific values at a later time.

The compositional grammar is designed to provide a common foundation for the additional functionality added by the other languages.

This document does not include a full description of how to implement an expression constraint parser, classifier or interpreter. It does not describe how to transform an expression constraint into other languages, such as OWL, SPARQL or SQL; or how to determine whether two expression constraints are equivalent. It also does not describe how to implement an EHR which uses expression constraints to constrain or query its content, or a terminology

³ <http://snomed.org/scg>

⁴ <http://snomed.org/scg>

⁵ <http://snomed.org/sts>

server which uses expression constraints to query its content. Instead, it provides a specification, examples and general guidance to assist in the implementation of expression constraints in any of these applications.

This document defines and describes the current version of the Expression Constraint Language - ECL 2.1.

1.0.4 History

Expression constraints have been used in projects and programs around the world for a number of years – for example [HL7 Terminfo](http://snomed.org/hl7terminfo)⁶, and the [NHS Logical Record Architecture](https://isd.hscic.gov.uk/trud3/user/guest/group/0/pack/12)⁷.

In 2013, a draft document on "SNOMED CT Expression Constraint Syntax Specification for Terminology Binding" was developed as an assignment for the SNOMED CT Implementation Advisor (SIA) scheme.

In 2014, this work was revised and extended to support a wider range of relevant use cases to produce version 1.0 of the Expression Constraint Language specification (2015). These updates included:

- Concrete values (e.g. integers, decimals and strings) are now permitted as attribute values. This is to provide alignment with the recent extensions to SNOMED CT Compositional Grammar;
- Cardinality constraints have been introduced, and as a result the optional operator (i.e. ~) is no longer provided;
- Attributes may now be preceded by a 'descendantOf' or 'descendantOrSelfOf' operator to indicate whether attribute descendants and/or the attribute itself should be used in the matching process;
- A reverse flag has been introduced, which allows relationships to be traversed in the reverse direction;
- Exclusion has been changed from a unary operator ('negation') to a binary operator ('minus');
- A wildcard character ('*') has been introduced to represent any concept in the substrate;
- A number of clarifications have been made, including the 'memberOf' operator and the default substrate upon which the expression constraints are executed.

An update to the Expression Constraint Language was then published in 2016 (version 1.1) to incorporate some additional features requested by implementers of the language. These updates include:

- Two new operators 'childOf' and 'parentOf' were added to support querying immediate children and immediate parents of a concept during user interface design;
- A new 'dot notation' was introduced (as an alternative to the Reverse flag) to refer to an attribute value for a concept or expression;
- The ability for a constraint operator (e.g. 'descendantOf') to be applied to a nested expression constraint was added;
- The ability to add comments within the text of an expression constraint was added;
- Additional optional brackets were allowed around subexpressions; and
- The non-normative syntax (previously named the 'Full Syntax') was renamed to the 'Long Syntax'.

Early in 2017 version 1.2 was published, to include a new feature requested by implementers: namely, the ability for the 'memberOf' function to be applied to a set of reference set concepts defined using an expression constraint. In this version, the explanation of *Operator Precedence* was also moved from section 6.7 to section 5.4. Version 1.3 was then published in mid 2017 to support a range of additional features - including allowing the refinement of subexpression constraints, permitting the use of subexpression constraints to represent a set of valid attribute names and simplifying the parsing of dotted expression constraints.

In mid 2020, version 1.4 was published to support boolean attribute values and to introduce the 'childOrSelfOf' and 'parentOrSelfOf' operators. Later that year, version 1.5 was published to support description filter constraints. These constraints filter the result set, by matching only on concepts which have a description that satisfies the filter criteria. Section 5.5 (Character Collation for Term Filters) and section 6.8 (Filter Constraints) were added in ECL version 1.5.

⁶ <http://snomed.org/hl7terminfo>

⁷ <https://isd.hscic.gov.uk/trud3/user/guest/group/0/pack/12>

In 2021, version 1.6 added concept filters, which allow the result set to be filtered based on the definition status, module, effectiveTime and active status of each concept.

And then in early 2022, version 2.0 was published. Version 2.0 includes a number of significant features, including:

- History supplements, to supplement the results with relevant inactive concepts,
- Reference set member filters, to filter the rows of a reference set, based on the value of specified fields,
- Support for returning multiple fields of a reference set, including fields other than the referencedComponentId,
- Support for module, effectiveTime and active filters on descriptions, and
- Support for word-prefix-any-order and wildcard searches for string-based concrete attribute values (for consistency with term searches in a Description filter).

Most significantly, version 2.0 is the first version of ECL that is specifically designed to support querying over historical patient records, which may contain inactive codes.

In August 2022, version 2.1 was published to allow description filters to filter results using description identifiers, and to harmonise the dialect alias filter (see [Appendix C](#)(see page 193)) with [BCP-47 \(Internet Best Current Practice Specification\)](#)⁸.

In November 2023 version 2.2 was published. This version added the ability to reference concepts using alternate identifiers, and also two convenience methods for finding the top (root) or bottom (leaf) concepts within a set.

For a list of previous PDF versions, please refer to [Previous Versions](#)(see page 211).

1.0.5 Audience

The target audiences of this document include:

- SNOMED National Release Centres;
- SNOMED CT designers and developers, including designers and developers of EHR systems, information models, data entry interfaces, storage systems, decision support systems, retrieval and analysis systems, communication standards and terminology services;
- SNOMED CT terminology developers, including concept model designers, content authors, map developers, subset and constraint developers and release process managers.

It should be noted that this document contains both technical and non-technical content. In particular, the detailed logical model and formal syntax is specifically focussed at more technical readers. Less technical readers are encouraged to read the introductory material (including the use cases and requirements) and the extensive set of examples that is presented. It should also be noted that even though complex expression constraints are possible, most expression constraints are likely to be very simple, such as those described in [Simple Expression Constraints](#)⁹.

1.0.6 Document Overview

This document defines the [SNOMED CT Expression Constraint Language](#)¹⁰ and describes how and where it may be implemented. [Chapter 2](#)¹¹ begins by describing the use cases in which it is anticipated that SNOMED CT Expression Constraint Language will be used. [Chapter 3](#)¹² then describes the requirements used to guide the definition of this language. In [Chapter 4](#)¹³, the logical model of the Expression Constraint Language is presented, while in [Chapter 5](#)

⁸ <https://www.rfc-editor.org/rfc/rfc5646.html>

⁹ <https://confluence.ihtsdotools.org/display/WIPECL/6.1+Simple+Expression+Constraints>

¹⁰ <http://snomed.org/ecl>

¹¹ <https://confluence.ihtsdotools.org/display/WIPECL/2.+Use+Cases>

¹² <https://confluence.ihtsdotools.org/display/WIPECL/3.+Requirements>

¹³ <https://confluence.ihtsdotools.org/display/WIPECL/4.+Logical+Model>

¹⁴ two syntaxes are defined using an ABNF serialisation of the logical model. [Chapter 6](#)¹⁵ then presents some examples of expression constraints that conform to the SNOMED CT Expression Constraint syntaxes, and [Chapter 7](#)¹⁶ discusses some implementation considerations. [Appendix A – Examples Of Valid Expressions](#)(see page 139) provides some examples of precoordinated and postcoordinated expressions that satisfy each of the expression constraints presented earlier in the document. [Appendix B – Examples Of Invalid Expressions](#)(see page 166) then provides some examples that do not satisfy these expression constraints. [Appendix C - Dialect Aliases](#)(see page 193) provides a list of example aliases that may be used to specify a particular dialect in an ECL filter constraint. [Appendix D - ECL Quick Reference](#)(see page 196) provides a quick reference to the key syntax features of the Expression Constraint Language. And finally, [Appendix E - Reference Set Fields](#)(see page 206) explains how reference set field names are used in ECL 2.0+.

¹⁴ <https://confluence.ihtsdotools.org/display/WIPECL/5.+Syntax+Specification>

¹⁵ <https://confluence.ihtsdotools.org/display/WIPECL/6.+Examples>

¹⁶ <https://confluence.ihtsdotools.org/display/WIPECL/7.+Implementation+Considerations>

2 2. Use Cases

The SNOMED CT Expression Constraint Language enables the intensional definition of a bounded set of clinical meanings. This is important for a number of use cases, including:

- [Terminology Binding](#)(see page 12);
- [Intensional Reference Set Definitions](#)(see page 12);
- [SNOMED CT Content Queries](#)(see page 12); and
- [SNOMED CT Concept Model](#)(see page 12).

In the following subsections, we describe each of these key use cases.

2.1 2.1 Terminology Binding

Most Electronic Health Records (EHRs) are designed and developed using one or more information models, which describe the information that is collected, stored, communicated and displayed. Some information models are designed for a specific proprietary system, while others are based on a common health information standard (e.g. HL7 FHIR resource, HL7 CDA template, ISO 13606 archetype). Information models may also be defined using a wide variety of representations (e.g. UML class diagram, database table design, Archetype Definition Language, or XML Schema). Irrespective of the purpose, design and representation of the information models, however, the use of clinical terminology is an important part of making the models complete and useful.

Terminology binding provides the links between the information model and the terminology. These links may be used to constrain the set of possible values which can populate a given coded data element in the information model, or they may define the meaning of an information model artefact using the terminology. Terminology binding is an important part of supporting the following clinical information system functions:

- Data capture;
- Retrieval and querying;
- Information model library management; and
- Semantic interoperability.

To enable terminology binding to be defined using intensional rules, a formal language must be used. The [SNOMED CT Expression Constraint Language](#)¹⁷ can be used in this way to define terminology bindings which constrain the set of possible coded values within an information model.

2.2 2.2 Intensional Reference Set Definitions

Reference sets are a flexible, extensible SNOMED CT file structure used to support a variety of requirements for the customization and enhancement of SNOMED CT content. These include the representation of subsets, language preferences, or maps to/from other code systems.

Some reference sets (using the Query Specification type) allow a serialised query to represent the membership of a subset of SNOMED CT components. A query contained in this reference set is executed against the content of SNOMED CT to produce a subset of concepts, descriptions or relationships. This query is referred to as an intensional definition of the subset. It can be run against future releases of SNOMED CT to generate a potentially different set of subset members. The members of the resulting subset may also be represented in an enumerated form as a Simple Reference Set. An enumerated representation of a subset is referred to as an extensional definition.

¹⁷ <http://snomed.org/ecl>

The [SNOMED CT Expression Constraint Language](#)¹⁸ can be used in this way to represent the intensional definition of a subset of SNOMED CT concepts that can be enumerated as a Simple Reference Set.

2.3 2.3 SNOMED CT Content Queries

SNOMED CT provides both hierarchies and formal concept definitions to allow a range of advanced query techniques. SNOMED CT queries can be performed over different sets of terminology artefacts (known as the substrate of the query), including:

- The precoordinated components distributed as part of the SNOMED CT international edition;
- The precoordinated components distributed by a local release centre as part of a national or local SNOMED CT edition;
- The postcoordinated expressions stored within an expression repository; or
- The SNOMED CT expressions stored within an Electronic Health Record (EHR).

The [SNOMED CT Expression Constraint Language](#)¹⁹ enables queries over SNOMED CT content to be expressed. These queries may be performed for a range of purposes, including the authoring and quality assurance of new SNOMED CT content, the design and development of extensional reference sets, and the design and display of SNOMED CT subsets in clinical user interfaces. While the language itself does not support querying over the full EHR content, the [SNOMED CT Expression Constraint Language](#)²⁰ could be embedded within record-based query languages (such as SQL) to represent the terminological aspects of these queries.

2.4 2.4 SNOMED CT Concept Model

The SNOMED CT Concept Model is the set of rules that determines the permitted sets of attributes and values that may be applied to particular types of concepts. There are also additional rules on the cardinality and grouping of each type of attribute. The SNOMED CT Concept Model includes the definition of the domain and range of each attribute. The domain is the set of concepts which are permitted to be used as the source of the attribute, while the range is the set of concepts which are permitted to be used as the target of the attribute. For example, the domain of the attribute [363698007 | Finding site](#)²¹ is the descendants and self of [404684003 | Clinical finding](#)²², while the range is the descendants and self of [442083009 | Anatomical or acquired body structure](#)²³. The SNOMED CT Concept Model rules are represented in a computable form in the [SNOMED CT Machine Readable Concept Model](#)²⁴.

18 <http://snomed.org/ecl>

19 <http://snomed.org/ecl>

20 <http://snomed.org/ecl>

21 <http://snomed.info/id/363698007>

22 <http://snomed.info/id/404684003>

23 <http://snomed.info/id/442083009>

24 <http://snomed.org/mrcm>

3 3. Requirements

In this chapter, we state the requirements of the [SNOMED CT Expression Constraint Language](#)²⁵. These requirements are grouped into [General SNOMED CT Language Requirements](#)(see page 14) (which are shared by all SNOMED CT computable languages), [Expression Constraint and Query Requirements](#)(see page 14), and [Concept Model Requirements](#)(see page 17).

3.1 3.1 General SNOMED CT Language Requirements

The general SNOMED CT language requirements include:

Requirement G.1: Backward compatibility

The language must be backwardly compatible with any version of the language that has previously been adopted as an SNOMED International standard.

Requirement G.2: Consistency

Each logical feature of the language should have a single, consistent meaning across all the languages in the SNOMED CT family of languages. Each logical feature should also have a consistent set of syntax representations.

Requirement G.3: Sufficient and necessary

Each language must be sufficiently expressive to meet the requirements of the use cases for which it was designed. However, functionality without a corresponding use case will not be included, as this increases the complexity of implementation unnecessarily.

Requirement G.4: Machine processability

In order to facilitate the easy adoption by technical audiences, instances of each language must be able to be parsed into a logical representation using a machine processable syntax specification. This requirement will be met by defining the language syntax in ABNF.

Requirement G.5: Human readability

Non-technical stakeholders require that the language is as human readable as possible, while still meeting the other requirements. This is essential for both the clinical validation of expressions, as well as for the education and training required to author expressions.

3.2 3.2 Expression Constraint and Query Requirements

The general expression constraint language requirements include:

Requirement E.1: Able to be evaluated against SNOMED CT content

Expression constraints must be able to be evaluated against a specific set of SNOMED CT content (referred to as the substrate). When evaluated against a finite set of precoordinated concepts or postcoordinated SNOMED CT expressions, a finite subset of the substrate can be found which satisfies the expression constraint.

Please note that the substrate over which the expression constraint is evaluated is not explicitly defined within the expression constraint, and must therefore be established by some other means. By default, the assumed substrate is the set of active components from the snapshot release (in distribution normal form) of the SNOMED CT versioned edition currently loaded into the given tool.

Requirement E.2: Expression constraint functional requirements

²⁵ <http://snomed.org/ecl>

The expression constraint language must support the following capabilities:

Function	Details
Concept reference	The ability to reference a precoordinated SNOMED CT concept using its identifier and optional human-readable term.
Concept hierarchy	The ability to refer to a set of concepts which is exactly equal to the descendants, descendants and self, ancestors, or ancestors and self of a given concept.
Immediate children and parents	The ability to refer to a set of concepts which are either immediate children or immediate parents of a given concept (based on non-redundant 116680003 is a ²⁶ relationships) (with or without the given concept itself).
Conjunction	The ability to connect two expression constraints, attribute groups or attribute sets via a logical AND operator.
Disjunction	The ability to connect two expression constraints, attribute groups or attribute sets via a logical OR operator.
Refinement	The ability to refine (or specialize) the meaning of an expression constraint using one or more attributes values.
Reverse	The ability to constrain the source concepts of a set of relationships, and refer to the destination concepts of these relationships.
Dotted attribute	The ability to refer to the value (or set of values) of an attribute that is included in the definition of a set of concepts.
Attribute group	The ability to group a collection of attributes which operate together as part of a refinement.
Attribute	The ability to specify an attribute name-value pair which further refines the meaning of the matching expressions.
Attribute descendants	The ability to define an attribute which may apply to either the descendants of the given attribute name, or the descendants and self of the given attribute name.

²⁶ <http://snomed.info/id/116680003>

Nesting	The ability to use an expression constraint to represent the valid set of attribute names and/or attribute values.
Concrete values	The ability to use integers, decimals, strings and booleans as attribute values.
Concrete value comparison	The ability to compare the attribute value of the matching expressions with the attribute value in the expression constraint using mathematical comparison operators (e.g. =, <, >, <=, >=, !=).
Member of	The ability to refer to a set of concepts that are referenced by members of a reference set (or set of reference sets).
Reference set field value selection	The ability to return the value of any non-metadata field of a reference set.
Exclusion	The ability to filter out a set of expressions from the result, by either removing expressions whose focus concept is in a specific set, or removing expressions whose attribute value matches a given value.
Any	The ability to refer to any concept in the substrate, without relying on the availability of a single root concept.
Description filter	The ability to filter the result set, based on the properties of each concept's descriptions. Expression constraints should be able to filter the concepts based on whether or not it has a description with a matching term, type, language, membership of a language reference set, and acceptability within that language reference set. Term matching approaches should include wildcard and word-prefix-any-order. Expression constraints should also be able to filter concepts based on the module, effectiveTime, active status and identifier of their descriptions.
Concept filter	The ability to filter the result set, based on the properties of each concept. Expression constraints should be able to restrict the definition status, module, effectiveTime and active status of matching concepts.
Member filter	The ability to filter rows of a reference set member, based on the value of specified fields.
History supplements	The ability to include inactive concepts that are associated with any active concept in a given result set, via an historical association reference set.

3.3 3.3 Concept Model Requirements

The SNOMED CT concept model requirements include:

Requirement C.1: The ability to express SNOMED CT concept model constraints

The language must support the ability to express SNOMED CT concept model constraints, such that the resulting expression constraint can be used to validate SNOMED CT concept definitions and postcoordinated expressions.

In particular, the language must support the ability to define the domain and cardinality of each attribute in the SNOMED CT concept model, and the range of all concept model **object** attributes (whose range is a set of SNOMED CT concepts). The domain of an attribute is the set of valid source concepts of relationships of that type. In most cases, this will be defined as the descendants and self of a given concept. The range of a concept model object attribute is the set of valid destination concepts of relationships of that type. This will be defined as the set of concepts that match a given expression constraint. The cardinality of an attribute constrains the number of times an active relationship of this type can be added to a concept in the SNOMED CT snapshot release (in necessary normal form). For more information about the SNOMED CT necessary normal form, please refer to [2.5. Generating Necessary Normal Form](#)²⁷ in the SNOMED CT OWL Guide (<http://snomed.org/owl>).

Please note that the range of a concept model **data** attribute (whose value is concrete) will be specified using a [value list constraint](#)²⁸ from the SNOMED CT Template Syntax (<http://snomed.org/sts>).

²⁷ <https://confluence.ihtsdotools.org/display/WIPOWL/2.5.+Generating+Necessary+Normal+Form+Relationships+from+the+OWL+Refsets>

²⁸ <https://confluence.ihtsdotools.org/display/DOCSTS/8.3.+Constrained+Replacement+Slots>

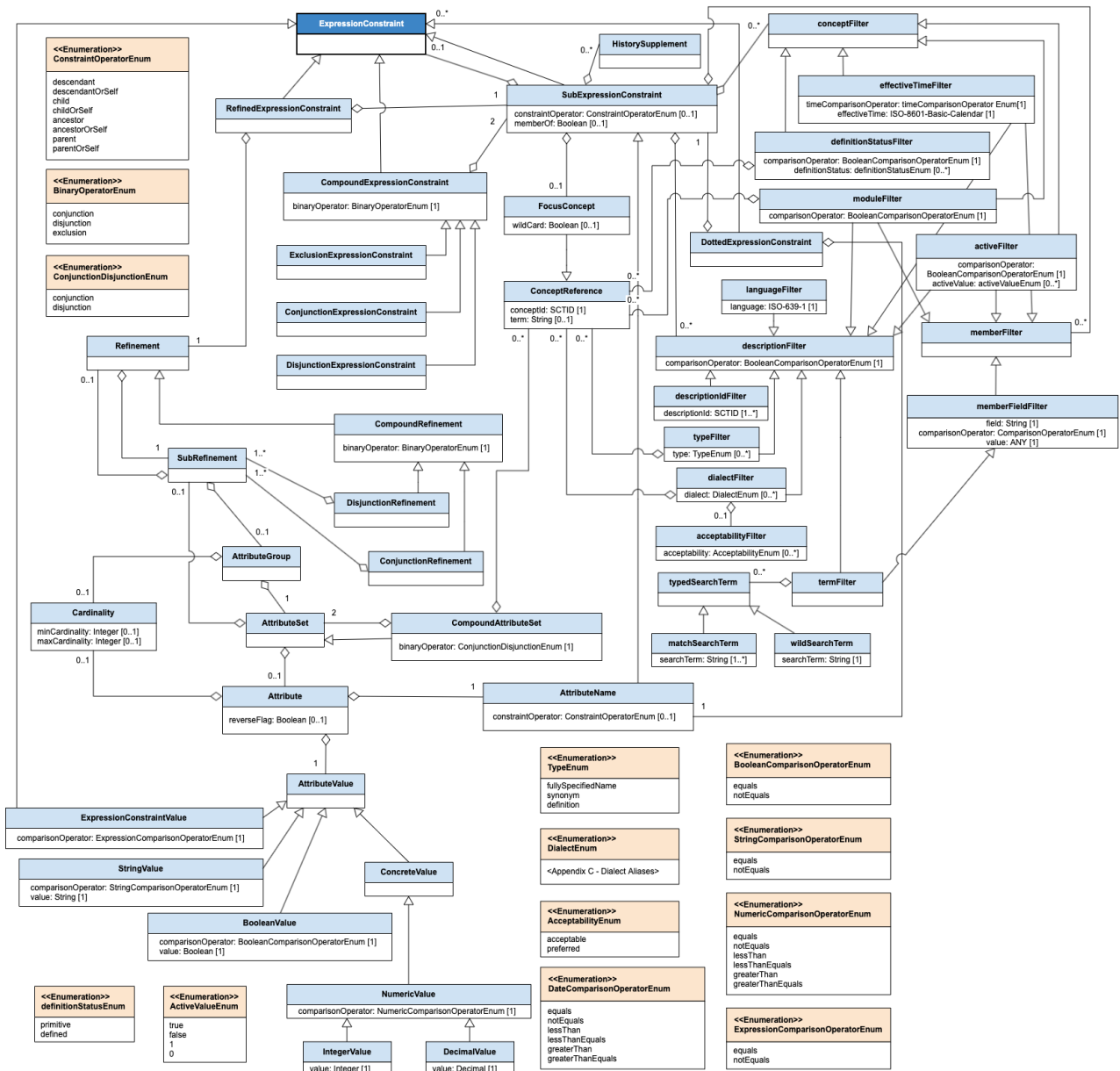


Figure 3: Logical Model of Expression Constraint Language

5 5. Syntax Specification

The following sections describe two syntaxes for use with the SNOMED CT Expression Constraint Language. These syntaxes are serialised representations of the logical model presented in the previous chapter, and are therefore logically equivalent.

The first of these syntaxes is referred to as the 'brief syntax' as it primarily uses a symbolic representation aimed to be as compact as possible. This syntax is considered to be the normative syntax, and is recommended for use in interoperable communications between systems.

The second syntax is referred to as the 'long syntax'. The long syntax introduces English-based textual alternatives to the symbols defined in the 'brief syntax', with the aim of increasing the human readability of the language. The textual alternatives provided in the 'long syntax' may (in theory) be translated into other languages to provide equivalent expression constraint representations that are human-readable by non-English speakers. Please note that the 'long syntax' (and any translations) is non-normative, and should only be used when a reliable mapping to the normative brief syntax is possible.

Please note that by default each expression constraint is evaluated against only the active components (and active members of each reference set) from the snapshot release (in distribution normal form) of a specified SNOMED CT versioned edition.

- [5.1 Brief Syntax \(Normative\)](#)(see page 21)
- [5.2 Long Syntax \(Informative\)](#)(see page 26)
- [5.3 Informative Comments](#)(see page 32)
- [5.4 Order of Operation](#)(see page 56)
- [5.5 Character Collation for Term Filters](#)(see page 60)

5.1 5.1 Brief Syntax (Normative)

The following ABNF definition specifies the Brief Syntax of the SNOMED CT Expression Constraint Language. This ABNF syntax and the ANTLR syntax is maintained in the [SNOMED Expression Constraint Language GitHub](#)³³ repository.

```

expressionConstraint = ws ( refinedExpressionConstraint /
compoundExpressionConstraint / dottedExpressionConstraint /
subExpressionConstraint ) ws
refinedExpressionConstraint = subExpressionConstraint ws ":" ws eclRefinement
compoundExpressionConstraint = conjunctionExpressionConstraint /
disjunctionExpressionConstraint / exclusionExpressionConstraint
conjunctionExpressionConstraint = subExpressionConstraint 1*(ws conjunction ws
subExpressionConstraint)
disjunctionExpressionConstraint = subExpressionConstraint 1*(ws disjunction ws
subExpressionConstraint)
exclusionExpressionConstraint = subExpressionConstraint ws exclusion ws
subExpressionConstraint
dottedExpressionConstraint = subExpressionConstraint 1*(ws
dottedExpressionAttribute)
dottedExpressionAttribute = dot ws eclAttributeName
subExpressionConstraint= [constraintOperator ws] ( ( [memberOf ws]
(eclFocusConcept / "(" ws expressionConstraint ws ")") ) *(ws
memberFilterConstraint) ) / (eclFocusConcept / "(" ws expressionConstraint ws
")") ) *(ws (descriptionFilterConstraint / conceptFilterConstraint)) [ws

```

³³ <https://github.com/IHTSDO/snomed-expression-constraint-language>

```

historySupplement]
eclFocusConcept = eclConceptReference / wildCard / altIdentifier
dot = "."
memberOf = "^" [ ws "[" ws (refsetFieldNameSet / wildCard) ws "]" ]
refsetFieldNameSet = refsetFieldName *(ws "," ws refsetFieldName)
refsetFieldName = 1*alpha
eclConceptReference = conceptId [ws "|" ws term ws "|"]
eclConceptReferenceSet = "(" ws eclConceptReference 1*(mws
eclConceptReference) ws ")"
conceptId = sctId
term = 1*nonwsNonPipe *( 1*SP 1*nonwsNonPipe )
altIdentifier = (QM altIdentifierSchemeAlias "#" altIdentifierCodeWithinQuotes
QM / altIdentifierSchemeAlias "#" altIdentifierCodeWithoutQuotes) [ws "|" ws
term ws "|"]
altIdentifierSchemeAlias = alpha *(dash / alpha / integerValue)
altIdentifierCodeWithinQuotes = 1*anyNonEscapedChar
altIdentifierCodeWithoutQuotes = 1*(alpha / digit / dash / "." / "_")
wildCard = "*"
constraintOperator = childOf / childOrSelfOf / descendantOrSelfOf /
descendantOf / parentOf / parentOrSelfOf / ancestorOrSelfOf / ancestorOf /
top / bottom
descendantOf = "<"
descendantOrSelfOf = "<<"
childOf = "<!"
childOrSelfOf = "<<!"
ancestorOf = ">"
ancestorOrSelfOf = ">>"
parentOf = ">!"
parentOrSelfOf = ">>!"
top = "!!>"
bottom = "!!<"
conjunction = (("a"/"A") ("n"/"N") ("d"/"D") mws) / ", "
disjunction = ("o"/"O") ("r"/"R") mws
exclusion = ("m"/"M") ("i"/"I") ("n"/"N") ("u"/"U") ("s"/"S") mws
eclRefinement = subRefinement ws [conjunctionRefinementSet /
disjunctionRefinementSet]
conjunctionRefinementSet = 1*(ws conjunction ws subRefinement)
disjunctionRefinementSet = 1*(ws disjunction ws subRefinement)
subRefinement = eclAttributeSet / eclAttributeGroup / "(" ws eclRefinement ws
")"
eclAttributeSet = subAttributeSet ws [conjunctionAttributeSet /
disjunctionAttributeSet]
conjunctionAttributeSet = 1*(ws conjunction ws subAttributeSet)
disjunctionAttributeSet = 1*(ws disjunction ws subAttributeSet)
subAttributeSet = eclAttribute / "(" ws eclAttributeSet ws ")"
eclAttributeGroup = "["[" cardinality "]" ws] "{" ws eclAttributeSet ws "}"
eclAttribute = "["[" cardinality "]" ws] [reverseFlag ws] eclAttributeName ws
(expressionComparisonOperator ws subExpressionConstraint /
numericComparisonOperator ws "#" numericValue / stringComparisonOperator ws
(typedSearchTerm / typedSearchTermSet) / booleanComparisonOperator ws
booleanValue)
cardinality = minValue to maxValue
minValue = nonNegativeIntegerValue

```

```

to = ".."
maxValue = nonNegativeIntegerValue / many
many = "*"
reverseFlag = "R"
eclAttributeName = subExpressionConstraint
expressionComparisonOperator = "=" / "!="
numericComparisonOperator = "=" / "!=" / "<=" / "<" / ">=" / ">"
timeComparisonOperator = "=" / "!=" / "<=" / "<" / ">=" / ">"
stringComparisonOperator = "=" / "!="
booleanComparisonOperator = "=" / "!="
idComparisonOperator = "=" / "!="
descriptionFilterConstraint = "{ws [ "d" / "D" ] ws descriptionFilter *(ws
", " ws descriptionFilter) ws }"
descriptionFilter = termFilter / languageFilter / typeFilter / dialectFilter /
moduleFilter / effectiveTimeFilter / activeFilter / descriptionIdFilter
descriptionIdFilter = descriptionIdKeyword ws idComparisonOperator ws
(descriptionId / descriptionIdSet)
descriptionIdKeyword = ("i"/"I") ("d"/"D")
descriptionId = sctId
descriptionIdSet = "(" ws descriptionId *(mws descriptionId) ws ")"
termFilter = termKeyword ws stringComparisonOperator ws (typedSearchTerm /
typedSearchTermSet)
termKeyword = ("t"/"T") ("e"/"E") ("r"/"R") ("m"/"M")
typedSearchTerm = ( [ matchKeyword ws ":" ws ] matchSearchTermSet ) / ( wild
ws ":" ws wildSearchTermSet )
typedSearchTermSet = "(" ws typedSearchTerm *(mws typedSearchTerm) ws ")"
wild = ("w"/"W") ("i"/"I") ("l"/"L") ("d"/"D")
matchKeyword = ("m"/"M") ("a"/"A") ("t"/"T") ("c"/"C") ("h"/"H")
matchSearchTerm = 1*(nonwsNonEscapedChar / escapedChar)
matchSearchTermSet = QM ws matchSearchTerm *(mws matchSearchTerm) ws QM
wildSearchTerm = 1*(anyNonEscapedChar / escapedWildChar)
wildSearchTermSet = QM wildSearchTerm QM
languageFilter = language ws booleanComparisonOperator ws (languageCode /
languageCodeSet)
language = ("l"/"L") ("a"/"A") ("n"/"N") ("g"/"G") ("u"/"U") ("a"/"A")
("g"/"G") ("e"/"E")
languageCode = 2alpha
languageCodeSet = "(" ws languageCode *(mws languageCode) ws ")"
typeFilter = typeIdFilter / typeTokenFilter
typeIdFilter = typeId ws booleanComparisonOperator ws
(subExpressionConstraint / eclConceptReferenceSet)
typeId = ("t"/"T") ("y"/"Y") ("p"/"P") ("e"/"E") ("i"/"I") ("d"/"D")
typeTokenFilter = type ws booleanComparisonOperator ws (typeToken /
typeTokenSet)
type = ("t"/"T") ("y"/"Y") ("p"/"P") ("e"/"E")
typeToken = synonym / fullySpecifiedName / definition
typeTokenSet = "(" ws typeToken *(mws typeToken) ws ")"
synonym = ("s"/"S") ("y"/"Y") ("n"/"N")
fullySpecifiedName = ("f"/"F") ("s"/"S") ("n"/"N")
definition = ("d"/"D") ("e"/"E") ("f"/"F")
dialectFilter = (dialectIdFilter / dialectAliasFilter) [ ws acceptabilitySet ]
dialectIdFilter = dialectId ws booleanComparisonOperator ws

```

```

(subExpressionConstraint / dialectIdSet)
dialectId = ("d"/"D") ("i"/"I") ("a"/"A") ("l"/"L") ("e"/"E") ("c"/"C")
("t"/"T") ("i"/"I") ("d"/"D")
dialectAliasFilter = dialect ws booleanComparisonOperator ws (dialectAlias /
dialectAliasSet)
dialect = ("d"/"D") ("i"/"I") ("a"/"A") ("l"/"L") ("e"/"E") ("c"/"C")
("t"/"T")
dialectAlias = alpha *( dash / alpha / integerValue)
dialectAliasSet = "(" ws dialectAlias [ws acceptabilitySet] *(mws dialectAlias
[ws acceptabilitySet] ) ws ")"
dialectIdSet = "(" ws eclConceptReference [ws acceptabilitySet] *(mws
eclConceptReference [ws acceptabilitySet] ) ws ")"
acceptabilitySet = acceptabilityConceptReferenceSet / acceptabilityTokenSet
acceptabilityConceptReferenceSet = "(" ws eclConceptReference *(mws
eclConceptReference) ws ")"
acceptabilityTokenSet = "(" ws acceptabilityToken *(mws acceptabilityToken) ws
")"
acceptabilityToken = acceptable / preferred
acceptable = ("a"/"A") ("c"/"C") ("c"/"C") ("e"/"E") ("p"/"P") ("t"/"T")
preferred = ("p"/"P") ("r"/"R") ("e"/"E") ("f"/"F") ("e"/"E") ("r"/"R")
conceptFilterConstraint = "{{{" ws ("c" / "C") ws conceptFilter *(ws "," ws
conceptFilter) ws "}}}"
conceptFilter = definitionStatusFilter / moduleFilter / effectiveTimeFilter /
activeFilter
definitionStatusFilter = definitionStatusIdFilter /
definitionStatusTokenFilter
definitionStatusIdFilter = definitionStatusIdKeyword ws
booleanComparisonOperator ws (subExpressionConstraint /
eclConceptReferenceSet)
definitionStatusIdKeyword = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N")
("i"/"I") ("t"/"T") ("i"/"I") ("o"/"O") ("n"/"N") ("s"/"S") ("t"/"T")
("a"/"A") ("t"/"T") ("u"/"U") ("s"/"S") ("i"/"I") ("d"/"D")
definitionStatusTokenFilter = definitionStatusKeyword ws
booleanComparisonOperator ws (definitionStatusToken /
definitionStatusTokenSet)
definitionStatusKeyword = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N")
("i"/"I") ("t"/"T") ("i"/"I") ("o"/"O") ("n"/"N") ("s"/"S") ("t"/"T")
("a"/"A") ("t"/"T") ("u"/"U") ("s"/"S")
definitionStatusToken = primitiveToken / definedToken
definitionStatusTokenSet = "(" ws definitionStatusToken *(mws
definitionStatusToken) ws ")"
primitiveToken = ("p"/"P") ("r"/"R") ("i"/"I") ("m"/"M") ("i"/"I") ("t"/"T")
("i"/"I") ("v"/"V") ("e"/"E")
definedToken = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N") ("e"/"E")
("d"/"D")
moduleFilter = moduleIdKeyword ws booleanComparisonOperator ws
(subExpressionConstraint / eclConceptReferenceSet)
moduleIdKeyword = ("m"/"M") ("o"/"O") ("d"/"D") ("u"/"U") ("l"/"L") ("e"/"E")
("i"/"I") ("d"/"D")
effectiveTimeFilter = effectiveTimeKeyword ws timeComparisonOperator ws
( timeValue / timeValueSet )
effectiveTimeKeyword = ("e"/"E") ("f"/"F") ("f"/"F") ("e"/"E") ("c"/"C")
("t"/"T") ("i"/"I") ("v"/"V") ("e"/"E") ("t"/"T") ("i"/"I") ("m"/"M")

```



```

("e"/"E")
timeValue = QM [ year month day ] QM
timeValueSet = "(" ws timeValue *(mws timeValue) ws ")"
year = digitNonZero digit digit digit
month = "01" / "02" / "03" / "04" / "05" / "06" / "07" / "08" / "09" / "10" /
"11" / "12"
day = "01" / "02" / "03" / "04" / "05" / "06" / "07" / "08" / "09" / "10" /
"11" / "12" / "13" / "14" / "15" / "16" / "17" / "18" / "19" / "20" / "21" /
"22" / "23" / "24" / "25" / "26" / "27" / "28" / "29" / "30" / "31"
activeFilter = activeKeyword ws booleanComparisonOperator ws activeValue
activeKeyword = ("a"/"A") ("c"/"C") ("t"/"T") ("i"/"I") ("v"/"V") ("e"/"E")
activeValue = activeTrueValue / activeFalseValue
activeTrueValue = "1" / "true"
activeFalseValue = "0" / "false"
memberFilterConstraint = "{{" ws ("m" / "M") ws memberFilter *(ws "," ws
memberFilter) ws "}}"
memberFilter = moduleFilter / effectiveTimeFilter / activeFilter /
memberFieldFilter
memberFieldFilter = refsetFieldName ws (expressionComparisonOperator ws
subExpressionConstraint / numericComparisonOperator ws "#" numericValue /
stringComparisonOperator ws (typedSearchTerm / typedSearchTermSet) /
booleanComparisonOperator ws booleanValue / ws timeComparisonOperator ws
(timeValue / timeValueSet) )
historySupplement = "{{" ws "+" ws historyKeyword [ historyProfileSuffix / ws
historySubset ] ws "}}"
historyKeyword = ("h"/"H") ("i"/"I") ("s"/"S") ("t"/"T") ("o"/"O") ("r"/"R")
("y"/"Y")
historyProfileSuffix = historyMinimumSuffix / historyModerateSuffix /
historyMaximumSuffix
historyMinimumSuffix = ("-"/"_" ) ("m"/"M") ("i"/"I") ("n"/"N")
historyModerateSuffix = ("-"/"_" ) ("m"/"M") ("o"/"O") ("d"/"D")
historyMaximumSuffix = ("-"/"_" ) ("m"/"M") ("a"/"A") ("x"/"X")
historySubset = "(" ws expressionConstraint ws ")"
numericValue = ["-"/"+"] (decimalValue / integerValue)
stringValue = 1*(anyNonEscapedChar / escapedChar)
integerValue = digitNonZero *digit / zero
decimalValue = integerValue "." 1*digit
booleanValue = true / false
true = ("t"/"T") ("r"/"R") ("u"/"U") ("e"/"E")
false = ("f"/"F") ("a"/"A") ("l"/"L") ("s"/"S") ("e"/"E")
nonNegativeIntegerValue = (digitNonZero *digit ) / zero
sctId = digitNonZero 5*17( digit )
ws = *( SP / HTAB / CR / LF / comment ) ; optional white space
mws = 1*( SP / HTAB / CR / LF / comment ) ; mandatory white space
comment = "/*" *(nonStarChar / starWithNonFSlash) "*/"
nonStarChar = SP / HTAB / CR / LF / %x21-29 / %x2B-7E / UTF8-2 / UTF8-3 /
UTF8-4
starWithNonFSlash = %x2A nonFSlash
nonFSlash = SP / HTAB / CR / LF / %x21-2E / %x30-7E / UTF8-2 / UTF8-3 / UTF8-4
SP = %x20 ; space
HTAB = %x09 ; tab
CR = %x0D ; carriage return
LF = %x0A ; line feed

```

```

QM = %x22 ; quotation mark
BS = %x5C ; back slash
star = %x2A ; asterisk
digit = %x30-39
zero = %x30
digitNonZero = %x31-39
nonwsNonPipe = %x21-7B / %x7D-7E / UTF8-2 / UTF8-3 / UTF8-4
anyNonEscapedChar = SP / HTAB / CR / LF / %x20-21 / %x23-5B / %x5D-7E /
UTF8-2 / UTF8-3 / UTF8-4
escapedChar = BS QM / BS BS
escapedWildChar = BS QM / BS BS / BS star
nonwsNonEscapedChar = %x21 / %x23-5B / %x5D-7E / UTF8-2 / UTF8-3 / UTF8-4
alpha = %x41-5A / %x61-7A
dash = %x2D
UTF8-2 = %xC2-DF UTF8-tail
UTF8-3 = %xE0 %xA0-BF UTF8-tail / %xE1-EC 2( UTF8-tail ) / %xED %x80-9F UTF8-
tail / %xEE-EF 2( UTF8-tail )
UTF8-4 = %xF0 %x90-BF 2( UTF8-tail ) / %xF1-F3 3( UTF8-tail ) / %xF4 %x80-8F
2( UTF8-tail )
UTF8-tail = %x80-BF

```

5.2 5.2 Long Syntax (Informative)

The following ABNF definition specifies the Long Syntax the [SNOMED CT Expression Constraint Language](#)³⁴. Please note that all keywords are case insensitive.

This ABNF syntax and the ANTLR syntax is maintained in the [SNOMED Expression Constraint Language GitHub](#)³⁵ repository.

```

expressionConstraint = ws ( refinedExpressionConstraint /
compoundExpressionConstraint / dottedExpressionConstraint /
subExpressionConstraint ) ws
refinedExpressionConstraint = subExpressionConstraint ws ":" ws eclRefinement
compoundExpressionConstraint = conjunctionExpressionConstraint /
disjunctionExpressionConstraint / exclusionExpressionConstraint
conjunctionExpressionConstraint = subExpressionConstraint 1*(ws conjunction ws
subExpressionConstraint)
disjunctionExpressionConstraint = subExpressionConstraint 1*(ws disjunction ws
subExpressionConstraint)
exclusionExpressionConstraint = subExpressionConstraint ws exclusion ws
subExpressionConstraint
dottedExpressionConstraint = subExpressionConstraint 1*(ws
dottedExpressionAttribute)
dottedExpressionAttribute = dot ws eclAttributeName
subExpressionConstraint= [constraintOperator ws] ( ( [memberOf ws]
(eclFocusConcept / "(" ws expressionConstraint ws ")") ) *(ws
memberFilterConstraint) ) / (eclFocusConcept / "(" ws expressionConstraint ws
")") ) *(ws (descriptionFilterConstraint / conceptFilterConstraint)) [ws
historySupplement]
eclFocusConcept = eclConceptReference / wildCard / altIdentifier

```

³⁴ <http://snomed.org/ecl>

³⁵ <https://github.com/IHTSDO/snomed-expression-constraint-language>

```

dot = "."
memberOf = ( "^" / ("m"/"M") ("e"/"E") ("m"/"M") ("b"/"B") ("e"/"E") ("r"/"R")
("o"/"O") ("f"/"F") ) [ ws "[" ws (refsetFieldNameSet / wildCard) ws "]" ]
refsetFieldNameSet = refsetFieldName *( ws "," ws refsetFieldName )
refsetFieldName = 1*alpha
eclConceptReference = conceptId [ws "|" ws term ws "|"]
eclConceptReferenceSet = "(" ws eclConceptReference 1*(mws
eclConceptReference) ws ")"
conceptId = sctId
term = 1*nonwsNonPipe *( 1*SP 1*nonwsNonPipe )
altIdentifier = (QM altIdentifierSchemeAlias "#" altIdentifierCodeWithinQuotes
QM / altIdentifierSchemeAlias "#" altIdentifierCodeWithoutQuotes) [ws "|" ws
term ws "|"]
altIdentifierSchemeAlias = alpha *(dash / alpha / integerValue)
altIdentifierCodeWithinQuotes = 1*anyNonEscapedChar
altIdentifierCodeWithoutQuotes = 1*(alpha / digit / dash / "." / "_")
wildCard = "*" / ( ("a"/"A") ("n"/"N") ("y"/"Y") )
constraintOperator = childOf / childOrSelfOf / descendantOrSelfOf /
descendantOf / parentOf / parentOrSelfOf / ancestorOrSelfOf / ancestorOf /
top / bottom
descendantOf = "<" / ( ("d"/"D") ("e"/"E") ("s"/"S") ("c"/"C") ("e"/"E")
("n"/"N") ("d"/"D") ("a"/"A") ("n"/"N") ("t"/"T") ("o"/"O") ("f"/"F") mws )
descendantOrSelfOf = "<<" / ( ("d"/"D") ("e"/"E") ("s"/"S") ("c"/"C")
("e"/"E") ("n"/"N") ("d"/"D") ("a"/"A") ("n"/"N") ("t"/"T") ("o"/"O")
("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F") ("o"/"O") ("f"/"F") mws )
childOf = "<!" / ((("c"/"C") ("h"/"H") ("i"/"I") ("l"/"L") ("d"/"D") ("o"/"O")
("f"/"F") mws )
childOrSelfOf = "<<!" / ((("c"/"C") ("h"/"H") ("i"/"I") ("l"/"L") ("d"/"D")
("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F") ("o"/"O")
("f"/"F") mws )
ancestorOf = ">" / ( ("a"/"A") ("n"/"N") ("c"/"C") ("e"/"E") ("s"/"S")
("t"/"T") ("o"/"O") ("r"/"R") ("o"/"O") ("f"/"F") mws )
ancestorOrSelfOf = ">>" / ( ("a"/"A") ("n"/"N") ("c"/"C") ("e"/"E") ("s"/"S")
("t"/"T") ("o"/"O") ("r"/"R") ("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E")
("l"/"L") ("f"/"F") ("o"/"O") ("f"/"F") mws )
parentOf = ">!" / ((("p"/"P") ("a"/"A") ("r"/"R") ("e"/"E") ("n"/"N") ("t"/"T")
("o"/"O") ("f"/"F") mws )
parentOrSelfOf = ">>!" / ((("p"/"P") ("a"/"A") ("r"/"R") ("e"/"E") ("n"/"N")
("t"/"T") ("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F")
("o"/"O") ("f"/"F") mws )
top = "!!>" / ((("t"/"T") ("o"/"O") ("p"/"P") mws )
bottom = "!!<" / ((("b"/"B") ("o"/"O") ("t"/"T") ("t"/"T") ("o"/"O") ("m"/"M")
mws )
conjunction = ((("a"/"A") ("n"/"N") ("d"/"D") mws) / ", "
disjunction = ("o"/"O") ("r"/"R") mws
exclusion = ("m"/"M") ("i"/"I") ("n"/"N") ("u"/"U") ("s"/"S") mws
eclRefinement = subRefinement ws [conjunctionRefinementSet /
disjunctionRefinementSet]
conjunctionRefinementSet = 1*(ws conjunction ws subRefinement)
disjunctionRefinementSet = 1*(ws disjunction ws subRefinement)
subRefinement = eclAttributeSet / eclAttributeGroup / "(" ws eclRefinement ws
")"

```

```

eclAttributeSet = subAttributeSet ws [conjunctionAttributeSet /
disjunctionAttributeSet]
conjunctionAttributeSet = 1*(ws conjunction ws subAttributeSet)
disjunctionAttributeSet = 1*(ws disjunction ws subAttributeSet)
subAttributeSet = eclAttribute / "(" ws eclAttributeSet ws ")"
eclAttributeGroup = "[" cardinality "]" ws "{" ws eclAttributeSet ws "}"
eclAttribute = "[" cardinality "]" ws [reverseFlag ws] eclAttributeName ws
(expressionComparisonOperator ws subExpressionConstraint /
numericComparisonOperator ws "#" numericValue / stringComparisonOperator ws
(typedSearchTerm / typedSearchTermSet) / booleanComparisonOperator ws
booleanValue)
cardinality = minValue to maxValue
minValue = nonNegativeIntegerValue
to = ".." / (mws ("t"/"T") ("o"/"O") mws)
maxValue = nonNegativeIntegerValue / many
many = "*" / ( ("m"/"M") ("a"/"A") ("n"/"N") ("y"/"Y"))
reverseFlag = ( ("r"/"R") ("e"/"E") ("v"/"V") ("e"/"E") ("r"/"R") ("s"/"S")
("e"/"E") ("o"/"O") ("f"/"F")) / "R"
eclAttributeName = subExpressionConstraint
expressionComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws
"=" / "<>"
numericComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws
"=" / "<>" / "<=" / "<" / ">=" / ">"
timeComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" /
"<>" / "<=" / "<" / ">=" / ">"
stringComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" /
"<>"
booleanComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws
"=" / "<>"
idComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" /
"<>"
descriptionFilterConstraint = "{" ws [ "d" / "D" ] ws descriptionFilter *(ws
"," ws descriptionFilter) ws "}"
descriptionFilter = termFilter / languageFilter / typeFilter / dialectFilter /
moduleFilter / effectiveTimeFilter / activeFilter / descriptionIdFilter
descriptionIdFilter = descriptionIdKeyword ws idComparisonOperator ws
(descriptionId / descriptionIdSet)
descriptionIdKeyword = ("i"/"I") ("d"/"D")
descriptionId = sctId
descriptionIdSet = "(" ws descriptionId *(mws descriptionId) ws ")"
termFilter = termKeyword ws stringComparisonOperator ws (typedSearchTerm /
typedSearchTermSet)
termKeyword = ("t"/"T") ("e"/"E") ("r"/"R") ("m"/"M")
typedSearchTerm = ( [ matchKeyword ws ":" ws ] matchSearchTermSet ) / ( wild
ws ":" ws wildSearchTermSet )
typedSearchTermSet = "(" ws typedSearchTerm *(mws typedSearchTerm) ws ")"
wild = ("w"/"W") ("i"/"I") ("l"/"L") ("d"/"D")
matchKeyword = ("m"/"M") ("a"/"A") ("t"/"T") ("c"/"C") ("h"/"H")
matchSearchTerm = 1*(nonwsNonEscapedChar / escapedChar)
matchSearchTermSet = QM ws matchSearchTerm *(mws matchSearchTerm) ws QM
wildSearchTerm = 1*(anyNonEscapedChar / escapedWildChar)
wildSearchTermSet = QM wildSearchTerm QM

```

```

languageFilter = language ws booleanComparisonOperator ws (languageCode /
languageCodeSet)
language = ("l"/"L") ("a"/"A") ("n"/"N") ("g"/"G") ("u"/"U") ("a"/"A")
("g"/"G") ("e"/"E")
languageCode = 2alpha
languageCodeSet = "(" ws languageCode *(mws languageCode) ws ")"
typeFilter = typeIdFilter / typeTokenFilter
typeIdFilter = typeId ws booleanComparisonOperator ws
(subExpressionConstraint / eclConceptReferenceSet)
typeId = ("t"/"T") ("y"/"Y") ("p"/"P") ("e"/"E") ("i"/"I") ("d"/"D")
typeTokenFilter = type ws booleanComparisonOperator ws (typeToken /
typeTokenSet)
type = ("t"/"T") ("y"/"Y") ("p"/"P") ("e"/"E")
typeToken = synonym / fullySpecifiedName / definition
typeTokenSet = "(" ws typeToken *(mws typeToken) ws ")"
synonym = ("s"/"S") ("y"/"Y") ("n"/"N") [ ("o"/"O") ("n"/"N") ("y"/"Y")
("m"/"M") ]
fullySpecifiedName = ( ("f"/"F") ("s"/"S") ("n"/"N") ) / ( ("f"/"F") ("u"/"U")
("l"/"L") ("l"/"L") ("y"/"Y") ("s"/"S") ("p"/"P") ("e"/"E") ("c"/"C")
("i"/"I") ("f"/"F") ("i"/"I") ("e"/"E") ("d"/"D") ("n"/"N") ("a"/"A")
("m"/"M") ("e"/"E") )
definition = ("d"/"D") ("e"/"E") ("f"/"F") [ ("i"/"I") ("n"/"N") ("i"/"I")
("t"/"T") ("i"/"I") ("o"/"O") ("n"/"N") ]
dialectFilter = (dialectIdFilter / dialectAliasFilter) [ ws acceptabilitySet ]
dialectIdFilter = dialectId ws booleanComparisonOperator ws
(subExpressionConstraint / dialectIdSet)
dialectId = ("d"/"D") ("i"/"I") ("a"/"A") ("l"/"L") ("e"/"E") ("c"/"C")
("t"/"T") ("i"/"I") ("d"/"D")
dialectAliasFilter = dialect ws booleanComparisonOperator ws (dialectAlias /
dialectAliasSet)
dialect = ("d"/"D") ("i"/"I") ("a"/"A") ("l"/"L") ("e"/"E") ("c"/"C")
("t"/"T")
dialectAlias = alpha *( dash / alpha / integerValue)
dialectAliasSet = "(" ws dialectAlias [ws acceptabilitySet] *(mws dialectAlias
[ws acceptabilitySet]) ws ")"
dialectIdSet = "(" ws eclConceptReference [ws acceptabilitySet] *(mws
eclConceptReference [ws acceptabilitySet] ) ws ")"
acceptabilitySet = acceptabilityConceptReferenceSet / acceptabilityTokenSet
acceptabilityConceptReferenceSet = "(" ws eclConceptReference *(mws
eclConceptReference) ws ")"
acceptabilityTokenSet = "(" ws acceptabilityToken *(mws acceptabilityToken) ws
")"
acceptabilityToken = acceptable / preferred
acceptable = ("a"/"A") ("c"/"C") ("c"/"C") ("e"/"E") ("p"/"P") ("t"/"T")
[ ("a"/"A") ("b"/"B") ("l"/"L") ("e"/"E") ]
preferred = ("p"/"P") ("r"/"R") ("e"/"E") ("f"/"F") ("e"/"E") ("r"/"R")
[ ("r"/"R") ("e"/"E") ("d"/"D") ]
conceptFilterConstraint = "{{" ws ("c" / "C") ws conceptFilter *(ws "," ws
conceptFilter) ws "}}"
conceptFilter = definitionStatusFilter / moduleFilter / effectiveTimeFilter /
activeFilter
definitionStatusFilter = definitionStatusIdFilter /

```

```

definitionStatusTokenFilter
definitionStatusIdFilter = definitionStatusIdKeyword ws
booleanComparisonOperator ws (subExpressionConstraint /
eclConceptReferenceSet)
definitionStatusIdKeyword = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N")
("i"/"I") ("t"/"T") ("j"/"J") ("o"/"O") ("n"/"N") ("s"/"S") ("t"/"T")
("a"/"A") ("t"/"T") ("u"/"U") ("s"/"S") ("i"/"I") ("d"/"D")
definitionStatusTokenFilter = definitionStatusKeyword ws
booleanComparisonOperator ws (definitionStatusToken /
definitionStatusTokenSet)
definitionStatusKeyword = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N")
("i"/"I") ("t"/"T") ("j"/"J") ("o"/"O") ("n"/"N") ("s"/"S") ("t"/"T")
("a"/"A") ("t"/"T") ("u"/"U") ("s"/"S")
definitionStatusToken = primitiveToken / definedToken
definitionStatusTokenSet = "(" ws definitionStatusToken *(mws
definitionStatusToken) ws ")"
primitiveToken = ("p"/"P") ("r"/"R") ("i"/"I") ("m"/"M") ("i"/"I") ("t"/"T")
("i"/"I") ("v"/"V") ("e"/"E")
definedToken = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N") ("e"/"E")
("d"/"D")
moduleFilter = moduleIdKeyword ws booleanComparisonOperator ws
(subExpressionConstraint / eclConceptReferenceSet)
moduleIdKeyword = ("m"/"M") ("o"/"O") ("d"/"D") ("u"/"U") ("l"/"L") ("e"/"E")
("i"/"I") ("d"/"D")
effectiveTimeFilter = effectiveTimeKeyword ws timeComparisonOperator ws
( timeValue / timeValueSet )
effectiveTimeKeyword = ("e"/"E") ("f"/"F") ("f"/"F") ("e"/"E") ("c"/"C")
("t"/"T") ("i"/"I") ("v"/"V") ("e"/"E") ("t"/"T") ("i"/"I") ("m"/"M")
("e"/"E")
timeValue = QM [ year month day ] QM
timeValueSet = "(" ws timeValue *(mws timeValue) ws ")"
year = digitNonZero digit digit digit
month = "01" / "02" / "03" / "04" / "05" / "06" / "07" / "08" / "09" / "10" /
"11" / "12"
day = "01" / "02" / "03" / "04" / "05" / "06" / "07" / "08" / "09" / "10" /
"11" / "12" / "13" / "14" / "15" / "16" / "17" / "18" / "19" / "20" / "21" /
"22" / "23" / "24" / "25" / "26" / "27" / "28" / "29" / "30" / "31"
activeFilter = activeKeyword ws booleanComparisonOperator ws activeValue
activeKeyword = ("a"/"A") ("c"/"C") ("t"/"T") ("i"/"I") ("v"/"V") ("e"/"E")
activeValue = activeTrueValue / activeFalseValue
activeTrueValue = "1" / "true"
activeFalseValue = "0" / "false"
memberFilterConstraint = "{{" ws ("m" / "M") ws memberFilter *(ws "," ws
memberFilter) ws "}}"
memberFilter = moduleFilter / effectiveTimeFilter / activeFilter /
memberFieldFilter
memberFieldFilter = refsetFieldName ws (expressionComparisonOperator ws
subExpressionConstraint / numericComparisonOperator ws "#" numericValue /
stringComparisonOperator ws (typedSearchTerm / typedSearchTermSet) /
booleanComparisonOperator ws booleanValue / ws timeComparisonOperator ws
(timeValue / timeValueSet) )
historySupplement = "{{" ws "+" ws historyKeyword [ historyProfileSuffix / ws
historySubset ] ws "}}"

```

```

historyKeyword = ("h"/"H") ("i"/"I") ("s"/"S") ("t"/"T") ("o"/"O") ("r"/"R")
("y"/"Y")
historyProfileSuffix = historyMinimumSuffix / historyModerateSuffix /
historyMaximumSuffix
historyMinimumSuffix = ("-"/"_" ) ("m"/"M") ("i"/"I") ("n"/"N")
historyModerateSuffix = ("-"/"_" ) ("m"/"M") ("o"/"O") ("d"/"D")
historyMaximumSuffix = ("-"/"_" ) ("m"/"M") ("a"/"A") ("x"/"X")
historySubset = "(" ws expressionConstraint ws ")"
numericValue = ["-"/"+"] (decimalValue / integerValue)
stringValue = 1*(anyNonEscapedChar / escapedChar)
integerValue = digitNonZero *digit / zero
decimalValue = integerValue "." 1*digit
booleanValue = true / false
true = ("t"/"T") ("r"/"R") ("u"/"U") ("e"/"E")
false = ("f"/"F") ("a"/"A") ("l"/"L") ("s"/"S") ("e"/"E")
nonNegativeIntegerValue = (digitNonZero *digit ) / zero
sctId = digitNonZero 5*17( digit )
ws = *( SP / HTAB / CR / LF / comment ) ; optional white space
mws = 1*( SP / HTAB / CR / LF / comment ) ; mandatory white space
comment = "/*" *(nonStarChar / starWithNonFSlash) "*/"
nonStarChar = SP / HTAB / CR / LF / %x21-29 / %x2B-7E / UTF8-2 / UTF8-3 /
UTF8-4
starWithNonFSlash = %x2A nonFSlash
nonFSlash = SP / HTAB / CR / LF / %x21-2E / %x30-7E / UTF8-2 / UTF8-3 / UTF8-4
SP = %x20 ; space
HTAB = %x09 ; tab
CR = %x0D ; carriage return
LF = %x0A ; line feed
QM = %x22 ; quotation mark
BS = %x5C ; back slash
star = %x2A ; asterisk
digit = %x30-39
zero = %x30
digitNonZero = %x31-39
nonwsNonPipe = %x21-7B / %x7D-7E / UTF8-2 / UTF8-3 / UTF8-4
anyNonEscapedChar = SP / HTAB / CR / LF / %x20-21 / %x23-5B / %x5D-7E /
UTF8-2 / UTF8-3 / UTF8-4
escapedChar = BS QM / BS BS
escapedWildChar = BS QM / BS BS / BS star
nonwsNonEscapedChar = %x21 / %x23-5B / %x5D-7E / UTF8-2 / UTF8-3 / UTF8-4
alpha = %x41-5A / %x61-7A
dash = %x2D
UTF8-2 = %xC2-DF UTF8-tail
UTF8-3 = %xE0 %xA0-BF UTF8-tail / %xE1-EC 2( UTF8-tail ) / %xED %x80-9F UTF8-
tail / %xEE-EF 2( UTF8-tail )
UTF8-4 = %xF0 %x90-BF 2( UTF8-tail ) / %xF1-F3 3( UTF8-tail ) / %xF4 %x80-8F
2( UTF8-tail )
UTF8-tail = %x80-BF

```

5.3 5.3 Informative Comments

This section provides a short description of each ABNF rule listed above. The related brief and long syntax rules are grouped together with the same description. Where the syntaxes are the same, the rule is listed once and preceded with the text "BS/LS". Where the brief and long syntaxes are different, both rules are listed separately and preceded with "BS" and "LS" respectively.

BS/LS: expressionConstraint = ws (refinedExpressionConstraint / compoundExpressionConstraint / dottedExpressionConstraint / subExpressionConstraint) ws	
	An expression constraint is either a refined expression constraint, a compound expression constraint, a dotted expression constraint, or a sub expression constraint.
BS/LS: refinedExpressionConstraint = subExpressionConstraint ws ":" ws eclRefinement	
	A refined expression constraint includes a subexpression constraint followed by a refinement.
BS/LS: compoundExpressionConstraint = conjunctionExpressionConstraint / disjunctionExpressionConstraint / exclusionExpressionConstraint	
	A compound expression constraint contains two or more expression constraints joined by either a conjunction, disjunction or exclusion. When potential ambiguity in binary operator precedence may occur, round brackets must be used to clearly disambiguate the order in which these operator are applied. Brackets are not required in expression constraints in which all binary operators are conjunctions, or all binary operators are disjunctions. Please note that unary operators (i.e. constraint operators and member of functions) are always applied before binary operators (i.e. conjunction, disjunction and exclusion).
BS/LS: conjunctionExpressionConstraint = subExpressionConstraint 1*(ws conjunction ws subExpressionConstraint)	
	A conjunction expression constraint combines two or more expression constraints with a conjunction ("and") operator. More than one conjunction may be used without brackets. However any compound expression constraint (using a different binary operator) that appears within a conjunction expression constraint must be enclosed by brackets.
BS/LS: disjunctionExpressionConstraint = subExpressionConstraint 1*(ws disjunction ws subExpressionConstraint)	
	A disjunction expression constraint combines two or more expression constraints with a disjunction ("or") operator. More than one disjunction may be used without brackets. However any compound expression constraint (using a different binary operator) that appears within a disjunction expression constraint must be enclosed by brackets.

BS/LS: exclusionExpressionConstraint = subExpressionConstraint ws exclusion ws subExpressionConstraint	
	An exclusion expression constraint combines two expression constraints with an exclusion ("minus") operator. A single exclusion operator may be used without brackets. However when the operands of the exclusion expression constraint are compound, these compound expression constraints must be enclosed by brackets.
BS/LS: dottedExpressionConstraint = subExpressionConstraint 1*(ws dottedExpressionAttribute)	
	A dotted expression constraint contains a sub expression constraint, followed by one or more dotted attributes. When a single dotted attribute is used, the result is the set of attribute values (for the given attribute name) of each concept that results from evaluating the subExpressionConstraint. When more than one dotted attribute is used, each dottedExpressionAttribute is sequentially evaluated (from left to right) against the given result set.
BS/LS: dottedExpressionAttribute = dot ws eclAttributeName	
	A dotted expression attribute consists of a 'dot', followed by an attribute name. Please note that the attribute name may be represented by any sub expression constraint.
BS/LS: subExpressionConstraint = [constraintOperator ws] (([memberOf ws] (eclFocusConcept / "(" ws expressionConstraint ws ")") *(ws memberFilterConstraint)) / (eclFocusConcept / "(" ws expressionConstraint ws ")")) *(ws (descriptionFilterConstraint / conceptFilterConstraint)) [ws historySupplement]	
	<p>A sub expression constraint optionally begins with a constraint operator and/or a memberOf function. It then includes either a single focus concept or an expression constraint (enclosed in brackets). If the memberOf function is applied, a member filter constraint may be used. A sub expression constraint may then optionally include one or more concept or description filter constraints, followed optionally by a history supplement.</p> <p>Notes: A memberOf function should be used only when the eclFocusConcept or expressionConstraint refers to a reference set concept, a set of reference set concepts, or a wild card. When both a constraintOperator and a memberOf function are used, they are applied from the inside to out (i.e. from right to left) - see 5.4 Order of Operation(see page 56). Therefore, if a constraintOperator is followed by a memberOf function, then the memberOf function is processed prior to the constraintOperator.</p>
BS/LS: eclFocusConcept = eclConceptReference / wildCard	
	A focus concept is a concept reference or a wild card.
BS/LS: dot = "."	

	A dot connects an expression constraint with an attribute whose values are included in the result.
BS: memberOf = "^" [ws "[" ws (refsetFieldNameSet / wildCard) ws "]"]	
LS: memberOf = ("^" / ("m"/"M") ("e"/"E") ("m"/"M") ("b"/"B") ("e"/"E") ("r"/"R") ("o"/"O") ("f"/"F")) [ws "[" ws (refsetFieldNameSet / wildCard) ws "]"]	
	By default, the 'memberOf' function returns the set of referenced components in the set of reference sets which follows. In the brief syntax, the memberOf function is represented using the "^" symbol. In the long syntax, the text "memberOf " (case insensitive and followed by at least one white space) is also allowed. If a set of reference set fields is listed in square brackets after the memberOf function, then the values of these fields are returned.
BS/LS: refsetFieldNameSet = refsetFieldName *(ws "," ws refsetFieldName)	
	A refsetFieldNameSet is a set of one or more reference set fields, separated by a comma and optional whitespace.
BS/LS: refsetFieldName = 1*alpha	
	A refsetFieldName is the set of alphabetic characters used to name a reference set field.
BS/LS: eclConceptReference = conceptId [ws " " ws term ws "]"]	
	A conceptReference is represented by a ConceptId, optionally followed by a term ³⁶ enclosed by a pair of " " characters. Whitespace before or after the ConceptId is ignored as is any whitespace between the initial " " characters and the first non-whitespace character in the term ³⁷ or between the last non-whitespace character and before second " " character.
BS/LS: eclConceptReferenceSet = "(" ws eclConceptReference 1*(mws eclConceptReference) ws ")"	
	A concept reference set includes two or more concept references separated by mandatory white space and enclosed in brackets.
BS/LS: conceptId = sctId	
	The ConceptId must be a valid SNOMED CT identifier ³⁸ for a concept ³⁹ . The initial digit may not be zero. The smallest number of digits is six, and the maximum is 18.

36 [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))

37 [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))

38 <https://confluence.ihtsdotools.org/display/DOCGLOSS/SNOMED+CT+identifier>

39 <https://confluence.ihtsdotools.org/display/DOCGLOSS/concept>

BS/LS: term = 1*nonwsnonpipe *(1*SP 1*nonwsnonpipe)	
	The term ⁴⁰ must be the term ⁴¹ from a SNOMED CT description ⁴² that is associated with the concept ⁴³ identified by the preceding concept identifier ⁴⁴ . For example, the term ⁴⁵ could be the preferred description ⁴⁶ , or the preferred description ⁴⁷ associated with a particular translation. The term ⁴⁸ may include valid UTF-8 ⁴⁹ characters except for the pipe "
BS: wildCard = "*" LS: wildCard = "*" / (("a"/"A") ("n"/"N") ("y"/"Y"))	
	A wild card represents any concept in the given substrate. In the brief syntax, a wildcard is represented using the "*" symbol. In the long syntax, the text "ANY" (case insensitive) is also allowed.
BS/LS: constraintOperator = childOf / childOrSelfOf / descendantOrSelfOf / descendantOf / parentOf / parentOrSelfOf / ancestorOrSelfOf / ancestorOf	
	A constraint operator is either 'childOf', 'childOrSelfOf', 'descendantOrSelfOf', 'descendantOf', 'parentOf', 'parentOrSelfOf', 'ancestorOrSelfOf', or 'ancestorOf'.
BS: descendantOf = "<" LS: descendantOf = "<" / (("d"/"D") ("e"/"E") ("s"/"S") ("c"/"C") ("e"/"E") ("n"/"N") ("d"/"D") ("a"/"A") ("n"/"N") ("t"/"T") ("o"/"O") ("f"/"F") mws)	
	The descendantOf operator returns the set of all subtypes of the given concept (or set of concepts). In the brief syntax, the descendantOf operator is represented using the symbol "<". In the long syntax, the text "descendantOf" (case insensitive and followed by at least one white space) is also allowed.
BS: descendantOrSelfOf = "<<" LS: descendantOrSelfOf = "<<" / (("d"/"D") ("e"/"E") ("s"/"S") ("c"/"C") ("e"/"E") ("n"/"N") ("d"/"D") ("a"/"A") ("n"/"N") ("t"/"T") ("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F") ("o"/"O") ("f"/"F") mws)	

40 [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))
 41 [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))
 42 <https://confluence.ihtsdotools.org/display/DOCGLOSS/SNOMED+CT+description>
 43 <https://confluence.ihtsdotools.org/display/DOCGLOSS/concept>
 44 <https://confluence.ihtsdotools.org/display/DOCGLOSS/concept+identifier>
 45 [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))
 46 <https://confluence.ihtsdotools.org/display/DOCGLOSS/description>
 47 <https://confluence.ihtsdotools.org/display/DOCGLOSS/description>
 48 [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))
 49 <https://confluence.ihtsdotools.org/display/DOCRELFMT/UTF-8>

	<p>The descendantOrSelfOf operator returns the set of all subtypes of the given concept (or set of concepts), plus the concept (or set of concepts) itself. In the brief syntax, the descendantOrSelfOf operator is represented using the symbols "<<". In the long syntax, the text "descendantOrSelfOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS: childOf = "<!" LS: childOf = "<!" / ((("c"/"C") ("h"/"H") ("i"/"I") ("l"/"L") ("d"/"D") ("o"/"O") ("f"/"F") mws)</p>	
	<p>The childOf operator returns the set of all immediate children of the given concept (or set of concepts). In the brief syntax, the childOf operator is represented using the symbols "<!". In the long syntax, the text "childOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS: childOrSelfOf = "<<!" LS: childOrSelfOf = "<<!" / ((("c"/"C") ("h"/"H") ("i"/"I") ("l"/"L") ("d"/"D") ("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F") ("o"/"O") ("f"/"F") mws)</p>	
	<p>The childOrSelfOf operator returns the set of all immediate children of the given concept (or set of concepts), plus the concept (or set of concepts) itself. In the brief syntax, the childOrSelfOf operator is represented using the symbols "<<!". In the long syntax, the text "childOrSelfOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS: ancestorOf = ">" LS: ancestorOf = ">" / ((("a"/"A") ("n"/"N") ("c"/"C") ("e"/"E") ("s"/"S") ("t"/"T") ("o"/"O") ("r"/"R") ("o"/"O") ("f"/"F") mws)</p>	
	<p>The ancestorOf operator returns the set of all supertypes of the given concept (or set of concepts). In the brief syntax, the ancestorOf operator is represented using the symbol ">". In the long syntax, the text "ancestorOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS: ancestorOrSelfOf = ">>" LS: ancestorOrSelfOf = ">>" / ((("a"/"A") ("n"/"N") ("c"/"C") ("e"/"E") ("s"/"S") ("t"/"T") ("o"/"O") ("r"/"R") ("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F") ("o"/"O") ("f"/"F") mws)</p>	
	<p>The ancestorOrSelfOf operator returns the set of all supertypes of the given concept (or set of concepts), plus the concept (or set of concepts) itself. In the brief syntax, the ancestorOrSelfOf operator is represented using the symbols ">>". In the long syntax, the text "ancestorOrSelfOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS: parentOf = ">!" LS: parentOf = ">!" / ((("p"/"P") ("a"/"A") ("r"/"R") ("e"/"E") ("n"/"N") ("t"/"T") ("o"/"O") ("f"/"F") mws)</p>	

	<p>The parentOf operator returns the set of all immediate parents of the given concept (or set of concepts). In the brief syntax, the parentOf operator is represented using the symbols ">!". In the long syntax, the text "parentOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS: parentOrSelfOf = ">>!" LS: parentOrSelfOf = ">>!" / (("p"/"P") ("a"/"A") ("r"/"R") ("e"/"E") ("n"/"N") ("t"/"T") ("o"/"O") ("r"/"R") ("s"/"S") ("e"/"E") ("l"/"L") ("f"/"F") ("o"/"O") ("f"/"F") mws)</p>	
	<p>The parentOrSelfOf operator returns the set of all immediate parents of the given concept (or set of concepts), plus the concept (or set of concepts) itself. In the brief syntax, the parentOrSelfOf operator is represented using the symbols ">>!". In the long syntax, the text "parentOrSelfOf" (case insensitive and followed by at least one white space) is also allowed.</p>
<p>BS/LS: conjunction = (("a"/"A") ("n"/"N") ("d"/"D") mws) / ","</p>	
	<p>A conjunction is represented either by the word "and" (case insensitive and followed by at least one white space), or by a comma.</p>
<p>BS/LS: disjunction = ("o"/"O") ("r"/"R") mws</p>	
	<p>A disjunction is represented by the word "or" (case insensitive and followed by at least one white space).</p>
<p>BS/LS: exclusion = ("m"/"M") ("i"/"I") ("n"/"N") ("u"/"U") ("s"/"S") mws</p>	
	<p>The exclusion operator is represented by the word "minus" (case insensitive and followed by at least one white space).</p>
<p>BS/LS: eclRefinement = subRefinement ws [conjunctionRefinementSet / disjunctionRefinementSet]</p>	
	<p>A refinement contains all the grouped and ungrouped attributes that refine the set of clinical meanings satisfied by the expression constraint. Refinements may represent the conjunction or disjunction of two smaller refinements, and may optionally be placed in brackets. Where both conjunction and disjunction are used, brackets are mandatory to disambiguate the intended meaning.</p>
<p>BS/LS: conjunctionRefinementSet = 1*(ws conjunction ws subRefinement)</p>	
	<p>A conjunction refinement set consists of one or more conjunction operators, each followed by a subRefinement.</p>
<p>BS/LS: disjunctionRefinementSet = 1*(ws disjunction ws subRefinement)</p>	

	A disjunction refinement set consists of one or more disjunction operators, each followed by a subRefinement.
BS/LS: subRefinement = eclAttributeSet / eclAttributeGroup / "(" ws eclRefinement ws ")"	
	A subRefinement is either an attribute set, an attribute group or a bracketed refinement.
BS/LS: eclAttributeSet = subAttributeSet ws [conjunctionAttributeSet / disjunctionAttributeSet]	
	An attribute set contains one or more attribute name ⁵⁰ -value pairs separated by a conjunction or disjunction operator. An attribute set may optionally be placed in brackets.
BS/LS: conjunctionAttributeSet = 1*(ws conjunction ws subAttributeSet)	
	A conjunction attribute set consists of one or more conjunction operators, each followed by a subAttributeSet.
BS/LS: disjunctionAttributeSet = 1*(ws disjunction ws subAttributeSet)	
	A disjunction attribute set consists of one or more disjunction operators, each followed by a subAttributeSet.
BS/LS: subAttributeSet = eclAttribute / "(" ws eclAttributeSet ws ")"	
	A subAttributeSet is either an attribute or a bracketed attribute set.
BS/LS: eclAttributeGroup = ["[" cardinality "]" ws] "{" ws eclAttributeSet ws "}"	
	An attribute group ⁵¹ contains a collection of attributes that operate together as part of the refinement ⁵² of the containing expression ⁵³ constraint. An attribute group may optionally be preceded by a cardinality. An attribute group cardinality indicates the minimum and maximum number of attribute groups that must satisfy the given attributeSet constraint for the expression constraint to be satisfied.
BS/LS: eclAttribute = ["[" cardinality "]" ws] [reverseFlag ws] eclAttributeName ws (expressionComparisonOperator ws subExpressionConstraint / numericComparisonOperator ws "#" numericValue / stringComparisonOperator ws (typedSearchTerm / typedSearchTermSet) / booleanComparisonOperator ws booleanValue)	

50 <https://confluence.ihtsdotools.org/display/DOCGLOSS/attribute+name>

51 <https://confluence.ihtsdotools.org/display/DOCGLOSS/attribute+group>

52 <https://confluence.ihtsdotools.org/display/DOCGLOSS/refinement>

53 <https://confluence.ihtsdotools.org/display/DOCGLOSS/expression>

	An attribute is a name ⁵⁴ -value pair expressing a single refinement ⁵⁵ of the containing expression ⁵⁶ constraint. Either the attribute value must satisfy (or not) the given expression constraint, the attribute value is compared with a given numeric value (integer or decimal) using a numeric comparison operator, the attribute value must match (or not match) the given typedSearchTerm or typedSearchTermSet, or the attribute value must be equal to (or not equal to) the given boolean value. The attribute may optionally be preceded by a cardinality constraint and/or a reverse flag.
BS/LS: cardinality = minValue to maxValue	
	The cardinality represents a constraint on the minimum and maximum number of times that the given attribute or attribute group may appear in a matching expression. The cardinality is enclosed in square brackets with the minimum cardinality appearing first, followed by a separator (two dots in the brief syntax), and then the maximum cardinality.
BS/LS: minValue = nonNegativeIntegerValue	
	A value that represents the minimum number of times that an attribute or attribute group may appear. The minimum cardinality must always be less than or equal to the maximum cardinality.
BS: to = ".." LS: to = ".." / (mws ("t"/"T") ("o"/"O") mws)	
	In the brief syntax, the minimum and maximum cardinality are separated by two dots (i.e. ".."). In the long syntax, the text "to" (case insensitive with at least one white space before and after) is also allowed between the two cardinalities.
BS/LS: maxValue = nonNegativeIntegerValue / many	
	A value that represents the maximum number of times that an attribute or attribute group may appear. A maximum cardinality of 'many' indicates that there is no limit on the number of times the attribute may appear.
BS: many = "*" LS: many = "*" / (("m"/"M") ("a"/"A") ("n"/"N") ("y"/"Y"))	
	In the brief syntax, a cardinality of 'many' is represented using the symbol "*". In the long syntax, the text "many" (case insensitive, with no trailing space) is also allowed.

⁵⁴ <https://confluence.ihtsdotools.org/display/DOCGLOSS/attribute+name>

⁵⁵ <https://confluence.ihtsdotools.org/display/DOCGLOSS/refinement>

⁵⁶ <https://confluence.ihtsdotools.org/display/DOCGLOSS/expression>

<p>BS: reverseFlag = "R"</p> <p>LS: reverseFlag = (("r"/"R") ("e"/"E") ("v"/"V") ("e"/"E") ("r"/"R") ("s"/"S") ("e"/"E") ("o"/"O") ("f"/"F")) / "R"</p>	
	<p>When a reverse flag is used on an attribute, the matching relationships are traversed in the reverse of the normal direction. This means that the target concept of each relationship must match the focus concept to which the attribute is applied, while the source concept of the relationship must match the attribute value. In the brief syntax, the reverse flag is represented using the character "R" (in uppercase). In the long syntax, the text "reverseOf" (case insensitive) is also allowed.</p>
<p>BS/LS: eclAttributeName = subExpressionConstraint</p>	
	<p>The attribute name is the name of an attribute (or relationship type) to which a value is applied to refine the meaning of a containing expression constraint. The attribute name is represented using a subExpressionConstraint, as defined above.</p>
<p>BS: expressionComparisonOperator = "=" / "!="</p> <p>LS: expressionComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" / "<>"</p>	
	<p>Attributes whose value is a concept may be compared to an expression constraint using either equals ("=") or not equals ("!="). In the long syntax "<>" and "not =" (case insensitive) are also valid ways to represent not equals.</p>
<p>BS: numericComparisonOperator = "=" / "!=" / "<=" / "<" / ">=" / ">"</p> <p>LS: numericComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" / "<>" / "<=" / "<" / ">=" / ">"</p>	
	<p>Attributes whose value is numeric (i.e. integer or decimal) may be compared to a specific concrete value using a variety of comparison operators, including equals ("="), less than ("<"), less than or equals ("<="), greater than (">"), greater than or equals (">=") and not equals ("!="). In the long syntax "<>" and "not =" (case insensitive) are also valid ways to represent not equals.</p>
<p>BS: timeComparisonOperator = "=" / "!=" / "<=" / "<" / ">=" / ">"</p> <p>LS: timeComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" / "<>" / "<=" / "<" / ">=" / ">"</p>	
	<p>Date and time values may be compared using a variety of comparison operators, including equals ("="), less than ("<"), less than or equals ("<="), greater than (">"), greater than or equals (">=") and not equals ("!="). In the long syntax "<>" and "not =" (case insensitive) are also valid ways to represent not equals.</p>
<p>BS: stringComparisonOperator = "=" / "!="</p> <p>LS: stringComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" / "<>"</p>	

	Attributes whose value is a string may be compared to an expression constraint using either equals ("=") or not equals ("!="). In the long syntax "<>" and "not =" (case insensitive) are also valid ways to represent not equals.
BS: booleanComparisonOperator = "=" / "!="	
LS: booleanComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" / "<>"	
	Attributes whose value is a boolean may be compared to an expression constraint using either equals ("=") or not equals ("!="). In the long syntax "<>" and "not =" (case insensitive) are also valid ways to represent not equals.
BS: idComparisonOperator = "=" / "!="	
LS: idComparisonOperator = "=" / "!=" / ("n"/"N") ("o"/"O") ("t"/"T") ws "=" / "<>"	
	Filter criteria whose value is a SCTID may be compared to a SNOMED CT identifier using either equals ("=") or not equals ("!="). In the long syntax "<>" and "not =" (case insensitive) are also valid ways to represent not equals.
BS/LS: descriptionFilterConstraint = "{" ws ["d", / "D"] ws descriptionFilter *(ws "," ws descriptionFilter) ws "}"	
	A descriptionFilterConstraint is a constraint used to filter the concepts in the result set, according to whether or not the given conditions match at least one of the concept's descriptions. A description filter constraint is always enclosed in double curly braces. Within these braces, it should (preferably) start with the letter 'D' followed by one or more description filters.
BS/LS: descriptionFilter = termFilter / languageFilter / typeFilter / dialectFilter / moduleFilter / effectiveTimeFilter / activeFilter / descriptionIdFilter	
	A description filter is either a term filter, a language filter, a type filter, a dialect filter, a module filter, an effective time filter, an active filter or a description id filter.
BS/LS: descriptionIdFilter = descriptionIdKeyword ws idComparisonOperator ws (descriptionId / descriptionIdSet)	
	A descriptionIdFilter starts with the 'id' keyword, followed by an id comparison operator and either a single description id or a set of description ids.
BS/LS: descriptionIdKeyword = ("i"/"I") ("d"/"D")	
	The description id keyword uses the text "id" (case insensitive)

BS/LS: descriptionId = sctId	
	The descriptionId must be a valid SNOMED CT identifier ⁵⁷ for a description ⁵⁸ . The initial digit may not be zero. The smallest number of digits is six, and the maximum is 18.
BS/LS: descriptionIdSet = "(" ws descriptionId *(mws descriptionId) ws ")"	
	A description id set consists of one or more description ids separated by mandatory white space and enclosed in brackets.
BS/LS: termFilter = termKeyword ws stringComparisonOperator ws (typedSearchTerm / typedSearchTermSet)	
	A termFilter starts with the 'term' keyword, followed by a string comparison operator and either a typed search term or a typed search term set (with optional white space between). For example: term = "respiratory".
BS/LS: termKeyword = ("t"/"T") ("e"/"E") ("r"/"R") ("m"/"M")	
	The term keyword uses the text "term" (case insensitive).
BS/LS: typedSearchTerm = ([matchKeyword ws ":" ws] matchSearchTermSet) / (wild ws ":" ws wildSearchTermSet)	
	A typed search term is either a match search term set or a wild search term set. A match search term set is optionally preceded by the text "match" and a colon. A wild search term set must be preceded by the text "wild" and a colon.
BS/LS: typedSearchTermSet = "(" ws typedSearchTerm *(mws typedSearchTerm) ws ")"	
	A typed search term set consists of one or more typed search terms separated by mandatory white space and enclosed in brackets.
BS/LS: wild = ("w"/"W") ("i"/"I") ("l"/"L") ("d"/"D")	
	A wildcard search type is indicated by the word "wild" (case insensitive).
BS/LS: matchKeyword = ("m"/"M") ("a"/"A") ("t"/"T") ("c"/"C") ("h"/"H")	
	A word prefix any order search is indicated by the word "match" (case insensitive).

⁵⁷ <https://confluence.ihtsdotools.org/display/DOCGLOSS/SNOMED+CT+identifier>

⁵⁸ <https://confluence.ihtsdotools.org/display/DOCGLOSS/description>

BS/LS: matchSearchTerm = 1*(nonwsNonEscapedChar / escapedChar)	
	A term used in a match search includes one or more of any non-whitespace printable character (other than double quotes or backslash) or an escaped character.
BS/LS: matchSearchTermSet = QM ws matchSearchTerm *(mws matchSearchTerm) ws QM	
	A term set in a match search includes one or more terms separated by mandatory whitespace and enclosed in quotation marks.
BS/LS: wildSearchTerm = 1*(anyNonEscapedChar / escapedWildChar)	
	A term used in a wildcard search includes one or more printable characters (other than double quotes or backslash) or an escaped character.
BS/LS: wildSearchTermSet = QM wildSearchTerm QM	
	A term set in a wildcard search includes a wildcard search term (optionally including whitespace) enclosed in quotation marks.
BS/LS: languageFilter = language ws booleanComparisonOperator ws (languageCode / languageCodeSet)	
	A language filter specifies the languages that a matching description may use. A language filter starts with the 'language' keyword, followed by a boolean comparison operator and either a single language code or a set of language codes.
BS/LS: language = ("l"/"L") ("a"/"A") ("n"/"N") ("g"/"G") ("u"/"U") ("a"/"A") ("g"/"G") ("e"/"E")	
	The 'language' keyword uses the text "LANGUAGE" (case insensitive).
BS/LS: languageCode = 2alpha	
	A language code is a 2 character alphanumeric string.
BS/LS: languageCodeSet = "(" ws languageCode *(mws languageCode) ws ")"	
	A language code set is one or more language codes, separated by mandatory whitespace, and enclosed in brackets.
BS/LS: typeFilter = typeldFilter / typeTokenFilter	

	A type filter specifies the description types that a matching description may have. A type filter is either a typeId filter or a typeToken filter.
BS/LS: typeIdFilter = typeId ws booleanComparisonOperator ws (subExpressionConstraint / eclConceptReferenceSet)	
	A typeId filter starts with the 'typeId' keyword, followed by a boolean comparison operator, and either a subExpressionConstraint or a set of concept references.
BS/LS: typeId = ("t"/"T") ("y"/"Y") ("p"/"P") ("e"/"E") ("i"/"I") ("d"/"D")	
	The 'typeId' keyword uses the text "TYPEID" (case insensitive).
BS/LS: typeTokenFilter = type ws booleanComparisonOperator ws (typeToken / typeTokenSet)	
	A typeToken filter starts with the 'type' keyword, followed by a boolean comparison operator, and either a single type token or a set of type tokens.
BS/LS: type = ("t"/"T") ("y"/"Y") ("p"/"P") ("e"/"E")	
	The 'type' keyword uses the text "TYPE" (case insensitive).
BS/LS: typeToken = synonym / fullySpecifiedName / definition	
	A type token is either a 'synonym' token, a 'fully specified name' token or a 'definition' token.
BS/LS: typeTokenSet = "(" ws typeToken *(mws typeToken) ws ")"	
	A type token set is one or more type tokens, separated by mandatory whitespace and enclosed in brackets.
BS: synonym = ("s"/"S") ("y"/"Y") ("n"/"N")	
LS: synonym = ("s"/"S") ("y"/"Y") ("n"/"N") [("o"/"O") ("n"/"N") ("y"/"Y") ("m"/"M")]	
	A 'synonym' token uses the text "SYN" (case insensitive). In the long syntax, the text "Synonym" (case insensitive) may be used instead.
BS: fullySpecifiedName = ("f"/"F") ("s"/"S") ("n"/"N")	
LS: fullySpecifiedName = (("f"/"F") ("s"/"S") ("n"/"N")) / (("f"/"F") ("u"/"U") ("l"/"L") ("l"/"L") ("y"/"Y") ("s"/"S") ("p"/"P") ("e"/"E") ("c"/"C") ("i"/"I") ("f"/"F") ("i"/"I") ("e"/"E") ("d"/"D") ("n"/"N") ("a"/"A") ("m"/"M") ("e"/"E"))	

	A 'fully specified name' token uses the text "FSN" (case insensitive). In the long syntax, the text "FullySpecifiedName" (case insensitive) may be used instead.
BS: definition = ("d"/"D") ("e"/"E") ("f"/"F")	
LS: definition = ("d"/"D") ("e"/"E") ("f"/"F") [("i"/"I") ("n"/"N") ("t"/"T") ("o"/"O") ("n"/"N")]	
	A 'definition' token uses the text "DEF" (case insensitive). In the long syntax, the text "Definition" (case insensitive) may be used instead.
BS/LS: dialectFilter = (dialectIdFilter / dialectAliasFilter) [ws acceptabilitySet]	
	A dialect filter specifies the language reference sets to which a matching description must belong. A dialect filter consists of either a dialectId filter or a dialectAlias filter, optionally followed by a set of acceptability values.
BS/LS: dialectIdFilter = dialectId ws booleanComparisonOperator ws (subExpressionConstraint / dialectIdSet)	
	A dialectId filter starts with the 'dialectId' keyword, followed by a boolean comparison operator, and either a subExpressionConstraint or a set of dialectIds.
BS/LS: dialectId = ("d"/"D") ("i"/"I") ("a"/"A") ("l"/"L") ("e"/"E") ("c"/"C") ("t"/"T") ("i"/"I") ("d"/"D")	
	A 'dialectId' keyword uses the text "DIALECTID" (case insensitive).
BS/LS: dialectAliasFilter = dialect ws booleanComparisonOperator ws (dialectAlias / dialectAliasSet)	
	A dialectAlias filter starts with the 'dialect' keyword, followed by a boolean comparison operator, and either a single dialect alias or a set of dialect aliases.
BS/LS: dialect = ("d"/"D") ("i"/"I") ("a"/"A") ("l"/"L") ("e"/"E") ("c"/"C") ("t"/"T")	
	A 'dialect' keyword uses the text "DIALECT" (case insensitive).
BS/LS: dialectAlias = alpha *(dash / alpha / integerValue)	
	A dialect alias consists of a single alphanumeric character followed by zero or more alphanumeric characters, integer values or dashes.
BS/LS: dialectAliasSet = "(" ws dialectAlias [ws acceptabilitySet] *(mws dialectAlias [ws acceptabilitySet]) ws ")"	
	A dialect alias set is one or more dialect aliases followed by an optional acceptability set, separated by mandatory white space, and enclosed in brackets.

BS/LS: dialectIdSet = "(" ws eclConceptReference [ws acceptabilitySet] *(mws eclConceptReference [ws acceptabilitySet]) ws ")"	
	A dialect id set is one or more concept references followed by an optional acceptability set, separated by mandatory white space, and enclosed in brackets.
BS/LS: acceptabilitySet = acceptabilityConceptReferenceSet / acceptabilityTokenSet	
	An acceptability set specifies the acceptabilities that a matching description must have in the language reference set specified by the preceding dialect filter. An acceptability set is either a set of one or more concept references or an acceptabilityToken set.
BS/LS: acceptabilityConceptReferenceSet = "(" ws eclConceptReference *(mws eclConceptReference) ws ")"	
	An acceptability concept reference set is a set of one or more references to concepts that are a < 900000000000511003 Acceptability .
BS/LS: acceptabilityTokenSet = "(" ws acceptabilityToken *(mws acceptabilityToken) ws ")"	
	An acceptability token set is one or more acceptability tokens, separated by mandatory whitespace, and enclosed in brackets.
BS/LS: acceptabilityToken = acceptable / preferred	
	An acceptability token is either an acceptable token and a preferred token.
BS: acceptable = ("a"/"A") ("c"/"C") ("c"/"C") ("e"/"E") ("p"/"P") ("t"/"T") LS: acceptable = ("a"/"A") ("c"/"C") ("c"/"C") ("e"/"E") ("p"/"P") ("t"/"T") [("a"/"A") ("b"/"B") ("l"/"L") ("e"/"E")]	
	An acceptable token uses the text "ACCEPT" (case insensitive). In the long syntax, the text "Acceptable" (case insensitive) may be used instead.
BS: preferred = ("p"/"P") ("r"/"R") ("e"/"E") ("f"/"F") ("e"/"E") ("r"/"R") LS: preferred = ("p"/"P") ("r"/"R") ("e"/"E") ("f"/"F") ("e"/"E") ("r"/"R") [("r"/"R") ("e"/"E") ("d"/"D")]	
	A preferred token uses the text "PREFER" (case insensitive). In the long syntax, the text "Preferred" (case insensitive) may be used instead.
BS/LS: conceptFilterConstraint = "{{" ws ("c" / "C") ws conceptFilter *(ws " , " ws conceptFilter) ws "}"	

	<p>A concept filter constraint is a constraint used to filter the concepts in the result set, according to whether or not the concept matches the given conditions. A concept filter constraint is always enclosed in double curly braces. Within these braces, it starts with the letter 'C' followed by one or more constraint filters.</p>
<p>BS/LS: conceptFilter = definitionStatusFilter / moduleFilter / effectiveTimeFilter / activeFilter</p>	
	<p>A concept filter is either a definition status filter, a module filter, an effective time filter or an active filter.</p>
<p>BS/LS: definitionStatusFilter = definitionStatusIdFilter / definitionStatusTokenFilter</p>	
	<p>A definition status filter is constraint that either filters the results of a query, based on each concept's definition status identifier or a token.</p>
<p>BS/LS: definitionStatusIdFilter = definitionStatusIdKeyword ws booleanComparisonOperator ws (subExpressionConstraint / eclConceptReferenceset)</p>	
	<p>A definition status filter is a constraint that filters the results of a query, based on whether or not each concept's definition status matches a given identifier. The filter starts with the keyword "definitionStatusId", followed by a boolean comparison operator and either a subexpression constraint or a set of concept references that are a subtype of 900000000000444006 Definition status⁵⁹.</p>
<p>BS/LS: definitionStatusIdKeyword = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N") ("i"/"I") ("t"/"T") ("i"/"I") ("o"/"O") ("n"/"N") ("s"/"S") ("t"/"T") ("a"/"A") ("t"/"T") ("u"/"U") ("s"/"S") ("i"/"I") ("d"/"D")</p>	
	<p>The definition status id keyword is the text "definitionStatusId" (in any combination of upper or lower case).</p>
<p>BS/LS: definitionStatusTokenFilter = definitionStatusKeyword ws booleanComparisonOperator ws (definitionStatusToken / definitionStatusTokenSet)</p>	
	<p>A definition status filter is a constraint that filters the results of a query, based on whether or not each concept's definition status matches a given token.</p>
<p>BS/LS: definitionStatusKeyword = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N") ("i"/"I") ("t"/"T") ("i"/"I") ("o"/"O") ("n"/"N") ("s"/"S") ("t"/"T") ("a"/"A") ("t"/"T") ("u"/"U") ("s"/"S")</p>	
	<p>The definition status keyword is the text "definitionStatus" (in any combination of upper or lower case).</p>

⁵⁹ <http://snomed.info/id/900000000000444006>

BS/LS: definitionStatusToken = primitiveToken / definedToken	
	A definition status token is either a primitive token or a defined token.
BS/LS: definitionStatusTokenSet = "(" ws definitionStatusToken *(mws definitionStatusToken) ws ")"	
	A definition status token set consists of one or more definition status tokens separated by mandatory white space and enclosed in brackets.
BS/LS: primitiveToken = ("p"/"P") ("r"/"R") ("i"/"I") ("m"/"M") ("t"/"T") ("v"/"V") ("e"/"E")	
	A primitive token represents the definition status 900000000000074008 Primitive ⁶⁰ using the text "primitive" (in any combination of upper and lower case characters).
BS/LS: definedToken = ("d"/"D") ("e"/"E") ("f"/"F") ("i"/"I") ("n"/"N") ("e"/"E") ("d"/"D")	
	A defined token represents the definition status 900000000000073002 Defined ⁶¹ using the text "defined" (in any combination of upper and lower case characters).
BS/LS: moduleFilter = moduleIdKeyword ws booleanComparisonOperator ws (subExpressionConstraint / eclConceptReferenceSet)	
	A module filter is a constraint that filters the results of a query based on the module to which each concept belongs. The filter starts with the keyword "moduleId", followed by a boolean comparison operator and either a subexpression constraint or a set of concept references that are a subtype of 9000000000000443000 Module ⁶² .
BS/LS: moduleIdKeyword = ("m"/"M") ("o"/"O") ("d"/"D") ("u"/"U") ("l"/"L") ("e"/"E") ("i"/"I") ("d"/"D")	
	The module id keyword is the text "moduleId" (in any combination of upper or lower case).
BS/LS: effectiveTimeFilter = effectiveTimeKeyword ws timeComparisonOperator ws (timeValue / timeValeSet)	
	An effective time filter is a constraint that filters the results of a query based on the effective time assigned to each concept.

60 <http://snomed.info/id/900000000000074008>

61 <http://snomed.info/id/900000000000073002>

62 <http://snomed.info/id/9000000000000443000>

BS/LS: effectiveTimeKeyword = ("e"/"E") ("f"/"F") ("f"/"F") ("e"/"E") ("c"/"C") ("t"/"T") ("i"/"I") ("v"/"V") ("e"/"E") ("t"/"T") ("i"/"I") ("m"/"M") ("e"/"E")	
	The effective time keyword is the text "effectiveTime" (in any combination of upper or lower case).
BS/LS: timeValue = QM [year month day] QM	
	A time value is a 8 digit string that represents the year, month and day of a specific date.
BS/LS: timeValueSet = "(" ws timeValue *(mws timeValue) ws ")"	
	A time value set consists of one or more time values separated by mandatory white space and enclosed in brackets.
BS/LS: year = digitNonZero digit digit digit	
	A year is a 4 digit string starting with a non-zero digit.
BS/LS: month = "01" / "02" / "03" / "04" / "05" / "06" / "07" / "08" / "09" / "10" / "11" / "12"	
	A month is a 2 digit string from "01" to "12" that represents a specific month of the year (e.g. "01" represents January)
BS/LS: day = "01" / "02" / "03" / "04" / "05" / "06" / "07" / "08" / "09" / "10" / "11" / "12" / "13" / "14" / "15" / "16" / "17" / "18" / "19" / "20" / "21" / "22" / "23" / "24" / "25" / "26" / "27" / "28" / "29" / "30" / "31"	
	A day is a 2 digit string from "01" to "31" that represents a specific day within a month of a year.
BS/LS: activeFilter = activeKeyword ws booleanComparisonOperator ws activeValue	
	An active filter is a constraint that filters the results of a query based on the active status of each concept
BS/LS: activeKeyword = ("a"/"A") ("c"/"C") ("t"/"T") ("i"/"I") ("v"/"V") ("e"/"E")	
	The active keyword is the text "active" (in any combination of upper or lower case).
BS/LS: activeValue = activeTrueValue / activeFalseValue	

	An active value represents the active status of a concept, and is either true (i.e. the concept is active) or false (i.e. the concept is inactive).
BS/LS: activeTrueValue = "1" / "true"	
	An active true value is a value that represents an active concept. This value is either "1" or "true".
BS/LS: activeFalseValue = "0" / "false"	
	An active false value is a value that represents an inactive concept. This value is either "0" or "false".
BS/LS: memberFilterConstraint = "{{" ws ("m" / "M") ws memberFilter *(ws "," ws memberFilter) ws "}"	
	A member filter constraint is a constraint used to filter the rows in one or more result sets, according to values of particular fields. A member filter constraint is always surrounded by double curly braces. Within these braces, it starts with the letter 'M' followed by one or more member filters.
BS/LS: memberFilter = moduleFilter / effectiveTimeFilter / activeFilter / memberFieldFilter	
	A member filter is either a module filter, an effective time filter, an active filter, or a member field filter.
BS/LS: memberFieldFilter = refsetFieldName ws (expressionComparisonOperator ws subExpressionConstraint / numericComparisonOperator ws "#" numericValue / stringComparisonOperator ws (typedSearchTerm / typedSearchTermSet) / booleanComparisonOperator ws booleanValue / ws timeComparisonOperator ws (timeValue / timeValueSet))	
	A member field filter always has three parts - (1) the reference set field name, (2) a comparison operator, and (3) the criteria on which to match the field's value. If the refset field is of type SNOMED CT concept, then an expression comparison operator is used, followed by a subexpression constraint. If the refset field is a numeric type, then a numeric comparison operator is used, followed by a hash symbol("#") and a numeric value. If the refset field is of type string, then a string comparison operator is used, followed by a typed search term or a typed search term set. If the refset field is of type boolean, then a boolean comparison operator is used, followed by a boolean value. And if the refset field is of type dateTime, then a time comparison operator is used, followed by a time value or time value set.
BS/LS: historySupplement = "{" ws "+" ws historyKeyword [historyProfileSuffix / ws historySubset] ws "}"	

	A history supplement augments the results of the expression constraint with relevant inactive concepts. A history supplement is always surrounded by double curly braces. Within these braces, it starts with a plus symbol (i.e. "+"), followed by the history keyword. The history keyword is optionally followed by either a profile suffix, or a history subset.
BS/LS: historyKeyword = ("h"/"H") ("i"/"I") ("s"/"S") ("t"/"T") ("o"/"O") ("r"/"R") ("y"/"Y")	
	The history keyword is the word "HISTORY" (case insensitive).
BS/LS: historyProfileSuffix = historyMinimumSuffix / historyModerateSuffix / historyMaximumSuffix	
	A history profile suffix is either the suffix for history minimum, history moderate or history maximum.
BS/LS: historyMinimumSuffix = ("-"/"_")("m"/"M") ("i"/"I") ("n"/"N")	
	The history minimum suffix is "-MIN" (case insensitive). The suffix may start with either a hyphen (i.e. "-") or an underscore (i.e. "_").
BS/LS: historyModerateSuffix = ("-"/"_")("m"/"M") ("o"/"O") ("d"/"D")	
	The history moderate suffix is "-MOD" (case insensitive). The suffix may start with either a hyphen (i.e. "-") or an underscore (i.e. "_").
BS/LS: historyMaximumSuffix = ("-"/"_")("m"/"M") ("a"/"A") ("x"/"X")	
	The history maximum suffix is "-MAX" (case insensitive). The suffix may start with either a hyphen (i.e. "-") or an underscore (i.e. "_").
BS/LS: historySubset = "(" ws expressionConstraint ws ")"	
	A history subset is an expression constraint that defines a set of historical association reference sets, surrounded by round brackets. Only descendants of 900000000000522004 Historical association reference set ⁶³ may be included in a history subset.
BS/LS: numericValue = ["-"/"+"] (decimalValue / integerValue)	
	A numeric value is either an integer or a decimal. Positive numbers optionally start with a plus sign ("+"), while negative integers begin with a minus sign ("-").
BS/LS: stringValue = 1*(anyNonEscapedChar / escapedChar)	

63 <http://snomed.info/id/900000000000522004>

	A string value includes one or more of any printable ASCII characters enclosed in quotation marks. Quotes and backslash characters within the string must be preceded by the escape character ("\").
BS/LS: integerValue = digitNonZero *digit / zero	
	An integer value is either starts with a non-zero digit followed by zero to many additional digits, or is the integer zero itself.
BS/LS: decimalValue = integerValue "." 1*digit	
	A decimal value starts with an integer. This is followed by a decimal point and one to many digits.
BS/LS: booleanValue = true / false	
	A boolean value is either true or false.
BS/LS: true = ("t"/"T") ("r"/"R") ("u"/"U") ("e"/"E")	
	A boolean value of true is represented by the word "true" (case insensitive).
BS/LS: false = ("f"/"F") ("a"/"A") ("l"/"L") ("s"/"S") ("e"/"E")	
	A boolean value of false is represented by the word "false" (case insensitive).
BS/LS: nonNegativeIntegerValue = (digitNonZero *digit) / zero	
	A non-negative integer value (i.e. positive integers or zero), without a preceding plus sign ("+").
BS/LS: sctId = digitNonZero 5*17(digit)	
	A SNOMED CT id is used to represent an attribute id or a concept ⁶⁴ id. The initial digit may not be zero. The smallest number of digits is six, and the maximum is 18.
BS/LS: ws = *(SP / HTAB / CR / LF / comment)	

⁶⁴ <https://confluence.ihtsdotools.org/display/DOCGLOSS/concept>

	<p>Optional whitespace characters (space, tab, carriage return, linefeed or a comment) are ignored everywhere in the expression⁶⁵ except:</p> <ol style="list-style-type: none"> 1. Whitespace within a conceptId is an error. Note: Whitespace before or after the last digit of a valid Identifier⁶⁶ is ignored. 2. Non-consecutive spaces within a term are treated as a significant character of the term. Note: Whitespace before the first or after the last non-whitespace character of a term⁶⁷ is ignored 3. Whitespace within the quotation marks of a concrete value is treated as a significant character.
<p>BS/LS: mws = 1*(SP / HTAB / CR / LF / comment)</p>	
	<p>Mandatory whitespace (i.e. space, tab, carriage return, linefeed or a comment) is required after certain keywords, including "And" and "Or".</p>
<p>BS/LS: comment = "/"* *(nonStarChar / starWithNonLSlash) "*" /</p>	
	<p>A comment, which provides additional human-readable details about the expression constraint. Comments begin with a forward slash directly followed by a star (i.e. "/"*) and end with a star directly followed by a forward slash (i.e. "*" /).</p>
<p>BS/LS: nonStarChar = SP / HTAB / CR / LF / %x21-29 / %x2B-7E / UTF8-2 / UTF8-3 / UTF8-4</p>	
	<p>A character that is not a star (i.e. not %x2A).</p>
<p>BS/LS: starWithNonLSlash = %x2A nonLSlash</p>	
	<p>A star (i.e. "*"*) followed by a character that is not a forward slash (i.e. not "/").</p>
<p>BS/LS: nonLSlash = SP / HTAB / CR / LF / %x21-2E / %x30-7E / UTF8-2 / UTF8-3 / UTF8-4</p>	
	<p>A character that is not a forward slash (i.e. not "/").</p>
<p>BS/LS: SP = %x20</p>	
	<p>Space character.</p>
<p>BS/LS: HTAB = %x09</p>	

⁶⁵ <https://confluence.ihtsdotools.org/display/DOCGLOSS/expression>

⁶⁶ <https://confluence.ihtsdotools.org/display/DOCGLOSS/Identifier>

⁶⁷ [https://confluence.ihtsdotools.org/display/DOCRELFMT/term+\(field\)](https://confluence.ihtsdotools.org/display/DOCRELFMT/term+(field))

	Tab character.
BS/LS: CR = %x0D	
	Carriage return character.
BS/LS: LF = %x0A	
	Line feed character.
BS/LS: QM = %x22	
	Quotation mark character.
BS/LS: BS = %x5C ; back slash	
	BS represents the backslash character "\".
BS/LS: star = %x2A ; asterisk	
	Star represents an asterisk "*".
BS/LS: digit = %x30-39	
	Any digit 0 through 9.
BS/LS: zero = %x30	
	The digit 0.
BS/LS: digitNonZero = %x31-39	
	Digits 1 through 9, but excluding 0. The first character of a concept identifier ⁶⁸ is constrained to a digit other than zero.
BS/LS: nonwsnonpipe = %x21-7B / %x7D-7E / UTF8-2 / UTF8-3 / UTF8-4	

⁶⁸ <https://confluence.ihtsdotools.org/display/DOCGLOSS/concept+identifier>

	Non whitespace (and non pipe) includes printable ASCII characters (these are also valid UTF8 characters encoded as one octet) and also includes all UTF8 characters encoded as 2- 3- or 4-octet sequences. It excludes space (which is %x20) and the pipe character "
BS/LS: anyNonEscapedChar = SP / HTAB / CR / LF / %x20-21 / %x23-5B / %x5D-7E / UTF8-2 / UTF8-3 / UTF8-4	
	anyNonEscapedChar includes any printable ASCII characters which do not need to be preceded by an escape character (i.e. "\"). This includes valid UTF8 characters encoded as one octet and all UTF8 characters encoded as 2, 3 or 4 octet sequences. It does, however, exclude the quotation mark (") and the backslash (). See RFC 3629 (UTF-8 ⁶⁹ , a transformation ⁷⁰ format of ISO ⁷¹ 10646 authored by the Network Working Group).
BS/LS: escapedChar = BS QM / BS BS	
	The double quotation mark and the back slash character must both be escaped within a string-based concrete value by preceding them with a back slash.
BS/LS: escapedWildChar = BS QM / BS BS / BS star	
	An escapedWildChar is one of the characters that must be escaped in a wildcard search term (i.e. " or \ or *), preceded by a backslash (i.e. \). The character sequence is therefore either \" or \\ or *.
BS/LS: nonwsNonEscapedChar = %x21 / %x23-5B / %x5D-7E / UTF8-2 / UTF8-3 / UTF8-4	
	A nonwsNonEscapedChar is any printable ASCII, UTF8-2, UTF8-3 or UTF8-4 character, excluding double quotes ("), backslash (\), and space ().
BS/LS: alpha = %x41-5A / %x61-7A	
	An alpha is any uppercase or lowercase character from "A" to "Z" (and "a" to "z") inclusive.
BS/LS: dash = %x2D	
	A dash is a hyphen (i.e. "-").
BS/LS: UTF8-2 = %xC2-DF UTF8-tail	
	UTF8 characters encoded as 2-octet sequences.

⁶⁹ <https://confluence.ihtsdotools.org/display/DOCRELFMT/UTF-8>

⁷⁰ <https://confluence.ihtsdotools.org/display/DOCGLOSS/transformation>

⁷¹ <https://confluence.ihtsdotools.org/display/DOCGLOSS/ISO>

BS/LS: UTF8-3 = %xE0 %xA0-BF UTF8-tail / %xE1-EC 2(UTF8-tail) / %xED %x80-9F UTF8-tail / %xEE-EF 2(UTF8-tail)	
	UTF8 characters encoded as 3-octet sequences.
BS/LS: UTF8-4 = %xF0 %x90-BF 2(UTF8-tail) / %xF1-F3 3(UTF8-tail) / %xF4 %x80-8F 2(UTF8-tail)	
	UTF8 characters encoded as 4-octet sequences.
BS/LS: UTF8-tail = %x80-BF	
	UTF8 characters encoded as 8-octet sequences.

5.4 5.4 Order of Operation

This section explains the correct order of operation for unary operators, binary operators, filters and supplements.

5.4.1 Unary Operators

Unary operators (e.g. descendantOf, descendantOrSelfOf, ancestorOf, ancestorOrSelfOf, memberOf) are applied from inside to out (i.e. from right to left). For example, when the following expression constraint is processed, the memberOf operator is applied first to the Example problem list concepts reference set, and then the descendants of the referenced components are determined.

```
< ^ 700043003 |Example problem list concepts reference set|72
```

5.4.2 Binary Operators

Whenever potential ambiguity in binary operator precedence may occur, round brackets must be used to clearly disambiguate the order in which these operators are applied. For example, the following expression constraint is not valid:

```
< 19829001 |Disorder of lung|73 OR ^ 700043003 |Example problem list concepts reference set|74
MINUS ^ 450976002 |Disorders and diseases reference set for GP/FP reason for encounter|75
```

And must be expressed using brackets, as either:

⁷² <http://snomed.info/id/700043003>

⁷³ <http://snomed.info/id/19829001>

⁷⁴ <http://snomed.info/id/700043003>

⁷⁵ <http://snomed.info/id/450976002>


```
( < 19829001 | Disorder of lung76 OR ^ 700043003 | Example problem list concepts reference set77 )
  MINUS ^ 450976002 | Disorders and diseases reference set for GP/FP reason for encounter78
```

or:

```
< 19829001 | Disorder of lung79 OR ( ^ 700043003 | Example problem list concepts reference set80
  MINUS ^ 450976002 | Disorders and diseases reference set for GP/FP reason for encounter81 )
```

When multiple exclusion operators (i.e. 'minus') are applied, brackets are similarly required. For example, the following expression constraint is not valid:

```
< 19829001 | Disorder of lung82 MINUS ^ 700043003 | Example problem list concepts reference set83
  MINUS ^ 450976002 | Disorders and diseases reference set for GP/FP reason for encounter84
```

And must be expressed using brackets, as either:

```
( < 19829001 | Disorder of lung85 MINUS ^ 700043003 | Example problem list concepts reference set86 )
  MINUS ^ 450976002 | Disorders and diseases reference set for GP/FP reason for encounter87
```

or:

```
< 19829001 | Disorder of lung88 MINUS ( ^ 700043003 | Example problem list concepts reference set89
  MINUS ^ 450976002 | Disorders and diseases reference set for GP/FP reason for encounter90 )
```

However, when only a single binary operator is used, or when all binary operators are either conjunction (i.e. 'and') or disjunction (i.e. 'or'), brackets are not required. For example, all of the following expression constraints are valid without brackets:

76 <http://snomed.info/id/19829001>

77 <http://snomed.info/id/700043003>

78 <http://snomed.info/id/450976002>

79 <http://snomed.info/id/19829001>

80 <http://snomed.info/id/700043003>

81 <http://snomed.info/id/450976002>

82 <http://snomed.info/id/19829001>

83 <http://snomed.info/id/700043003>

84 <http://snomed.info/id/450976002>

85 <http://snomed.info/id/19829001>

86 <http://snomed.info/id/700043003>

87 <http://snomed.info/id/450976002>

88 <http://snomed.info/id/19829001>

89 <http://snomed.info/id/700043003>

90 <http://snomed.info/id/450976002>

< 19829001 |Disorder of lung|⁹¹ AND ^ 700043003 |Example problem list concepts reference set|⁹²

< 19829001 |Disorder of lung|⁹³ OR ^ 700043003 |Example problem list concepts reference set|⁹⁴

< 19829001 |Disorder of lung|⁹⁵ MINUS ^ 700043003 |Example problem list concepts reference set|⁹⁶

< 19829001 |Disorder of lung|⁹⁷ OR ^ 700043003 |Example problem list concepts reference set|⁹⁸
OR ^ 450976002 |Disorders and diseases reference set for GP/FP reason for encounter|⁹⁹

< 19829001 |Disorder of lung|¹⁰⁰ AND ^ 700043003 |Example problem list concepts reference set|¹⁰¹
AND ^ 450976002 |Disorders and diseases reference set for GP/FP reason for encounter|¹⁰²

Please note that unary operators are always applied before binary operators.

5.4.3 Filter Constraints

Filter constraints (e.g. concept, description, or member filters) apply only to the sub-expression constraint part that is directly to the left of the filter.

For example, the following expression constraint will apply the term filter to only the descendants or self of 415582006 |Stenosis|¹⁰³. This expression constraint will match descendants of 404684003 |Clinical finding|¹⁰⁴ with

91 <http://snomed.info/id/19829001>

92 <http://snomed.info/id/700043003>

93 <http://snomed.info/id/19829001>

94 <http://snomed.info/id/700043003>

95 <http://snomed.info/id/19829001>

96 <http://snomed.info/id/700043003>

97 <http://snomed.info/id/19829001>

98 <http://snomed.info/id/700043003>

99 <http://snomed.info/id/450976002>

100 <http://snomed.info/id/19829001>

101 <http://snomed.info/id/700043003>

102 <http://snomed.info/id/450976002>

103 <http://snomed.info/id/415582006>

104 <http://snomed.info/id/404684003>

a finding site that is a descendant or self of 39057004 | Pulmonary valve structure¹⁰⁵, and an associated morphology that is any descendant or self of 415582006 | Stenosis¹⁰⁶ which has a description matching the term "insufficiency". Therefore, the concept 123801008 | Heart valve stenosis and regurgitation (disorder)¹⁰⁷ will match this expression constraint because it has the associated morphology 708027006 | Valvular stenosis with valvular insufficiency¹⁰⁸.

```
< 404684003 |Clinical finding109 :
  363698007 |Finding site110 = << 39057004 |Pulmonary valve structure111 ,
  116676008 |Associated morphology112 = << 415582006 |Stenosis113 {{ term = "insufficiency" }}
```

To apply a filter to a sub-expression constraint, which includes a refinement or binary operators, the subexpression must be enclosed in brackets. For example, the following expression constraint will find all the descendants of clinical finding, with a finding site that is a descendant or self of 39057004 | Pulmonary valve structure¹¹⁴ and an associated morphology that is a descendant or self of 415582006 | Stenosis¹¹⁵, and will then match only those clinical finding concepts that have a description that matches the term "insufficiency". Therefore, the concept 123801008 | Heart valve stenosis and regurgitation (disorder)¹¹⁶ will **not** match this expression constraints, as it does not have a description that matches the term "insufficiency".

```
(< 404684003 |Clinical finding117 :
  363698007 |Finding site118 = << 39057004 |Pulmonary valve structure119 ,
  116676008 |Associated morphology120 = << 415582006 |Stenosis121) {{ term = "insufficiency" }}
```

5.4.4 History Supplements

History supplements are applied only to the sub-expression constraint part that is directly to its left, after any filter constraints on this sub-expression constraint part have been applied.

For example, the following expression constraint will match all concepts that are **both** an active member of the 734139008 | Anatomy structure and part association reference set¹²² **and** also either an active member of the 734138000 | Anatomy structure and entire association reference set¹²³ or an inactive concept associated with an

105 <http://snomed.info/id/39057004>

106 <http://snomed.info/id/415582006>

107 <http://snomed.info/id/123801008>

108 <http://snomed.info/id/708027006>

109 <http://snomed.info/id/404684003>

110 <http://snomed.info/id/363698007>

111 <http://snomed.info/id/39057004>

112 <http://snomed.info/id/116676008>

113 <http://snomed.info/id/415582006>

114 <http://snomed.info/id/39057004>

115 <http://snomed.info/id/415582006>

116 <http://snomed.info/id/123801008>

117 <http://snomed.info/id/404684003>

118 <http://snomed.info/id/363698007>

119 <http://snomed.info/id/39057004>

120 <http://snomed.info/id/116676008>

121 <http://snomed.info/id/415582006>

122 <http://snomed.info/id/734139008>

123 <http://snomed.info/id/734138000>

active member of the 734138000 | Anatomy structure and entire association reference set¹²⁴ via the 900000000000527005 | SAME AS association reference set¹²⁵. Because all active members of the 734139008 | Anatomy structure and part association reference set¹²⁶ are active, there will be no inactive concepts in the result set.

```

^ 734139008 |Anatomy structure and part association reference set127
AND ^ 734138000 |Anatomy structure and entire association reference set128
  {{ + HISTORY ( 900000000000527005 |SAME AS association reference set129 ) }}

```


To apply the history supplement to the entire sub-expression constraint above, the sub-expression constraint must be enclosed in round brackets. For example, the following expression constraint will match concepts that are **both** members of the 734139008 | Anatomy structure and part association reference set¹³⁰ **and** also members of the 734138000 | Anatomy structure and entire association reference set¹³¹; and it will also match on any inactive concept that is associated via a 900000000000527005 | SAME AS association reference set¹³² to a member of both reference sets.

```

( ^ 734139008 |Anatomy structure and part association reference set133
  AND ^ 734138000 |Anatomy structure and entire association reference set134 )
  {{ + HISTORY ( 900000000000527005 |SAME AS association reference set135 ) }}

```

5.5 5.5 Character Collation for Term Filters

 This page is published as **Draft for Trial Use**. The recommendations on this page will be reviewed and may be updated following feedback from implementation experiences.

To promote consistency between implementations of ECL, the following collation principles are recommended:

- **Search and match** - The default behaviour of a system implementing ECL queries with term filters, is to use locale-specific asymmetric searching at the secondary comparison strength level -as specified in the [Unicode Technical Standard #10 - Unicode Collation Algorithm](#)¹³⁶. This means that the search is, by default, case insensitive, with some language-specific character normalization behaviour.
 - *Asymmetric*: Asymmetric searches require characters in the query that are unmarked (i.e. the 'base letters') to match characters in the target that are either *marked* or *unmarked* (with the same base letter). However, a character in the query that is *marked* will only match a character in the target that is *marked* in the same way.

124 <http://snomed.info/id/734138000>

125 <http://snomed.info/id/900000000000527005>

126 <http://snomed.info/id/734139008>

127 <http://snomed.info/id/734139008>

128 <http://snomed.info/id/734138000>

129 <http://snomed.info/id/900000000000527005>

130 <http://snomed.info/id/734139008>

131 <http://snomed.info/id/734138000>

132 <http://snomed.info/id/900000000000527005>

133 <http://snomed.info/id/734139008>

134 <http://snomed.info/id/734138000>

135 <http://snomed.info/id/900000000000527005>

136 http://www.unicode.org/reports/tr10/#Asymmetric_Search_Secondary

- *Secondary strength*: Searches with a strength of secondary will only consider level 1 differences (e.g. "d" vs "e") and level 2 differences (e.g. "e" vs "é" in English). However, level 3 differences (e.g. "e" vs "E") are not considered. This provides the same effect as queries being case insensitive. For example, in English, "e" in the query will match both "e" and "E" in the target; and "E" in the query will similarly match both "e" and "E" in the target.
- **Language customizations** - Locale-based customizations of the standard are specified in the [Unicode Common Locale Data Repository \(CLDR\)](#)¹³⁷. The unicode CLDR specifies the characters that are considered to be 'marked' variants of the base letters, identical base letters, and/or contractions in each specified language. The description terms in the substrate should be indexed separately for each language supported.

For example, the following search behaviour is expected in the locales specified below.

- In **English, Swedish and Danish**, the following search behaviour is expected:

Note: No customizations are made in these 3 locales for the characters used in these searches. Therefore, the [CLDR root collation order](#)¹³⁸ is used.

Search Term	Target Matches	Target does NOT Match
resume	resume, Resume, RESUME, résumé, rèsùmè, Résumé, RÉSUMÉ, ...	-
Resume	resume, Resume, RESUME, résumé, rèsùmè, Résumé, RÉSUMÉ, ...	-
résumé	résumé, Résumé, RÉSUMÉ, ...	resume, Resume, RESUME, ...
Résumé	résumé, Résumé, RÉSUMÉ, ...	resume, Resume, RESUME, ...

- In **English**, the following search behaviour is expected (based on the [CLDR 'en' locale](#)¹³⁹, which uses the [CLDR root collation order](#)¹⁴⁰):

Search Term	Target Matches	Target does NOT Match
sjogren	sjogren, Sjogren, SJOGREN, sjögren, Sjögren, SJÖGREN, sjøgren, Sjøgren, SJØGREN, ...	-
sjögren	sjögren, Sjögren, SJÖGREN, ...	sjogren, Sjogren, SJOGREN, sjøgren, Sjøgren, SJØGREN, ...
Angstrom	angstrom, Angstrom, ANGSTROM, ångström, Ångström, ÅNGSTRÖM, ångstrøm, Ångstrøm, ÅNGSTRØM, ...	ångstrœm, Ångstrœm, ÅNGSTRÆM, ...

¹³⁷ <http://cldr.unicode.org/index/cldr-spec/collation-guidelines>

¹³⁸ https://unicode.org/reports/tr35/tr35-collation.html#Root_Collation

¹³⁹ <https://github.com/unicode-org/cldr/blob/master/common/collation/en.xml>

¹⁴⁰ https://unicode.org/reports/tr35/tr35-collation.html#Root_Collation

Search Term	Target Matches	Target does NOT Match
Ångström	ångström, Ångström, ÅNGSTRÖM, ...	angstrom, Angstrom, ANGSTROM, ångstrøm, Ångstrøm, ÅNGSTRØM, ...
Ångstrøm	ångstrøm, Ångstrøm, ÅNGSTRØM, ...	angstrom, Angstrom, ANGSTROM, ångström, Ångström, ÅNGSTRÖM, ...
aangstrøm	aangstrøm, Aangstrøm, AANGSTRØM, ...	angstrom, Angstrom, ANGSTROM, ångström, Ångström, ÅNGSTRÖM, ångstrøm, Ångstrøm, ÅNGSTRØM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...

- In **Swedish**, the following search behaviour is expected (based on the customizations in the [CLDR 'sv' locale](#)¹⁴¹):

Search Term	Target Matches	Target does NOT Match
sjogren	sjogren, Sjogren, SJOGREN, ...	sjögren, Sjögren, SJÖGREN, sjøgren, Sjøgren, SJØGREN, ...
sjögren	sjögren, Sjögren, SJÖGREN, sjøgren, Sjøgren, SJØGREN, ...	sjogren, Sjogren, SJOGREN, ...
Angstrom	angstrom, Angstrom, ANGSTROM, ...	ångström, Ångström, ÅNGSTRÖM, ångstrøm, Ångstrøm, ÅNGSTRØM, ångstrøem, Ångstrøem, ÅNGSTRÆM, aangström, Aangstrøm, AANGSTRÖM, ...
Ångström	ångström, Ångström, ÅNGSTRÖM, ångstrøm, Ångstrøm, ÅNGSTRØM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...	angstrom, Angstrom, ANGSTROM, aangström, Aangstrøm, AANGSTRÖM, ...
Ångstrøm	ångstrøm, Ångstrøm, ÅNGSTRØM, ...	angstrom, Angstrom, ANGSTROM, ångström, Ångström, ÅNGSTRÖM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...

141 <https://github.com/unicode-org/cldr/blob/master/common/collation/sv.xml>

Search Term	Target Matches	Target does NOT Match
aangstrøm	aangstrøm, Aangstrøm, AANGSTRØM, ...	angstrom, Angstrom, ANGSTROM, ångström, Ångström, ÅNGSTRÖM, ångstrøm, Ångstrøm, ÅNGSTRØM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...

- And in **Danish**, the following search behaviour is expected (based on the customizations in the CLDR 'da' locale¹⁴²):

Search Term	Target Matches	Target does NOT Match
sjogren	sjogren, Sjogren, SJOGREN, ...	sjögren, Sjögren, SJÖGREN, sjøgren, Sjøgren, SJØGREN, ...
sjögren	sjögren, Sjögren, SJÖGREN, ...	sjogren, Sjogren, SJOGREN, sjøgren, Sjøgren, SJØGREN, ...
Angstrom	angstrom, Angstrom, ANGSTROM, ...	ångström, Ångström, ÅNGSTRÖM, ångstrøm, Ångstrøm, ÅNGSTRØM, ångstrøem, Ångstrøem, ÅNGSTRÆM, aangstrøm, Aangstrøm, AANGSTRØM ...
Ångström	ångström, Ångström, ÅNGSTRÖM, aangström, Aangström, AANGSTRÖM, ...	angstrom, Angstrom, ANGSTROM, ångstrøm, Ångstrøm, ÅNGSTRØM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...
Ångstrøm	ångstrøm, Ångstrøm, ÅNGSTRØM, ångström, Ångström, ÅNGSTRÖM, aangstrøm, Aangstrøm, AANGSTRØM, aangström, Aangström, AANGSTRÖM, ...	angstrom, Angstrom, ANGSTROM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...
aangstrøm	ångstrøm, Ångstrøm, ÅNGSTRØM, ångström, Ångström, ÅNGSTRÖM, aangstrøm, Aangstrøm, AANGSTRØM, aangström, Aangström, AANGSTRÖM, ...	angstrom, Angstrom, ANGSTROM, ångstrøem, Ångstrøem, ÅNGSTRÆM, ...

142 <https://github.com/unicode-org/clldr/blob/master/common/collation/da.xml>

6 6. Examples

The examples in this section illustrate the syntaxes proposed in [Section 5](#)(see page 21).

- [6.1 Simple Expression Constraints](#)(see page 64)
- [6.2 Refinements](#)(see page 72)
- [6.3 Cardinality](#)(see page 82)
- [6.4 Conjunction and Disjunction](#)(see page 90)
- [6.5 Exclusion and Not Equals](#)(see page 97)
- [6.6 Constraint Comments](#)(see page 99)
- [6.7 Nested Expression Constraints](#)(see page 100)
- [6.8 Description Filters](#)(see page 105)
- [6.9 Concept Filters](#)(see page 116)
- [6.10 Member Filters](#)(see page 121)
- [6.11 History Supplements](#)(see page 124)
- [6.12 Top and Bottom](#)(see page 129)

6.1 6.1 Simple Expression Constraints

The simplest type of expression constraint contains a single concept optionally preceded by an expression constraint operator and/or membership function. Expression constraint operators (e.g. descendant of) traverse the hierarchical relationships in SNOMED CT to return the set of concepts that are directly or transitively connected to the focus concept. Membership functions return the set of concepts referenced by a reference set.

In this section we consider some of these simple examples.

6.1.1 Self

If no expression constraint operator or membership function is applied, the expression constraint is satisfied only by the specified concept. For example, the expression constraint below is satisfied only by the concept [404684003 | Clinical finding](#)¹⁴³.

404684003 | Clinical finding¹⁴⁴

Please note that this expression constraint is equivalent to an expression that looks the same but is written in [SNOMED CT Compositional Grammar](#)¹⁴⁵.

6.1.2 Descendant of

A single 'less than' sign (i.e. "<") indicates that the expression constraint is satisfied by all descendants of the specified concept. The expression constraint below evaluates to the set of all subtypes (both direct children and transitive subtypes) of [404684003 | Clinical finding](#)¹⁴⁶, using the brief syntax.

¹⁴³ <http://snomed.info/id/404684003>

¹⁴⁴ <http://snomed.info/id/404684003>

¹⁴⁵ <http://snomed.org/scg>

¹⁴⁶ <http://snomed.info/id/404684003>


```
< 404684003 |Clinical finding|147
```

Using the long syntax, the above expression constraint may be represented as:

```
descendantOf 404684003 |Clinical finding|148
```

The descendantOf function is primarily used on concepts, which serve as the 'grouper' of a set of values (e.g. |Clinical finding (finding)|¹⁴⁹, |Severities (qualifier value)|¹⁵⁰, |Unit (qualifier value)|¹⁵¹). The descendantOf function may also be applied to other concepts, or to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

6.1.3 Descendant or Self of

Two consecutive 'less than' signs (i.e. "<<") indicates that the expression constraint is satisfied by all descendants of the specified concept plus the specified concept itself. The expression constraint below evaluates to the set of descendants of 73211009 |Diabetes mellitus|¹⁵², plus the concept 73211009 |Diabetes mellitus|¹⁵³ itself.

```
<< 73211009 |Diabetes mellitus|154
```

Using the long syntax, the above expression constraint may be represented as:

```
descendantOrSelfOf 73211009 |Diabetes mellitus|155
```

The descendantOrSelfOf function is primarily used for attribute values, which refer to a specific clinical value (e.g. 73211009 |Diabetes mellitus|¹⁵⁶, 73761001 |Colonoscopy|¹⁵⁷, 385055001 |Tablet dose form|¹⁵⁸), but any specialization of this value is also acceptable. The descendantOrSelfOf function may also be applied to other concepts, or to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

147 <http://snomed.info/id/404684003>

148 <http://snomed.info/id/404684003>

149 <http://snomed.info/id/404684003>

150 <http://snomed.info/id/272141005>

151 <http://snomed.info/id/258666001>

152 <http://snomed.info/id/73211009>

153 <http://snomed.info/id/73211009>

154 <http://snomed.info/id/73211009>

155 <http://snomed.info/id/73211009>

156 <http://snomed.info/id/73211009>

157 <http://snomed.info/id/73761001>

158 <http://snomed.info/id/385055001>

6.1.4 Child of

A 'less than' sign directly followed by an exclamation mark (i.e. "<!") indicates that the expression constraint is satisfied by the set of proximal children of the specified concept. The children of a concept are those concepts that are the source of a non-redundant [116680003](http://snomed.info/id/116680003) | is a¹⁵⁹ relationship whose target is the given concept. The expression constraint below, represented using the brief syntax, evaluates to the set of immediate children of the concept [404684003](http://snomed.info/id/404684003) | Clinical finding¹⁶⁰.

```
<! 404684003 |Clinical finding161
```

Using the long syntax, the above expression constraint may be represented as:

```
childOf 404684003 |Clinical finding162
```

Please note that the childOf function may only be executed against a finite and pre-classified substrate, and that the results of this function are specific to the substrate used. The childOf function may also be applied to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

6.1.5 Child or Self of

Two consecutive 'less than' signs directly followed by an exclamation mark (i.e. "<<!") indicates that the expression constraint is satisfied by the set of proximal children of the specified concept plus the specified concept itself. The children of a concept are those concepts that are the source of a non-redundant [116680003](http://snomed.info/id/116680003) | is a¹⁶³ relationship whose target is the given concept. The expression constraint below, represented using the brief syntax, evaluates to the set of immediate children of the concept [404684003](http://snomed.info/id/404684003) | Clinical finding¹⁶⁴, plus the concept [404684003](http://snomed.info/id/404684003) | Clinical finding¹⁶⁵ itself.

```
<<! 404684003 |Clinical finding166
```

Using the long syntax, the above expression constraint may be represented as:

```
childOrSelfOf 404684003 |Clinical finding167
```

Please note that the childOrSelfOf function may only be executed against a finite and pre-classified substrate, and that the results of this function are specific to the substrate used. The childOrSelfOf function may also be applied to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

159 <http://snomed.info/id/116680003>

160 <http://snomed.info/id/404684003>

161 <http://snomed.info/id/404684003>

162 <http://snomed.info/id/404684003>

163 <http://snomed.info/id/116680003>

164 <http://snomed.info/id/404684003>

165 <http://snomed.info/id/404684003>

166 <http://snomed.info/id/404684003>

167 <http://snomed.info/id/404684003>

6.1.6 Ancestor of

A single 'greater than' sign (i.e. ">") indicates that the expression constraint is satisfied by all ancestors of the specified concept. The expression constraint below, using the brief syntax, evaluates to the set of all supertypes (both direct parents and transitive supertypes) of [40541001 | Acute pulmonary edema](http://snomed.info/id/40541001)¹⁶⁸ :

```
> 40541001 |Acute pulmonary edema|169
```

Using the long syntax, the above expression constraint may be represented as:

```
ancestorOf 40541001 |Acute pulmonary edema|170
```

Please note that the ancestorOf function may also be applied to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

6.1.7 Ancestor or Self of

Two consecutive 'greater than' signs (i.e. ">>") indicates that the expression constraint is satisfied by all ancestors of the specified concept plus the specified concept itself. The expression constraint below evaluates to the set of ancestors of [40541001 | Acute pulmonary edema](http://snomed.info/id/40541001)¹⁷¹, plus the concept [40541001 | Acute pulmonary edema](http://snomed.info/id/40541001)¹⁷² .

```
>> 40541001 |Acute pulmonary edema|173
```

Using the long syntax, the above expression constraint may be represented as:

```
ancestorOrSelfOf 40541001 |Acute pulmonary edema|174
```

Please note that the ancestorOrSelfOf function may also be applied to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

6.1.8 Parent of

A 'greater than' sign directly followed by an exclamation mark (i.e. ">!") indicates that the expression constraint is satisfied by the set of proximal parents of the specified concept. The parents of a concept are those concepts that

¹⁶⁸ <http://snomed.info/id/40541001>

¹⁶⁹ <http://snomed.info/id/40541001>

¹⁷⁰ <http://snomed.info/id/40541001>

¹⁷¹ <http://snomed.info/id/40541001>

¹⁷² <http://snomed.info/id/40541001>

¹⁷³ <http://snomed.info/id/40541001>

¹⁷⁴ <http://snomed.info/id/40541001>

are the target of a non-redundant `| is a`¹⁷⁵ relationship whose source is the given concept. The expression constraint below, represented using the brief syntax, evaluates to the set of immediate parents of the concept `40541001 | Acute pulmonary edema`¹⁷⁶.

```
>! 40541001 |Acute pulmonary edema177
```

Using the long syntax, the above expression constraint may be represented as:

```
parentOf 40541001 |Acute pulmonary edema178
```

Please note that the `parentOf` function should only be executed against a finite and pre-classified substrate, and that the results of this function are specific to the substrate used. The `parentOf` function may also be applied to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

6.1.9 Parent or Self of

Two consecutive 'greater than' signs directly followed by an exclamation mark (i.e. ">>!") indicates that the expression constraint is satisfied by the set of proximal parents of the specified concept plus the specified concept itself. The parents of a concept are those concepts that are the target of a non-redundant `| is a`¹⁷⁹ relationship whose source is the given concept. The expression constraint below, represented using the brief syntax, evaluates to the set of immediate parents of the concept `40541001 | Acute pulmonary edema`¹⁸⁰, plus the concept `40541001 | Acute pulmonary edema`¹⁸¹ itself.

```
>>! 40541001 |Acute pulmonary edema182
```

Using the long syntax, the above expression constraint may be represented as:

```
parentOrSelfOf 40541001 |Acute pulmonary edema183
```

Please note that the `parentOrSelfOf` function should only be executed against a finite and pre-classified substrate, and that the results of this function are specific to the substrate used. The `parentOrSelfOf` function may also be applied to nested expression constraints (as discussed in [6.7 Nested Expression Constraints](#)(see page 100)).

175 <http://snomed.info/id/116680003>

176 <http://snomed.info/id/40541001>

177 <http://snomed.info/id/40541001>

178 <http://snomed.info/id/40541001>

179 <http://snomed.info/id/116680003>

180 <http://snomed.info/id/40541001>

181 <http://snomed.info/id/40541001>

182 <http://snomed.info/id/40541001>

183 <http://snomed.info/id/40541001>

6.1.10 Member of

The `memberOf` function (by default) evaluates to the set of concepts that are referenced by the given reference set (i.e. the set of `referencedComponentIds`). Please note that this function may be applied only to reference sets whose referenced components are concepts. The SNOMED CT Expression Constraint Language does not support use of the `memberOf` function on reference sets whose referenced components are not concepts (i.e. descriptions or relationships).

The `memberOf` function is represented in the brief syntax using a 'caret' character (i.e. "^") and is usually followed by a single concept id for a concept-based reference set. For example, the following expression constraint is satisfied by the set of concepts which are members of `700043003 | Example problem list concepts reference set`¹⁸⁴:

```
^ 700043003 | Example problem list concepts reference set185
```

Using the long syntax the expression constraint is represented as:

```
memberOf 700043003 | Example problem list concepts reference set186
```

The expression constraints above both return the values in the `referencedComponentId` field of the given reference sets. However, it is also possible to specify one or more fields, whose values will be returned, by including the relevant field names in square brackets after the `memberOf` operator ("^" or "`memberOf`"). For example, the following expression constraint is equivalent to the brief syntax example above.

```
^ [referencedComponentId] 700043003 | Example problem list concepts reference set187
```

The value of other fields can also be returned by an expression constraint¹ (see page 0). For example, the following expression constraint will return the `targetComponentId` values (i.e. the 'Entire' anatomy concepts) from the `734138000 | Anatomy structure and entire association reference set`¹⁸⁸.

```
^ [targetComponentId] 734138000 | Anatomy structure and entire association reference set189
```

It is also possible to return the values of more than one field in a reference set (e.g. a pair or tuple of values).² (see page 0) ³ (see page 0) ⁴ (see page 0) For example, to return both the source and target of the `816210007 | SNOMED CT to MedDRA simple map reference set`¹⁹⁰, the following expression constraint could be used:

```
^ [referencedComponentId, mapTarget] 816210007 | SNOMED CT to MedDRA simple map reference set191
```

184 <http://snomed.info/id/700043003>

185 <http://snomed.info/id/700043003>

186 <http://snomed.info/id/700043003>

187 <http://snomed.info/id/700043003>

188 <http://snomed.info/id/734138000>

189 <http://snomed.info/id/734138000>

190 <http://snomed.info/id/816210007>

191 <http://snomed.info/id/816210007>

To return all the non-metadata fields of a referenceSet (i.e. the values of the referencedComponentId and additional fields), a wildcard (i.e. "*" in the brief syntax, and "*" or "Any" in the long syntax) can be used. For example, the following expression constraint will return the referencedComponentId, mapGroup, mapPriority, mapRule, mapAdvice, mapTarget and correlationId for each row of the 447562003 | ICD-10 complex map reference set¹⁹².

```
^ [*] 447562003 | ICD-10 complex map reference set193
```

For more information on the use of reference set field names in ECL, please refer to [Appendix E - Reference Set Fields](#)(see page 206).

Please note that it is also possible to apply the memberOf function to an expression constraint that returns a set of concept-based reference set concepts. For more information, please refer to [6.7 Nested Expression Constraints](#)(see page 100).

And for information about applying filter constraints to reference set members, please refer to [6.10 Member Filters](#)(see page 121).

6.1.11 Any

A single 'star' (i.e. "*") may be used in the place of a concept reference to represent any concept in the substrate. The expression constraint below evaluates to the set of all concepts in the given substrate.

```
*
```

Using the long syntax, the above expression constraint may also be represented as:

```
ANY
```

This wildcard character (or 'ANY' keyword) may be used anywhere within an expression constraint that a concept reference may be used. In many situations, the wildcard is equivalent to the following expression constraint:

```
<< 138875005 | SNOMED CT concept194
```

However, some situations exist in which the concept 138875005 | SNOMED CT concept¹⁹⁵ is not included in the substrate, and therefore cannot be used to determine the full set of concepts available. In other cases, the single character wildcard may serve as a convenient shortcut for the longer expression constraint above.

Please note that the following three expression constraints evaluate to the same set of concepts:

¹⁹² <http://snomed.info/id/447562003>

¹⁹³ <http://snomed.info/id/447562003>

¹⁹⁴ <http://snomed.info/id/138875005>

¹⁹⁵ <http://snomed.info/id/138875005>

*

<< *

>> *

The two expression constraints below evaluate to all concepts in the substrate minus the root concept:

< *

<! *

And the two expression constraints below evaluate to all non-leaf concepts in the substrate:

> *

>! *

Finally, the expression constraint below evaluates to all concepts that are referenced by any reference set in the substrate:

^ *

6.1.12 Alternate Identifier

If an alternate identifier exists for a concept, in an identifier scheme other than SNOMED CT, then this can be used to refer to that concept [\[see page 0\]](#) (see [4.2.4 Identifier File Specification](#)¹⁹⁶). For example there may be SNOMED CT concept for "Type of hemoglobin in blood at point in time" with an alternate identifier with the scheme alias "LOINC" and the code "54486-6":

LOINC#54486-6

¹⁹⁶ <https://confluence.ihtsdotools.org/display/DOCRELFMT/4.2.4+Identifier+File+Specification>

The alternate identifier code can be surrounded by quotes.

The alternate identifier code can be surrounded by quotes.

The alternate identifier code can be surrounded by double quotes. Quoting may be necessary to ensure correct parsing of the constraint depending on the characters used in the code:

```
LOINC#"54486-6"
```

Please note: ECL is always evaluated against SNOMED CT concepts. When you run a query using alternate identifiers from another code system only concepts that are in the SNOMED CT representation of that other code system will be returned. For example the descendants of a concept in the SNOMED CT version of a code system is likely to be different from the code system of the alternate identifier.

¹(see page 69) **Note:** If a reference set field is selected that does not exist in any of the identified reference sets, then the expression constraint should be considered erroneous.

²(see page 69) **Note:** Returning the values of more than one reference set field may only be done as the final operation of an expression constraint.

³(see page 69) **Note:** Attempts to select more than one reference set field on an inner subquery should result in an execution error.

⁴(see page 69) **Note:** In some implementation contexts, the `memberOf` function may be restricted to return only a single field.

⁵(see page 71) **Note:** No alternate identifiers exist in the International Edition at this time.

6.2 6.2 Refinements

In this section, we illustrate how the set of matching concepts can be filtered using one or more simple attribute refinements. For more information on applying refinements to nested expression constraints, using nested attribute names and using nested attribute values, please refer to [6.7 Nested Expression Constraints](#)(see page 100).

6.2.1 Attributes

Adding an attribute refinement to an expression constraint restricts the set of valid clinical meanings to only those whose defining attributes satisfy the given refinement condition. Similarly to [SNOMED CT Compositional Grammar](#)¹⁹⁷, attribute refinements are placed after a 'colon' (i.e. ":") in the expression constraint.

The example below is satisfied only by the set of lung disorders, which have an associated morphology that is exactly equal to `79654002 | Edema`¹⁹⁸.

¹⁹⁷ <http://snomed.org/scg>

¹⁹⁸ <http://snomed.info/id/79654002>


```
< 19829001 |Disorder of lung|199 :
  116676008 |Associated morphology|200 = 79654002 |Edema|201
```

Using the long syntax, the above expression is represented as:

```
descendantOf 19829001 |Disorder of lung|202 :
  116676008 |Associated morphology|203 = 79654002 |Edema|204
```

In many cases, however, the value of the matching attribute is allowed to be either the concept itself, *or* a descendant of that concept. In these cases, the descendantOrSelfOf operator is used prior to the concept representing the attribute value. For example, the expression constraint below (in brief and long syntaxes respectively) is satisfied only by the set of lung disorders, which have an associated morphology of 79654002 |Edema|²⁰⁵ or any descendant of 79654002 |Edema|²⁰⁶.

```
< 19829001 |Disorder of lung|207 :
  116676008 |Associated morphology|208 = << 79654002 |Edema|209
```

```
descendantOf 19829001 |Disorder of lung|210 :
  116676008 |Associated morphology|211 = descendantOrSelfOf 79654002 |Edema|212
```

When more than one attribute is defined in an expression constraint, the attributes are normally separated by a comma. A comma between two attributes indicates a conjunction and implies that both attribute conditions must be true. For example, the expression constraint below, written in brief syntax, is satisfied only by the set of clinical findings, which have both a finding site of 39057004 |Pulmonary valve structure|²¹³ (or a subtype of 39057004 |Pulmonary valve structure|²¹⁴) and an associated morphology of 'stenosis' (or a subtype of 'stenosis').

199 <http://snomed.info/id/19829001>

200 <http://snomed.info/id/116676008>

201 <http://snomed.info/id/79654002>

202 <http://snomed.info/id/19829001>

203 <http://snomed.info/id/116676008>

204 <http://snomed.info/id/79654002>

205 <http://snomed.info/id/79654002>

206 <http://snomed.info/id/79654002>

207 <http://snomed.info/id/19829001>

208 <http://snomed.info/id/116676008>

209 <http://snomed.info/id/79654002>

210 <http://snomed.info/id/19829001>

211 <http://snomed.info/id/116676008>

212 <http://snomed.info/id/79654002>

213 <http://snomed.info/id/39057004>

214 <http://snomed.info/id/39057004>

```
< 404684003 |Clinical finding215 :
  363698007 |Finding site216 = << 39057004 |Pulmonary valve structure217 ,
  116676008 |Associated morphology218 = << 415582006 |Stenosis219
```

Please note that attribute refinements may also be used when the focus concept is '*' (or ANY). The following expression constraint represents any concept that has a 246075003 |Causative agent²²⁰ attribute whose value is 387517004 |Paracetamol²²¹.

```
* : 246075003 |Causative agent222 = 387517004 |Paracetamol223
```

Using the long syntax, the above expression may also be represented as:

```
ANY : 246075003 |Causative agent224 = 387517004 |Paracetamol225
```

6.2.2 Attribute Groups

Similarly to SNOMED CT compositional grammar, expression constraints use curly braces (i.e. "{..}") to indicate that a set of attributes should be grouped together in an attribute group. For example, the expression constraint below is satisfied only by the set of clinical findings with an associated morphology of 'stenosis' (or descendant) at the finding site 'pulmonary valve structure' (or descendant), and also with an associated morphology of 'hypertrophy' (or descendant) at the finding site 'right ventricular structure' (or descendant).

```
< 404684003 |Clinical finding226 :
  { 363698007 |Finding site227 = << 39057004 |Pulmonary valve structure228 ,
    116676008 |Associated morphology229 = << 415582006 |Stenosis230 },
```

215 <http://snomed.info/id/404684003>

216 <http://snomed.info/id/363698007>

217 <http://snomed.info/id/39057004>

218 <http://snomed.info/id/116676008>

219 <http://snomed.info/id/415582006>

220 <http://snomed.info/id/246075003>

221 <http://snomed.info/id/387517004>

222 <http://snomed.info/id/246075003>

223 <http://snomed.info/id/387517004>

224 <http://snomed.info/id/246075003>

225 <http://snomed.info/id/387517004>

226 <http://snomed.info/id/404684003>

227 <http://snomed.info/id/363698007>

228 <http://snomed.info/id/39057004>

229 <http://snomed.info/id/116676008>

230 <http://snomed.info/id/415582006>

```
{ 363698007 |Finding site231 = << 53085002 |Right ventricular structure232,
  116676008 |Associated morphology233 = << 56246009 |Hypertrophy234 }
```

Using the 'long syntax', the above expression constraint is represented as:

```
descendantOf 404684003 |Clinical finding235 :
  { 363698007 |Finding site236 = descendantOrSelfOf 39057004 |Pulmonary valve structure237,
    116676008 |Associated morphology238 = descendantOrSelfOf 415582006 |Stenosis239 },
  { 363698007 |Finding site240 = descendantOrSelfOf 53085002 |Right ventricular structure241,
    116676008 |Associated morphology242 = descendantOrSelfOf 56246009 |Hypertrophy243 }
```

6.2.3 Attribute Constraint Operators

In some cases, an attribute concept has subtypes or supertypes in the | Concept model attribute²⁴⁴ hierarchy. Where this occurs, it is possible to indicate that an attribute condition may be satisfied by matching one of the subtypes or supertypes of the given attribute. This is done adding a constraint operator directly before the attribute name concept. For example, the expression constraint below will not only match clinical findings that are | Associated with²⁴⁵ a type of | Edema²⁴⁶, but also those that are | Due to²⁴⁷, | After²⁴⁸ or the | Causative agent²⁴⁹ of a type of | Edema²⁵⁰. This result occurs because the 47429007 | Associated with²⁵¹ attribute concept has three subtypes: 255234002 | After²⁵², 246075003 | Causative agent²⁵³ and 42752001 | Due to²⁵⁴.

```
<< 404684003 |Clinical finding255 :
  << 47429007 |Associated with256 = << 267038008 |Edema257
```

231 <http://snomed.info/id/363698007>

232 <http://snomed.info/id/53085002>

233 <http://snomed.info/id/116676008>

234 <http://snomed.info/id/56246009>

235 <http://snomed.info/id/404684003>

236 <http://snomed.info/id/363698007>

237 <http://snomed.info/id/39057004>

238 <http://snomed.info/id/116676008>

239 <http://snomed.info/id/415582006>

240 <http://snomed.info/id/363698007>

241 <http://snomed.info/id/53085002>

242 <http://snomed.info/id/116676008>

243 <http://snomed.info/id/56246009>

244 <http://snomed.info/id/410662002>

245 <http://snomed.info/id/47429007>

246 <http://snomed.info/id/267038008>

247 <http://snomed.info/id/42752001>

248 <http://snomed.info/id/255234002>

249 <http://snomed.info/id/246075003>

250 <http://snomed.info/id/267038008>

251 <http://snomed.info/id/47429007>

252 <http://snomed.info/id/255234002>

253 <http://snomed.info/id/246075003>

254 <http://snomed.info/id/42752001>

255 <http://snomed.info/id/404684003>

256 <http://snomed.info/id/47429007>

257 <http://snomed.info/id/267038008>

This expression constraint is represented in the long syntax as:

```
descendantOrSelfOf 404684003 |Clinical finding258 :
  descendantOrSelfOf 47429007 |Associated with259 = descendantOrSelfOf 267038008 |Edema260
```

Similarly, the expression constraint below will not only match clinical findings that are | Due to²⁶¹ a type of | Edema²⁶², but also those that have an | Associated with²⁶³ relationship whose value is a type of | Edema²⁶⁴.

```
<< 404684003 |Clinical finding265 :
  >> 246075003 |Causative agent266 = << 267038008 |Edema267
```

This expression constraint is represented in the long syntax as:

```
descendantOrSelfOf 404684003 |Clinical finding268 :
  ancestorOrSelfOf 246075003 |Causative agent269 = descendantOrSelfOf 267038008 |Edema270
```

6.2.4 Concrete Values

The revised [SNOMED CT Compositional Grammar](#)²⁷¹ allows attributes to be given concrete values (e.g. Strings, Integers, Decimal, Boolean). The [SNOMED CT Expression Constraint Language](#)²⁷² supports the ability to compare these attribute values with a given concrete value.

When numeric concrete values (i.e. Integers and Decimals) are compared, a set of standard mathematical operators may be used. These mathematical operators are:

Operator	Name
=	Equals
!=	Not equals

²⁵⁸ <http://snomed.info/id/404684003>

²⁵⁹ <http://snomed.info/id/47429007>

²⁶⁰ <http://snomed.info/id/267038008>

²⁶¹ <http://snomed.info/id/42752001>

²⁶² <http://snomed.info/id/267038008>

²⁶³ <http://snomed.info/id/47429007>

²⁶⁴ <http://snomed.info/id/267038008>

²⁶⁵ <http://snomed.info/id/404684003>

²⁶⁶ <http://snomed.info/id/246075003>

²⁶⁷ <http://snomed.info/id/267038008>

²⁶⁸ <http://snomed.info/id/404684003>

²⁶⁹ <http://snomed.info/id/246075003>

²⁷⁰ <http://snomed.info/id/267038008>

²⁷¹ <http://snomed.org/scg>

²⁷² <http://snomed.org/ecl>

<	Less than
<=	Less than or equals
>	Greater than
>=	Greater than or equals

Please note that the 'not equals' operator may alternatively be represented as "<>" and "not =" (case insensitive) in the long syntax.

The following expression constraint is satisfied by oral medicinal products, which contain amoxicillin and have a presentation strength greater than or equal to 250 mg.

```
< 763158003 |Medicinal product (product)|273 :
  411116001 |Has manufactured dose form (attribute)|274 = << 385268001 |Oral dose form (dose form)|275 ,
  { << 127489000 |Has active ingredient (attribute)|276 = << 372687004 |Amoxicillin (substance)|277 ,
    1142135004 |Has presentation strength numerator value (attribute)|278 >= #250,
    732945000 |Has presentation strength numerator unit (attribute)|279 = 258684004 |milligram (qualifier
value)|280 }
```

Please note that, as per SNOMED CT Compositional Grammar, integer and decimal values are preceded by a hash character (e.g. "#500"), while string values are surrounded by double quotes (e.g. "PANADOL").

To find those oral amoxicillin products that have a strength between 250 and 800 mg (inclusive), the following expression constraint may be used:

```
< 763158003 |Medicinal product (product)|281 :
  411116001 |Has manufactured dose form (attribute)|282 = << 385268001 |Oral dose form (dose form)|283 ,
  { << 127489000 |Has active ingredient (attribute)|284 = << 372687004 |Amoxicillin (substance)|285 ,
    1142135004 |Has presentation strength numerator value (attribute)|286 >= #250,
    1142135004 |Has presentation strength numerator value (attribute)|287 <= #800,
```

273 <http://snomed.info/id/763158003>

274 <http://snomed.info/id/411116001>

275 <http://snomed.info/id/385268001>

276 <http://snomed.info/id/127489000>

277 <http://snomed.info/id/372687004>

278 <http://snomed.info/id/1142135004>

279 <http://snomed.info/id/732945000>

280 <http://snomed.info/id/258684004>

281 <http://snomed.info/id/763158003>

282 <http://snomed.info/id/411116001>

283 <http://snomed.info/id/385268001>

284 <http://snomed.info/id/127489000>

285 <http://snomed.info/id/372687004>

286 <http://snomed.info/id/1142135004>

287 <http://snomed.info/id/1142135004>

```
732945000 |Has presentation strength numerator unit (attribute)288 = 258684004 |milligram (qualifier value)289 }
```

Concrete values of type string and boolean may also be included in an expression constraint, and compared using an 'equal to' (i.e. "=") or 'not equal to' (i.e. "!=") operator. The following expression constraint is satisfied only by products with a product name equal to "PANADOL"¹ [\(see page 0\)](#).

```
< 373873005 |Pharmaceutical / biologic product290 :
  3460481009 |Has product name291 = "PANADOL"
```

The following expression constraint is satisfied only by products that are in the national benefit scheme (of the given country)² [\(see page 0\)](#).

```
< 373873005 |Pharmaceutical / biologic product292 :
  859999999102 |Is in national benefit scheme293 = TRUE
```

6.2.5 Reverse Attributes

In most cases, an attribute refinement is satisfied by those concepts, which are the source concept of a defining relationship whose destination concept matches the attribute value. In some cases, however, it may be necessary to select the destination concept of a relationship and constrain the source concept to a given attribute value. To achieve this, an expression constraint indicates that an attribute is to be constrained in the reverse order using a 'reverse flag'³ [\(see page 0\)](#). In the brief syntax, the reverse flag is represented by preceding the name of the attribute with a capital letter 'R'.

For example, the expression constraint below finds the set of anatomical structures, which are the finding site of a type of bone fracture (e.g. 85050009 |Humerus²⁹⁴, 71341001 |Femur²⁹⁵).

```
< 91723000 |Anatomical structure296 :
  R 363698007 |Finding site297 =< 125605004 |Fracture of bone298
```

The above expression constraint is represented in the long syntax as:

²⁸⁸ <http://snomed.info/id/732945000>

²⁸⁹ <http://snomed.info/id/258684004>

²⁹⁰ <http://snomed.info/id/373873005>

²⁹¹ <http://snomed.info/id/3460481009>

²⁹² <http://snomed.info/id/373873005>

²⁹³ <http://snomed.org/fictid#859999999102>

²⁹⁴ <http://snomed.info/id/85050009>

²⁹⁵ <http://snomed.info/id/71341001>

²⁹⁶ <http://snomed.info/id/91723000>

²⁹⁷ <http://snomed.info/id/363698007>

²⁹⁸ <http://snomed.info/id/125605004>

```

descendantOf 91723000 |Anatomical structure299 :
reverseOf 363698007 |Finding site300 = descendantOf 125605004 |Fracture of bone301

```

6.2.6 Dotted Attributes

An alternative way of representing 'reversed attributes' is by applying the *dot notation* to represent them as *dotted attributes*. Using this alternative notation, "`< 123456 123456 |X302 .234567 234567 |Y303`" represents the set of attribute values (i.e. destination concepts) of the attribute "Y" for descendants or self of concept "X". This is therefore equivalent to "`* : R 234567 234567 |Y304 = < 123456 123456 |X305`" using the reverse flag.

The previous expression constraint (which finds the set of body sites for any subtype of bone fracture) has an equivalent representation using the 'dot notation' of:

```

< 91723000 |Anatomical structure306 AND (< 125605004 |Fracture of bone307 . 363698007 |Finding site308 )

```

Because all values of `363698007 |Finding site309` must be `< 91723000 |Anatomical structure310` (according to the [SNOMED CT concept model³¹¹](https://confluence.ihtsdotools.org/display/DOCGLOSS/SNOMED+CT+concept+model)), this expression constraint can be further simplified to:

```

< 125605004 |Fracture of bone312 . 363698007 |Finding site313

```

The next example finds the set of substances, which are an active ingredient in any product containing amoxicillin.

```

< 105590001 |Substance314 :
R << 127489000 |Has active ingredient315 = < 27658006 |Product containing amoxicillin316

```

This expression constraint is represented in the long syntax as:

299 <http://snomed.info/id/91723000>
300 <http://snomed.info/id/363698007>
301 <http://snomed.info/id/125605004>
302 <http://snomed.info/id/123456>
303 <http://snomed.info/id/234567>
304 <http://snomed.info/id/234567>
305 <http://snomed.info/id/123456>
306 <http://snomed.info/id/91723000>
307 <http://snomed.info/id/125605004>
308 <http://snomed.info/id/363698007>
309 <http://snomed.info/id/363698007>
310 <http://snomed.info/id/91723000>
311 <https://confluence.ihtsdotools.org/display/DOCGLOSS/SNOMED+CT+concept+model>
312 <http://snomed.info/id/125605004>
313 <http://snomed.info/id/363698007>
314 <http://snomed.info/id/105590001>
315 <http://snomed.info/id/127489000>
316 <http://snomed.info/id/27658006>

```

descendantOf 105590001 |Substance317 :
ReverseOf descendantOrSelfOf 127489000 |Has active ingredient318 = descendantOf 27658006 |Product
containing amoxicillin319

```

An equivalent way of representing this constraint, using the 'dot notation' is:

```

< 105590001 |Substance320 AND ( < 27658006 |Product containing amoxicillin321 . << 127489000 |Has
active ingredient322 )

```

or (using the [SNOMED CT concept model](#)³²³ to simplify):

```

< 27658006 |Product containing amoxicillin324 . << 127489000 |Has active ingredient325

```

When more than one dot attribute is used in sequence, the dot notation is evaluated sequentially from left to right. For example, the following expression constraint represents the set of | Finding sites³²⁶ of any concept that is | Associated with³²⁷ a subtype of | Disorder of lung³²⁸.

```

< 19829001 |Disorder of lung329 . < 47429007 |Associated with330 . 363698007 |Finding site331

```

This expression constraint is evaluated by first finding the descendants of | Disorder of lung³³², then finding the set of attribute values for these concepts (with an attribute type that is any subtype of | Associated with³³³), and then from these attribute value concepts, finding the value of any | Finding sites³³⁴ attribute. Please note that the expression constraint above (with no brackets) is equivalent to the one below (with brackets added).

```

(( < 19829001 |Disorder of lung335 ) . < 47429007 |Associated with336 ) . 363698007 |Finding site337

```

317 <http://snomed.info/id/105590001>

318 <http://snomed.info/id/127489000>

319 <http://snomed.info/id/27658006>

320 <http://snomed.info/id/105590001>

321 <http://snomed.info/id/27658006>

322 <http://snomed.info/id/127489000>

323 <https://confluence.ihtsdotools.org/display/DOCGLOSS/SNOMED+CT+concept+model>

324 <http://snomed.info/id/27658006>

325 <http://snomed.info/id/127489000>

326 <http://snomed.info/id/363698007>

327 <http://snomed.info/id/47429007>

328 <http://snomed.info/id/19829001>

329 <http://snomed.info/id/19829001>

330 <http://snomed.info/id/47429007>

331 <http://snomed.info/id/363698007>

332 <http://snomed.info/id/19829001>

333 <http://snomed.info/id/47429007>

334 <http://snomed.info/id/363698007>

335 <http://snomed.info/id/19829001>

336 <http://snomed.info/id/47429007>

337 <http://snomed.info/id/363698007>

6.2.7 Any Attribute Name and Value

A single 'star' (i.e. "*") may be used in the place of an attribute name to represent any attribute in the substrate. The expression constraint below evaluates to the set of clinical findings which have any attribute with a value of 79654002 | Edema³³⁸.

```
< 404684003 |Clinical finding339 : * = 79654002 |Edema340
```

Using the long syntax, the above expression constraint may also be represented as:

```
descendantOf 404684003 |Clinical finding341 : ANY = 79654002 |Edema342
```

The 'star' symbol (i.e. "*") may also be used to represent any attribute value (either with or without refinement). The following expression constraint evaluates to the set of clinical findings which have an associated morphology (with any value).

```
< 404684003 |Clinical finding343 : 116676008 |Associated morphology344 = *
```

Using the long syntax, the above expression constraint may also be represented as:

```
descendantOf 404684003 |Clinical finding345 : 116676008 |Associated morphology346 = ANY
```

¹[see page 78](#) Concrete values of type string are case sensitive and compared using the Unicode Collation Algorithm (<http://www.unicode.org/reports/tr10/>).

²[see page 78](#) Please note that the concept 859999999102 |Is in national benefit scheme| is a fictitious attribute used here to illustrate boolean values.

³[see page 78](#) It should be noted that using a reversed attribute joined by conjunction with a non-reversed attribute may lead to a nonsensical constraint (e.g. "<<a: {b=c, Rd=e}"). This is because the target concept of the reversed attribute must be matched with the source concept of the non-reversed attribute, which in turn must be the same as the source concept of the reversed attribute (being in the same attribute group). This would require the reversed attribute to be reflexive (i.e. the source and target concept to be the same).

338 <http://snomed.info/id/79654002>

339 <http://snomed.info/id/404684003>

340 <http://snomed.info/id/79654002>

341 <http://snomed.info/id/404684003>

342 <http://snomed.info/id/79654002>

343 <http://snomed.info/id/404684003>

344 <http://snomed.info/id/116676008>

345 <http://snomed.info/id/404684003>

346 <http://snomed.info/id/116676008>

6.3 6.3 Cardinality

6.3.1 Attribute cardinality

6.3.1.1 Overview

To support use cases such as the SNOMED CT concept model and terminology binding, expression constraints may constrain the number of times an attribute can be included in an expression or concept definition represented in the SNOMED CT distribution view [1](#). This is done using a cardinality constraint, which consists of a minimum cardinality and a maximum cardinality (written "[X..Y]"). A minimum cardinality of X constrains the valid clinical meanings to those which have at least (i.e. >=) X non-redundant [2](#) attributes that match the given attribute criteria. A maximum cardinality of Y constrains the valid clinical meanings to those which have at most (i.e. <=) Y non-redundant [2](#) attributes that match the given attribute criteria. For example, a cardinality of "[1..5]" indicates that all clinical meanings that satisfy the given expression constraint must have at least one and at most five attributes that match the given attribute criteria.

The expression constraint below is satisfied only by products with one, two or three active ingredients.

```
< 373873005 |Pharmaceutical / biologic product347 :
  [1..3] 127489000 |Has active ingredient348 =< 105590001 |Substance349
```

Using the long syntax, this expression constraint may be represented as:

```
descendantOf 373873005 |Pharmaceutical / biologic product350 :
  [1 to 3] 127489000 |Has active ingredient351 = descendantOf 105590001 |Substance352
```

The following expression constraint is satisfied only by products which have exactly one active ingredient:

```
< 373873005 |Pharmaceutical / biologic product353 :
  [1..1] 127489000 |Has active ingredient354 =< 105590001 |Substance355
```

347 <http://snomed.info/id/373873005>

348 <http://snomed.info/id/127489000>

349 <http://snomed.info/id/105590001>

350 <http://snomed.info/id/373873005>

351 <http://snomed.info/id/127489000>

352 <http://snomed.info/id/105590001>

353 <http://snomed.info/id/373873005>

354 <http://snomed.info/id/127489000>

355 <http://snomed.info/id/105590001>

6.3.1.2 Unconstrained Cardinalities

A minimum cardinality of '0' indicates that there is *no* constraint on the minimum number of attributes that may match the given attribute criteria. For example, the following expression constraint is satisfied only by products with at most one active ingredient (i.e. the maximum cardinality is '1' and the minimum cardinality is unconstrained).

```
< 373873005 |Pharmaceutical / biologic product356 :
  [0..1] 127489000 |Has active ingredient357 =< 105590001 |Substance358
```

Using the long syntax, this may be represented as:

```
descendantOf 373873005 |Pharmaceutical / biologic product359 :
  [0 to 1] 127489000 |Has active ingredient360 = descendantOf 105590001 |Substance361
```

A maximum cardinality of '*' (or 'many') indicates that there is *no* constraint on the maximum number of attributes that may match the given attribute criteria. For example, the following expression constraint is satisfied only by products that have at least one active ingredient (i.e. the minimum cardinality is '1' and the maximum cardinality is unconstrained).

```
< 373873005 |Pharmaceutical / biologic product362 :
  [1..*] 127489000 |Has active ingredient363 =< 105590001 |Substance364
```

Using the long syntax, this may be represented as:

```
descendantOf 373873005 |Pharmaceutical / biologic product365 :
  [1 to many] 127489000 |Has active ingredient366 = descendantOf 105590001 |Substance367
```

A cardinality of [0..*] should therefore never be used as this indicates that the given attribute is not being constrained in any way, and is therefore a redundant part of the expression constraint.

356 <http://snomed.info/id/373873005>

357 <http://snomed.info/id/127489000>

358 <http://snomed.info/id/105590001>

359 <http://snomed.info/id/373873005>

360 <http://snomed.info/id/127489000>

361 <http://snomed.info/id/105590001>

362 <http://snomed.info/id/373873005>

363 <http://snomed.info/id/127489000>

364 <http://snomed.info/id/105590001>

365 <http://snomed.info/id/373873005>

366 <http://snomed.info/id/127489000>

367 <http://snomed.info/id/105590001>

6.3.1.3 Default Cardinalities

The default cardinality of each attribute, where not explicitly stated, is [1..*]. Therefore, the following two expression constraints are equivalent.

```
< 373873005 |Pharmaceutical / biologic product368 :
  [1..*] 127489000 |Has active ingredient369 =< 105590001 |Substance370
```

```
< 373873005 |Pharmaceutical / biologic product371 :
  127489000 |Has active ingredient372 =< 105590001 |Substance373
```

6.3.1.4 Non-redundant Attributes

As mentioned above, only non-redundant defining attributes are included in the cardinality count. Therefore, the following postcoordinated expression:

```
404684003 |Clinical finding374 :
  { 116676008 |Associated morphology375 = 72704001 |Fracture376 ,
    363698007 |Finding site377 = 299701004 |Bone of forearm378 ,
    363698007 |Finding site379 = 62413002 |Bone structure of radius380 }
```

will successfully satisfy the expression constraint:

```
< 404684003 |Clinical finding381 :
  [1..1] 363698007 |Finding site382 =< 91723000 |Anatomical structure383
```

368 <http://snomed.info/id/373873005>

369 <http://snomed.info/id/127489000>

370 <http://snomed.info/id/105590001>

371 <http://snomed.info/id/373873005>

372 <http://snomed.info/id/127489000>

373 <http://snomed.info/id/105590001>

374 <http://snomed.info/id/404684003>

375 <http://snomed.info/id/116676008>

376 <http://snomed.info/id/72704001>

377 <http://snomed.info/id/363698007>

378 <http://snomed.info/id/299701004>

379 <http://snomed.info/id/363698007>

380 <http://snomed.info/id/62413002>

381 <http://snomed.info/id/404684003>

382 <http://snomed.info/id/363698007>

383 <http://snomed.info/id/91723000>

This is because 299701004 | Bone of forearm³⁸⁴ is a supertype of 62413002 | Bone structure of radius³⁸⁵ and therefore the attribute " 363698007 | Finding site³⁸⁶ = 299701004 | Bone of forearm³⁸⁷ " is redundant.

6.3.1.5 Attribute Cardinality in Groups

When the attributes to which cardinality are applied can be grouped, but braces are not used in the expression constraint, the cardinality constrains the number of times the attribute may be included in *any* attribute group. For example, the following expression constraint is satisfied by any clinical finding whose definition has two or more non-redundant finding sites, irrespective of which attribute group they are contained in.

```
< 404684003 | Clinical finding388 :
  [2..*] 363698007 | Finding site389 =< 91723000 | Anatomical structure390
```

In contrast, when braces are placed around an attribute with a given cardinality, there must exist at least one attribute group for which the given cardinality is satisfied by attributes in that group. For example, the following expression constraint is satisfied by any clinical finding whose definition contains an attribute group with two or more non-redundant finding sites.

```
< 404684003 | Clinical finding391 :
  { [2..*] 363698007 | Finding site392 =< 91723000 | Anatomical structure393 }
```

6.3.2 Attribute Group Cardinality

Minimum and maximum cardinalities may also be applied to attribute groups. A minimum attribute group cardinality of X constrains the valid clinical meanings to those which have at least (i.e. >=) X non-redundant attribute groups that match the given attribute group criteria. A maximum cardinality of Y constrains the valid clinical meanings to those which have at most (i.e. <=) Y non-redundant attribute groups that match the given attribute group criteria. For example, a cardinality of "[1..2]" indicates that all clinical meanings that satisfy the given expression constraint must have at least one and at most two attribute groups that match the given attribute group criteria.

The expression constraint below is satisfied only by products with one, two or three attribute groups, which each contain at least one active ingredient relationship.

384 <http://snomed.info/id/299701004>

385 <http://snomed.info/id/62413002>

386 <http://snomed.info/id/363698007>

387 <http://snomed.info/id/299701004>

388 <http://snomed.info/id/404684003>

389 <http://snomed.info/id/363698007>

390 <http://snomed.info/id/91723000>

391 <http://snomed.info/id/404684003>

392 <http://snomed.info/id/363698007>

393 <http://snomed.info/id/91723000>

```
< 373873005 |Pharmaceutical / biologic product394 :
  [1..3] { [1..*] 127489000 |Has active ingredient395 =< 105590001 |Substance396 }
```

Please note that the above expression constraint is equivalent to:

```
< 373873005 |Pharmaceutical / biologic product397 :
  [1..3] { 127489000 |Has active ingredient398 =< 105590001 |Substance399 }
```

And may be written using the long syntax as:

```
descendantOf 373873005 |Pharmaceutical / biologic product400 :
  [1 to 3] { [1 to many] 127489000 |Has active ingredient401 =
    descendantOf 105590001 |Substance402 }
```

6.3.2.1 Unconstrained Cardinalities

As with attribute cardinalities, a minimum cardinality of '0' indicates that there is *no* constraint on the minimum number of attribute groups that may match the given attribute group criteria. For example, the following expression constraint is satisfied only by products with at most one attribute group containing an active ingredient relationship (i.e. the maximum attribute group cardinality is '1' and the minimum attribute group cardinality is unconstrained).

```
< 373873005 |Pharmaceutical / biologic product403 :
  [0..1] { 127489000 |Has active ingredient404 =< 105590001 |Substance405 }
```

Using the long syntax, this may be represented as:

```
descendantOf 373873005 |Pharmaceutical / biologic product406 :
  [0 to 1] { 127489000 |Has active ingredient407 = descendantOf 105590001 |Substance408 }
```

394 <http://snomed.info/id/373873005>

395 <http://snomed.info/id/127489000>

396 <http://snomed.info/id/105590001>

397 <http://snomed.info/id/373873005>

398 <http://snomed.info/id/127489000>

399 <http://snomed.info/id/105590001>

400 <http://snomed.info/id/373873005>

401 <http://snomed.info/id/127489000>

402 <http://snomed.info/id/105590001>

403 <http://snomed.info/id/373873005>

404 <http://snomed.info/id/127489000>

405 <http://snomed.info/id/105590001>

406 <http://snomed.info/id/373873005>

407 <http://snomed.info/id/127489000>

408 <http://snomed.info/id/105590001>

A maximum cardinality of '*' (or 'many') indicates that there is *no* constraint on the maximum number of attribute groups that may match the given attribute group criteria. For example, the following expression constraint is satisfied only by products that have at least one attribute group containing an active ingredient relationship (i.e. the minimum attribute group cardinality is '1' and the maximum attribute group cardinality is unconstrained).

```
< 373873005 |Pharmaceutical / biologic product409 :
  [1..*]{ 127489000 |Has active ingredient410 =< 105590001 |Substance411 }
```

Using the long syntax, this may be represented as:

```
descendantOf 373873005 |Pharmaceutical / biologic product412 :
  [1 to *]{ 127489000 |Has active ingredient413 = descendantOf 105590001 |Substance414 }
```

A cardinality of [0..*] should therefore never be used as this indicates that the given attribute group is not being constrained in any way, and is therefore a redundant part of the expression constraint.

6.3.2.2 Default Cardinalities

As with attribute cardinality, the default attribute group cardinality, where not explicitly stated, is [1..*]. Therefore, the following four expression constraints are equivalent.

```
< 373873005 |Pharmaceutical / biologic product415 :
  { 127489000 |Has active ingredient416 =< 105590001 |Substance417 }
```

```
< 373873005 |Pharmaceutical / biologic product418 :
  {[1..*] 127489000 |Has active ingredient419 =< 105590001 |Substance420 }
```

```
< 373873005 |Pharmaceutical / biologic product421 :
  [1..*]{ 127489000 |Has active ingredient422 =< 105590001 |Substance423 }
```

409 <http://snomed.info/id/373873005>

410 <http://snomed.info/id/127489000>

411 <http://snomed.info/id/105590001>

412 <http://snomed.info/id/373873005>

413 <http://snomed.info/id/127489000>

414 <http://snomed.info/id/105590001>

415 <http://snomed.info/id/373873005>

416 <http://snomed.info/id/127489000>

417 <http://snomed.info/id/105590001>

418 <http://snomed.info/id/373873005>

419 <http://snomed.info/id/127489000>

420 <http://snomed.info/id/105590001>

421 <http://snomed.info/id/373873005>

422 <http://snomed.info/id/127489000>

423 <http://snomed.info/id/105590001>

```
< 373873005 |Pharmaceutical / biologic product424 :
  [1..*] { [1..*] 127489000 |Has active ingredient425 = < 105590001 |Substance426 }
```

6.3.2.3 Non-redundant Attribute Groups

As mentioned above, only non-redundant defining attributes are included in the cardinality count. Therefore, the following postcoordinated expression:

```
< 404684003 |Clinical finding427 :
  { 363698007 |Finding site428 = 299701004 |Bone of forearm429 },
  { 363698007 |Finding site430 = 62413002 |Bone structure of radius431 }
```

will successfully satisfy the expression constraint:

```
< 404684003 |Clinical finding432 :
  [1..1] { 363698007 |Finding site433 = < 91723000 |Anatomical structure434 }
```

This is because 299701004 | Bone of forearm⁴³⁵ is a supertype of 62413002 | Bone structure of radius⁴³⁶ and therefore the attribute group " { 363698007 |Finding site⁴³⁷ = 299701004 |Bone of forearm⁴³⁸ } " is redundant.

6.3.2.4 Attribute and Attribute Group Cardinalities

Attribute cardinalities and attribute group cardinalities can be used together to achieve a combined effect. For example, to represent the set of clinical findings which have *no* attribute groups that contain two or more finding site attributes (in the same attribute group), the following expression constraint can be used:

424 <http://snomed.info/id/373873005>

425 <http://snomed.info/id/127489000>

426 <http://snomed.info/id/105590001>

427 <http://snomed.info/id/404684003>

428 <http://snomed.info/id/363698007>

429 <http://snomed.info/id/299701004>

430 <http://snomed.info/id/363698007>

431 <http://snomed.info/id/62413002>

432 <http://snomed.info/id/404684003>

433 <http://snomed.info/id/363698007>

434 <http://snomed.info/id/91723000>

435 <http://snomed.info/id/299701004>

436 <http://snomed.info/id/62413002>

437 <http://snomed.info/id/363698007>

438 <http://snomed.info/id/299701004>


```
< 404684003 |Clinical finding439 :
  [0..0] { [2..*] 363698007 |Finding site440 =< 91723000 |Anatomical structure441 }
```

6.3.3 Reverse Cardinalities

When a cardinality constraint is applied to a reversed refinement, it constrains the number of source concepts (matching the given criteria) for which each destination concept may be relevant attribute value.

For example, the following expression constraint represents the substances, which are the active ingredient of exactly three products.

```
< 105590001 |Substance442 : [3..3] R 127489000 |Has active ingredient443 = *
```

If this expression constraint was executed against a simplified substrate containing the following seven relationships:

Source concept	Attribute	Destination concept
412458007 Orphenadrine + aspirin + caffeine ⁴⁴⁴	127489000 Has active ingredient ⁴⁴⁵	372714007 Orphenadrine ⁴⁴⁶
412458007 Orphenadrine + aspirin + caffeine ⁴⁴⁷	127489000 Has active ingredient ⁴⁴⁸	387458008 Aspirin ⁴⁴⁹
412458007 Orphenadrine + aspirin + caffeine ⁴⁵⁰	127489000 Has active ingredient ⁴⁵¹	255641001 Caffeine ⁴⁵²
412096001 Aspirin + codeine ⁴⁵³	127489000 Has active ingredient ⁴⁵⁴	387458008 Aspirin ⁴⁵⁵

439 <http://snomed.info/id/404684003>

440 <http://snomed.info/id/363698007>

441 <http://snomed.info/id/91723000>

442 <http://snomed.info/id/105590001>

443 <http://snomed.info/id/127489000>

444 <http://snomed.info/id/412458007>

445 <http://snomed.info/id/127489000>

446 <http://snomed.info/id/372714007>

447 <http://snomed.info/id/412458007>

448 <http://snomed.info/id/127489000>

449 <http://snomed.info/id/387458008>

450 <http://snomed.info/id/412458007>

451 <http://snomed.info/id/127489000>

452 <http://snomed.info/id/255641001>

453 <http://snomed.info/id/412096001>

454 <http://snomed.info/id/127489000>

455 <http://snomed.info/id/387458008>

Source concept	Attribute	Destination concept
412096001 Aspirin + codeine ⁴⁵⁶	127489000 Has active ingredient ⁴⁵⁷	387494007 Codeine ⁴⁵⁸
424102008 Acetaminophen+aspirin ⁴⁵⁹	127489000 Has active ingredient ⁴⁶⁰	387517004 Acetaminophen ⁴⁶¹
424102008 Acetaminophen+aspirin ⁴⁶²	127489000 Has active ingredient ⁴⁶³	387458008 Aspirin ⁴⁶⁴

then the result would include only the concept 387458008 | Aspirin⁴⁶⁵.

¹[\(see page 82\)](#) For more information about the SNOMED CT distribution view, please refer to the [SNOMED CT Technical Implementation Guide](#)⁴⁶⁶. Please note that full normalization of expressions (as would be performed by a Description Logic classifier) is required prior to evaluation.

²[\(see page 82\)](#) As defined in the [SNOMED CT Technical Implementation Guide](#)⁴⁶⁷. [[a\(see page 82\)](#) [b\(see page 82\)](#)]

6.4 6.4 Conjunction and Disjunction

6.4.1 Compound Expression Constraints

Expression constraints can be built up from smaller parts using conjunction (i.e. AND) and disjunction (i.e. OR). The simplest example of this is where the conjunction or disjunction is used between two simple expressions. For example, the following expression constraint is satisfied only by clinical findings which are *both* a disorder of the lung *and* an edema of the trunk. This gives the same result as a mathematical *intersection* between the set of 19829001 | Disorder of lung⁴⁶⁸ descendants and the set of 301867009 | Edema of trunk⁴⁶⁹ descendants.

< 19829001 |Disorder of lung⁴⁷⁰ AND < 301867009 |Edema of trunk⁴⁷¹

Please note that all keywords are case insensitive, so the following two expression constraints are equivalent to the above:

⁴⁵⁶ <http://snomed.info/id/412096001>

⁴⁵⁷ <http://snomed.info/id/127489000>

⁴⁵⁸ <http://snomed.info/id/387494007>

⁴⁵⁹ <http://snomed.info/id/424102008>

⁴⁶⁰ <http://snomed.info/id/127489000>

⁴⁶¹ <http://snomed.info/id/387517004>

⁴⁶² <http://snomed.info/id/424102008>

⁴⁶³ <http://snomed.info/id/127489000>

⁴⁶⁴ <http://snomed.info/id/387458008>

⁴⁶⁵ <http://snomed.info/id/387458008>

⁴⁶⁶ <http://snomed.org/tig>

⁴⁶⁷ <http://snomed.org/tig>

⁴⁶⁸ <http://snomed.info/id/19829001>

⁴⁶⁹ <http://snomed.info/id/301867009>

⁴⁷⁰ <http://snomed.info/id/19829001>

⁴⁷¹ <http://snomed.info/id/301867009>

```
< 19829001 |Disorder of lung|472 and < 301867009 |Edema of trunk|473
```

```
< 19829001 |Disorder of lung|474 And < 301867009 |Edema of trunk|475
```

The next expression constraint is satisfied only by clinical findings which are *either* a disorder of the lung *or* an edema of the trunk. This gives the same result as a mathematical *union* of the set of 19829001 | Disorder of lung|⁴⁷⁶ descendants and the set of 301867009 | Edema of trunk|⁴⁷⁷ descendants. For this reason, an *OR* operator will usually allow more valid clinical meanings than an *AND* operator.

```
< 19829001 |Disorder of lung|478 OR < 301867009 |Edema of trunk|479
```

Conjunction and disjunction operators may also be combined with the use of the 'member of' function, as shown below:

```
< 19829001 |Disorder of lung|480 AND ^ 700043003 |Example problem list concepts reference set|481
```

This expression constraint is satisfied only by concepts that belong to the 19829001 | Disorder of lung|⁴⁸² hierarchy *and* are also members of the 700043003 | Example problem list concepts reference set|⁴⁸³.

When more than one conjunction or more than one disjunction is used, round brackets can be optionally applied. For example, the following expression constraints are all valid and equivalent to each other:

```
< 19829001 |Disorder of lung|484 AND < 301867009 |Edema of trunk|485 AND  
^ 700043003 |Example problem list concepts reference set|486
```

472 <http://snomed.info/id/19829001>

473 <http://snomed.info/id/301867009>

474 <http://snomed.info/id/19829001>

475 <http://snomed.info/id/301867009>

476 <http://snomed.info/id/19829001>

477 <http://snomed.info/id/301867009>

478 <http://snomed.info/id/19829001>

479 <http://snomed.info/id/301867009>

480 <http://snomed.info/id/19829001>

481 <http://snomed.info/id/700043003>

482 <http://snomed.info/id/19829001>

483 <http://snomed.info/id/700043003>

484 <http://snomed.info/id/19829001>

485 <http://snomed.info/id/301867009>

486 <http://snomed.info/id/700043003>

```
( < 19829001 |Disorder of lung|487 AND < 301867009 |Edema of trunk|488 ) AND
  ^ 700043003 |Example problem list concepts reference set|489
```

```
< 19829001 |Disorder of lung|490 AND ( < 301867009 |Edema of trunk|491 AND
  ^ 700043003 |Example problem list concepts reference set|492 )
```

However, where a conjunction and disjunction are both used together, it is mandatory to use round brackets to disambiguate the meaning of the expression constraint. For example, the following expression constraint is **not** valid:

```
< 19829001 |Disorder of lung|493 AND < 301867009 |Edema of trunk|494 OR
  ^ 700043003 |Example problem list concepts reference set|495
```

And must be expressed (depending on the intended meaning) as either:

```
( < 19829001 |Disorder of lung|496 AND < 301867009 |Edema of trunk|497 ) OR
  ^ 700043003 |Example problem list concepts reference set|498
```

Or as:

```
< 19829001 |Disorder of lung|499 AND ( < 301867009 |Edema of trunk|500 OR
  ^ 700043003 |Example problem list concepts reference set|501 )
```

487 <http://snomed.info/id/19829001>

488 <http://snomed.info/id/301867009>

489 <http://snomed.info/id/700043003>

490 <http://snomed.info/id/19829001>

491 <http://snomed.info/id/301867009>

492 <http://snomed.info/id/700043003>

493 <http://snomed.info/id/19829001>

494 <http://snomed.info/id/301867009>

495 <http://snomed.info/id/700043003>

496 <http://snomed.info/id/19829001>

497 <http://snomed.info/id/301867009>

498 <http://snomed.info/id/700043003>

499 <http://snomed.info/id/19829001>

500 <http://snomed.info/id/301867009>

501 <http://snomed.info/id/700043003>

6.4.2 Attribute Conjunction and Disjunction

Conjunction and disjunction may be used within refinements in a variety of ways. The most common way of using these operators in a refinement is to define the conjunction or disjunction of individual attributes.

For example, the expression constraint below, in which the comma between the two attributes represents conjunction, is satisfied only by clinical findings which have *both* a finding site of pulmonary valve structure (or subtype) *and* an associated morphology of stenosis (or subtype).

```
< 404684003 |Clinical finding|502 :
  363698007 |Finding site|503 = << 39057004 |Pulmonary valve structure|504 ,
  116676008 |Associated morphology|505 = << 415582006 |Stenosis|506
```

This expression constraint can equivalently be expressed as:

```
< 404684003 |Clinical finding|507 :
  363698007 |Finding site|508 = << 39057004 |Pulmonary valve structure|509 AND
  116676008 |Associated morphology|510 = << 415582006 |Stenosis|511
```

The following example uses the disjunction operator (OR) to represent the disjunction of two attributes. This constraint is satisfied only by clinical findings which have *either* an associated morphology of 'infarct' (or subtype) *or* are due to a myocardial infarction (or subtype).

```
< 404684003 |Clinical finding|512 :
  116676008 |Associated morphology|513 = << 55641003 |Infarct|514 OR
  42752001 |Due to|515 = << 22298006 |Myocardial infarction|516
```

When more than one conjunction or more than one disjunction is used in a refinement, round brackets can be optionally applied. For example, the following expression constraints are all valid and equivalent to each other:

502 <http://snomed.info/id/404684003>

503 <http://snomed.info/id/363698007>

504 <http://snomed.info/id/39057004>

505 <http://snomed.info/id/116676008>

506 <http://snomed.info/id/415582006>

507 <http://snomed.info/id/404684003>

508 <http://snomed.info/id/363698007>

509 <http://snomed.info/id/39057004>

510 <http://snomed.info/id/116676008>

511 <http://snomed.info/id/415582006>

512 <http://snomed.info/id/404684003>

513 <http://snomed.info/id/116676008>

514 <http://snomed.info/id/55641003>

515 <http://snomed.info/id/42752001>

516 <http://snomed.info/id/22298006>

```
< 404684003 |Clinical finding|517 :
  363698007 |Finding site|518 = << 39057004 |Pulmonary valve structure|519 AND
  116676008 |Associated morphology|520 = << 415582006 |Stenosis|521 AND
  42752001 |Due to|522 = << 445238008 |Malignant carcinoid tumor|523
```

```
< 404684003 |Clinical finding|524 :
  ( 363698007 |Finding site|525 = << 39057004 |Pulmonary valve structure|526 AND
  116676008 |Associated morphology|527 = << 415582006 |Stenosis|528 ) AND
  42752001 |Due to|529 = << 445238008 |Malignant carcinoid tumor|530
```

```
< 404684003 |Clinical finding|531 :
  363698007 |Finding site|532 = << 39057004 |Pulmonary valve structure|533 AND
  ( 116676008 |Associated morphology|534 = << 415582006 |Stenosis|535 AND
  42752001 |Due to|536 = << 445238008 |Malignant carcinoid tumor|537 )
```

However, where a conjunction and disjunction are both used together in a refinement, it is mandatory to use brackets to disambiguate the meaning of the expression constraint.

For example, the following expression constraint is **not** valid:

517 <http://snomed.info/id/404684003>
 518 <http://snomed.info/id/363698007>
 519 <http://snomed.info/id/39057004>
 520 <http://snomed.info/id/116676008>
 521 <http://snomed.info/id/415582006>
 522 <http://snomed.info/id/42752001>
 523 <http://snomed.info/id/445238008>
 524 <http://snomed.info/id/404684003>
 525 <http://snomed.info/id/363698007>
 526 <http://snomed.info/id/39057004>
 527 <http://snomed.info/id/116676008>
 528 <http://snomed.info/id/415582006>
 529 <http://snomed.info/id/42752001>
 530 <http://snomed.info/id/445238008>
 531 <http://snomed.info/id/404684003>
 532 <http://snomed.info/id/363698007>
 533 <http://snomed.info/id/39057004>
 534 <http://snomed.info/id/116676008>
 535 <http://snomed.info/id/415582006>
 536 <http://snomed.info/id/42752001>
 537 <http://snomed.info/id/445238008>

```
< 404684003 |Clinical finding|538 :
  363698007 |Finding site|539 = << 39057004 |Pulmonary valve structure|540 AND
  116676008 |Associated morphology|541 = << 415582006 |Stenosis|542 OR
  42752001 |Due to|543 = << 445238008 |Malignant carcinoid tumor|544
```

And must be expressed (depending on the intended meaning) as either:

```
< 404684003 |Clinical finding|545 :
  ( 363698007 |Finding site|546 = << 39057004 |Pulmonary valve structure|547 AND
  116676008 |Associated morphology|548 = << 415582006 |Stenosis|549 ) OR
  42752001 |Due to|550 = << 445238008 |Malignant carcinoid tumor|551
```

Or as:

```
< 404684003 |Clinical finding|552 :
  363698007 |Finding site|553 = << 39057004 |Pulmonary valve structure|554 AND
  ( 116676008 |Associated morphology|555 = << 415582006 |Stenosis|556 OR
  42752001 |Due to|557 = << 445238008 |Malignant carcinoid tumor|558 )
```

6.4.3 Attribute Group Conjunction and Disjunction

Similarly, conjunction and disjunction may be defined between attribute groups. The following expression constraint is satisfied only by clinical findings which *either* have a finding site of pulmonary valve structure (or subtype) and an associated morphology of stenosis (or subtype), *OR* have a finding site of right ventricular structure (or subtype) and an associated morphology of hypertrophy (or subtype).

538 <http://snomed.info/id/404684003>

539 <http://snomed.info/id/363698007>

540 <http://snomed.info/id/39057004>

541 <http://snomed.info/id/116676008>

542 <http://snomed.info/id/415582006>

543 <http://snomed.info/id/42752001>

544 <http://snomed.info/id/445238008>

545 <http://snomed.info/id/404684003>

546 <http://snomed.info/id/363698007>

547 <http://snomed.info/id/39057004>

548 <http://snomed.info/id/116676008>

549 <http://snomed.info/id/415582006>

550 <http://snomed.info/id/42752001>

551 <http://snomed.info/id/445238008>

552 <http://snomed.info/id/404684003>

553 <http://snomed.info/id/363698007>

554 <http://snomed.info/id/39057004>

555 <http://snomed.info/id/116676008>

556 <http://snomed.info/id/415582006>

557 <http://snomed.info/id/42752001>

558 <http://snomed.info/id/445238008>

```
< 404684003 |Clinical finding559 :
  { 363698007 |Finding site560 = << 39057004 |Pulmonary valve structure561 ,
    116676008 |Associated morphology562 = << 415582006 |Stenosis563 } OR
  { 363698007 |Finding site564 = << 53085002 |Right ventricular structure565 ,
    116676008 |Associated morphology566 = << 56246009 |Hypertrophy567 }
```

6.4.4 Attribute Value Conjunction and Disjunction

Conjunction and disjunction can also be applied to attribute values. The example below is satisfied only by members of the adverse drug reactions reference set for GP/FP health issue, which have a causative agent that is *either* a subtype of pharmaceutical / biologic product *or* a subtype of substance.

```
^ 450990004 |Adverse drug reactions reference set for GP/FP health issue568 :
  246075003 |Causative agent569 = (< 373873005 |Pharmaceutical / biologic product570 OR < 105590001 |
  Substance571 )
```

Similarly, attribute values can also use conjunction. The following expression constraint is satisfied only by clinical findings with an associated morphology whose value is *both* a subtype (or self) of ulcer *and* a subtype (or self) of hemorrhage.

```
< 404684003 |Clinical finding572 : 116676008 |Associated morphology573 =
  (<< 56208002 |Ulcer574 AND << 50960005 |Hemorrhage575 )
```

For more information about nested attribute values and nested compound expression constraints, please refer to [6.7 Nested Expression Constraints](#)(see page 100).

559 <http://snomed.info/id/404684003>

560 <http://snomed.info/id/363698007>

561 <http://snomed.info/id/39057004>

562 <http://snomed.info/id/116676008>

563 <http://snomed.info/id/415582006>

564 <http://snomed.info/id/363698007>

565 <http://snomed.info/id/53085002>

566 <http://snomed.info/id/116676008>

567 <http://snomed.info/id/56246009>

568 <http://snomed.info/id/450990004>

569 <http://snomed.info/id/246075003>

570 <http://snomed.info/id/373873005>

571 <http://snomed.info/id/105590001>

572 <http://snomed.info/id/404684003>

573 <http://snomed.info/id/116676008>

574 <http://snomed.info/id/56208002>

575 <http://snomed.info/id/50960005>

6.5 6.5 Exclusion and Not Equals

6.5.1 Exclusion of Simple Expressions

Exclusion is supported in the SNOMED CT Expression Constraint Language by the binary operator 'MINUS'. Exclusion works in a similar manner to mathematical subtraction. For example, the following expression constraint returns the set of lung disorders which are not a descendant or self of edema of the trunk.

```
<< 19829001 |Disorder of lung|576 MINUS << 301867009 |Edema of trunk|577
```

Logically, this expression constraint takes the set of descendants of 'disorder of lung' and subtracts the set of descendants of 'edema of trunk'. Please note that the keyword 'MINUS' is case insensitive.

Exclusion can also be applied to the membership of a reference set. For example, the following expression constraint returns the set of lung disorders which are not members of the cardiology reference set. That is, the set of descendants or self of 'disorder of lung' minus the set of members of the 'cardiology reference set'.

```
<< 19829001 |Disorder of lung|578 MINUS ^ 700043003 |Example problem list concepts reference set|579
```

Please note that when more than one exclusion operator is used, or when an exclusion operator is used together with a conjunction or disjunction, round brackets must be used to disambiguate the intended meaning.

6.5.2 Exclusion of Attribute Values

Attribute values, represented by compound expression constraints, may also contain exclusions. When this occurs, the expression constraint is satisfied by any concept or expression which has at least one attribute (of the given type) whose value is satisfied by the compound constraint defined in the attribute value. For example, the expression constraint below represents the set of clinical findings, which have an associated morphology that is a descendant or self of ulcer and a descendant or self of hemorrhage, but not a descendant or self of obstruction.

```
< 404684003 |Clinical finding|580 : 116676008 |Associated morphology|581 =  
(( << 56208002 |Ulcer|582 AND << 50960005 |Hemorrhage|583 ) MINUS << 26036001 |Obstruction|584 )
```

⁵⁷⁶ <http://snomed.info/id/19829001>

⁵⁷⁷ <http://snomed.info/id/301867009>

⁵⁷⁸ <http://snomed.info/id/19829001>

⁵⁷⁹ <http://snomed.info/id/700043003>

⁵⁸⁰ <http://snomed.info/id/404684003>

⁵⁸¹ <http://snomed.info/id/116676008>

⁵⁸² <http://snomed.info/id/56208002>

⁵⁸³ <http://snomed.info/id/50960005>

⁵⁸⁴ <http://snomed.info/id/26036001>

6.5.3 Not Equal to Attribute Value

It is also possible to simply state that an attribute value should not fall in a particular range. The example below is satisfied only by clinical findings which have an associated morphology that is not a descendant (or self) of obstruction.

```
< 404684003 |Clinical finding|585 :
  116676008 |Associated morphology|586 != << 26036001 |Obstruction|587
```

Using the long syntax, this expression constraint can be represented as:

```
descendantOf 404684003 |Clinical finding|588 :
  116676008 |Associated morphology|589 NOT = descendantOrSelfOf 26036001 |Obstruction|590
```

To prohibit an attribute from having a value in a particular range, a cardinality of [0..0] must be used. For example, the following expression constraint represents the set of clinical findings which have exactly zero (i.e. they do not have any) associated morphologies that are a descendant or self of obstruction.

```
< 404684003 |Clinical finding|591 :
  [0..0] 116676008 |Associated morphology|592 = << 26036001 |Obstruction|593
```

To prohibit an attribute from having a value *outside* a particular range, a cardinality of [0..0] is used in conjunction with the 'not equal to' comparison operator. For example, the following expression constraint represents the set of clinical findings which have exactly zero associated morphologies that are *not* a descendant or self of obstruction. In other words, clinical findings for which *all* associated morphologies (if any exist) are descendants (or self) of obstruction.

```
< 404684003 |Clinical finding|594 :
  [0..0] 116676008 |Associated morphology|595 != << 26036001 |Obstruction|596
```

585 <http://snomed.info/id/404684003>

586 <http://snomed.info/id/116676008>

587 <http://snomed.info/id/26036001>

588 <http://snomed.info/id/404684003>

589 <http://snomed.info/id/116676008>

590 <http://snomed.info/id/26036001>

591 <http://snomed.info/id/404684003>

592 <http://snomed.info/id/116676008>

593 <http://snomed.info/id/26036001>

594 <http://snomed.info/id/404684003>

595 <http://snomed.info/id/116676008>

596 <http://snomed.info/id/26036001>

If we also want to ensure that at least one associated morphology does exist (and all of these have a value which is a descendant or self of obstruction), then the following expression constraint can be used:

```
< 404684003 |Clinical finding|597 :
  [0..0] 116676008 |Associated morphology|598 != << 26036001 |Obstruction|599 and
  [1..*] 116676008 |Associated morphology|600 = << 26036001 |Obstruction|601
```

Note that the cardinality on the second attribute may be omitted, as [1..*] is assumed by default.

6.6 6.6 Constraint Comments

6.6.1 Comments

SNOMED CT Expression Constraints may also include comments inline within the constraint string to explain, describe or document different aspects of the expression constraints. Each comment begins with a forward slash directly followed by a star (i.e. "/*") and ends with a star directly followed by a forward slash (i.e. "*/"). Comments may be placed anywhere in an expression constraint where whitespace (i.e. "ws") or mandatory whitespace (i.e. "mws") is allowed.

Comments have no effect on the machine processable interpretation of an expression constraint, as they should be ignored during evaluation. For example, the following two expression constraints (the first with comments, and the second without), will evaluate to exactly the same set of concepts:

```
/* Disorders of lung with edema */
< 19829001 |Disorder of lung|602 : /* Descendants of disorder of lung */
  116676008 |Associated morphology|603 = << 79654002 |Edema|604
  /* Where the associated morphology is edema or a subtype */
```

```
< 19829001 |Disorder of lung|605 :
  116676008 |Associated morphology|606 = << 79654002 |Edema|607
```

A comment may include both stars and forward slashes. However a star may never be directly followed by a forward slash within the middle of a comment, as this combination denotes the end of the comment.

⁵⁹⁷ <http://snomed.info/id/404684003>

⁵⁹⁸ <http://snomed.info/id/116676008>

⁵⁹⁹ <http://snomed.info/id/26036001>

⁶⁰⁰ <http://snomed.info/id/116676008>

⁶⁰¹ <http://snomed.info/id/26036001>

⁶⁰² <http://snomed.info/id/19829001>

⁶⁰³ <http://snomed.info/id/116676008>

⁶⁰⁴ <http://snomed.info/id/79654002>

⁶⁰⁵ <http://snomed.info/id/19829001>

⁶⁰⁶ <http://snomed.info/id/116676008>

⁶⁰⁷ <http://snomed.info/id/79654002>

6.7 6.7 Nested Expression Constraints

Expression constraints can be nested in a variety of ways to form nested expression constraints. These nested expression constraints use subexpressions, enclosed in round brackets, in the place of a simple concept reference.

Nested expression constraints can be created by:

- Applying constraint operators to an expression constraint
- Applying the `memberOf` function to an expression constraint
- Combining expression constraints using binary operators
- Adding dotted attributes to expression constraints
- Adding refinements to expression constraints
- Using expression constraints to represent valid attribute names
- Using expression constraints to represent valid attribute values

In this section, we describe each of these approaches to creating nested expression constraints.

6.7.1 Constraint Operators

When a constraint operator is applied to an expression constraint, the resulting set of matching expressions is the union of applying the constraint operator to each of its members.

For example, the following expression constraint represents all the members of the `| Example problem list concepts reference set`⁶⁰⁸ plus the union of the descendants of each of these members.

```
<< ( ^ 700043003 | Example problem list concepts reference set609 )
```

Please note that the brackets in the above expression constraint are optional. In this particular case, removing the brackets does not change the meaning of the constraint.

As another example, the following expression constraint represents the set of all descendants of the `| Finding site`⁶¹⁰ of `| Fracture of bone`⁶¹¹.

```
< ( 125605004 | Fracture of bone612 . 363698007 | Finding site613 )
```

Because the `| Finding site`⁶¹⁴ of `| Fracture of bone`⁶¹⁵ is `272673000 | Bone structure`⁶¹⁶, the above expression constraint is equivalent to:

608 <http://snomed.info/id/700043003>

609 <http://snomed.info/id/700043003>

610 <http://snomed.info/id/363698007>

611 <http://snomed.info/id/125605004>

612 <http://snomed.info/id/125605004>

613 <http://snomed.info/id/363698007>

614 <http://snomed.info/id/363698007>

615 <http://snomed.info/id/125605004>

616 <http://snomed.info/id/272673000>

```
< 272673000 |Bone structure|617
```

Please note that this is *not* the same as the expression constraint:

```
< 125605004 |Fracture of bone|618 . 363698007 |Finding site|619
```

which refers to the set of | Finding site|⁶²⁰ values for any descendant of | Fracture of bone|⁶²¹, and is instead equivalent to:

```
(< 125605004 |Fracture of bone|622 ). 363698007 |Finding site|623
```

See the subsection below on [Dotted Attributes](#)(see page 103) for more information about expression constraints of this form.

6.7.2 MemberOf Function

The memberOf function may also be applied to an expression constraint that returns a set of concept-based reference set concepts. When this is done, the nested expression constraint (to which the memberOf function is applied) must always be enclosed in round brackets.

For example, the expression constraint below is satisfied by the set of concepts which are members of any subtype of | GP/FP health issue reference set|⁶²⁴. In other words, it represents the union of applying the memberOf function to each of the descendants of | GP/FP health issue reference set|⁶²⁵.

```
^ (< 450973005 |GP/FP health issue reference set|626 )
```

The expression constraint above evaluates to the same set of concepts as applying the memberOf function to each individual subtype of 450973005 | GP/FP health issue reference set|⁶²⁷ and then taking the union of these sets. Therefore, when applied to the 20170131 international edition of SNOMED CT, the above expression constraint evaluates to the same set of concepts as the following expression constraint.

617 <http://snomed.info/id/272673000>

618 <http://snomed.info/id/125605004>

619 <http://snomed.info/id/363698007>

620 <http://snomed.info/id/363698007>

621 <http://snomed.info/id/125605004>

622 <http://snomed.info/id/125605004>

623 <http://snomed.info/id/363698007>

624 <http://snomed.info/id/450973005>

625 <http://snomed.info/id/450973005>

626 <http://snomed.info/id/450973005>

627 <http://snomed.info/id/450973005>

```

^ 450990004 | Adverse drug reactions reference set for GP/FP health issue628
OR ^ 450989008 | Allergies reference set for GP/FP health issue629
OR ^ 450985002 | Disorders and diseases reference set for GP/FP health issue630
OR ^ 450988000 | Family history reference set for GP/FP health issue631
OR ^ 450991000 | Processes and procedures reference set for GP/FP health issue632
OR ^ 450986001 | Results reference set for GP/FP health issue633
OR ^ 450992007 | Social history reference set for GP/FP health issue634
OR ^ 450984003 | Symptoms and signs reference set for GP/FP health issue635

```

6.7.3 Compound Expression Constraints

When conjunction (i.e. AND), disjunction (i.e. OR) or exclusion (i.e. MINUS) are applied to one or more complex subexpression constraints, brackets are usually required to nest the subexpression constraints.

For example, the following expression constraint uses brackets around the first complex operand (< 404684003 | Clinical finding⁶³⁶ : 363698007 | Finding site⁶³⁷ = << 39057004 | Pulmonary valve structure⁶³⁸) to apply the 'AND' operator to two expression constraints.

```

(< 404684003 | Clinical finding639 :
  363698007 | Finding site640 = << 39057004 | Pulmonary valve structure641 )
AND ^ 700043003 | Example problem list concepts reference set642

```

An equivalent expression constraint can be achieved by swapping the order of the operands, as shown below.

```

^ 700043003 | Example problem list concepts reference set643
AND (< 404684003 | Clinical finding644 :
  363698007 | Finding site645 = << 39057004 | Pulmonary valve structure646 )

```

628 <http://snomed.info/id/450990004>

629 <http://snomed.info/id/450989008>

630 <http://snomed.info/id/450985002>

631 <http://snomed.info/id/450988000>

632 <http://snomed.info/id/450991000>

633 <http://snomed.info/id/450986001>

634 <http://snomed.info/id/450992007>

635 <http://snomed.info/id/450984003>

636 <http://snomed.info/id/404684003>

637 <http://snomed.info/id/363698007>

638 <http://snomed.info/id/39057004>

639 <http://snomed.info/id/404684003>

640 <http://snomed.info/id/363698007>

641 <http://snomed.info/id/39057004>

642 <http://snomed.info/id/700043003>

643 <http://snomed.info/id/700043003>

644 <http://snomed.info/id/404684003>

645 <http://snomed.info/id/363698007>

646 <http://snomed.info/id/39057004>

Similarly, if both sides of the compound expression are complex expression constraints, then brackets may be required on both sides. For example:

```
( < 404684003 |Clinical finding|647 : 363698007 |Finding site|648 = << 39057004 |Pulmonary valve structure|649
  AND ( < 64572001 |Disease|650 : 116676008 |Associated morphology|651 = << 415582006 |Stenosis|652 ) )
```

6.7.4 Dotted Attributes

Dotted attributes can also be applied to a nested subexpression constraint. When this is done, the resulting subexpression represents the union of the values of the given dotted attribute for any expression that matches the given nested subexpression constraint.

For example, the following expression constraint represents the set of all substances that are the | Direct substance|⁶⁵³ of a | Specimen collection|⁶⁵⁴ procedure that is | Using device|⁶⁵⁵ equal to a subtype (or self) of | Catheter|⁶⁵⁶.

```
( << 17636008 |Specimen collection|657 : 424226004 |Using device|658 = << 19923001 |Catheter|659 ) .
363701004 |Direct substance|660
```

When executed against the 20170131 international edition of SNOMED CT, the above expression constraint matches the following three concepts:

```
78014005 |Urine|661
87612001 |Blood|662
4635002 |Arterial blood|663
```

647 <http://snomed.info/id/404684003>

648 <http://snomed.info/id/363698007>

649 <http://snomed.info/id/39057004>

650 <http://snomed.info/id/64572001>

651 <http://snomed.info/id/116676008>

652 <http://snomed.info/id/415582006>

653 <http://snomed.info/id/363701004>

654 <http://snomed.info/id/17636008>

655 <http://snomed.info/id/424226004>

656 <http://snomed.info/id/19923001>

657 <http://snomed.info/id/17636008>

658 <http://snomed.info/id/424226004>

659 <http://snomed.info/id/19923001>

660 <http://snomed.info/id/363701004>

661 <http://snomed.info/id/78014005>

662 <http://snomed.info/id/87612001>

663 <http://snomed.info/id/4635002>

6.7.5 Refinement

As mentioned in [6.2 Refinements](#)(see page 72), it is possible to apply refinements to nested expression constraints. When a refinement is applied to a complex subexpression constraint, the subexpression constraint must be enclosed in brackets.

For example, the expression constraint below represents the set of all clinical findings and events which occur after some procedure.

```
(<< 404684003 |Clinical finding (finding)|664 OR << 272379006 |Event (event)|665):
  255234002 |After|666 = << 71388002 |Procedure (procedure)|667
```

Attribute Names

In some cases, the valid set of attribute names can be represented using an expression constraint. For example, the expression constraint below represents the set of bone fractures that have no additional defining attributes (besides |Finding site|⁶⁶⁸ and |Associated morphology|⁶⁶⁹).

```
<< 125605004 |Fracture of bone|670 :
  [0..0] ((<< 410662002 |Concept model attribute|671 MINUS 363698007 |Finding site|672)
  MINUS 116676008 |Associated morphology|673) = *
```

Within this expression constraint, the subexpression:

```
(<< 410662002 |Concept model attribute|674 MINUS 363698007 |Finding site|675) MINUS 116676008 |
Associated morphology|676
```

represents the set of attributes that must match the given refinement condition (in this case, these attributes must not appear in the concept definition of matching concepts due to the cardinality of [0..0]).

664 <http://snomed.info/id/404684003>

665 <http://snomed.info/id/272379006>

666 <http://snomed.info/id/255234002>

667 <http://snomed.info/id/71388002>

668 <http://snomed.info/id/363698007>

669 <http://snomed.info/id/116676008>

670 <http://snomed.info/id/125605004>

671 <http://snomed.info/id/410662002>

672 <http://snomed.info/id/363698007>

673 <http://snomed.info/id/116676008>

674 <http://snomed.info/id/410662002>

675 <http://snomed.info/id/363698007>

676 <http://snomed.info/id/116676008>

6.7.6 Attribute Values

Similarly to the SNOMED CT Compositional Grammar, it is also possible to nest expression constraints within an attribute value. Please note that when the attribute value is a simple expression constraint (as per the above examples), brackets are not required around the value. However, when the attribute value is either an expression constraint with a refinement, or a compound expression constraint with a binary operator, then brackets must be placed around the attribute value. For example, the following expression constraint represents the set of clinical findings which are associated with another clinical finding that has an associated morphology of 'infarct' (or subtype).

```
< 404684003 |Clinical finding|677 :
  47429007 |Associated with|678 = (< 404684003 |Clinical finding|679 :
    116676008 |Associated morphology|680 = << 55641003 |Infarct|681 )
```

In this example, brackets are required around the nested attribute value "`< 404684003 |Clinical finding|682 : 116676008 |Associated morphology|683 = << 55641003 |Infarct|684 "`.

6.8 6.8 Description Filters

In this section, we illustrate how description filters can be applied to expression constraints to further restrict the matching concepts.

6.8.1 Overview

Description filter constraints provide the ability to limit the set of concepts, that satisfy a given expression constraint, based on the descriptions associated with each concept. Only concepts that have at least one matching description for each filter criteria will be included in the set of matching concepts. Descriptions can be filtered based on their term, type, language, dialect, acceptability in a given dialect, module, effectiveTime, active status and description identifier. Description filters are specified inside double curly braces, and optionally being with the letter "D". Any filter that does not specify its type is, by default, assumed to be a description filter.

In the following sections, we explain each type of description filter criteria.

6.8.2 Term Filter

Term filters enable an expression constraint to match on only those concepts with an associated description whose term matches the given search term. For example, the following expression constraint is satisfied by SNOMED CT concepts with a description matching the search terms "heart" and "att". This expression constraint works like a term search performed in a SNOMED CT browser. Please note that the "D" (either upper or lower case) at the start of the filter indicates that this is a description filter constraint, rather than a concept filter constraint (see [6.9 Concept](#)

⁶⁷⁷ <http://snomed.info/id/404684003>

⁶⁷⁸ <http://snomed.info/id/47429007>

⁶⁷⁹ <http://snomed.info/id/404684003>

⁶⁸⁰ <http://snomed.info/id/116676008>

⁶⁸¹ <http://snomed.info/id/55641003>

⁶⁸² <http://snomed.info/id/404684003>

⁶⁸³ <http://snomed.info/id/116676008>

⁶⁸⁴ <http://snomed.info/id/55641003>

[Filters](#)(see page 116). If the type of a filter constraint is not specified (as in most of the examples below), then it is assumed that the constraint is a description constraint.

```
* {{ D term = "heart att" }}
```

By default, term filters match using a word-prefix-any-order match technique. This means that each string value in the search term must match the start of a word in the concept's description term, but that these words may appear in any order. This word-prefix-any-order match technique can be explicitly specified in the term filter, using the keyword "match:" before the search term. For example, the following four expression constraints are equivalent, and are each satisfied only by diseases with a description term that includes both a word starting with "heart" **and** a word starting with "att" (in any order).

```
< 64572001 |Disease685 {{ term = "heart att"}}
```

```
< 64572001 |Disease686 {{ term = "heart", term = "att"}}
```

```
< 64572001 |Disease687 {{ term = match:"heart att"}}
```

```
< 64572001 |Disease688 {{ term = "att heart"}}
```

To indicate that a matching description may match either one search term or another, a search term set may be used.

The example below matches only those diseases with a description term containing **either** a word starting with "heart" **or** a word starting with "card" (or both).

```
< 64572001 |Disease689 {{ term = ("heart" "card")}}
```

The other technique that may be used is a wildcard search. This technique is specified using the keyword "wild:" and matches the search term in the expression constraint against the entire candidate description term (rather than just individual words). An asterisk (*) is used as a wildcard to indicate that any (zero to many) characters may appear in the given position.

For example, the expression constraint below will match only diseases with a description term starting with "cardi" and ending with "opathy" with any number of characters between. This term filter would therefore match on terms such as "cardiopathy", "cardiomyopathy" and "cardiac channelopathy", but would **not** match on terms like "atrial cardiopathy" or "Cardiomyopathy (disorder)".

685 <http://snomed.info/id/64572001>

686 <http://snomed.info/id/64572001>

687 <http://snomed.info/id/64572001>

688 <http://snomed.info/id/64572001>

689 <http://snomed.info/id/64572001>

```
< 64572001 |Disease690 {{ term = wild:"cardi*opathy"}}
```

Please note that to perform an exact string match on an entire term, a wildcard search without a wildcard can be used (e.g. term=wild:"cardiopathy"). For example, the following expression constraint will match only diseases with a description term that exactly matches the full string "cardiopathy". This expression constraint will therefore match the concept 56265001 | Heart disease (disorder)⁶⁹¹ (with synonym "Cardiopathy"), but will **not** match the concept 870575001 | Disorder of cardiac atrium (disorder)⁶⁹² (with synonym "Atrial cardiopathy")

```
< 64572001 |Disease693 {{ term = wild:"cardiopathy"}}
```

It is also possible to mix the match techniques in a search term set. For example, the expression constraint below will match those diseases with a description term that either contains a word starting with "gas", or ending with "itis" - e.g. "gastric flu", "gastritis", or "tonsillitis".

```
< 64572001 |Disease694 {{ term = (match:"gas" wild:"*itis")}}
```

If more than one filter is applied, then **all** filters (surrounded in double braces) must match at least one description of a concept, for that concept to satisfy the constraint. The descriptions that match each of the filters can either be the same description, or different descriptions on the same concept.

The expression constraint below matches those diseases which have **both** a description that contains a word starting "eye" **and** a description that ends with "itis". For example, this constraint would match the concept 9826008 | Conjunctivitis (disorder)⁶⁹⁵ (with synonyms "Pink eye disease" and "Conjunctivitis") and the concept 15680481000119104 | Viral conjunctivitis of bilateral eyes (disorder)⁶⁹⁶ (with synonyms "Bilateral viral conjunctivitis" and "Viral conjunctivitis of both eyes"), but would **not** match the concept 45261009 | Viral conjunctivitis (disorder)⁶⁹⁷ (which does not have a synonym matching the word prefix "eye").

```
< 64572001 |Disease698 {{ term = "eye"}} {{ term = wild:"*itis"}}
```

6.8.3 Language Filter

Language filters enable an expression constraint to match on only those concepts with a matching description in a specified language. Language filters use the keyword "language", followed by a comparison operator (e.g. "=" or "!="), and the ISO 639-1 two-character language code (in upper or lowercase).

The expression constraint below matches only those diseases with a Swedish description containing the word prefix "hjärt" - e.g. 41884003 | hjärtpolyp⁶⁹⁹ from the Swedish Edition (20200531)

690 <http://snomed.info/id/64572001>

691 <http://snomed.info/id/56265001>

692 <http://snomed.info/id/870575001>

693 <http://snomed.info/id/64572001>

694 <http://snomed.info/id/64572001>

695 <http://snomed.info/id/9826008>

696 <http://snomed.info/id/15680481000119104>

697 <http://snomed.info/id/45261009>

698 <http://snomed.info/id/64572001>

699 <http://snomed.info/id/41884003>

```
< 64572001 |Disease700 {{ term = "hjärt", language = sv }}
```

The expression constraint below matches only those diseases with a Swedish description containing the word prefix "hjärt" and an English description containing the word prefix "heart" - e.g. 84114007 | hjärtsvikt⁷⁰¹ (with English synonym "Heart failure") from the Swedish Edition (20200531).

```
< 64572001 |Disease702 {{ term = "hjärt", language = sv }} {{ term = "heart", language = en }}
```

6.8.4 Description Type Filter

Type filters enable an expression constraint to match on only those concepts with a matching description of a specified type. Type filters may either use the keyword "type" with the values "fsn", "syn" or "def", or may use the keyword "typeld" with a concept value that is < 900000000000446008 |Description type⁷⁰³.

The following table lists the valid description type keywords in both the brief and full syntax, and their equivalent concept reference alternatives. Please note that the full syntax accepts both the brief and full syntax keywords. If additional description types are required, these must be specified in a filter using the 'typeld' keyword with the corresponding concept reference.

Type Keyword		Typeld Concept Reference
Brief Syntax	Full Syntax	
fsn	fullySpecifiedName	90000000000003001 Fully specified name
syn	synonym	90000000000013009 Synonym
def	definition	90000000000055004 Definition

For example, the expression constraint below matches all the subtypes of |Heart disease⁷⁰⁴, that have a fully specified name containing the word prefix "heart".

```
< 56265001 |Heart disease705 {{ term = "heart", type = fsn }}
```

The following two expression constraints are equivalent, and both match only the subtypes of |Heart disease⁷⁰⁶, which have a Swedish synonym containing the word prefix "hjärt".

⁷⁰⁰ <http://snomed.info/id/64572001>

⁷⁰¹ <http://snomed.info/id/84114007>

⁷⁰² <http://snomed.info/id/64572001>

⁷⁰³ <http://snomed.info/id/900000000000446008>

⁷⁰⁴ <http://snomed.info/id/56265001>

⁷⁰⁵ <http://snomed.info/id/56265001>

⁷⁰⁶ <http://snomed.info/id/56265001>

```
< 56265001 |Heart disease707 {{ term = "hjärt", language = SV, type = syn }}
```

```
< 56265001 |Heart disease708 {{ term = "hjärta", language = sv, typeId = 900000000000013009 |synonym709 }}
```

The two equivalent expression constraints below match the subtypes of |Heart disease⁷¹⁰, which either have a synonym containing the word prefix "heart", or a fully specified name containing the word prefix "heart".

```
< 56265001 |Heart disease711 {{ term = "heart", type = (syn fsn) }}
```

```
< 56265001 |Heart disease712 {{ term = "heart", typeId = ( 900000000000013009 |Synonym713
90000000000003001 |Fully specified name714 ) }}
```

6.8.5 Dialect Filter

Dialect filters enable an expression constraint to match on only those concepts with a matching description in a specified language reference set. Dialect filters may either use the keyword "dialect" with a value that represents a valid alias for a specific language reference set, or may use the keyword "dialectId" with a concept value that is < 9000000000000506000 |Language type reference set⁷¹⁵. Please refer to [Appendix C - Dialect Aliases](#)(see page 193) for a selection of valid dialect aliases for known language reference sets.

For example, the two equivalent expression constraints below will match all subtypes of |Disease⁷¹⁶ that have a description in the Australian English language reference set.

```
< 64572001 |Disease717 {{ dialect = en-au }}
```

```
< 64572001 |Disease718 {{ dialectId = 32570271000036106 |Australian English language reference set719 }}
```

The expression constraint below matches all diseases with a description in the New Zealand English language reference set that has a word starting with "cardio".

707 <http://snomed.info/id/56265001>

708 <http://snomed.info/id/56265001>

709 <http://snomed.info/id/900000000000013009>

710 <http://snomed.info/id/56265001>

711 <http://snomed.info/id/56265001>

712 <http://snomed.info/id/56265001>

713 <http://snomed.info/id/900000000000013009>

714 <http://snomed.info/id/90000000000003001>

715 <http://snomed.info/id/9000000000000506000>

716 <http://snomed.info/id/64572001>

717 <http://snomed.info/id/64572001>

718 <http://snomed.info/id/64572001>

719 <http://snomed.info/id/32570271000036106>

```
< 64572001 |Disease|720 {{ term = "cardio", dialect = en-nz }}
```

In some situations, multiple language reference sets need to be used together to identify an appropriate set of concepts. A filter constraint may include a list of dialects to specify that a matching description may belong to any of the given language reference sets.

For example, the following expression constraint matches all diseases that have a description in either the en-nhs-clinical or en-nhs-pharmacy language reference sets, where that description contains a word starting with the prefix "card".

```
< 64572001 |Disease|721 {{ term = "card", dialect = ( en-nhs-clinical en-nhs-pharmacy ) }}
```

6.8.6 Acceptability Filter

Acceptability filters enable an expression constraint to match on only those concepts with a matching description that has the specified acceptability in the specified language reference set. Acceptability filters must always be applied to a specified dialect. As such, they are represented by placing the required acceptability in brackets after the value of the dialect filter. Acceptabilities can be indicated using either one of the keywords below, or using a concept value that is `< 900000000000511003 |Acceptability|722`. The following table lists the valid acceptability keywords in both the brief and full syntax, and their equivalent concept reference alternatives. Please note that the full syntax accepts both the brief and full syntax keywords.

Acceptability Keyword		AcceptabilityId Concept Reference
Brief Syntax	Full Syntax	
prefer	preferred	900000000000548007 Preferred
accept	acceptable	900000000000549004 Acceptable

For example, the following two expression constraints both match all descendants of disease with a description that matches the word prefix 'box', has the type 'synonym', and has an acceptability of 'preferred' in the en-us language reference set. In other words, this expression constraint matches diseases with a US English preferred term that uses the word prefix 'box'.

```
< 64572001 |Disease|723 {{ term = "box", type = syn, dialect = en-us (prefer) }}
```

⁷²⁰ <http://snomed.info/id/64572001>

⁷²¹ <http://snomed.info/id/64572001>

⁷²² <http://snomed.info/id/900000000000511003>

⁷²³ <http://snomed.info/id/64572001>

```
< 64572001 |Disease724 {{ term = "box", typeId = 90000000000013009 |Synonym725, dialect = en-us (
900000000000548007 |Preferred726 ) }}
```

Multiple dialect filters may be used with different acceptabilities applied to each. For example, the expression constraint below matches on diseases, which have a synonym with word prefix "box" that is preferred in the en-nhs-clinical language reference set **and** is acceptable in the en-gb language reference set.

```
< 64572001 |Disease727 {{ term = "box", type = syn, dialect = en-nhs-clinical (prefer), dialect = en-
gb (accept) }}
```

To support alternative acceptabilities in more than one language reference set, a dialect set can be used. For example, the following two equivalent expression constraints match on diseases, which have a synonym with word prefix "box" that is **either** preferred in the en-gb language reference set **or** preferred in the en-nhs-clinical language reference set.

```
< 64572001 |Disease728 {{ term = "box", type = syn, dialect = ( en-gb (prefer) en-nhs-clinical (prefer) ) }}
```

```
< 64572001 |Disease729 {{ term = "box", type = syn, dialect = ( en-gb en-nhs-clinical ) (prefer) }}
```

6.8.7 Filters with Negation

Filters can use negation in a number of ways. The simplest approach is to use the 'not equal to' comparison operator (e.g. "!=") before the value.

For example, the following expression constraint matches on subtypes of | Fracture of bone⁷³⁰ that do not use the word prefix "fracture" in their US English preferred term.

```
< 125605004 |Fracture of bone731 {{ term != "fracture", type = syn, dialect = en-us (prefer)}}
```

If we remove the type and acceptability filters, as shown below, the remaining expression constraint matches on those subtypes of | Fracture of bone⁷³² which have any US English description that does not contain the word prefix "fracture". Concepts including 263171005 | Fractured nasal bones⁷³³ (with synonym "Broken nose") will match the constraint below.

⁷²⁴ <http://snomed.info/id/64572001>

⁷²⁵ <http://snomed.info/id/90000000000013009>

⁷²⁶ <http://snomed.info/id/900000000000548007>

⁷²⁷ <http://snomed.info/id/64572001>

⁷²⁸ <http://snomed.info/id/64572001>

⁷²⁹ <http://snomed.info/id/64572001>

⁷³⁰ <http://snomed.info/id/125605004>

⁷³¹ <http://snomed.info/id/125605004>

⁷³² <http://snomed.info/id/125605004>

⁷³³ <http://snomed.info/id/263171005>

```
< 125605004 |Fracture of bone734 {{ term != "fracture", dialect = en-us}}
```

To find the set of concepts, for which **all** descriptions match some specified criteria, the expression constraint must use the MINUS operation to exclude concepts that have a non-matching description. For example, the expression constraint below matches all subtypes of |Fracture of bone⁷³⁵, for which **every** description contains the word prefix "fracture". Please note that the filter only applies to the descendants of 125605004 |Fracture of bone⁷³⁶ (i.e. the subexpression directly preceding the filter).

```
< 125605004 |Fracture of bone737 MINUS < 125605004 |Fracture of bone738 {{ term != "fracture"}}
```

This expression constraint can be simplified to the equivalent one below, using the wildcard character '*' (which represents any concept in the substrate).

```
< 125605004 |Fracture of bone739 MINUS * {{ term != "fracture"}}
```

Using a similar principle, the expression constraint below matches all concepts that do not have a preferred term specified in the en-nz language reference set.

```
* MINUS * {{ type = syn, dialect = en-nz (prefer) }}
```

6.8.8 Module Filter

Description module filters enable an expression constraint to match on only those concepts with a matching description that belongs to a specified module. Module filters use the keyword "moduleid" with a concept reference that is < 900000000000443000 |Module⁷⁴⁰.

For example, the expression constraint below matches all subtypes of 195967001 |Asthma⁷⁴¹ with a description that belongs to the US National Library of Medicine maintained module.

```
< 195967001 |Asthma742 {{ D moduleid = 731000124108 |US National Library of Medicine maintained module743 }}
```

And the expression constraint below matches all subtypes of 404684003 |Clinical finding⁷⁴⁴ with a definition that belongs to the international core module.

734 <http://snomed.info/id/125605004>

735 <http://snomed.info/id/125605004>

736 <http://snomed.info/id/125605004>

737 <http://snomed.info/id/125605004>

738 <http://snomed.info/id/125605004>

739 <http://snomed.info/id/125605004>

740 <http://snomed.info/id/900000000000443000>

741 <http://snomed.info/id/195967001>

742 <http://snomed.info/id/195967001>

743 <http://snomed.info/id/731000124108>

744 <http://snomed.info/id/404684003>


```
< 404684003 |Clinical finding745 {{ D type = def, moduleId = 900000000000207008 |SNOMED CT core module|  
746 }}
```

6.8.9 Effective Time Filter

Description effective time filters enable an expression constraint to match on only those concepts with a description that has an effectiveTime matching the specified criteria. Effective time filters may use any of the date comparison operators shown below:

Operator	Name
=	Equals
!=	Not equals
<	Before the given date
<=	Before or on the given date
>	After the given date
>=	After or on the given date

Please note that the value of an effective time filter (if present) must be a 8 digit date, formatted according to ISO 8601's basic calendar date format (i.e. YYYYMMDD). If the effectiveTime of the description in the substrate includes a time and/or time zone designator, these should be ignored when performing the comparison.

For example, the following expression constraint matches all subtypes of 125605004 |Fracture of bone⁷⁴⁷ with a description that has an effective time of 31st January 2021.

```
< 125605004 |Fracture of bone748 {{ D effectiveTime = "20210131" }}
```

And the following expression constraint matches all subtypes of 125605004 |Fracture of bone⁷⁴⁹ with a description that has any effective time that is *not* 31st January 2021.

```
< 125605004 |Fracture of bone750 {{ D effectiveTime != "20210131" }}
```

⁷⁴⁵ <http://snomed.info/id/404684003>

⁷⁴⁶ <http://snomed.info/id/900000000000207008>

⁷⁴⁷ <http://snomed.info/id/125605004>

⁷⁴⁸ <http://snomed.info/id/125605004>

⁷⁴⁹ <http://snomed.info/id/125605004>

⁷⁵⁰ <http://snomed.info/id/125605004>

Similarly, greater than, less than, greater than or equals and less than or equals operators may be used in an effectiveTime filter. For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁵¹ with a description that has an effectiveTime of 31st July 2019 or later (i.e. more recent).

```
< 125605004 |Fracture of bone752 {{ D effectiveTime >= "20190731" }}
```

And the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁵³ with a description that has an effective time of 31st July 2019 or earlier.

```
< 125605004 |Fracture of bone754 {{ D effectiveTime <= "20190731" }}
```

The effectiveTime filter can also use sets of effective times. For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁵⁵ with a description that has an effectiveTime of either 31st January 2019, 31st July 2019, 31st January 2020, or 31st July 2020.

```
< 125605004 |Fracture of bone756 {{ D effectiveTime = ("20190131" "20190731" "20200131" "20200731" )}}
```

And the expression constraint below matches all subtypes of 125605004 | Fracture of bone⁷⁵⁷ with a description, which does *not* have any of the following effective times: 31st January 2019, 31st July 2019, 31st January 2020 or 31st July 2020.

```
< 125605004 |Fracture of bone758 {{ D effectiveTime != ("20190131" "20190731" "20200131" "20200731" )}}
```

To match concepts with unpublished descriptions, to which an effectiveTime has not been assigned, an effectiveTime value of "" can be used. For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁵⁹ with a description to which an effectiveTime has not yet been assigned.

```
< 125605004 |Fracture of bone760 {{ D effectiveTime = "" }}
```

Please note that description effectiveTime filters, which use the comparison operators "<" and ">", will **not** match any descriptions with an effectiveTime = "".

751 <http://snomed.info/id/125605004>

752 <http://snomed.info/id/125605004>

753 <http://snomed.info/id/125605004>

754 <http://snomed.info/id/125605004>

755 <http://snomed.info/id/125605004>

756 <http://snomed.info/id/125605004>

757 <http://snomed.info/id/125605004>

758 <http://snomed.info/id/125605004>

759 <http://snomed.info/id/125605004>

760 <http://snomed.info/id/125605004>

6.8.10 Active Filter

Description active filters enable an expression constraint to match on only those concepts with a description that has a matching active status. Descriptions are either active (i.e. active = 1 or active = "true") or inactive (i.e. active = 0 or active = "false"). By default, only active descriptions are included in the substrate.

For example, the following expression constraints return all concepts in the International Patient Summary reference set, which have an active description.

```
^ 816080008 |International Patient Summary|761 {{ D active = 1 }}
```

```
^ 816080008 |International Patient Summary|762 {{ D active = true }}
```

And the following expression constraints return all concepts in the International Patient Summary reference set, which have an inactive description.

```
^ 816080008 |International Patient Summary|763 {{ D active = 0 }}
```

```
^ 816080008 |International Patient Summary|764 {{ D active = false }}
```

6.8.11 Description Id Filter

Description id filters enable an expression constraint to match on only those concepts with a description that has a matching description identifier. For example, the following expression constraint matches any concept, which has an associated description with the identifier "3032638017". The only concept that matches this expression constraint is 707444001 | Uncomplicated asthma (disorder)⁷⁶⁵.

```
* {{ D id = 3032638017 }}
```

Description id filters can also be applied to other expression constraints, to check whether the concept with the matching description id is in a given set of concepts. For example, the following expression constraint will match any descendant of 195967001 | Asthma (disorder)⁷⁶⁶, which has a description with identifier "3032638017". This can be used to check if the concept with the given description id is a descendant of 195967001 | Asthma (disorder)⁷⁶⁷.

761 <http://snomed.info/id/816080008>

762 <http://snomed.info/id/816080008>

763 <http://snomed.info/id/816080008>

764 <http://snomed.info/id/816080008>

765 <http://snomed.info/id/707444001>

766 <http://snomed.info/id/195967001>

767 <http://snomed.info/id/195967001>

```
< 195967001 |Asthma (disorder)|768 {{ D id = 3032638017 }}
```

Description id filters may also include a set of description identifiers. The following expression constraint will match any descendant of 195967001 |Asthma (disorder)|⁷⁶⁹, with a description whose identifier is either "1208972017", "2674140012" or "3043971012".

```
< 195967001 |Asthma (disorder)|770 {{ D id = (1208972017 2674140012 3043971012) }}
```

Please note that inactive concepts can have active descriptions, so the description id filter can be applied *after* inactive concepts are added to the query results via a [history supplement](#)⁷⁷¹. For example, the following expression constraint matches any descendant of the concept 195967001 |Asthma (disorder)|⁷⁷², or any inactive concept that is associated with a descendant of 195967001 |Asthma (disorder)|⁷⁷³ via an historical association, as long as the concept has a description with the identifier "264553015". The only concept that matches this expression constraint is the inactive concept 170644007 |Mild asthma|⁷⁷⁴.

```
(< 195967001 |Asthma (disorder)|775 {{+HISTORY}}) {{ D id = 264553015 }}
```

For more information on history supplements, please refer to [6.11 History Supplements](#)(see page 124).

6.9 6.9 Concept Filters

In this section, we illustrate how concept filters can be applied to expression constraints to further restrict the matching concepts.

6.9.1 Overview

Concept filter constraints provide the ability to limit the set of concepts that satisfy a given expression constraint, based on the properties of each concept. Only concepts with properties that match the criteria specified in the concept filter constraint will be included in the set of matching concepts. Concepts can be filtered based on their definition status, module, effectiveTime, and active status. In the following sections we explain each of these concept filter criteria.

Definition Status Filter

Definition status filters enable an expression constraint to match on only those concepts with a matching definition status. Definition status filters may either use the keyword 'definitionStatus' with the values "defined" or "primitive", or may use the keyword "definitionStatusId" with a concept value that is < 900000000000444006 |
Definition status|⁷⁷⁶.

⁷⁶⁸ <http://snomed.info/id/195967001>

⁷⁶⁹ <http://snomed.info/id/195967001>

⁷⁷⁰ <http://snomed.info/id/195967001>

⁷⁷¹ <https://confluence.ihtsdotools.org/display/ECL/6.11+History+Supplements>

⁷⁷² <http://snomed.info/id/195967001>

⁷⁷³ <http://snomed.info/id/195967001>

⁷⁷⁴ <http://snomed.info/id/170644007>

⁷⁷⁵ <http://snomed.info/id/195967001>

⁷⁷⁶ <http://snomed.info/id/900000000000444006>

The following table lists the valid `definitionStatus` tokens and their equivalent `definitionStatusId` concept reference alternatives. If additional definition statuses are required, these must be specified in a filter using the '`definitionStatusId`' keyword with the corresponding concept reference.

<code>definitionStatus</code> (token)	<code>definitionStatusId</code> (concept reference)
primitive	900000000000074008 Not sufficiently defined by necessary conditions definition status
defined	900000000000073002 Sufficiently defined by necessary conditions definition status

For example, the expression constraints below match all the primitive subtypes of |Heart disease⁷⁷⁷.

```
< 56265001 |Heart disease778 {{ C definitionStatus = primitive }}
```

```
< 56265001 |Heart disease779 {{ C definitionStatusId = 900000000000074008 |Primitive780 }}
```

Similarly, the two expression constraints below match all the fully defined subtypes of |Heart disease⁷⁸¹.

```
< 56265001 |Heart disease782 {{ C definitionStatus = defined }}
```

```
< 56265001 |Heart disease783 {{ C definitionStatusId = 900000000000073002 |Defined784 }}
```

Please note that Concept filters and [Description Filters](#)(see page 105) can be used together to filter the results of an expression constraint based on both the properties of each concept and the properties of their descriptions. For example the following expression constraint matches all primitive subtypes of 64572001 |Disease⁷⁸⁵, which have at least one description term that includes a word starting with "heart".

```
< 64572001 |Disease786 {{ C definitionStatus = primitive }} {{ D term = "heart" }}
```

⁷⁷⁷ <http://snomed.info/id/56265001>

⁷⁷⁸ <http://snomed.info/id/56265001>

⁷⁷⁹ <http://snomed.info/id/56265001>

⁷⁸⁰ <http://snomed.info/id/900000000000074008>

⁷⁸¹ <http://snomed.info/id/56265001>

⁷⁸² <http://snomed.info/id/56265001>

⁷⁸³ <http://snomed.info/id/56265001>

⁷⁸⁴ <http://snomed.info/id/900000000000073002>

⁷⁸⁵ <http://snomed.info/id/64572001>

⁷⁸⁶ <http://snomed.info/id/64572001>

6.9.2 Module Filter

Module filters enable an expression constraint to match on only those concepts that belong to a specified module [\[see page 0\]](#). Module filters use the keyword "moduleid" with a concept reference that is <

900000000000443000 |Module|⁷⁸⁷ .

For example, the expression constraint below matches all subtypes of 195967001 |Asthma|⁷⁸⁸ that belong to the US National Library of Medicine maintained module.

```
< 195967001 |Asthma|789 {{ C moduleid = 731000124108 |US National Library of Medicine maintained module|  
790 }}
```

And the expression constraint below matches all primitive subtypes of 195967001 |Asthma|⁷⁹¹ that belong to the international core module.

```
< 195967001 |Asthma|792 {{ C definitionStatus = primitive, moduleid = 900000000000207008 |SNOMED CT  
core module|793 }}
```

6.9.3 Effective Time Filter

Effective time filters enable an expression constraint to match on only those concepts with an effectiveTime that matches the specified criteria. Effective time filters may use any of the date comparison operators shown below:

Operator	Name
=	Equals
!=	Not equals
<	Before the given date
<=	Before or on the given date
>	After the given date
>=	After or on the given date

787 <http://snomed.info/id/900000000000443000>

788 <http://snomed.info/id/195967001>

789 <http://snomed.info/id/195967001>

790 <http://snomed.info/id/731000124108>

791 <http://snomed.info/id/195967001>

792 <http://snomed.info/id/195967001>

793 <http://snomed.info/id/900000000000207008>

Please note that the value of an effective time filter (if present) must be a 8 digit date, formatted according to ISO 8601's basic calendar date format (i.e. YYYYMMDD). If the effectiveTime of the concept in the substrate includes a time and/or time zone designator, these should be ignored when performing the comparison.

For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁹⁴ with an effective time of 31st January 2021.

```
< 125605004 | Fracture of bone795 {{ C effectiveTime = "20210131" }}
```

And the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁹⁶ with any effective time that is *not* 31st January 2021.

```
< 125605004 | Fracture of bone797 {{ C effectiveTime != "20210131" }}
```

Similarly, greater than, less than, greater than or equals and less than or equals operators may be used in an effectiveTime filter. For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁷⁹⁸ with an effectiveTime of 31st July 2019 or later (i.e. more recent).

```
< 125605004 | Fracture of bone799 {{ C effectiveTime >= "20190731" }}
```

And the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁸⁰⁰ with an effective time of 31st July 2019 or earlier.

```
< 125605004 | Fracture of bone801 {{ C effectiveTime <= "20190731" }}
```

The effectiveTime filter can also use sets of effective times. For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁸⁰² with an effectiveTime of either 31st January 2019, 31st July 2019, 31st January 2020, or 31st July 2020.

```
< 125605004 | Fracture of bone803 {{ C effectiveTime = ("20190131" "20190731" "20200131" "20200731" ) }}
```

And the expression constraint below matches all subtypes of 125605004 | Fracture of bone⁸⁰⁴ which does *not* have any of the following effective times: 31st January 2019, 31st July 2019, 31st January 2020 or 31st July 2020.

794 <http://snomed.info/id/125605004>

795 <http://snomed.info/id/125605004>

796 <http://snomed.info/id/125605004>

797 <http://snomed.info/id/125605004>

798 <http://snomed.info/id/125605004>

799 <http://snomed.info/id/125605004>

800 <http://snomed.info/id/125605004>

801 <http://snomed.info/id/125605004>

802 <http://snomed.info/id/125605004>

803 <http://snomed.info/id/125605004>

804 <http://snomed.info/id/125605004>

```
< 125605004 |Fracture of bone805 {{ C effectiveTime != ("20190131" "20190731" "20200131" "20200731" )}}
```

To match unpublished concepts to which an effectiveTime has not been assigned, an effectiveTime value of "" can be used. For example, the following expression constraint matches all subtypes of 125605004 | Fracture of bone⁸⁰⁶ to which an effectiveTime has not yet been assigned.

```
< 125605004 |Fracture of bone807 {{ C effectiveTime = "" }}
```

Please note that effectiveTime filters, which use the comparison operators "<" and ">", will **not** match any concepts with an effectiveTime = "".

6.9.4 Active Filter

Active filters enable an expression constraint to match on only those concepts with a matching active status. Concepts are either active (i.e. active = 1 or active = "true") or inactive (i.e. active = 0 or active = "false"). By default, both active and inactive concepts are included in the substrate. This allows inactive members of a reference set to be retrieved (e.g. for historical reference sets, in which the referenced component is intended to be inactive). However, because only active relationships are included in the default substrate, as soon as a refinement or hierarchical operator is used, only active concepts are matched.

For example, the following expression constraints returns only active concepts in the International Patient Summary reference set.

```
^ 816080008 |International Patient Summary808 {{ C active = 1 }}
```

```
^ 816080008 |International Patient Summary809 {{ C active = true }}
```

And the following expression constraints return only inactive concepts in the International Patient Summary reference set.

```
^ 816080008 |International Patient Summary810 {{ C active = 0 }}
```

```
^ 816080008 |International Patient Summary811 {{ C active = false }}
```

805 <http://snomed.info/id/125605004>

806 <http://snomed.info/id/125605004>


807 <http://snomed.info/id/125605004>

808 <http://snomed.info/id/816080008>

809 <http://snomed.info/id/816080008>

810 <http://snomed.info/id/816080008>

811 <http://snomed.info/id/816080008>

 Please note that module filters are not intended to replace the use of simple reference sets to organize content of a particular type. Module filters are instead intended to be used for purposes related to the management of extensions or editions.

6.10 6.10 Member Filters

In this section, we illustrate how filters can be applied to a set of reference set members to restrict the matching values.

6.10.1 Overview

Member filters provide the ability to filter the rows of a reference set, based on the value of specific fields in the reference set. These filters are specified inside double curly braces, and begin with the letter "M".

6.10.2 Member Field Filters

To apply a member filter to one or more reference sets, the fields of those reference sets are matched against specified criteria. Only reference set members whose field values match the given criteria will be included in the results.

For example, the following expression constraint will match all referencedComponentIds (i.e. SNOMED CT concept id) from the active 447562003 | ICD-10 complex map reference set⁸¹² rows, which map to the ICD-10 code "J45.9" (as a word prefix). When applied to the July 2021 international edition, this will match 59 concepts, including 195967001 | Asthma⁸¹³, 707447008 | Exacerbation of severe persistent asthma (disorder)⁸¹⁴ and 401193004 | Asthma confirmed (situation)⁸¹⁵.

```
^ 447562003 | ICD-10 complex map reference set816 {{ M mapTarget = "J45.9" }}
```

Please note that by default, a word-prefix-any-order match is performed. Therefore the following expression constraint will match on rows that have a mapTarget of "J45.0", "J45.1", ..., "J45.8", "J45.9" etc.

```
^ 447562003 | ICD-10 complex map reference set817 {{ M mapTarget = "J45" }}
```

Alternatively, a wildcard search can be performed, to achieve similar results. For example, the expression constraint below will match on rows that have a mapTarget starting with "J45" followed by zero or more other characters (e.g. "J45.0", "J45.1", ..., "J45.8", "J45.9")

```
^ 447562003 | ICD-10 complex map reference set818 {{ M mapTarget = wild:"J45*" }}
```

812 <http://snomed.info/id/447562003>

813 <http://snomed.info/id/195967001>

814 <http://snomed.info/id/707447008>

815 <http://snomed.info/id/401193004>

816 <http://snomed.info/id/447562003>

817 <http://snomed.info/id/447562003>

818 <http://snomed.info/id/447562003>

To achieve an exact string match, a wildcard search (without an '*') can be used. For example, the expression constraint below will match only rows that have a mapTarget of "J45.9".

```
^ 447562003 |ICD-10 complex map reference set819 {{ M mapTarget = wild:"J45.9" }}
```

For more information on wildcard and word-prefix-any-order searching, please refer to [6.8 Description Filters](#)(see page 105).

Multiple field constraints can be applied within a reference set member filter. For example, the following expression constraint will return the referencedComponentId from the 447562003 | ICD-10 complex map reference set⁸²⁰ rows, which have a mapGroup of "2", a mapPriority of "1" and a mapTarget of "J45.9".

```
^ 447562003 |ICD-10 complex map reference set821
{{ M mapGroup = #2, mapPriority = #1, mapTarget = "J45.9" }}
```

Other comparison operators may also be used, when defining field criteria. The available operators depend on the field's datatype, as shown in the table below.

Data type	Comparison Operators	
	Brief syntax	Long syntax
SCTID / Expression	=, !=	=, !=, NOT =, <>
Integer / Decimal	=, !=, <=, <, >=, >	=, !=, NOT =, <>, <=, <, >=, >
String	=, !=	=, !=, NOT =, <>
Boolean	=, !=	=, !=, NOT =, <>
Time	=, !=, <=, <, >=, >	=, !=, NOT =, <>, <=, <, >=, >

In addition, reference set fields of type 'string' may be filtered using the same word-prefix-any-order and wildcard techniques used by the description term filters. For example, the following expression constraint will match all referencedComponentId from the active 447562003 | ICD-10 complex map reference set⁸²² rows that have a mapGroup not equal to 2, a mapPriority less than 2, and a mapTarget that starts with the letter "J".

```
^ 447562003 |ICD-10 complex map reference set823 {{ M mapGroup !
= #2, mapPriority < #2, mapTarget = wild:"J*" }}
```

819 <http://snomed.info/id/447562003>
 820 <http://snomed.info/id/447562003>
 821 <http://snomed.info/id/447562003>
 822 <http://snomed.info/id/447562003>
 823 <http://snomed.info/id/447562003>

Member filters can also be used in combination with the `memberOf` function to support the selection of other fields of a reference set (see [6.1 Simple Expression Constraints](#)(see page 64)). For example, the following expression constraint returns the active SNOMED CT concept that is considered to be the same as the inactive concept [67415000 | Hay asthma](#)⁸²⁴

```
^ [targetComponentId] 900000000000527005 |SAME AS association reference set825
{{ M referencedComponentId = 67415000 |Hay asthma826 }}
```

For more information on the use of reference set field names in ECL, please refer to [Appendix E - Reference Set Fields](#)(see page 206).

For additional ways of specifying queries over the historical association reference sets, please refer to [6.11 History Supplements](#)(see page 124).

6.10.3 Module Filter

Module filters enable an expression constraint to match on only those rows of a reference set that belong to a specified module. Module filters use the keyword "moduleId" with a concept reference that is [900000000000443000 |Module](#)⁸²⁷.

For example, the expression constraint below matches all members of the [900000000000534007 | Module dependency reference set](#)⁸²⁸ that belong to an Australian maintained module.

```
^ 900000000000534007 |Module dependency reference set829 {{ M moduleId = << 32570231000036109 |
Australian maintained module830 }}
```

6.10.4 Effective Time Filter

Effective time filters enable an expression constraint to match on only those rows of a reference set with an `effectiveTime` that matches the specified criteria.

For example, the following expression constraint matches all rows of the [816080008 | International Patient Summary](#)⁸³¹ which have been updated since 31st July 2021 (inclusive). Note that the `referencedComponentId` is the only field returned.

```
^ 816080008 |International Patient Summary832 {{ M effectiveTime >= "20210731" }}
```

824 <http://snomed.info/id/67415000>

825 <http://snomed.info/id/900000000000527005>

826 <http://snomed.info/id/67415000>

827 <http://snomed.info/id/900000000000443000>

828 <http://snomed.info/id/900000000000534007>

829 <http://snomed.info/id/900000000000534007>

830 <http://snomed.info/id/32570231000036109>

831 <http://snomed.info/id/816080008>

832 <http://snomed.info/id/816080008>

6.10.5 Active Filter

Active filters enable an expression constraint to match on only those members of a reference set with a matching active status. Reference set rows are either active (i.e. active = 1 or active = "true") or inactive (i.e. active = 0 or active = "false"). By default, only active members of a reference set are included in the substrate.

For example, the following expression constraints returns the inactive members of the [816080008 | International Patient Summary](http://snomed.info/id/816080008)⁸³³.

```
^ 816080008 |International Patient Summary|834 {{ M active = 0 }}
```

6.11 6.11 History Supplements

In this section, we illustrate how history supplements can be applied to an expression constraint to supplement the results with relevant inactive concepts. History supplements are specified inside double curly braces and begin with a plus sign (i.e. "+") followed by the word "HISTORY".

6.11.1 Background

When capturing new clinical data in an electronic health record (EHR), it is good practice to only allow active SNOMED CT concept identifiers to be recorded. However, SNOMED CT is a dynamic and evolving terminology that must remain consistent with current clinical practice and our evolving understanding of disease processes and treatments. As a result, content may change, become outdated, or need remodelling. As SNOMED CT evolves, concepts that were previously recorded in the EHR may subsequently be inactivated. For legal reasons, it is important that the concepts used at the time the data was recorded should persist in the health records. For this reason, the number of inactive SNOMED CT identifiers in an EHR may increase over time.

As most ECL queries typically return only active SNOMED CT concept identifiers, it may not be possible to retrieve health records containing inactive identifiers using a standard expression constraint. One solution to this challenge, is to execute the expression constraint over an old SNOMED CT edition, in which all required concepts were active. However, given that the logical definitions in SNOMED CT typically improve over time, it is generally accepted that the best ECL results can be obtained using the most recent edition. Therefore, a query approach utilising the most recent edition of SNOMED CT is preferred in many cases.

When a SNOMED CT concept is inactivated, the author first allocates an appropriate reason for the inactivation, and then links the inactivated concept to one or more replacements using historical association reference sets. These historical associations provide a clear understanding of the level of semantic equivalence between the inactivated concept and its replacements where they exist. Vendors can use these historical associations to supplement the active concepts in their query results, with inactive concepts which are linked via appropriate historical associations to the active query results.

On this page, we describe how 'history supplements' can be added to an ECL query, to augment the query results with relevant inactive concepts, and how the resulting queries can be used to retrieve a more complete set of matching health records.

⁸³³ <http://snomed.info/id/816080008>

⁸³⁴ <http://snomed.info/id/816080008>

6.11.2 History Supplements

6.11.2.1 Overview

The member filter syntax, described in [6.10 Member Filters](#) (see page 121), can be used to augment the results of an expression constraint with a set of inactive concepts that are related via an historical association reference set. For example, the following expression constraint can be used to find all the active descendants (and self) of the concept [195967001 | Asthma](#)⁸³⁵, plus any inactive concept that is linked to an active descendant (or self) of [195967001 | Asthma](#)⁸³⁶ via a historical [SAME AS association reference set](#)⁸³⁷ member.

```
<< 195967001 |Asthma838 OR
^ 9000000000000527005 |SAME AS association reference set839 {{ M targetComponentId = << 195967001 |
Asthma840 }}
```

The ECL **history supplement** syntax can be used to simplify queries with this structure. For example, the above query can be expressed in a shorter form as:

```
<< 195967001 |Asthma841 {{ + HISTORY ( 9000000000000527005 |SAME AS association reference set842 ) }}
```

6.11.2.2 Template

The general template [1](#) (see page 0) for history supplements is shown below.

```
[[+ecl @ecl_query]] {{ + HISTORY ( [[+ecl @history_refset_query]] ) }}
```

This general template for history supplements is equivalent to the expanded version shown below. Please note that the first and last slot in this template have the same name, which indicates that they must be populated with the same value (which in this case is the ECL query being performed).

```
[[+ecl @ecl_query]] OR
^ [[+ecl @history_refset_query]] {{ M targetComponentId = [[+ecl @ecl_query]] }}
```

835 <http://snomed.info/id/195967001>

836 <http://snomed.info/id/195967001>

837 <http://snomed.info/id/9000000000000527005>

838 <http://snomed.info/id/195967001>

839 <http://snomed.info/id/9000000000000527005>

840 <http://snomed.info/id/195967001>

841 <http://snomed.info/id/195967001>

842 <http://snomed.info/id/9000000000000527005>

Please note that this history template does not support the [90000000000525002](http://snomed.info/id/90000000000525002) | MOVED FROM association reference set⁸⁴³, as the referencedComponentId refers to the active concept, while the targetComponentId refers to the inactive concept (which is the opposite of typical historical associations). If supporting [MOVED FROM](http://snomed.info/id/90000000000527005)⁸⁴⁴ historical associations, it is recommended that these be added to the [90000000000527005](http://snomed.info/id/90000000000527005) | SAME AS association reference set⁸⁴⁵, to ensure that the template pattern above can be consistently applied.

Also note that the [90000000000524003](http://snomed.info/id/90000000000524003) | MOVED TO association reference set⁸⁴⁶ can be ignored for the purposes of executing historical ECL queries.

6.11.2.3 Profiles

To help implementers of clinical systems write suitable ECL queries that include an appropriate set of inactive concepts, three history supplement profiles are provided. These profiles are designed to support a range of use cases, depending on the level of precision and recall required for inactive content. The three history supplement profiles are described in the table below.

History Profile	Purpose	Historical Association Reference Sets
HISTORY-MIN	<p>Minimum: To support use cases requiring a high level of precision, only historical associations that have a one-to-one equivalence with their replacement are used.</p> <p>Example use case: Clinical decision support</p>	90000000000527005 SAME AS association reference set ⁸⁴⁷
HISTORY-MOD	<p><i>Moderate:</i> To support use cases that must balance precision with recall, only historical associations that</p> <ul style="list-style-type: none"> • Have a one-to-one equivalence with their replacement • Have a one-to-many equivalence with their replacement, or • Are replaced by a concept that represents the intended original meaning closely enough to be clinically useful <p>are used.</p> <p>Example use cases: Clinical research, clinical audit</p>	90000000000527005 SAME AS association reference set ⁸⁴⁸
		90000000000526001 REPLACED BY association reference set ⁸⁴⁹
		90000000000528000 WAS A association reference set ⁸⁵⁰
		1186924009 PARTIALLY EQUIVALENT TO association reference set ⁸⁵¹

843 <http://snomed.info/id/90000000000525002>

844 <http://snomed.info/id/90000000000525002>

845 <http://snomed.info/id/90000000000527005>

846 <http://snomed.info/id/90000000000524003>

847 <http://snomed.info/id/90000000000527005>

848 <http://snomed.info/id/90000000000527005>

849 <http://snomed.info/id/90000000000526001>

850 <http://snomed.info/id/90000000000528000>

851 <http://snomed.info/id/1186924009>

History Profile	Purpose	Historical Association Reference Sets
HISTORY-MAX HISTORY (*)	<p><i>Maximum:</i> To support use cases that require the highest level of recall, where precision is not as important, all possible historical associations are used.</p> <p>Example use case: Identifying patients for manual review.</p>	< 900000000000522004 Historical association reference set ⁸⁵²

For example, if a high level of precision is required, then the HISTORY-MIN profile may be used. The expression constraint below matches descendants or self of 195967001 | Asthma⁸⁵³, plus any inactive concept that is associated with a descendant or self of 195967001 | Asthma⁸⁵⁴ in the 900000000000527005 | SAME AS association reference set⁸⁵⁵ or the 900000000000525002 | MOVED FROM association reference set⁸⁵⁶

```
<< 195967001 |Asthma857 {{ + HISTORY-MIN }}
```

The above expression constraint is equivalent to the one below, with an expanded history supplement.

```
<< 195967001 |Asthma858 {{ + HISTORY ( 900000000000527005 |SAME AS association reference set859 ) }}
```

Use cases that must balance the precision of associated inactive concepts with the level of recall, may use the HISTORY-MOD supplement. The following two expression constraint, which use the history supplement profile and the expanded history supplement respectively, are equivalent.

```
<< 195967001 |Asthma860 {{ + HISTORY-MOD }}
```

```
<< 195967001 |Asthma861 {{ + HISTORY ( 900000000000527005 |SAME AS association reference set862 OR 900000000000526001 |REPLACED BY association reference set863 OR 900000000000528000 |WAS A association reference set864 OR 1186924009 |PARTIALLY EQUIVALENT TO association reference set865 ) }}
```

852 <http://snomed.info/id/900000000000522004>
 853 <http://snomed.info/id/195967001>
 854 <http://snomed.info/id/195967001>
 855 <http://snomed.info/id/900000000000527005>
 856 <http://snomed.info/id/900000000000525002>
 857 <http://snomed.info/id/195967001>
 858 <http://snomed.info/id/195967001>
 859 <http://snomed.info/id/900000000000527005>
 860 <http://snomed.info/id/195967001>
 861 <http://snomed.info/id/195967001>
 862 <http://snomed.info/id/900000000000527005>
 863 <http://snomed.info/id/900000000000526001>
 864 <http://snomed.info/id/900000000000528000>
 865 <http://snomed.info/id/1186924009>

And finally, use cases that require the highest level of recall, may use the HISTORY-MAX supplement profile. This profile uses all possible historical association reference sets to find any potentially relevant inactive concept. The following four expression constraints, which use (a) the history supplement profile, (b) the expanded history supplement, (c) the ANY wildcard symbol ('*'), and (d) the 'history' keyword on its own, are all equivalent. Please note that the 90000000000524003 | MOVED TO association reference set⁸⁶⁶ does not need to be included in the execution of this query, because the targetComponentId is assigned a namespace concept

```
<< 195967001 |Asthma867 {{ + HISTORY-MAX }}
```

```
<< 195967001 |Asthma868 {{ + HISTORY (< 90000000000522004 |Historical association reference set869 ) }}
```

```
<< 195967001 |Asthma870 {{ + HISTORY (*) }}
```

```
<< 195967001 |Asthma871 {{ + HISTORY }}
```

6.11.3 Use Case Examples

Here are two use cases that illustrate how these history supplements may be used in practice:

6.11.3.1 Use Case 1

A clinical system is trying to count the number of patients who have had any type of referral to a service. The system attempts to use the following ECL query to find patient records with a matching procedure.

```
<< 306206005 |Referral to service (procedure)872
```

This query is successfully used to find patient records containing active referral concepts, such as 308461008 | Referral to radiology service (procedure)⁸⁷³.

However, it is discovered that there are 738,090 patient records coded with the inactive SNOMED CT concept 183598009 | Refer to Radiology department (procedure)⁸⁷⁴, which should also be included in the patient count. The clinical system, therefore, adjusts its expression constraint query as shown below, to add a history supplement that includes all inactive concepts with the same meaning as one of the active referral concepts.

866 <http://snomed.info/id/90000000000524003>

867 <http://snomed.info/id/195967001>

868 <http://snomed.info/id/195967001>

869 <http://snomed.info/id/90000000000522004>

870 <http://snomed.info/id/195967001>

871 <http://snomed.info/id/195967001>

872 <http://snomed.info/id/306206005>

873 <http://snomed.info/id/308461008>

874 <http://snomed.info/id/183598009>


```
<< 306206005 |Referral to service (procedure)875 {{ + HISTORY-MIN }}
```

Because the expression constraint "`<< 306206005 |Referral to service (procedure)876`" matches the active concept `308461008 | Referral to radiology service (procedure)877`, and a SAME AS association exists between the inactive concept `183598009 | Refer to Radiology department (procedure)878` and the active concept `308461008 | Referral to radiology service (procedure)879`, the above expression constraint will include the inactive concept `183598009 | Refer to Radiology department (procedure)880`, and therefore successfully find the additional 738,090 patient records in which this inactive referral procedure is recorded.

6.11.3.2 Use Case 2

A clinician is trying to find all patients with any type of breast pain. Knowing that she will be reviewing the patient records prior to acting upon the information, she decides to use a maximal approach to searching historical records. She therefore uses the following ECL query:

```
<< 53430007 |Pain of breast (finding)881 {{ + HISTORY-MAX }}
```

She is delighted to see that patient records containing the inactive concept `315251009 | Unilateral mastalgia (situation)882` are retrieved, as these are indeed relevant to her query. Behind the scenes, the clinical system was able to identify that this inactive concept may be relevant, because it is linked to the active concepts `1010235008 | Pain of left breast883` and `1010237000 | Pain of right breast884` (which are both a type of `| Pain of breast885`) via the POSSIBLY EQUIVALENT TO association reference set⁸⁸⁶.

[\[see page 125\]](#) Note that this template uses the template syntax defined in the [SNOMED CT Template Syntax specification⁸⁸⁷](#), with the addition of an 'ECL' replacement type to indicate that the respective slot must be replaced by a valid ECL expression constraint. This extended template slot syntax is then used within an expression constraint to informally illustrate the pattern required when expanding a history supplement.

6.12 6.12 Top and Bottom

In this section we illustrate how a set of concepts can be filtered, using the top or bottom operators, to find the concepts that are the highest or lowest in the hierarchy within the set.

875 <http://snomed.info/id/306206005>

876 <http://snomed.info/id/306206005>

877 <http://snomed.info/id/308461008>

878 <http://snomed.info/id/183598009>

879 <http://snomed.info/id/308461008>

880 <http://snomed.info/id/183598009>

881 <http://snomed.info/id/53430007>

882 <http://snomed.info/id/315251009>

883 <http://snomed.info/id/1010235008>

884 <http://snomed.info/id/1010237000>

885 <http://snomed.info/id/53430007>

886 <http://snomed.info/id/9000000000000523009>

887 <http://snomed.org/sts>

6.12.1 Top of set

Two consecutive exclamation marks followed by a 'greater than' sign (i.e. "!!>") indicates that the expression constraint is satisfied by the concepts from the results of the subexpression that have the highest position in the hierarchy, relative to one another. In other words the set of concepts, that is the result of the subexpression, will be filtered by concepts that have no ancestors within that set.

For example the following expression constraint represents the highest, or most general, concepts in the hierarchy within the set of 363698007 | Finding sites⁸⁸⁸ from the subtypes of 386617003 | Digestive system finding⁸⁸⁹.

```
!!> ( < 386617003 |Digestive system finding890 . 363698007 |Finding site891 )
```

Using the long syntax, the above expression constraint may be represented as:

```
top ( < 386617003 |Digestive system finding892 . 363698007 |Finding site893 )
```

An equivalent expression constraint without using the top operator can be written:

```
( < 386617003 |Digestive system finding894 . 363698007 |Finding site895 )  
MINUS < ( < 386617003 |Digestive system finding896 . 363698007 |Finding site897 )
```

6.12.2 Bottom of set

Two consecutive exclamation marks followed by a 'less than' sign (i.e. "!!<") indicates that the expression constraint is satisfied by all concepts from the results of the subexpression that have the lowest position in the hierarchy, relative to one another. In other words the set of concepts, that is the result of the subexpression, will be filtered by concepts that have no descendants within that set.

For example, the following expression constraint represents the lowest, or most specific, concepts in the hierarchy within the set of concepts that are both ancestor-and-self of 427089005 | Diabetes mellitus due to cystic fibrosis⁸⁹⁸ and also within the 816080008 | International Patient Summary⁸⁹⁹ reference set:

888 <http://snomed.info/id/363698007>

889 <http://snomed.info/id/386617003>

890 <http://snomed.info/id/386617003>

891 <http://snomed.info/id/363698007>

892 <http://snomed.info/id/386617003>

893 <http://snomed.info/id/363698007>

894 <http://snomed.info/id/386617003>

895 <http://snomed.info/id/363698007>

896 <http://snomed.info/id/386617003>

897 <http://snomed.info/id/363698007>

898 <http://snomed.info/id/427089005>

899 <http://snomed.info/id/816080008>

```
!!< (>> 427089005 |Diabetes mellitus due to cystic fibrosis|900 AND ^ 816080008 |International Patient Summary|901 )
```

Using the long syntax, the above expression constraint may be represented as:

```
bottom (>> 427089005 |Diabetes mellitus due to cystic fibrosis|902 AND ^ 816080008 |International Patient Summary|903 )
```

An equivalent expression constraint without using the bottom operator would be:

```
(>> 427089005 |Diabetes mellitus due to cystic fibrosis|904 AND ^ 816080008 |International Patient Summary|905 )
MINUS > (>> 427089005 |Diabetes mellitus due to cystic fibrosis|906 AND ^ 816080008 |International Patient Summary|907 )
```

6.12.3 Use Case Examples

Here are use cases that illustrate how the top and bottom operators may be used in practice:

6.12.3.1 Use Cases for Bottom

Not all clinical information systems use the same set of SNOMED CT concepts, for various reasons; many countries use their own national editions, countries without a national license may use a SNOMED CT freeset, national guidelines may dictate information model bindings that constrain the set of concepts used.

There are scenarios where it may be necessary to transform clinical records from one information system to another. For example a cross-border research project. If the source system has a broader or different set of concepts than the target system can use then a common strategy is to walk up the hierarchy to find the first common ancestor that is shared by both systems.

One example is development and use of value sets for cross-border sharing, as in MyHealth@EU, where not all participants have a SNOMED CT Affiliate License. The countries who are a member would like to use and share full-SNOMED value sets whereas non-members should only use concepts from the SNOMED CT freeset.

Another example is the NHS Emergency Care Data Set. This is a collection of UK nationally defined subsets for use in a specific context. The bottom operator could be used to transform a specific concept like 45133009 | Neurotoxic shellfish poisoning|⁹⁰⁸ into a less specific concept that is within the 991411000000109 | Emergency care diagnosis simple reference set|⁹⁰⁹ :

900 <http://snomed.info/id/427089005>

901 <http://snomed.info/id/816080008>

902 <http://snomed.info/id/427089005>

903 <http://snomed.info/id/816080008>

904 <http://snomed.info/id/427089005>

905 <http://snomed.info/id/816080008>

906 <http://snomed.info/id/427089005>

907 <http://snomed.info/id/816080008>

908 <http://snomed.info/id/45133009>

909 <http://snomed.info/id/991411000000109>

```
!!< (>> 45133009 | Neurotoxic shellfish poisoning910 AND ^ 991411000000109 | Emergency care diagnosis  
simple reference set911 )
```

This would result in the set of concepts: 118940003 | Disorder of nervous system⁹¹² and 75258004 | Food poisoning⁹¹³.

910 <http://snomed.info/id/45133009>

911 <http://snomed.info/id/991411000000109>

912 <http://snomed.info/id/118940003>

913 <http://snomed.info/id/75258004>

7 7. Implementation Considerations

When implementing the SNOMED CT Expression Constraint Language, the factors that need to be taken into consideration depend on what tasks are being performed. For example, implementations may require expression constraints to be authored, parsed, validated, executed, stored, displayed or exchanged.

The subsections below look at each of these tasks individually and provide a summary of the factors that should be considered prior to implementation. Please note that the guidance provided below is not a step-by-step how-to manual, but instead provides some general insights that we hope are helpful in implementing this language specification.

- [7.1 Authoring](#)(see page 133)
- [7.2 Parsing](#)(see page 135)
- [7.3 Validating](#)(see page 136)
- [7.4 Executing](#)(see page 136)
- [7.5 Storing](#)(see page 136)
- [7.6 Displaying](#)(see page 137)
- [7.7 Exchanging](#)(see page 137)

7.1 7.1 Authoring

Authoring SNOMED CT Expression Constraints can be performed using two main techniques:

1. *Language-based authoring*: This technique involves the author constructing a SNOMED CT Expression Constraint using one of the syntaxes defined in Chapter 5.
2. *Form-based authoring*: This technique involves the author entering values into separate fields of a form, and the clinical system automatically composing the values together into a syntactically correct SNOMED CT Expression Constraint.

7.1.1 Language-Based Authoring

Language-based authoring is useful for situations in which ad hoc expression constraints must be defined which don't necessarily conform to a consistent structure. For example, some expression constraints (e.g. those that define terminology bindings or predefined queries) may be authored by software developers during the design, development or customization of a clinical application. Other expression constraints (e.g. those used to define intentional reference sets or validation queries) may be defined by terminologists during the process of developing a SNOMED CT extension. Expression constraints may also be authored by users who wish to retrieve or analyse information stored in patient records using SNOMED CT (e.g. for clinical, epidemiological or research queries).

To use language-based authoring, the user must be familiar with the basic features of the Expression Constraint Language syntax. There are, however, a number of ways in which a tool can support the user while creating expression constraints, including:

- Validating the syntactical correctness of the expression constraint as it is authored;
- Checking the expression constraint for conformance against the concept model;
- Automatically populating or correcting the term associated with a concept reference;
- Providing integrated tools to search the SNOMED CT hierarchy for concept references to include in the expression constraint;
- Filtering the concept search to those concepts which are valid to use at the given point in the expression constraint (e.g. only showing attribute concepts, or those within the valid range of the given attribute); and
- Suggesting the set of valid operators or characters that may be used at a given point in the expression constraint;

7.1.2 Form-Based Authoring

Form-based authoring is particularly useful when non-technical users need to create constraints or queries which have a consistent structure. In these situations, it may be useful to either:

- Create an 'expression constraint template' in which the attribute values are populated with the values that the user enters into the associated fields of the form;
- Create a form-driven query tool to support a useful subset of possible query structures.

One scenario in which the first form-based approach may be used is when there is a terminology-based dependency between the values of two fields on a user interface. For example, Figure 4 illustrates a simplified Procedures form in which the coded value entered into the *Procedure Type* field must be a descendant of the coded value entered into the *Procedure Category* field. When a *Procedure Category* of "Surgery" (i.e. 387713003 | Surgical procedure⁹¹⁴) is selected, the expression constraint "< 387713003 | Surgical procedure⁹¹⁵" is used to populate the value list for the *Procedure Type* field.

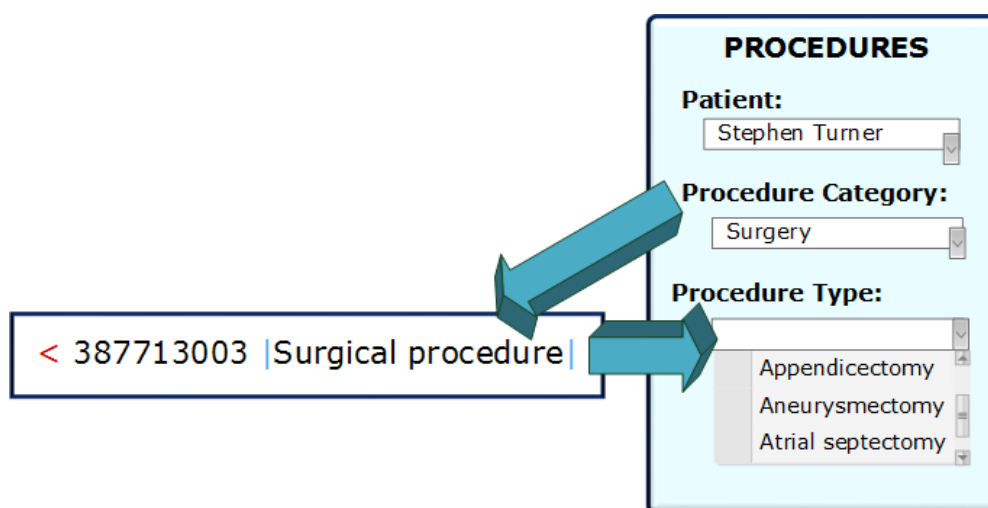


Figure 4: Authoring using expression constraint templates

The second form-based authoring technique mentioned above is a form-driven query tool. Figure 5 below illustrates a very simple form-driven query tool, in which the user selects the required operator (e.g. 'ancestorOf', 'descendantOf', 'memberOf') and operand (e.g. 'Example Problem List') and then defines one or more attribute refinements.

⁹¹⁴ <http://snomed.info/id/387713003>

⁹¹⁵ <http://snomed.info/id/387713003>

QUERY FORM

<p>Operator</p> <div style="border: 1px solid gray; padding: 2px;">Members of</div> <ul style="list-style-type: none"> <li style="border: 1px solid gray; padding: 2px;">Ancestors of <li style="border: 1px solid gray; padding: 2px;">Descendants of <li style="border: 1px solid gray; padding: 2px;">Members of 	<p>Concept</p> <div style="border: 1px solid gray; padding: 2px;">Example Problem</div> <ul style="list-style-type: none"> <li style="border: 1px solid gray; padding: 2px;">Observables <li style="border: 1px solid gray; padding: 2px;">Procedures <li style="border: 1px solid gray; padding: 2px;">Qualifiers
---	--

Refinements	
Name	Value
Finding site	Endocrine system

[^] 700043003 | Example Problem List Subset | :
 363698007 | Finding site | =
 113331007 | Endocrine system |

Figure 5: Authoring using a form-driven query tool

7.2 7.2 Parsing

Parsing is the process of analysing a string of characters according to the rules of a formal grammar. Parsing a SNOMED CT Expression Constraint involves processing the expression constraint string using one of the ABNF syntax specifications defined in [Chapter 5](#) (see page 21), and breaking it into its constituent parts. This creates a representation of the expression constraint that can be further processed. Parsing an expression constraint is required to perform syntactic validation, concept model validation or execution. It should be noted, when parsing, that all keywords in the language are case insensitive.

A number of parser development tools are available which can generate a parser from a context-free grammar written in ABNF, such as the one defined in this document. These tools include:

- APG
- aParse
- abnfgn

Please note, the ABNF syntax defined in this specification was tested using the APG Parser Generator [\[see page 0\]](#).

Other non-ABNF parser generators are also available which can be used with an alternate syntax representation – for example:

- ANTLR
- XText
- ACE

Some of these tools (e.g. XText and ACE) can also be used to generate authoring environments with features such as syntax highlighting and autocompletion.

Alternatively, an expression constraint parser can be created manually using a programming language such as Perl or C++.

www.coasttocoastresearch.com⁹¹⁶

7.3 7.3 Validating

SNOMED CT Expression Constraints can be automatically validated to ensure that they conform to a variety of rules, including:

- Expression constraints must conform to one of the syntaxes defined in [Chapter 5](#)([see page 21](#)). Syntactic validation can be performed using an expression parser, as described in [Section 7.2](#)([see page 135](#));
- Expression constraints must conform to the concept model. This validation can be performed by comparing the parsed expression constraint against the rules defined in the SNOMED CT concept model;
- All concept references included in the expression constraint must be valid. In most cases this means that the concept references must refer to active concepts in the given version and edition of SNOMED CT;
- All concept references used to refer to attribute names must be a descendant of [246061005 | Attribute](#)⁹¹⁷ ;
- All concept references to which a `memberOf` function is applied must be a descendant of [9000000000000455006 | Reference set](#)⁹¹⁸ ;
- All concept references to which a `memberOf` function is applied must contain only `referencedComponentIds` that refer to concepts.

Please note that some of these rules may not apply in all environments.

7.4 7.4 Executing

SNOMED CT Expression Constraints must be evaluated against a given SNOMED CT substrate in order to instantiate the matching set of concepts or expressions. There are a number of possible implementation strategies for the execution of SNOMED CT Expression Constraints, which depend in part on the storage format of the substrate. For example:

- Store SNOMED CT in a relational database, and translate each SNOMED CT Expression Constraint into one or more SQL statements;
- Store SNOMED CT in an RDF store, and translate each SNOMED CT Expression Constraint into a SPARQL query;
- Store SNOMED CT in an XML database, and translate each SNOMED CT Expression Constraint into one or more XQL statements;
- Write a bespoke query execution engine (e.g. in Java or C++) to return matching concepts or expressions.

Each of these strategies requires that the expression constraints are first parsed (and preferably validated) prior to execution.

7.5 7.5 Storing

Storing SNOMED CT Expression Constraints in an expression constraint library may be done for a variety of purposes, including:

⁹¹⁶ <http://www.coasttocoastresearch.com>

⁹¹⁷ <http://snomed.info/id/246061005>

⁹¹⁸ <http://snomed.info/id/9000000000000455006>

- To enable expression constraints to be re-executed (without re-authoring) after updates are made to the SNOMED CT substrate or the expression constraint itself;
- To provide a library of terminology binding constraints against which record instances will be validated;
- To provide a library of concept model constraints against which terminology artefacts (e.g. extensions, expressions) will be validated;
- To provide a library of predefined queries that may be shared by multiple users;
- To provide a library of terminology binding constraints that may be shared within a standards community.

A library of SNOMED CT Expression Constraints may be implemented using a number of techniques, including:

- Creating a Query specification reference set that records the expression constraint as the 'query';
- Creating a customized RF2 reference set with one or more new attributes that allow the expression constraint string and relevant metadata to be recorded;
- Creating a table in a relational database to store the SNOMED CT Expression Constraint and associated metadata;
- Creating a text file with a consistent structural format to store the SNOMED CT Expression Constraint and associated metadata;

In many cases it is useful to assign a unique identifier to each expression constraint in the library, so that they can be indexed and referenced for faster retrieval.

7.6 7.6 Displaying

A number of options exist for displaying SNOMED CT Expression Constraints, including:

- Displaying the expression constraint using SNOMED CT Expression Constraint Language in its originally authored and stored form;
- Converting the expression constraint to use either all symbols (as per the Brief Syntax), or all human-readable operators (as per alternate text introduced in the Long Syntax);
- Enhancing the expression constraint by adding in terms that may have been omitted, or replacing the existing terms with either local-dialect Preferred Terms or Fully Specified Names;
- Hiding the SNOMED CT identifiers for each concept and displaying only the Preferred Terms;
- Enhancing the display by using different font colors for each different part of the expression constraint (e.g. identifiers, terms, vertical bars, and operators), and by using whitespace in a way that improves the readability of the expression;
- Automatically transforming the expression constraint into a human-readable string using a predefined algorithm. For example, a simple algorithm may convert the symbols to text and remove the concept identifiers – e.g. "Descendants of fracture of bone: Finding site = Descendants or self of arm". More sophisticated algorithms may use pattern matching and predefined templates to construct a more natural string;
- Representing the operators, operands and attribute values of the expression constraint by populating a structured form. This approach is primarily suited to expression constraints with a consistent template, where the form can be pre-designed.

Which of these options is most appropriate to use when displaying expression constraints, will depend on a number of factors, including the type of users that will be viewing the constraints, the scope of the required constraint functionality, and the capabilities of the system implementation.

7.7 7.7 Exchanging

SNOMED CT Expression Constraints can be shared between systems and users via a number of methods, including:

- Exchanging an expression constraint string which conforms to the Brief Syntax of the [Expression Constraint Language](#)⁹¹⁹;
- Exchanging an expression constraint identifier, which can be unambiguously interpreted by the receiving system. If this approach is adopted it is recommended that an expression constraint repository is used to ensure that both the sending and receiving systems have a shared and consistent understanding of the meaning of each expression constraint.

Irrespective of the method used, it is recommended that the Brief Syntax of the [SNOMED CT Expression Constraint Language](#)⁹²⁰ be used as the normative syntax for the interoperable sharing of expression constraints.

919 <http://snomed.org/ecl>

920 <http://snomed.org/ecl>

8 Appendix A – Examples Of Valid Expressions

This appendix provides examples of expressions (both precoordinated and postcoordinated) which satisfy each of the expression constraints that were introduced in [Chapter 6](#) (see page 64). This list of examples is not intended to be exhaustive, but rather to provide a representative sample to help clarify the meaning of each constraint. It is assumed that each particular usage of an expression constraint will clearly identify whether or not postcoordinated expressions are part of the valid substrate. Please refer to the [SNOMED CT Languages Github repository](#)⁹²¹ for a set of text files containing each of these examples.

- [A.1 Simple Expression Constraints - Valid Expressions](#) (see page 139)
- [A.2 Refinements - Valid Expressions](#) (see page 142)
- [A.3 Cardinality - Valid Expressions](#) (see page 149)
- [A.4 Conjunction and Disjunction - Valid Expressions](#) (see page 155)
- [A.5 Exclusion and Not Equals - Valid Expressions](#) (see page 159)
- [A.6 Nested Expression Constraints - Valid Expressions](#) (see page 163)

8.1 A.1 Simple Expression Constraints - Valid Expressions

Expression Constraint	Valid Expression (see page 0)	
	Precoordinated	Postcoordinated
404684003 Clinical finding ⁹²²	404684003 Clinical finding ⁹²³	-
< 404684003 Clinical finding ⁹²⁴	64572001 Disease ⁹²⁵	404684003 Clinical finding ⁹²⁶ : 363698007 Finding site ⁹²⁷ = 80891009 Heart structure ⁹²⁸
	56265001 Heart disease ⁹²⁹	
<< 73211009 Diabetes mellitus ⁹³⁰	73211009 Diabetes mellitus ⁹³¹	73211009 Diabetes mellitus ⁹³² : 42752001 Due to ⁹³³ = 61823004 Injury of pancreas ⁹³⁴
	46635009 Diabetes mellitus type 1 ⁹³⁵	

921 <https://github.com/IHTSDO/SNOMEDCT-Languages>

922 <http://snomed.info/id/404684003>

923 <http://snomed.info/id/404684003>

924 <http://snomed.info/id/404684003>

925 <http://snomed.info/id/64572001>

926 <http://snomed.info/id/404684003>

927 <http://snomed.info/id/363698007>

928 <http://snomed.info/id/80891009>

929 <http://snomed.info/id/56265001>

930 <http://snomed.info/id/73211009>

931 <http://snomed.info/id/73211009>

932 <http://snomed.info/id/73211009>

933 <http://snomed.info/id/42752001>

934 <http://snomed.info/id/61823004>

935 <http://snomed.info/id/46635009>

	105401000119101 Diabetes mellitus due to pancreatic injury ⁹³⁶	
<! 404684003 Clinical finding ⁹³⁷	64572001 Disease ⁹³⁸	404684003 Clinical finding ⁹³⁹ : 116676008 Associated morphology ⁹⁴⁰ = 79654002 Edema ⁹⁴¹ [see page 0]
	267038008 Edema ⁹⁴²	
> 40541001 Acute pulmonary edema ⁹⁴³	111273006 Acute respiratory disease ⁹⁴⁴	64572001 Disease ⁹⁴⁵ : 116676008 Associated morphology ⁹⁴⁶ = 79654002 Edema ⁹⁴⁷ , 363698007 Finding site ⁹⁴⁸ = 39607008 Lung structure ⁹⁴⁹
	404684003 Clinical finding ⁹⁵⁰	
	138875005 SNOMED CT concept ⁹⁵¹	
>> 40541001 Acute pulmonary edema ⁹⁵²	40541001 Acute pulmonary edema ⁹⁵³	64572001 Disease ⁹⁵⁴ : 263502005 Clinical course ⁹⁵⁵ = 424124008 Sudden onset AND/OR short duration ⁹⁵⁶ , { 116676008 Associated morphology ⁹⁵⁷ = 40829002 Acute edema ⁹⁵⁸ , 363698007 Finding site ⁹⁵⁹ = 39607008 Lung structure ⁹⁶⁰ }
	111273006 Acute respiratory disease ⁹⁶¹	
	404684003 Clinical finding ⁹⁶²	

936 <http://snomed.info/id/105401000119101>
937 <http://snomed.info/id/404684003>
938 <http://snomed.info/id/64572001>
939 <http://snomed.info/id/404684003>
940 <http://snomed.info/id/116676008>
941 <http://snomed.info/id/79654002>
942 <http://snomed.info/id/267038008>
943 <http://snomed.info/id/40541001>
944 <http://snomed.info/id/111273006>
945 <http://snomed.info/id/64572001>
946 <http://snomed.info/id/116676008>
947 <http://snomed.info/id/79654002>
948 <http://snomed.info/id/363698007>
949 <http://snomed.info/id/39607008>
950 <http://snomed.info/id/404684003>
951 <http://snomed.info/id/138875005>
952 <http://snomed.info/id/40541001>
953 <http://snomed.info/id/40541001>
954 <http://snomed.info/id/64572001>
955 <http://snomed.info/id/263502005>
956 <http://snomed.info/id/424124008>
957 <http://snomed.info/id/116676008>
958 <http://snomed.info/id/40829002>
959 <http://snomed.info/id/363698007>
960 <http://snomed.info/id/39607008>
961 <http://snomed.info/id/111273006>
962 <http://snomed.info/id/404684003>

	138875005 SNOMED CT concept ⁹⁶³	
>! 40541001 Acute pulmonary edema ⁹⁶⁴	111273006 Acute respiratory disease ⁹⁶⁵	19829001 Disorder of lung ⁹⁶⁶ : { 116676008 Associated morphology ⁹⁶⁷ = 79654002 Edema ⁹⁶⁸ , 363698007 Finding site ⁹⁶⁹ = 39607008 Lung structure ⁹⁷⁰ }
	19242006 Pulmonary edema ⁹⁷¹	
^ 700043003 Example problem list concepts reference set ⁹⁷²	394659003 Acute coronary syndrome ⁹⁷³	-
	194828000 Angina ⁹⁷⁴	
	29857009 Chest pain ⁹⁷⁵	
*	138875005 SNOMED CT concept ⁹⁷⁶	404684003 Clinical finding ⁹⁷⁷ : 363698007 Finding site ⁹⁷⁸ = 80891009 Heart structure ⁹⁷⁹
	404684003 Clinical finding ⁹⁸⁰	71388002 Procedure ⁹⁸¹ : 405813007 Procedure site - Direct ⁹⁸² = 66754008 Appendix structure ⁹⁸³

963 <http://snomed.info/id/138875005>
 964 <http://snomed.info/id/40541001>
 965 <http://snomed.info/id/111273006>
 966 <http://snomed.info/id/19829001>
 967 <http://snomed.info/id/116676008>
 968 <http://snomed.info/id/79654002>
 969 <http://snomed.info/id/363698007>
 970 <http://snomed.info/id/39607008>
 971 <http://snomed.info/id/19242006>
 972 <http://snomed.info/id/700043003>
 973 <http://snomed.info/id/394659003>
 974 <http://snomed.info/id/194828000>
 975 <http://snomed.info/id/29857009>
 976 <http://snomed.info/id/138875005>
 977 <http://snomed.info/id/404684003>
 978 <http://snomed.info/id/363698007>
 979 <http://snomed.info/id/80891009>
 980 <http://snomed.info/id/404684003>
 981 <http://snomed.info/id/71388002>
 982 <http://snomed.info/id/405813007>
 983 <http://snomed.info/id/66754008>

	322236009 Paracetamol 500mg tablet ⁹⁸⁴	373873005 Pharmaceutical / biologic product ⁹⁸⁵ : { 127489000 Has active ingredient ⁹⁸⁶ = 412031009 Paracetamol or derivative ⁹⁸⁷ }
--	--	---

¹(see page 139) Where necessary, these examples make some assumptions about the membership of the example reference sets.

²(see page 140) Please note that this makes the assumption that the given expression constraint is executed against a finite set of expressions that has been pre-classified (e.g. in an expression repository), and that after classification there are no intermediate expressions between this expression and 404684003 | Clinical finding⁹⁸⁸.

³(see page 141) Please note that this makes the assumption that the given expression constraint is executed against a finite set of expressions that has been pre-classified (e.g. in an expression repository), and that after classification there are no intermediate expressions between 40541001 | Acute pulmonary edema⁹⁸⁹ and this expression.

8.2 A.2 Refinements - Valid Expressions

Expression Constraint	Valid Expression ¹ (see page 0) ² (see page 0)	
	Precoordinated	Postcoordinated
< 19829001 Disorder of lung ⁹⁹⁰ : 116676008 Associated morphology ⁹⁹¹ = 79654002 Edema ⁹⁹²	11468004 Postoperative pulmonary edema ⁹⁹³	210051003 Injury to heart and lung ⁹⁹⁴ : 116676008 Associated morphology ⁹⁹⁵ = 79654002 Edema ⁹⁹⁶
	276637009 Hemorrhagic pulmonary edema ⁹⁹⁷	

984 <http://snomed.info/id/322236009>

985 <http://snomed.info/id/373873005>

986 <http://snomed.info/id/127489000>

987 <http://snomed.info/id/412031009>

988 <http://snomed.info/id/404684003>

989 <http://snomed.info/id/40541001>

990 <http://snomed.info/id/19829001>

991 <http://snomed.info/id/116676008>

992 <http://snomed.info/id/79654002>

993 <http://snomed.info/id/11468004>

994 <http://snomed.info/id/210051003>

995 <http://snomed.info/id/116676008>

996 <http://snomed.info/id/79654002>

997 <http://snomed.info/id/276637009>

<p>< 19829001 Disorder of lung⁹⁹⁸ : 116676008 Associated morphology⁹⁹⁹ = << 79654002 Edema¹⁰⁰⁰</p>	<p>233709006 Toxic pulmonary edema¹⁰⁰¹</p>	<p>275504005 Lung cyst¹⁰⁰² : 116676008 Associated morphology¹⁰⁰³ = 103619005 Inflammatory edema¹⁰⁰⁴</p>
	<p>233711002 Oxygen-induced pulmonary edema¹⁰⁰⁵</p>	<p>19829001 Disorder of lung¹⁰⁰⁶ : 116676008 Associated morphology¹⁰⁰⁷ = 40829002 Acute edema¹⁰⁰⁸</p>
<p>< 404684003 Clinical finding¹⁰⁰⁹ : 363698007 Finding site¹⁰¹⁰ = << 39057004 Pulmonary valve structure¹⁰¹¹ , 116676008 Associated morphology¹⁰¹² = << 415582006 Stenosis¹⁰¹³</p>	<p>56786000 Pulmonic valve stenosis¹⁰¹⁴</p>	<p>56786000 Pulmonic valve stenosis¹⁰¹⁵ : 363698007 Finding site¹⁰¹⁶ = 90318009 Structure of anulus fibrosus of pulmonary artery¹⁰¹⁷ , 116676008 Associated morphology¹⁰¹⁸ = 88015002 Partial stenosis¹⁰¹⁹</p>
	<p>86299006 Tetralogy of Fallot¹⁰²⁰</p>	<p>404684003 Clinical finding¹⁰²¹ : 363698007 Finding site¹⁰²² = 39057004 Pulmonary valve structure¹⁰²³ ,</p>

998 <http://snomed.info/id/19829001>
 999 <http://snomed.info/id/116676008>
 1000 <http://snomed.info/id/79654002>
 1001 <http://snomed.info/id/233709006>
 1002 <http://snomed.info/id/275504005>
 1003 <http://snomed.info/id/116676008>
 1004 <http://snomed.info/id/103619005>
 1005 <http://snomed.info/id/233711002>
 1006 <http://snomed.info/id/19829001>
 1007 <http://snomed.info/id/116676008>
 1008 <http://snomed.info/id/40829002>
 1009 <http://snomed.info/id/404684003>
 1010 <http://snomed.info/id/363698007>
 1011 <http://snomed.info/id/39057004>
 1012 <http://snomed.info/id/116676008>
 1013 <http://snomed.info/id/415582006>
 1014 <http://snomed.info/id/56786000>
 1015 <http://snomed.info/id/56786000>
 1016 <http://snomed.info/id/363698007>
 1017 <http://snomed.info/id/90318009>
 1018 <http://snomed.info/id/116676008>
 1019 <http://snomed.info/id/88015002>
 1020 <http://snomed.info/id/86299006>
 1021 <http://snomed.info/id/404684003>
 1022 <http://snomed.info/id/363698007>
 1023 <http://snomed.info/id/39057004>

		116676008 Associated morphology ¹⁰²⁴ = 415582006 Stenosis ¹⁰²⁵
* : 246075003 Causative agent ¹⁰²⁶ = 387517004 Paracetamol ¹⁰²⁷	295124009 Paracetamol overdose ¹⁰²⁸	404684003 Clinical finding ¹⁰²⁹ : 246075003 Causative agent ¹⁰³⁰ = 387517004 Paracetamol ¹⁰³¹
	292042007 Adverse reaction to paracetamol ¹⁰³²	
< 404684003 Clinical finding ¹⁰³³ : { 363698007 Finding site ¹⁰³⁴ = << 39057004 Pulmonary valve structure ¹⁰³⁵ , 116676008 Associated morphology ¹⁰³⁶ = << 415582006 Stenosis ¹⁰³⁷ } , { 363698007 Finding site ¹⁰³⁸ = << 53085002 Right ventricular structure ¹⁰³⁹ , 116676008 Associated morphology ¹⁰⁴⁰ = << 56246009 Hypertrophy ¹⁰⁴¹ }	86299006 Tetralogy of Fallot ¹⁰⁴²	404684003 Clinical finding ¹⁰⁴³ : { 363698007 Finding site ¹⁰⁴⁴ = 31689007 Structure of cusp of pulmonic valve ¹⁰⁴⁵ , 116676008 Associated morphology ¹⁰⁴⁶ = 415582006 Stenosis ¹⁰⁴⁷ } , { 363698007 Finding site ¹⁰⁴⁸ = 53085002 Right ventricular structure ¹⁰⁴⁹ , 116676008 Associated morphology ¹⁰⁵⁰ = 125521000 Acute hypertrophy ¹⁰⁵¹ }
	204351007 Fallot's trilogy ¹⁰⁵²	

- 1024 <http://snomed.info/id/116676008>
1025 <http://snomed.info/id/415582006>
1026 <http://snomed.info/id/246075003>
1027 <http://snomed.info/id/387517004>
1028 <http://snomed.info/id/295124009>
1029 <http://snomed.info/id/404684003>
1030 <http://snomed.info/id/246075003>
1031 <http://snomed.info/id/387517004>
1032 <http://snomed.info/id/292042007>
1033 <http://snomed.info/id/404684003>
1034 <http://snomed.info/id/363698007>
1035 <http://snomed.info/id/39057004>
1036 <http://snomed.info/id/116676008>
1037 <http://snomed.info/id/415582006>
1038 <http://snomed.info/id/363698007>
1039 <http://snomed.info/id/53085002>
1040 <http://snomed.info/id/116676008>
1041 <http://snomed.info/id/56246009>
1042 <http://snomed.info/id/86299006>
1043 <http://snomed.info/id/404684003>
1044 <http://snomed.info/id/363698007>
1045 <http://snomed.info/id/31689007>
1046 <http://snomed.info/id/116676008>
1047 <http://snomed.info/id/415582006>
1048 <http://snomed.info/id/363698007>
1049 <http://snomed.info/id/53085002>
1050 <http://snomed.info/id/116676008>
1051 <http://snomed.info/id/125521000>
1052 <http://snomed.info/id/204351007>

<pre><< 404684003 Clinical finding¹⁰⁵³ : << 47429007 Associated with¹⁰⁵⁴ = << 267038008 Edema¹⁰⁵⁵</pre>	<pre>230580009 Myxedema neuropathy¹⁰⁵⁶</pre>	<pre>95356008 Mucosal ulcer¹⁰⁵⁷ : 42752001 Due to¹⁰⁵⁸ = 19242006 Pulmonary edema¹⁰⁵⁹</pre>
<pre>< 27658006 Amoxicillin¹⁰⁶⁰ : 411116001 Has dose form¹⁰⁶¹ = << 385055001 Tablet dose form¹⁰⁶² , { 17999999100 Has basis of strength¹⁰⁶³ = (21999999102 Amoxicillin only¹⁰⁶⁴ : 18999999103 Has strength magnitude¹⁰⁶⁵ >= #200, 19999999101 Has strength unit¹⁰⁶⁶ = 258684004 mg¹⁰⁶⁷ }}</pre>	<pre>374644001 Amoxicillin trihydrate 200 mg tablet¹⁰⁶⁸</pre>	<pre>27658006 Amoxicillin¹⁰⁶⁹ : 411116001 Has dose form¹⁰⁷⁰ = 421026006 Oral tablet¹⁰⁷¹ , { 127489000 Has active ingredient¹⁰⁷² = 96068000 Amoxicillin trihydrate¹⁰⁷³ , 17999999100 Has basis of strength¹⁰⁷⁴ = (21999999102 Amoxicillin only¹⁰⁷⁵ : 18999999103 Has strength magnitude¹⁰⁷⁶ = #500, 19999999101 Has strength unit¹⁰⁷⁷ = 258684004 mg¹⁰⁷⁸)}</pre>

1053 <http://snomed.info/id/404684003>
1054 <http://snomed.info/id/47429007>
1055 <http://snomed.info/id/267038008>
1056 <http://snomed.info/id/230580009>
1057 <http://snomed.info/id/95356008>
1058 <http://snomed.info/id/42752001>
1059 <http://snomed.info/id/19242006>
1060 <http://snomed.info/id/27658006>
1061 <http://snomed.info/id/411116001>
1062 <http://snomed.info/id/385055001>
1063 <http://snomed.org/fictid#17999999100>
1064 <http://snomed.org/fictid#21999999102>
1065 <http://snomed.org/fictid#18999999103>
1066 <http://snomed.org/fictid#19999999101>
1067 <http://snomed.info/id/258684004>
1068 <http://snomed.info/id/374644001>
1069 <http://snomed.info/id/27658006>
1070 <http://snomed.info/id/411116001>
1071 <http://snomed.info/id/421026006>
1072 <http://snomed.info/id/127489000>
1073 <http://snomed.info/id/96068000>
1074 <http://snomed.org/fictid#17999999100>
1075 <http://snomed.org/fictid#21999999102>
1076 <http://snomed.org/fictid#18999999103>
1077 <http://snomed.org/fictid#19999999101>
1078 <http://snomed.info/id/258684004>

<pre>< 27658006 Amoxicillin ¹⁰⁷⁹ : 411116001 Has dose form ¹⁰⁸⁰ = << 385055001 Tablet dose form ¹⁰⁸¹ , { 179999999100 Has basis of strength ¹⁰⁸² ¹⁰⁸³ = (219999999102 Amoxicillin only 189999999103 Has strength magnitude ¹⁰⁸⁴ >= #500, 189999999103 Has strength magnitude ¹⁰⁸⁵ <= #800, 199999999101 Has strength unit ¹⁰⁸⁶ = 258684004 mg ¹⁰⁸⁷)}</pre>	<pre>374646004 Amoxicillin 500 mg tablet ¹⁰⁸⁸</pre>	<pre>27658006 Amoxicillin ¹⁰⁸⁹ : 411116001 Has dose form ¹⁰⁹⁰ = 421026006 Oral tablet ¹⁰⁹¹ , { 179999999100 Has basis of strength ¹⁰⁹² = (219999999102 Amoxicillin only ¹⁰⁹³ : 189999999103 Has strength magnitude ¹⁰⁹⁴ = #750, 199999999101 Has strength unit ¹⁰⁹⁵ = 258684004 mg ¹⁰⁹⁶)}</pre>
<pre>< 373873005 Pharmaceutical / biologic product ¹⁰⁹⁷ : 209999999104 Has trade name ¹⁰⁹⁸ = "PANADOL"</pre>	<pre>259999999103 PANADOL [paracetamol] tablet ¹⁰⁹⁹</pre>	<pre>373873005 Pharmaceutical / biologic product ¹¹⁰⁰ : { 127489000 Has active ingredient ¹¹⁰¹ = 412031009 Paracetamol or derivative ¹¹⁰² }, 209999999104 Has trade name ¹¹⁰³ = "PANADOL"</pre>

1079 <http://snomed.info/id/27658006>
1080 <http://snomed.info/id/411116001>
1081 <http://snomed.info/id/385055001>
1082 <http://snomed.org/fictid#179999999100>
1083 <http://snomed.org/fictid#219999999102>
1084 <http://snomed.org/fictid#189999999103>
1085 <http://snomed.org/fictid#189999999103>
1086 <http://snomed.org/fictid#199999999101>
1087 <http://snomed.info/id/258684004>
1088 <http://snomed.info/id/374646004>
1089 <http://snomed.info/id/27658006>
1090 <http://snomed.info/id/411116001>
1091 <http://snomed.info/id/421026006>
1092 <http://snomed.org/fictid#179999999100>
1093 <http://snomed.org/fictid#219999999102>
1094 <http://snomed.org/fictid#189999999103>
1095 <http://snomed.org/fictid#199999999101>
1096 <http://snomed.info/id/258684004>
1097 <http://snomed.info/id/373873005>
1098 <http://snomed.org/fictid#209999999104>
1099 <http://snomed.org/fictid#259999999103>
1100 <http://snomed.info/id/373873005>
1101 <http://snomed.info/id/127489000>
1102 <http://snomed.info/id/412031009>
1103 <http://snomed.org/fictid#209999999104>

<p>< 91723000 Anatomical structure¹¹⁰⁴ : R 363698007 Finding site¹¹⁰⁵ = < 125605004 Fracture of bone¹¹⁰⁶</p>	<p>85050009 Humerus¹¹⁰⁷</p>	<p>85050009 Humerus¹¹⁰⁸ : 272741003 Laterality¹¹⁰⁹ = 7771000 Left¹¹¹⁰</p>
	<p>71341001 Femur¹¹¹¹</p>	<p>71341001 Femur¹¹¹² : 272741003 Laterality¹¹¹³ = 24028007 Right¹¹¹⁴</p>
<p>< 125605004 Fracture of bone¹¹¹⁵ . 363698007 Finding site¹¹¹⁶</p>	<p>85050009 Humerus¹¹¹⁷</p>	<p>85050009 Humerus¹¹¹⁸ : 272741003 Laterality¹¹¹⁹ = 7771000 Left¹¹²⁰</p>
	<p>71341001 Femur¹¹²¹</p>	<p>71341001 Femur¹¹²² : 272741003 Laterality¹¹²³ = 24028007 Right¹¹²⁴</p>
<p>< 105590001 Substance¹¹²⁵ : R << 127489000 Has active ingredient¹¹²⁶ = < 27658006 Product containing amoxicillin¹¹²⁷</p>	<p>395938000 Clavulanate potassium¹¹²⁸</p>	<p>-</p>

1104 <http://snomed.info/id/91723000>
1105 <http://snomed.info/id/363698007>
1106 <http://snomed.info/id/125605004>
1107 <http://snomed.info/id/85050009>
1108 <http://snomed.info/id/85050009>
1109 <http://snomed.info/id/272741003>
1110 <http://snomed.info/id/7771000>
1111 <http://snomed.info/id/71341001>
1112 <http://snomed.info/id/71341001>
1113 <http://snomed.info/id/272741003>
1114 <http://snomed.info/id/24028007>
1115 <http://snomed.info/id/125605004>
1116 <http://snomed.info/id/363698007>
1117 <http://snomed.info/id/85050009>
1118 <http://snomed.info/id/85050009>
1119 <http://snomed.info/id/272741003>
1120 <http://snomed.info/id/7771000>
1121 <http://snomed.info/id/71341001>
1122 <http://snomed.info/id/71341001>
1123 <http://snomed.info/id/272741003>
1124 <http://snomed.info/id/24028007>
1125 <http://snomed.info/id/105590001>
1126 <http://snomed.info/id/127489000>
1127 <http://snomed.info/id/27658006>
1128 <http://snomed.info/id/395938000>

	387137007 Omeprazole ¹¹²⁹	
< 27658006 Product containing amoxicillin ¹¹³⁰ . << 127489000 Has active ingredient ¹¹³¹	395938000 Clavulanate potassium ¹¹³²	-
	387137007 Omeprazole ¹¹³³	
< 404684003 Clinical finding ¹¹³⁴ : * = 79654002 Edema ¹¹³⁵	19242006 Pulmonary edema ¹¹³⁶	404684003 Clinical finding ¹¹³⁷ : 116676008 Associated morphology ¹¹³⁸ = 79654002 Edema ¹¹³⁹
	97341000119105 P roliferative retinopathy with retinal edema due to type	
< 404684003 Clinical finding ¹¹⁴⁰ : 116676008 Associated morphology ¹¹⁴¹ = *	19242006 Pulmonary edema ¹¹⁴²	404684003 Clinical finding ¹¹⁴³ : 116676008 Associated morphology ¹¹⁴⁴ = 79654002 Edema ¹¹⁴⁵
	263225007 Hip fracture ¹¹⁴⁶	404684003 Clinical finding ¹¹⁴⁷ : 116676008 Associated morphology ¹¹⁴⁸ = 72704001 Fracture ¹¹⁴⁹

1129 <http://snomed.info/id/387137007>
 1130 <http://snomed.info/id/27658006>
 1131 <http://snomed.info/id/127489000>
 1132 <http://snomed.info/id/395938000>
 1133 <http://snomed.info/id/387137007>
 1134 <http://snomed.info/id/404684003>
 1135 <http://snomed.info/id/79654002>
 1136 <http://snomed.info/id/19242006>
 1137 <http://snomed.info/id/404684003>
 1138 <http://snomed.info/id/116676008>
 1139 <http://snomed.info/id/79654002>
 1140 <http://snomed.info/id/404684003>
 1141 <http://snomed.info/id/116676008>
 1142 <http://snomed.info/id/19242006>
 1143 <http://snomed.info/id/404684003>
 1144 <http://snomed.info/id/116676008>
 1145 <http://snomed.info/id/79654002>
 1146 <http://snomed.info/id/263225007>
 1147 <http://snomed.info/id/404684003>
 1148 <http://snomed.info/id/116676008>
 1149 <http://snomed.info/id/72704001>

¹[\(see page 142\)](#) Please note that some of these examples are based on a hypothetical drug concept model. These examples are not intended to reflect any specific drug model.

²[\(see page 142\)](#) SNOMED CT identifiers with the '9999999' namespace were created for example only, and should not be used in a production environment.

8.3 A.3 Cardinality - Valid Expressions

Expression Constraint	Valid Expression ¹ (see page 0)	
	Precoordinated	Postcoordinated
<code>< 373873005 Pharmaceutical / biologic product¹¹⁵⁰ : [1..3] 127489000 Has active ingredient¹¹⁵¹ = < 105590001 Substance¹¹⁵²</code>	<code>322236009 Paracetamol 500mg tablet¹¹⁵³</code>	<code>373873005 Pharmaceutical / biologic product¹¹⁵⁴ : { 127489000 Has active ingredient¹¹⁵⁵ = 412031009 Paracetamol or derivative¹¹⁵⁶ }</code>
	<code>404826002 Benzocaine + butamben + tetracaine hydrochloride¹¹⁵⁷</code>	<code>373873005 Pharmaceutical / biologic product¹¹⁵⁸ : { 127489000 Has active ingredient¹¹⁵⁹ = 412031009 Paracetamol or derivative¹¹⁶⁰ }, { 127489000 Has active ingredient¹¹⁶¹ = 387494007 Codeine¹¹⁶² }</code>
<code>< 373873005 Pharmaceutical / biologic product¹¹⁶³ : [1..1]</code>	<code>370166004 Aspirin 325mg tablet¹¹⁶⁶</code>	<code>373873005 Pharmaceutical / biologic product¹¹⁶⁷ : { 127489000 Has active ingredient</code>

1150 <http://snomed.info/id/373873005>
 1151 <http://snomed.info/id/127489000>
 1152 <http://snomed.info/id/105590001>
 1153 <http://snomed.info/id/322236009>
 1154 <http://snomed.info/id/373873005>
 1155 <http://snomed.info/id/127489000>
 1156 <http://snomed.info/id/412031009>
 1157 <http://snomed.info/id/404826002>
 1158 <http://snomed.info/id/373873005>
 1159 <http://snomed.info/id/127489000>
 1160 <http://snomed.info/id/412031009>
 1161 <http://snomed.info/id/127489000>
 1162 <http://snomed.info/id/387494007>
 1163 <http://snomed.info/id/373873005>
 1166 <http://snomed.info/id/370166004>
 1167 <http://snomed.info/id/373873005>

127489000 Has active ingredient 1164 = < 105590001 Substance ¹¹⁶⁵		¹¹⁶⁸ = 412031009 Paracetamol or derivative ¹¹⁶⁹ }
< 373873005 Pharmaceutical / biologic product ¹¹⁷⁰ : [0..1] 127489000 Has active ingredient 1171 = < 105590001 Substance ¹¹⁷²	279999999108 Inert tablet ¹¹⁷³	373873005 Pharmaceutical / biologic product ¹¹⁷⁴ : { 127489000 Has active ingredient 1175 = 412031009 Paracetamol or derivative ¹¹⁷⁶ }
	370166004 Aspirin 325mg tablet 1177	
< 373873005 Pharmaceutical / biologic product ¹¹⁷⁸ : [1..*] 127489000 Has active ingredient 1179 = < 105590001 Substance ¹¹⁸⁰	7947003 Aspirin ¹¹⁸¹	373873005 Pharmaceutical / biologic product ¹¹⁸² : { 127489000 Has active ingredient 1183 = 412031009 Paracetamol or derivative ¹¹⁸⁴ }, { 127489000 Has active ingredient 1185 = 255641001 Caffeine ¹¹⁸⁶ }, { 127489000 Has active ingredient 1187 = 387458008 Aspirin ¹¹⁸⁸ }
	437867004 Chlorphenamine + dextromethorphan + paracetamol + pseudoephedrine 1189	

1164 <http://snomed.info/id/127489000>
1165 <http://snomed.info/id/105590001>
1168 <http://snomed.info/id/127489000>
1169 <http://snomed.info/id/412031009>
1170 <http://snomed.info/id/373873005>
1171 <http://snomed.info/id/127489000>
1172 <http://snomed.info/id/105590001>
1173 <http://snomed.org/fictid#279999999108>
1174 <http://snomed.info/id/373873005>
1175 <http://snomed.info/id/127489000>
1176 <http://snomed.info/id/412031009>
1177 <http://snomed.info/id/370166004>
1178 <http://snomed.info/id/373873005>
1179 <http://snomed.info/id/127489000>
1180 <http://snomed.info/id/105590001>
1181 <http://snomed.info/id/7947003>
1182 <http://snomed.info/id/373873005>
1183 <http://snomed.info/id/127489000>
1184 <http://snomed.info/id/412031009>
1185 <http://snomed.info/id/127489000>
1186 <http://snomed.info/id/255641001>
1187 <http://snomed.info/id/127489000>
1188 <http://snomed.info/id/387458008>
1189 <http://snomed.info/id/437867004>

<p>< 404684003 Clinical finding¹¹⁹⁰ : [1..1] 363698007 Finding site¹¹⁹¹ = < 91723000 Anatomical structure¹¹⁹²</p>	<p>125596004 Injury of elbow¹¹⁹³</p>	<p>404684003 Clinical finding¹¹⁹⁴ : { 116676008 Associated morphology¹¹⁹⁵ = 72704001 Fracture¹¹⁹⁶ , 363698007 Finding site¹¹⁹⁷ = 299701004 Bone of forearm¹¹⁹⁸ , 363698007 Finding site¹¹⁹⁹ = 62413002 Bone structure of radius¹²⁰⁰ } [2] (see page 0)</p>
<p>< 404684003 Clinical finding¹²⁰¹ : [2..*] 363698007 Finding site¹²⁰² = < 91723000 Anatomical structure¹²⁰³</p>	<p>86299006 Tetralogy of Fallot¹²⁰⁴</p>	<p>404684003 Clinical finding¹²⁰⁵ : { 116676008 Associated morphology¹²⁰⁶ = 72704001 Fracture¹²⁰⁷ , 363698007 Finding site¹²⁰⁸ = 299701004 Bone of forearm¹²⁰⁹ } , { 116676008 Associated morphology¹²¹⁰ = 72704001 Fracture¹²¹¹ , 363698007 Finding site¹²¹² = 702468001 Bone structure of lower leg¹²¹³ }</p>

1190 <http://snomed.info/id/404684003>
 1191 <http://snomed.info/id/363698007>
 1192 <http://snomed.info/id/91723000>
 1193 <http://snomed.info/id/125596004>
 1194 <http://snomed.info/id/404684003>
 1195 <http://snomed.info/id/116676008>
 1196 <http://snomed.info/id/72704001>
 1197 <http://snomed.info/id/363698007>
 1198 <http://snomed.info/id/299701004>
 1199 <http://snomed.info/id/363698007>
 1200 <http://snomed.info/id/62413002>
 1201 <http://snomed.info/id/404684003>
 1202 <http://snomed.info/id/363698007>
 1203 <http://snomed.info/id/91723000>
 1204 <http://snomed.info/id/86299006>
 1205 <http://snomed.info/id/404684003>
 1206 <http://snomed.info/id/116676008>
 1207 <http://snomed.info/id/72704001>
 1208 <http://snomed.info/id/363698007>
 1209 <http://snomed.info/id/299701004>
 1210 <http://snomed.info/id/116676008>
 1211 <http://snomed.info/id/72704001>
 1212 <http://snomed.info/id/363698007>
 1213 <http://snomed.info/id/702468001>

<pre>< 404684003 Clinical finding¹²¹⁴ : { [2..*] 363698007 finding site¹²¹⁵ = < 91723000 Anatomical structure¹²¹⁶ }</pre>	<p>-</p>	<pre>64572001 Disease¹²¹⁷ : { 116676008 Associated morphology¹²¹⁸ = 396351009 Congenital septal defect¹²¹⁹ , 363698007 Finding site¹²²⁰ = 25943004 Structure of atrioventricular node¹²²¹ , 363698007 Finding site¹²²² = 113262008 Thoracic aorta structure¹²²³ } { 116676008 Associated morphology¹²²⁴ = 90141005 Congenital hypertrophy¹²²⁵ , 363698007 Finding site¹²²⁶ = 244384009 Entire right ventricle¹²²⁷ }</pre>
<pre>< 373873005 Pharmaceutical / biologic product¹²²⁸ : [1..3] { [1..*] 127489000 Has active ingredient¹²²⁹ = < 105590001 Substance¹²³⁰ }</pre>	<pre>322236009 Paracetamol 500mg tablet¹²³¹</pre>	<pre>373873005 Pharmaceutical / biologic product¹²³² : { 127489000 Has active ingredient¹²³³ = 412031009 Paracetamol or derivative¹²³⁴ }</pre>
	<pre>404826002 Benzocaine + butamben + tetracaine hydrochloride¹²³⁵</pre>	<pre>373873005 Pharmaceutical / biologic product¹²³⁶ : { 127489000 Has active ingredient¹²³⁶ }</pre>

1214 <http://snomed.info/id/404684003>
1215 <http://snomed.info/id/363698007>
1216 <http://snomed.info/id/91723000>
1217 <http://snomed.info/id/64572001>
1218 <http://snomed.info/id/116676008>
1219 <http://snomed.info/id/396351009>
1220 <http://snomed.info/id/363698007>
1221 <http://snomed.info/id/25943004>
1222 <http://snomed.info/id/363698007>
1223 <http://snomed.info/id/113262008>
1224 <http://snomed.info/id/116676008>
1225 <http://snomed.info/id/90141005>
1226 <http://snomed.info/id/363698007>
1227 <http://snomed.info/id/244384009>
1228 <http://snomed.info/id/373873005>
1229 <http://snomed.info/id/127489000>
1230 <http://snomed.info/id/105590001>
1231 <http://snomed.info/id/322236009>
1232 <http://snomed.info/id/373873005>
1233 <http://snomed.info/id/127489000>
1234 <http://snomed.info/id/412031009>
1235 <http://snomed.info/id/404826002>
1236 <http://snomed.info/id/373873005>

		¹²³⁷ = 412031009 Paracetamol or derivative ¹²³⁸ }, { 127489000 Has active ingredient ¹²³⁹ = 387494007 Codeine ¹²⁴⁰ }
< 373873005 Pharmaceutical / biologic product ¹²⁴¹ : [0..1] { 127489000 Has active ingredient ¹²⁴² = < 105590001 Substance ¹²⁴³ }	111115279999999108 Inert tablet ¹²⁴⁴	373873005 Pharmaceutical / biologic product ¹²⁴⁵ : { 127489000 Has active ingredient ¹²⁴⁶ = 412031009 Paracetamol or derivative ¹²⁴⁷ }
	370166004 Aspirin 325mg tablet ¹²⁴⁸	373873005 Pharmaceutical / biologic product ¹²⁵³ : { 127489000 Has active ingredient ¹²⁵⁴ = 412031009 Paracetamol or derivative ¹²⁵⁵ }, { 127489000 Has active ingredient ¹²⁵⁶ = 387494007 Codeine ¹²⁵⁷ }
< 373873005 Pharmaceutical / biologic product ¹²⁴⁹ : [1..*] { 127489000 Has active ingredient ¹²⁵⁰ = < 105590001 Substance ¹²⁵¹ }	370166004 Aspirin 325mg tablet ¹²⁵²	

1237 <http://snomed.info/id/127489000>
1238 <http://snomed.info/id/412031009>
1239 <http://snomed.info/id/127489000>
1240 <http://snomed.info/id/387494007>
1241 <http://snomed.info/id/373873005>
1242 <http://snomed.info/id/127489000>
1243 <http://snomed.info/id/105590001>
1244 <http://snomed.org/fictid#111115279999999108>
1245 <http://snomed.info/id/373873005>
1246 <http://snomed.info/id/127489000>
1247 <http://snomed.info/id/412031009>
1248 <http://snomed.info/id/370166004>
1249 <http://snomed.info/id/373873005>
1250 <http://snomed.info/id/127489000>
1251 <http://snomed.info/id/105590001>
1252 <http://snomed.info/id/370166004>
1253 <http://snomed.info/id/373873005>
1254 <http://snomed.info/id/127489000>
1255 <http://snomed.info/id/412031009>
1256 <http://snomed.info/id/127489000>
1257 <http://snomed.info/id/387494007>

<pre>< 404684003 Clinical finding¹²⁵⁸ : [1..1] { 363698007 Finding site¹²⁵⁹ = < 91723000 Anatomical structure¹²⁶⁰ }</pre>	<pre>125596004 Injury of elbow¹²⁶¹</pre>	<pre>404684003 Clinical finding¹²⁶² : { 363698007 Finding site¹²⁶³ = 299701004 Bone of forearm¹²⁶⁴ }, { 363698007 Finding site¹²⁶⁵ = 62413002 Bone structure of radius¹²⁶⁶ }</pre>
<pre>< 404684003 Clinical finding¹²⁶⁷ : [0..0] { [2..*] 363698007 Finding site¹²⁶⁸ = < 91723000 Anatomical structure¹²⁶⁹ }</pre>	<pre>86299006 Tetralogy of Fallot¹²⁷⁰</pre>	<pre>404684003 Clinical finding¹²⁷¹ : 363698007 Finding site¹²⁷² = 39057004 Pulmonary valve structure¹²⁷³ , 116676008 Associated morphology¹²⁷⁴ = 415582006 Stenosis¹²⁷⁵</pre>

¹[\(see page 149\)](#) The SNOMED CT identifiers created with the '9999999' namespace are for example only, and should not be used in a production environment.

²[\(see page 151\)](#) As mentioned earlier, only non-redundant defining attributes are included in the cardinality count. Because `62413002|Bone structure of radius|` is a subtype of `299701004|Bone of forearm|`, the refinement `"363698007|Finding site|=299701004|Bone of forearm|` " is redundant.

- 1258 <http://snomed.info/id/404684003>
- 1259 <http://snomed.info/id/363698007>
- 1260 <http://snomed.info/id/91723000>
- 1261 <http://snomed.info/id/125596004>
- 1262 <http://snomed.info/id/404684003>
- 1263 <http://snomed.info/id/363698007>
- 1264 <http://snomed.info/id/299701004>
- 1265 <http://snomed.info/id/363698007>
- 1266 <http://snomed.info/id/62413002>
- 1267 <http://snomed.info/id/404684003>
- 1268 <http://snomed.info/id/363698007>
- 1269 <http://snomed.info/id/91723000>
- 1270 <http://snomed.info/id/86299006>
- 1271 <http://snomed.info/id/404684003>
- 1272 <http://snomed.info/id/363698007>
- 1273 <http://snomed.info/id/39057004>
- 1274 <http://snomed.info/id/116676008>
- 1275 <http://snomed.info/id/415582006>

8.4 A.4 Conjunction and Disjunction - Valid Expressions

Expression Constraint	Valid Expression [see page 0]	
	Precoordinated	Postcoordinated
< 19829001 Disorder of lung ¹²⁷⁶ AND < 301867009 Edema of trunk ¹²⁷⁷	233709006 Toxic pulmonary edema ¹²⁷⁸ 61233003 Silo-fillers' disease ¹²⁸⁴	233709006 Toxic pulmonary edema ¹²⁷⁹ : 116676008 Associated morphology ¹²⁸⁰ = 40829002 Acute edema ¹²⁸¹ , 363698007 Finding site ¹²⁸² = 278985004 Fissure of right lung ¹²⁸³
< 19829001 Disorder of lung ¹²⁸⁵ OR < 301867009 Edema of trunk ¹²⁸⁶	363358000 Malignant tumour of lung ¹²⁸⁷ 19242006 Pulmonary edema ¹²⁹¹	233709006 Toxic pulmonary edema ¹²⁸⁸ : 116676008 Associated morphology ¹²⁸⁹ = 40829002 Acute edema ¹²⁹⁰
< 19829001 Disorder of lung ¹²⁹² AND ^ 700043003 Example	100100011 9102 Pulmonary embolism with	

1276 <http://snomed.info/id/19829001>
 1277 <http://snomed.info/id/301867009>
 1278 <http://snomed.info/id/233709006>
 1279 <http://snomed.info/id/233709006>
 1280 <http://snomed.info/id/116676008>
 1281 <http://snomed.info/id/40829002>
 1282 <http://snomed.info/id/363698007>
 1283 <http://snomed.info/id/278985004>
 1284 <http://snomed.info/id/61233003>
 1285 <http://snomed.info/id/19829001>
 1286 <http://snomed.info/id/301867009>
 1287 <http://snomed.info/id/363358000>
 1288 <http://snomed.info/id/233709006>
 1289 <http://snomed.info/id/116676008>
 1290 <http://snomed.info/id/40829002>
 1291 <http://snomed.info/id/19242006>
 1292 <http://snomed.info/id/19829001>

<p>problem list concepts reference set¹²⁹³</p>	<p>pulmonary infarction¹²⁹⁴</p>	
<p>< 404684003 Clinical finding¹²⁹⁵ : 363698007 Finding site¹²⁹⁶ = << 39057004 Pulmonary valve structure¹²⁹⁷ AND 116676008 Associated morphology¹²⁹⁸ = << 415582006 Stenosis¹²⁹⁹</p>	<p>91442002 Rheumatic pulmonary valve stenosis¹³⁰⁰</p> <p>86299006 Tetralogy of Fallot¹³⁰⁶</p>	<p>56786000 Pulmonic valve stenosis¹³⁰¹ : 363698007 Finding site¹³⁰² = 90318009 Structure of anulus fibrosus of pulmonary artery¹³⁰³ , 116676008 Associated morphology¹³⁰⁴ = 88015002 Partial stenosis¹³⁰⁵</p>

1293 <http://snomed.info/id/700043003>
 1294 <http://snomed.info/id/1001000119102>
 1295 <http://snomed.info/id/404684003>
 1296 <http://snomed.info/id/363698007>
 1297 <http://snomed.info/id/39057004>
 1298 <http://snomed.info/id/116676008>
 1299 <http://snomed.info/id/415582006>
 1300 <http://snomed.info/id/91442002>
 1301 <http://snomed.info/id/56786000>
 1302 <http://snomed.info/id/363698007>
 1303 <http://snomed.info/id/90318009>
 1304 <http://snomed.info/id/116676008>
 1305 <http://snomed.info/id/88015002>
 1306 <http://snomed.info/id/86299006>

<p>< 404684003 Clinical finding¹³⁰⁷ : 116676008 Associated morphology¹³⁰⁸ = << 55641003 Infarct¹³⁰⁹ OR 42752001 Due to¹³¹⁰ = << 22298006 Myocardial infarction¹³¹¹</p>	<p>45456005 Renal infarct¹³¹²</p> <hr/> <p>703326006 Mitral regurgitati on due to acute myocardia l infarction ¹³¹⁶</p>	<p>95281009 Sudden cardiac death¹³¹³ : 42752001 Due to¹³¹⁴ = 22298006 Myocardial infarction¹³¹⁵</p>
--	---	--

1307 <http://snomed.info/id/404684003>

1308 <http://snomed.info/id/116676008>

1309 <http://snomed.info/id/55641003>

1310 <http://snomed.info/id/42752001>

1311 <http://snomed.info/id/22298006>

1312 <http://snomed.info/id/45456005>

1313 <http://snomed.info/id/95281009>

1314 <http://snomed.info/id/42752001>

1315 <http://snomed.info/id/22298006>

1316 <http://snomed.info/id/703326006>

<p>< 404684003 Clinical finding¹³¹⁷ : { 363698007 Finding site¹³¹⁸ =<< 39057004 Pulmonary valve structure¹³¹⁹ , 116676008 Associated morphology¹³²⁰ =<< 415582006 Stenosis¹³²¹ } OR { 363698007 Finding site¹³²² = << 53085002 Right ventricular structure¹³²³ , 116676008 Associated morphology¹³²⁴ =<< 56246009 Hypertrophy¹³²⁵ }</p>	<p>85971001 Rheumatic pulmonary valve stenosis with insufficien cy¹³²⁶</p> <hr/> <p>86299006 Tetralogy of Fallot¹³³²</p>	<p>56786000 Pulmonic valve stenosis¹³²⁷ : 363698007 Finding site¹³²⁸ = 90318009 Structure of anulus fibrosus of pulmonary artery¹³²⁹ , 116676008 Associated morphology¹³³⁰ = 88015002 Partial stenosis¹³³¹</p>
<p>^ 450990004 Adverse drug reactions reference set for GP/FP health</p>	<p>294811002 Corticotro phic hormone allergy¹³³⁷</p>	<p>-</p>

1317 <http://snomed.info/id/404684003>
1318 <http://snomed.info/id/363698007>
1319 <http://snomed.info/id/39057004>
1320 <http://snomed.info/id/116676008>
1321 <http://snomed.info/id/415582006>
1322 <http://snomed.info/id/363698007>
1323 <http://snomed.info/id/53085002>
1324 <http://snomed.info/id/116676008>
1325 <http://snomed.info/id/56246009>
1326 <http://snomed.info/id/85971001>
1327 <http://snomed.info/id/56786000>
1328 <http://snomed.info/id/363698007>
1329 <http://snomed.info/id/90318009>
1330 <http://snomed.info/id/116676008>
1331 <http://snomed.info/id/88015002>
1332 <http://snomed.info/id/86299006>
1337 <http://snomed.info/id/294811002>

<p>issue¹³³³ : 246075003 Causative agent¹³³⁴ = (< 373873005 Pharmaceutical / biologic product¹³³⁵ OR < 105590001 Substance¹³³⁶)</p>	<p>293584003 Paracetam ol allergy¹³³⁸ 293585002 Salicylate allergy¹³³⁹</p>	
<p>< 404684003 Clinical finding¹³⁴⁰ : 116676008 Associated morphology¹³⁴¹ = (<< 56208002 Ulcer¹³⁴² AND << 50960005 Hemorrhage¹³⁴³)</p>	<p>12847006 Acute duodenal ulcer with hemorrhag e¹³⁴⁴</p>	<p>64572001 Disease¹³⁴⁵ : { 116676008 Associated morphology¹³⁴⁶ = 55075001 Bleeding ulcer¹³⁴⁷ , 363698007 Finding site¹³⁴⁸ = 14374004 Structure of lymphatic vessel of oesophagus¹³⁴⁹ }</p>

[\[see page 155\]](#) Where necessary, these examples make some assumptions about the membership of the example reference sets.

8.5 A.5 Exclusion and Not Equals - Valid Expressions

Expression Constraint	Valid Expression [see page 0]	
	Precoordinated	Postcoordinated

- 1333 <http://snomed.info/id/450990004>
- 1334 <http://snomed.info/id/246075003>
- 1335 <http://snomed.info/id/373873005>
- 1336 <http://snomed.info/id/105590001>
- 1338 <http://snomed.info/id/293584003>
- 1339 <http://snomed.info/id/293585002>
- 1340 <http://snomed.info/id/404684003>
- 1341 <http://snomed.info/id/116676008>
- 1342 <http://snomed.info/id/56208002>
- 1343 <http://snomed.info/id/50960005>
- 1344 <http://snomed.info/id/12847006>
- 1345 <http://snomed.info/id/64572001>
- 1346 <http://snomed.info/id/116676008>
- 1347 <http://snomed.info/id/55075001>
- 1348 <http://snomed.info/id/363698007>
- 1349 <http://snomed.info/id/14374004>

<pre><< 19829001 Disorder of lung¹³⁵⁰ MINUS << 301867009 Edema of trunk¹³⁵¹</pre>	<pre>372146004 Acute chest syndrome¹³⁵² 413839001 Chronic lung disease¹³⁵⁸</pre>	<pre>27819004 Pulmonary ossification¹³⁵³ : { 116676008 Associated morphology¹³⁵⁴ = 18115005 Pathologic calcification¹³⁵⁵ , 363698007 Finding site¹³⁵⁶ = 31094006 Structure of lobe of lung¹³⁵⁷ }</pre>
<pre><< 19829001 Disorder of lung¹³⁵⁹ MINUS ^ 700043003 Example problem list concepts reference set¹³⁶⁰</pre>	<pre>233613009 Fungal pneumonia¹³⁶¹</pre>	<pre>27819004 Pulmonary ossification¹³⁶² : { 116676008 Associated morphology¹³⁶³ = 18115005 Pathologic calcification¹³⁶⁴ , 363698007 Finding site¹³⁶⁵ = 31094006 Structure of lobe of lung¹³⁶⁶ }</pre>
<pre>< 404684003 Clinical finding¹³⁶⁷ : 116676008 Associated morphology¹³⁶⁸ = ((<< 56208002 Ulcer¹³⁶⁹ AND << 50960005 Hemorrhage¹³⁷⁰) MINUS << 26036001 Obstruction¹³⁷¹)</pre>	<pre>15902003 Gastric ulcer with hemorrhage¹³⁷²</pre>	<pre>64572001 Disease¹³⁷³ : { 116676008 Associated morphology¹³⁷⁴ = 55075001 Bleeding ulcer¹³⁷⁵ , 363698007 Finding site¹³⁷⁶ = 14374004 Structure of lymphatic vessel of esophagus¹³⁷⁷ }</pre>

1350 <http://snomed.info/id/19829001>
1351 <http://snomed.info/id/301867009>
1352 <http://snomed.info/id/372146004>
1353 <http://snomed.info/id/27819004>
1354 <http://snomed.info/id/116676008>
1355 <http://snomed.info/id/18115005>
1356 <http://snomed.info/id/363698007>
1357 <http://snomed.info/id/31094006>
1358 <http://snomed.info/id/413839001>
1359 <http://snomed.info/id/19829001>
1360 <http://snomed.info/id/700043003>
1361 <http://snomed.info/id/233613009>
1362 <http://snomed.info/id/27819004>
1363 <http://snomed.info/id/116676008>
1364 <http://snomed.info/id/18115005>
1365 <http://snomed.info/id/363698007>
1366 <http://snomed.info/id/31094006>
1367 <http://snomed.info/id/404684003>
1368 <http://snomed.info/id/116676008>
1369 <http://snomed.info/id/56208002>
1370 <http://snomed.info/id/50960005>
1371 <http://snomed.info/id/26036001>
1372 <http://snomed.info/id/15902003>
1373 <http://snomed.info/id/64572001>
1374 <http://snomed.info/id/116676008>
1375 <http://snomed.info/id/55075001>
1376 <http://snomed.info/id/363698007>
1377 <http://snomed.info/id/14374004>

<p>< 404684003 Clinical finding¹³⁷⁸ : 116676008 Associated morphology¹³⁷⁹ != << 26036001 Obstruction¹³⁸⁰</p>	<p>233613009 Fungal pneumonia¹³⁸¹</p> <hr/> <p>46708007 Acute gastric ulcer with hemorrhage AND obstruction¹³⁹¹</p>	<p>64572001 Disease¹³⁸² : { 116676008 Associated morphology¹³⁸³ = 26036001 Obstruction¹³⁸⁴ , 363698007 Finding site¹³⁸⁵ = 422897007 Vascular structure of stomach¹³⁸⁶ } { 116676008 Associated morphology¹³⁸⁷ = 45771005 Acute bleeding ulcer¹³⁸⁸ , 363698007 Finding site¹³⁸⁹ = 422897007 Vascular structure of stomach¹³⁹⁰ }</p>
<p>< 404684003 Clinical finding¹³⁹² : [0..0] 116676008 Associated morphology¹³⁹³ = << 26036001 Obstruction¹³⁹⁴</p>	<p>233613009 Fungal pneumonia¹³⁹⁵</p> <hr/> <p>15902003 Gastric ulcer with hemorrhage¹⁴⁰¹</p>	<p>64572001 Disease¹³⁹⁶ : { 116676008 Associated morphology¹³⁹⁷ = 55075001 Bleeding ulcer¹³⁹⁸ , 363698007 Finding site¹³⁹⁹ = 14374004 Structure of lymphatic vessel of oesophagus¹⁴⁰⁰ }</p>

1378 <http://snomed.info/id/404684003>
 1379 <http://snomed.info/id/116676008>
 1380 <http://snomed.info/id/26036001>
 1381 <http://snomed.info/id/233613009>
 1382 <http://snomed.info/id/64572001>
 1383 <http://snomed.info/id/116676008>
 1384 <http://snomed.info/id/26036001>
 1385 <http://snomed.info/id/363698007>
 1386 <http://snomed.info/id/422897007>
 1387 <http://snomed.info/id/116676008>
 1388 <http://snomed.info/id/45771005>
 1389 <http://snomed.info/id/363698007>
 1390 <http://snomed.info/id/422897007>
 1391 <http://snomed.info/id/46708007>
 1392 <http://snomed.info/id/404684003>
 1393 <http://snomed.info/id/116676008>
 1394 <http://snomed.info/id/26036001>
 1395 <http://snomed.info/id/233613009>
 1396 <http://snomed.info/id/64572001>
 1397 <http://snomed.info/id/116676008>
 1398 <http://snomed.info/id/55075001>
 1399 <http://snomed.info/id/363698007>
 1400 <http://snomed.info/id/14374004>
 1401 <http://snomed.info/id/15902003>

<pre>< 404684003 Clinical finding¹⁴⁰² : [0..0] 116676008 Associated morphology¹⁴⁰³ != << 26036001 Obstruction¹⁴⁰⁴</pre>	<pre>244815007 Pyloric obstruction¹⁴⁰⁵ 84906002 Local cyanosis¹⁴¹¹</pre>	<pre>64572001 Disease¹⁴⁰⁶ : { 116676008 Associated morphology¹⁴⁰⁷ = 26036001 Obstruction¹⁴⁰⁸ , 363698007 Finding site¹⁴⁰⁹ = 314600001 Cholechoenterostomy stoma¹⁴¹⁰ }</pre>
<pre>< 404684003 Clinical finding¹⁴¹² : [0..0] 116676008 Associated morphology¹⁴¹³ != << 26036001 Obstruction¹⁴¹⁴ AND [1..*] 116676008 Associated morphology¹⁴¹⁵ = << 26036001 Obstruction¹⁴¹⁶</pre>	<pre>244815007 Pyloric obstruction¹⁴¹⁷</pre>	<pre>64572001 Disease¹⁴¹⁸ : { 116676008 Associated morphology¹⁴¹⁹ = 26036001 Obstruction¹⁴²⁰ , 363698007 Finding site¹⁴²¹ = 314600001 Cholechoenterostomy stoma¹⁴²² }</pre>

[\[see page 159\]](#) Where necessary, these examples make some assumptions about the membership of the example reference sets.

- 1402 <http://snomed.info/id/404684003>
- 1403 <http://snomed.info/id/116676008>
- 1404 <http://snomed.info/id/26036001>
- 1405 <http://snomed.info/id/244815007>
- 1406 <http://snomed.info/id/64572001>
- 1407 <http://snomed.info/id/116676008>
- 1408 <http://snomed.info/id/26036001>
- 1409 <http://snomed.info/id/363698007>
- 1410 <http://snomed.info/id/314600001>
- 1411 <http://snomed.info/id/84906002>
- 1412 <http://snomed.info/id/404684003>
- 1413 <http://snomed.info/id/116676008>
- 1414 <http://snomed.info/id/26036001>
- 1415 <http://snomed.info/id/116676008>
- 1416 <http://snomed.info/id/26036001>
- 1417 <http://snomed.info/id/244815007>
- 1418 <http://snomed.info/id/64572001>
- 1419 <http://snomed.info/id/116676008>
- 1420 <http://snomed.info/id/26036001>
- 1421 <http://snomed.info/id/363698007>
- 1422 <http://snomed.info/id/314600001>

8.6 A.6 Nested Expression Constraints - Valid Expressions

Expression Constraint	Valid Expression <small>[see page 0]</small>	
	Precoordinated	Postcoordinated
$\ll (\wedge 700043003 \mid \text{Example problem list concepts reference set}^{1423})$	394659003 \mid Acute coronary syndrome ¹⁴²⁴	194828000 \mid Angina ¹⁴²⁵ : 255234002 \mid After ¹⁴²⁶ = 22298006 \mid Myocardial infarction ¹⁴²⁷
	194828000 \mid Angina ¹⁴²⁸	
	371807002 \mid Atypical angina ¹⁴²⁹	
$\wedge (< 450973005 \mid \text{GP/FP health issue reference set}^{1430})$	140004 \mid Chronic pharyngitis ¹⁴³¹	-
	297009 \mid Acute myringitis ¹⁴³²	
$(< 404684003 \mid \text{Clinical finding}^{1433} : 363698007 \mid \text{Finding site}^{1434} = \ll 39057004 \mid \text{Pulmonary valve structure}^{1435}) \text{ AND } \wedge 700043003 \mid \text{Example problem list concepts reference set}^{1436}$	204351007 \mid Fallot's trilogyl ¹⁴³⁷	-
	457652006 \mid Calcification of pulmonary valve ¹⁴³⁸	

1423 <http://snomed.info/id/700043003>

1424 <http://snomed.info/id/394659003>

1425 <http://snomed.info/id/194828000>

1426 <http://snomed.info/id/255234002>

1427 <http://snomed.info/id/22298006>

1428 <http://snomed.info/id/194828000>

1429 <http://snomed.info/id/371807002>

1430 <http://snomed.info/id/450973005>

1431 <http://snomed.info/id/140004>

1432 <http://snomed.info/id/297009>

1433 <http://snomed.info/id/404684003>

1434 <http://snomed.info/id/363698007>

1435 <http://snomed.info/id/39057004>

1436 <http://snomed.info/id/700043003>

1437 <http://snomed.info/id/204351007>

1438 <http://snomed.info/id/457652006>

<p>(< 404684003 Clinical finding¹⁴³⁹ : 363698007 Finding site¹⁴⁴⁰ = << 39057004 Pulmonary valve structure¹⁴⁴¹) AND (< 64572001 Disease¹⁴⁴² : 116676008 Associated morphology¹⁴⁴³ = << 415582006 Stenosis¹⁴⁴⁴)</p>	<p>204351007 Fallot&#39;s trilogy¹⁴⁴⁵ 56786000 Pulmonic valve stenosis¹⁴⁵¹</p>	<p>19036004 Rheumatic heart valve stenosis¹⁴⁴⁶ : { 363698007 Finding site¹⁴⁴⁷ = 39057004 Pulmonary valve structure¹⁴⁴⁸ , 116676008 Associated morphology¹⁴⁴⁹ = 415582006 Stenosis¹⁴⁵⁰ }</p>
<p>(<< 17636008 Specimen collection¹⁴⁵² : 424226004 Using device¹⁴⁵³ = << 19923001 Catheter¹⁴⁵⁴) . 363701004 Direct substance¹⁴⁵⁵</p>	<p>78014005 Urine¹⁴⁵⁶ 87612001 Blood¹⁴⁵⁷</p>	<p>-</p>
<p>(<< 404684003 Clinical finding (finding)¹⁴⁵⁸ OR << 272379006 Event (event)¹⁴⁵⁹): 255234002 After¹⁴⁶⁰ = << 71388002 Procedure (procedure)¹⁴⁶¹</p>	<p>235948002 Postoperative acute pancreatitis¹⁴⁶²</p>	<p>64572001 Disease¹⁴⁶³ : { 370135005 Pathological process¹⁴⁶⁴ = 441862004 Infectious process¹⁴⁶⁵ , 255234002 After¹⁴⁶⁶ = 387713003 Surgical procedure</p>

1439 <http://snomed.info/id/404684003>
1440 <http://snomed.info/id/363698007>
1441 <http://snomed.info/id/39057004>
1442 <http://snomed.info/id/64572001>
1443 <http://snomed.info/id/116676008>
1444 <http://snomed.info/id/415582006>
1445 <http://snomed.info/id/204351007>
1446 <http://snomed.info/id/19036004>
1447 <http://snomed.info/id/363698007>
1448 <http://snomed.info/id/39057004>
1449 <http://snomed.info/id/116676008>
1450 <http://snomed.info/id/415582006>
1451 <http://snomed.info/id/56786000>
1452 <http://snomed.info/id/17636008>
1453 <http://snomed.info/id/424226004>
1454 <http://snomed.info/id/19923001>
1455 <http://snomed.info/id/363701004>
1456 <http://snomed.info/id/78014005>
1457 <http://snomed.info/id/87612001>
1458 <http://snomed.info/id/404684003>
1459 <http://snomed.info/id/272379006>
1460 <http://snomed.info/id/255234002>
1461 <http://snomed.info/id/71388002>
1462 <http://snomed.info/id/235948002>
1463 <http://snomed.info/id/64572001>
1464 <http://snomed.info/id/370135005>
1465 <http://snomed.info/id/441862004>
1466 <http://snomed.info/id/255234002>

	441795000 Infected seroma after surgical procedure ¹⁴⁷⁰	¹⁴⁶⁷ , 116676008 Associated morphology ¹⁴⁶⁸ = 112633009 Surgical would ¹⁴⁶⁹ }
<< 125605004 Fracture of bone ¹⁴⁷¹ : [0..0] ((<< 410662002 Concept model attribute ¹⁴⁷² MINUS 363698007 Finding site ¹⁴⁷³) MINUS 116676008 Associated morphology ¹⁴⁷⁴) = *	125605004 Fracture of bone ¹⁴⁷⁵ 439987009 Open fracture of bone ¹⁴⁸¹	64572001 Disease ¹⁴⁷⁶ : { 363698007 Finding site ¹⁴⁷⁷ = 71341001 Bone structure of femur ¹⁴⁷⁸ , 116676008 Associated morphology ¹⁴⁷⁹ = 20946005 Fracture, closed ¹⁴⁸⁰ }
< 404684003 Clinical finding ¹⁴⁸² : 47429007 Associated with ¹⁴⁸³ = (< 404684003 Clinical finding ¹⁴⁸⁴ : 116676008 Associated morphology ¹⁴⁸⁵ = << 55641003 Infarct ¹⁴⁸⁶)	71023004 Pericarditis secondary to acute myocardial infarction ¹⁴⁸⁷	3238004 Pericarditis (disorder) ¹⁴⁸⁸ : 47429007 Associated with ¹⁴⁸⁹ = 57054005 Acute myocardial infarction ¹⁴⁹⁰

[\[see page 163\]](#) Where necessary, these examples make some assumptions about the membership of the example reference sets.

- 1470 <http://snomed.info/id/441795000>
- 1467 <http://snomed.info/id/387713003>
- 1468 <http://snomed.info/id/116676008>
- 1469 <http://snomed.info/id/112633009>
- 1471 <http://snomed.info/id/125605004>
- 1472 <http://snomed.info/id/410662002>
- 1473 <http://snomed.info/id/363698007>
- 1474 <http://snomed.info/id/116676008>
- 1475 <http://snomed.info/id/125605004>
- 1476 <http://snomed.info/id/64572001>
- 1477 <http://snomed.info/id/363698007>
- 1478 <http://snomed.info/id/71341001>
- 1479 <http://snomed.info/id/116676008>
- 1480 <http://snomed.info/id/20946005>
- 1481 <http://snomed.info/id/439987009>
- 1482 <http://snomed.info/id/404684003>
- 1483 <http://snomed.info/id/47429007>
- 1484 <http://snomed.info/id/404684003>
- 1485 <http://snomed.info/id/116676008>
- 1486 <http://snomed.info/id/55641003>
- 1487 <http://snomed.info/id/71023004>
- 1488 <http://snomed.info/id/3238004>
- 1489 <http://snomed.info/id/47429007>
- 1490 <http://snomed.info/id/57054005>

9 Appendix B – Examples Of Invalid Expressions

This appendix provides examples of expressions (both precoordinated and postcoordinated) which **do not** satisfy the given expression constraints from [Chapter 6](#) (see page 64). This list of examples is not intended to be exhaustive, but rather to provide a useful sample to help clarify the meaning of these constraint. Please refer to the [SNOMED CT Languages Github repository](#)¹⁴⁹¹ for a set of text files containing each of these examples.

- [B.1 Simple Expression Constraints - Invalid Expressions](#) (see page 166)
- [B.2 Refinements - Invalid Expressions](#) (see page 168)
- [B.3 Cardinality - Invalid Expressions](#) (see page 177)
- [B.4 Conjunction and Disjunction - Invalid Expressions](#) (see page 183)
- [B.5 Exclusion and Not Equals - Invalid Expressions](#) (see page 186)
- [B.6 Nested Expression Constraints - Invalid Expressions](#) (see page 190)

9.1 B.1 Simple Expression Constraints - Invalid Expressions

Expression Constraint	INVALID Expression ¹ (see page 0)	
	Precoordinated	Postcoordinated
404684003 Clinical finding ¹⁴⁹²	56265001 Heart disease ¹⁴⁹³ 71388002 Procedure ¹⁴⁹⁷	404684003 Clinical finding ¹⁴⁹⁴ : 363698007 Finding site ¹⁴⁹⁵ = 80891009 Heart structure ¹⁴⁹⁶
< 404684003 Clinical finding ¹⁴⁹⁸	404684003 Clinical finding ¹⁴⁹⁹ 71388002 Procedure ¹⁵⁰³	71388002 Procedure ¹⁵⁰⁰ : 405813007 Procedure site - Direct ¹⁵⁰¹ = 80891009 Heart structure ¹⁵⁰²
<< 73211009 Diabetes mellitus ¹⁵⁰⁴	71388002 Procedure ¹⁵⁰⁵	404684003 Clinical finding ¹⁵⁰⁶ : 363698007 Finding site ¹⁵⁰⁷ =

¹⁴⁹¹ <https://github.com/IHTSDO/SNOMEDCT-Languages>

¹⁴⁹² <http://snomed.info/id/404684003>

¹⁴⁹³ <http://snomed.info/id/56265001>

¹⁴⁹⁴ <http://snomed.info/id/404684003>

¹⁴⁹⁵ <http://snomed.info/id/363698007>

¹⁴⁹⁶ <http://snomed.info/id/80891009>

¹⁴⁹⁷ <http://snomed.info/id/71388002>

¹⁴⁹⁸ <http://snomed.info/id/404684003>

¹⁴⁹⁹ <http://snomed.info/id/404684003>

¹⁵⁰⁰ <http://snomed.info/id/71388002>

¹⁵⁰¹ <http://snomed.info/id/405813007>

¹⁵⁰² <http://snomed.info/id/80891009>

¹⁵⁰³ <http://snomed.info/id/71388002>

¹⁵⁰⁴ <http://snomed.info/id/73211009>

¹⁵⁰⁵ <http://snomed.info/id/71388002>

¹⁵⁰⁶ <http://snomed.info/id/404684003>

¹⁵⁰⁷ <http://snomed.info/id/363698007>

	362969004 Disorder of endocrine system ¹⁵⁰⁹	113331007 Structure of endocrine system ¹⁵⁰⁸
<! 404684003 Clinical finding ¹⁵¹⁰	404684003 Clinical finding ¹⁵¹¹	404684003 Clinical finding ¹⁵¹² : 116676008 Associated morphology ¹⁵¹³ = 79654002 Edema ¹⁵¹⁴ , 363698007 Finding site ¹⁵¹⁵ = 80891009 Heart structure ¹⁵¹⁶ 2 (see page 0)
	233709006 Toxic pulmonary edema ¹⁵¹⁷	
> 40541001 Acute pulmonary edema ¹⁵¹⁸	40541001 Acute pulmonary edema ¹⁵¹⁹	40541001 Acute pulmonary edema ¹⁵²⁰ : 246112005 Severity ¹⁵²¹ = 24484000 Severe ¹⁵²²
	233709006 Toxic pulmonary edema ¹⁵²³	
	304527002 Acute asthma ¹⁵²⁴	
>> 40541001 Acute pulmonary edema ¹⁵²⁵	233709006 Toxic pulmonary edema ¹⁵²⁶	40541001 Acute pulmonary edema ¹⁵²⁷ : 246112005 Severity ¹⁵²⁸ = 24484000 Severe ¹⁵²⁹
	304527002 Acute asthma ¹⁵³⁰	

1509 <http://snomed.info/id/362969004>
 1508 <http://snomed.info/id/113331007>
 1510 <http://snomed.info/id/404684003>
 1511 <http://snomed.info/id/404684003>
 1512 <http://snomed.info/id/404684003>
 1513 <http://snomed.info/id/116676008>
 1514 <http://snomed.info/id/79654002>
 1515 <http://snomed.info/id/363698007>
 1516 <http://snomed.info/id/80891009>
 1517 <http://snomed.info/id/233709006>
 1518 <http://snomed.info/id/40541001>
 1519 <http://snomed.info/id/40541001>
 1520 <http://snomed.info/id/40541001>
 1521 <http://snomed.info/id/246112005>
 1522 <http://snomed.info/id/24484000>
 1523 <http://snomed.info/id/233709006>
 1524 <http://snomed.info/id/304527002>
 1525 <http://snomed.info/id/40541001>
 1526 <http://snomed.info/id/233709006>
 1527 <http://snomed.info/id/40541001>
 1528 <http://snomed.info/id/246112005>
 1529 <http://snomed.info/id/24484000>
 1530 <http://snomed.info/id/304527002>

>! 40541001 Acute pulmonary edema 1531	404684003 Clinical finding 1532	64572001 Disease ¹⁵³³ : 263502005 Clinical course ¹⁵³⁴ = 424124008 Sudden onset AND/ OR short duration ¹⁵³⁵ 3(see page 0)
	267038008 Edema ¹⁵³⁶	
^ 700043003 Example problem list concepts reference set ¹⁵³⁷	6143009 Diabetic education 1538	71388002 Procedure ¹⁵³⁹ : 405813007 Procedure site - Direct ¹⁵⁴⁰ = 80891009 Heart structure ¹⁵⁴¹
	75367002 Blood pressure ¹⁵⁴²	
*	-	-
	-	-
	-	-

[1\(see page 166\)](#) Where necessary, these examples make some assumptions about the membership of the example reference sets.

[2\(see page 167\)](#) Please note that this makes the assumption that the given expression constraint is executed against a finite set of expressions that has been pre-classified (e.g. in an expression repository), and that after classification there is at least one intermediate expression between this expression and 404684003 | Clinical finding|¹⁵⁴³.

[3\(see page 168\)](#) Please note that this makes the assumption that the given expression constraint is executed against a finite set of expressions that has been pre-classified (e.g. in an expression repository), and that after classification there is at least one intermediate expression between 40541001 | Acute pulmonary edema|¹⁵⁴⁴ and this expression.

9.2 B.2 Refinements - Invalid Expressions

Expression Constraint	INVALID Expression 1(see page 0) 2(see page 0)
------------------------------	---

1531 <http://snomed.info/id/40541001>
 1532 <http://snomed.info/id/404684003>
 1533 <http://snomed.info/id/64572001>
 1534 <http://snomed.info/id/263502005>
 1535 <http://snomed.info/id/424124008>
 1536 <http://snomed.info/id/267038008>
 1537 <http://snomed.info/id/700043003>
 1538 <http://snomed.info/id/6143009>
 1539 <http://snomed.info/id/71388002>
 1540 <http://snomed.info/id/405813007>
 1541 <http://snomed.info/id/80891009>
 1542 <http://snomed.info/id/75367002>
 1543 <http://snomed.info/id/404684003>
 1544 <http://snomed.info/id/40541001>

	Precoordinated	Postcoordinated
< 19829001 Disorder of lung ¹⁵⁴⁵ : 116676008 Associated morphology ¹⁵⁴⁶ = 79654002 Edema ¹⁵⁴⁷	19829001 Disorder of lung ¹⁵⁴⁸	19829001 Disorder of lung ¹⁵⁴⁹ : 116676008 Associated morphology ¹⁵⁵⁰ = 44132006 Abscess ¹⁵⁵¹
	73452002 Abscess of lung ¹⁵⁵²	19829001 Disorder of lung ¹⁵⁵³ : 116676008 Associated morphology ¹⁵⁵⁴ = 40829002 Acute edema ¹⁵⁵⁵
	233711002 Oxygen-induced pulmonary edema ¹⁵⁵⁶	
< 19829001 Disorder of lung ¹⁵⁵⁷ : 116676008 Associated morphology ¹⁵⁵⁸ = << 79654002 Edema ¹⁵⁵⁹	19829001 Disorder of lung ¹⁵⁶⁰	6141006 Retinal edema ¹⁵⁶¹ : 116676008 Associated morphology ¹⁵⁶² = 103619005 Inflammatory edema ¹⁵⁶³
	73452002 Abscess of lung ¹⁵⁶⁴	19829001 Disorder of lung ¹⁵⁶⁵ : 116676008 Associated morphology ¹⁵⁶⁶ = 44132006 Abscess ¹⁵⁶⁷
	6141006 Retinal edema ¹⁵⁶⁸	

1545 <http://snomed.info/id/19829001>
 1546 <http://snomed.info/id/116676008>
 1547 <http://snomed.info/id/79654002>
 1548 <http://snomed.info/id/19829001>
 1549 <http://snomed.info/id/19829001>
 1550 <http://snomed.info/id/116676008>
 1551 <http://snomed.info/id/44132006>
 1552 <http://snomed.info/id/73452002>
 1553 <http://snomed.info/id/19829001>
 1554 <http://snomed.info/id/116676008>
 1555 <http://snomed.info/id/40829002>
 1556 <http://snomed.info/id/233711002>
 1557 <http://snomed.info/id/19829001>
 1558 <http://snomed.info/id/116676008>
 1559 <http://snomed.info/id/79654002>
 1560 <http://snomed.info/id/19829001>
 1561 <http://snomed.info/id/6141006>
 1562 <http://snomed.info/id/116676008>
 1563 <http://snomed.info/id/103619005>
 1564 <http://snomed.info/id/73452002>
 1565 <http://snomed.info/id/19829001>
 1566 <http://snomed.info/id/116676008>
 1567 <http://snomed.info/id/44132006>
 1568 <http://snomed.info/id/6141006>

<p>< 404684003 Clinical finding¹⁵⁶⁹ : 363698007 Finding site¹⁵⁷⁰ = << 39057004 Pulmonary valve structure¹⁵⁷¹ , 116676008 Associated morphology¹⁵⁷² = << 415582006 Stenosis¹⁵⁷³</p>	<p>404684003 Clinical finding¹⁵⁷⁴</p> <p>448643005 Abnormality of pulmonary valve¹⁵⁷⁸</p> <p>431238002 Abscess of pulmonary valve¹⁵⁸⁴</p>	<p>448643005 Abnormality of pulmonary valve¹⁵⁷⁵ : 116676008 Associated morphology¹⁵⁷⁶ = 44132006 Abscess¹⁵⁷⁷</p> <p>404684003 Clinical finding¹⁵⁷⁹ : 363698007 Finding site¹⁵⁸⁰ = 61853006 Spinal canal structure¹⁵⁸¹ , 116676008 Associated morphology¹⁵⁸² = 415582006 Stenosis¹⁵⁸³</p>
<p>* : 246075003 Causative agent¹⁵⁸⁵ = 387517004 Paracetamol¹⁵⁸⁶</p>	<p>46093004 Paracetamol measurement¹⁵⁸⁷</p>	<p>404684003 Clinical finding¹⁵⁸⁸ : 246075003 Causative agent¹⁵⁸⁹ = 372687004 Amoxicillin¹⁵⁹⁰</p>

1569 <http://snomed.info/id/404684003>
 1570 <http://snomed.info/id/363698007>
 1571 <http://snomed.info/id/39057004>
 1572 <http://snomed.info/id/116676008>
 1573 <http://snomed.info/id/415582006>
 1574 <http://snomed.info/id/404684003>
 1575 <http://snomed.info/id/448643005>
 1576 <http://snomed.info/id/116676008>
 1577 <http://snomed.info/id/44132006>
 1578 <http://snomed.info/id/448643005>
 1579 <http://snomed.info/id/404684003>
 1580 <http://snomed.info/id/363698007>
 1581 <http://snomed.info/id/61853006>
 1582 <http://snomed.info/id/116676008>
 1583 <http://snomed.info/id/415582006>
 1584 <http://snomed.info/id/431238002>
 1585 <http://snomed.info/id/246075003>
 1586 <http://snomed.info/id/387517004>
 1587 <http://snomed.info/id/46093004>
 1588 <http://snomed.info/id/404684003>
 1589 <http://snomed.info/id/246075003>
 1590 <http://snomed.info/id/372687004>

<pre> < 404684003 Clinical finding¹⁵⁹¹ : { 363698007 Finding site¹⁵⁹² = << 39057004 Pulmonary valve structure¹⁵⁹³ , 116676008 Associated morphology¹⁵⁹⁴ = << 415582006 Stenosis¹⁵⁹⁵ }, { 363698007 Finding site¹⁵⁹⁶ = << 53085002 Right ventricular structure¹⁵⁹⁷ , 116676008 Associated morphology¹⁵⁹⁸ = << 56246009 Hypertrophy¹⁵⁹⁹ } </pre>	<pre> 404684003 Clinical finding¹⁶⁰⁰ } 56786000 Pulmonary valve stenosis¹⁶¹⁰ } </pre>	<pre> 404684003 Clinical finding¹⁶⁰¹ : { 363698007 Finding site¹⁶⁰² = 39057004 Pulmonary valve structure¹⁶⁰³ , 116676008 Associated morphology¹⁶⁰⁴ = 56246009 Hypertrophy¹⁶⁰⁵ }, { 363698007 Finding site¹⁶⁰⁶ = 53085002 Right ventricular structure¹⁶⁰⁷ , 116676008 Associated morphology¹⁶⁰⁸ = 415582006 Stenosis¹⁶⁰⁹ } </pre>
--	---	---

1591 <http://snomed.info/id/404684003>
1592 <http://snomed.info/id/363698007>
1593 <http://snomed.info/id/39057004>
1594 <http://snomed.info/id/116676008>
1595 <http://snomed.info/id/415582006>
1596 <http://snomed.info/id/363698007>
1597 <http://snomed.info/id/53085002>
1598 <http://snomed.info/id/116676008>
1599 <http://snomed.info/id/56246009>
1600 <http://snomed.info/id/404684003>
1601 <http://snomed.info/id/404684003>
1602 <http://snomed.info/id/363698007>
1603 <http://snomed.info/id/39057004>
1604 <http://snomed.info/id/116676008>
1605 <http://snomed.info/id/56246009>
1606 <http://snomed.info/id/363698007>
1607 <http://snomed.info/id/53085002>
1608 <http://snomed.info/id/116676008>
1609 <http://snomed.info/id/415582006>
1610 <http://snomed.info/id/56786000>

<p><< 404684003 Clinical finding¹⁶¹¹ : << 47429007 Associated with¹⁶¹² = << 267038008 Edema¹⁶¹³</p>	<p>404684003 Clinical finding¹⁶¹⁴</p>	<p>95356008 Mucosal ulcer¹⁶¹⁵ : 42752001 Due to¹⁶¹⁶ = 59901004 Cheek biting¹⁶¹⁷</p>
---	--	--

1611 <http://snomed.info/id/404684003>

1612 <http://snomed.info/id/47429007>

1613 <http://snomed.info/id/267038008>

1614 <http://snomed.info/id/404684003>

1615 <http://snomed.info/id/95356008>

1616 <http://snomed.info/id/42752001>

1617 <http://snomed.info/id/59901004>

<pre> < 27658006 Amoxicillin 1618 : 411116001 Has dose form 1619 =<< 385055001 Tablet dose form 1620 , { 17999999910 0 Has basis of strength 1621 = (21999999910 2 Amoxicillin only 1622 : 18999999910 3 Has strength magnitude 1623 >= #200, 19999999910 1 Has strength unit 1624 = 258684004 mg 1625 }} </pre>	<pre> 2699999991 00 Amoxicillin capsule 1626 374233002 Amoxicillin trihydrate 125 mg chewable tablet 1635 </pre>	<pre> 27658006 Amoxicillin 1627 : 411116001 Has dose form 1628 = 421026006 Oral tablet 1629 , { 179999999100 Has basis of strength 1630 = (219999999102 Amoxicillin only 1631 : 189999999103 Has strength magnitude 1632 = 175, 199999999101 Has strength unit 1633 = 258684004 mg 1634)} </pre>
---	--	--

- 1618 <http://snomed.info/id/27658006>
- 1619 <http://snomed.info/id/411116001>
- 1620 <http://snomed.info/id/385055001>
- 1621 <http://snomed.org/fictid#179999999100>
- 1622 <http://snomed.org/fictid#219999999102>
- 1623 <http://snomed.org/fictid#189999999103>
- 1624 <http://snomed.org/fictid#199999999101>
- 1625 <http://snomed.info/id/258684004>
- 1626 <http://snomed.org/fictid#269999999100>
- 1627 <http://snomed.info/id/27658006>
- 1628 <http://snomed.info/id/411116001>
- 1629 <http://snomed.info/id/421026006>
- 1630 <http://snomed.org/fictid#179999999100>
- 1631 <http://snomed.org/fictid#219999999102>
- 1632 <http://snomed.org/fictid#189999999103>
- 1633 <http://snomed.org/fictid#199999999101>
- 1634 <http://snomed.info/id/258684004>
- 1635 <http://snomed.info/id/374233002>

<pre>< 27658006 Amoxicillin 1636 : 411116001 Has dose form ¹⁶³⁷ =<< 385055001 Tablet dose form ¹⁶³⁸, { 17999999910 0 Has basis of strength 1639 = (21999999910 2 Amoxicillin only ¹⁶⁴⁰ : 18999999910 3 Has strength magnitude 1641 >= #500, 18999999910 3 Has strength magnitude 1642 <= #800, 19999999910 1 Has strength unit 1643 = 258684004 mg ¹⁶⁴⁴)}</pre>	<pre>2699999991 00 Amoxicillin capsule ¹⁶⁴⁵</pre>	<pre>27658006 Amoxicillin ¹⁶⁴⁶ : 411116001 Has dose form ¹⁶⁴⁷ = 421026006 Oral tablet ¹⁶⁴⁸ , { 179999999100 Has basis of strength ¹⁶⁴⁹ = (219999999102 Amoxicillin only ¹⁶⁵⁰ : 189999999103 Has strength magnitude ¹⁶⁵¹ = #850, 199999999101 Has strength unit ¹⁶⁵² = 258684004 mg ¹⁶⁵³)}</pre>
---	--	---

- 1636 <http://snomed.info/id/27658006>
- 1637 <http://snomed.info/id/411116001>
- 1638 <http://snomed.info/id/385055001>
- 1639 <http://snomed.org/fictid#179999999100>
- 1640 <http://snomed.org/fictid#219999999102>
- 1641 <http://snomed.org/fictid#189999999103>
- 1642 <http://snomed.org/fictid#189999999103>
- 1643 <http://snomed.org/fictid#199999999101>
- 1644 <http://snomed.info/id/258684004>
- 1645 <http://snomed.org/fictid#269999999100>
- 1646 <http://snomed.info/id/27658006>
- 1647 <http://snomed.info/id/411116001>
- 1648 <http://snomed.info/id/421026006>
- 1649 <http://snomed.org/fictid#179999999100>
- 1650 <http://snomed.org/fictid#219999999102>
- 1651 <http://snomed.org/fictid#189999999103>
- 1652 <http://snomed.org/fictid#199999999101>
- 1653 <http://snomed.info/id/258684004>

	374647008 Amoxicillin 875 mg tablet ¹⁶⁵⁴	
< 373873005 Pharmaceutical / biologic product ¹⁶⁵⁵ : 209999999104 Has trade name ¹⁶⁵⁶ = "PANADOL"	373873005 Pharmaceutical / biologic product ¹⁶⁵⁷ 322236009 Paracetamol 500mg tablet ¹⁶⁶²	373873005 Pharmaceutical / biologic product ¹⁶⁵⁸ : { 127489000 Has active ingredient ¹⁶⁵⁹ = 412031009 Paracetamol or derivative ¹⁶⁶⁰ , 209999999104 Has trade name ¹⁶⁶¹ = "PANADEINE" }
< 91723000 Anatomical structure ¹⁶⁶³ : R 363698007 Finding site ¹⁶⁶⁴ = < 125605004 Fracture of bone ¹⁶⁶⁵	34080009 Malleus structure ¹⁶⁶⁶ 10200004 Liver structure ¹⁶⁷⁰	34080009 Malleus structure ¹⁶⁶⁷ : 272741003 Laterality ¹⁶⁶⁸ = 7771000 Left ¹⁶⁶⁹ 10200004 Liver structure ¹⁶⁷¹ : 272741003 Laterality ¹⁶⁷² = 24028007 Right ¹⁶⁷³

1654 <http://snomed.info/id/374647008>
 1655 <http://snomed.info/id/373873005>
 1656 <http://snomed.org/fictid#209999999104>
 1657 <http://snomed.info/id/373873005>
 1658 <http://snomed.info/id/373873005>
 1659 <http://snomed.info/id/127489000>
 1660 <http://snomed.info/id/412031009>
 1661 <http://snomed.org/fictid#209999999104>
 1662 <http://snomed.info/id/322236009>
 1663 <http://snomed.info/id/91723000>
 1664 <http://snomed.info/id/363698007>
 1665 <http://snomed.info/id/125605004>
 1666 <http://snomed.info/id/34080009>
 1667 <http://snomed.info/id/34080009>
 1668 <http://snomed.info/id/272741003>
 1669 <http://snomed.info/id/7771000>
 1670 <http://snomed.info/id/10200004>
 1671 <http://snomed.info/id/10200004>
 1672 <http://snomed.info/id/272741003>
 1673 <http://snomed.info/id/24028007>

< 125605004 Fracture of bone ¹⁶⁷⁴ . 363698007 Finding site ¹⁶⁷⁵	34080009 Malleus structure ¹⁶⁷⁶	34080009 Malleus structure ¹⁶⁷⁷ : 272741003 Laterality ¹⁶⁷⁸ = 7771000 Left ¹⁶⁷⁹
	10200004 Liver structure ¹⁶⁸⁰	10200004 Liver structure ¹⁶⁸¹ : 272741003 Laterality ¹⁶⁸² = 24028007 Right ¹⁶⁸³
< 105590001 Substance ¹⁶⁸⁴ : R << 127489000 Has active ingredient ¹⁶⁸⁵ = < 27658006 Product containing amoxicillin ¹⁶⁸⁶	105590001 Substance ¹⁶⁸⁷	373873005 Pharmaceutical / biologic product ¹⁶⁸⁸ : 127489000 Has active ingredient ¹⁶⁸⁹ = 372687004 Amoxicillin ¹⁶⁹⁰
	387517004 Paracetamo l ¹⁶⁹¹	
24999999910 1 TRIPHASIL tablet ¹⁶⁹² . 127489000 Has active ingredient ¹⁶⁹³	105590001 Substance ¹⁶⁹⁴	373873005 Pharmaceutical / biologic product ¹⁶⁹⁵ : 127489000 Has active ingredient ¹⁶⁹⁶ = 126109000 Levonorgestrel ¹⁶⁹⁷

1674 <http://snomed.info/id/125605004>
 1675 <http://snomed.info/id/363698007>
 1676 <http://snomed.info/id/34080009>
 1677 <http://snomed.info/id/34080009>
 1678 <http://snomed.info/id/272741003>
 1679 <http://snomed.info/id/7771000>
 1680 <http://snomed.info/id/10200004>
 1681 <http://snomed.info/id/10200004>
 1682 <http://snomed.info/id/272741003>
 1683 <http://snomed.info/id/24028007>
 1684 <http://snomed.info/id/105590001>
 1685 <http://snomed.info/id/127489000>
 1686 <http://snomed.info/id/27658006>
 1687 <http://snomed.info/id/105590001>
 1688 <http://snomed.info/id/373873005>
 1689 <http://snomed.info/id/127489000>
 1690 <http://snomed.info/id/372687004>
 1691 <http://snomed.info/id/387517004>
 1692 <http://snomed.org/fictid#249999999101>
 1693 <http://snomed.info/id/127489000>
 1694 <http://snomed.info/id/105590001>
 1695 <http://snomed.info/id/373873005>
 1696 <http://snomed.info/id/127489000>
 1697 <http://snomed.info/id/126109000>

	387517004 Paracetamol ¹⁶⁹⁸	
< 404684003 Clinical finding ¹⁶⁹⁹ : * = 79654002 Edema ¹⁷⁰⁰	263225007 Hip fracture ¹⁷⁰¹	404684003 Clinical finding ¹⁷⁰² : 116676008 Associated morphology ¹⁷⁰³ = 72704001 Fracture ¹⁷⁰⁴
	385933006 Edema control education ¹⁷⁰⁵	
< 404684003 Clinical finding ¹⁷⁰⁶ : 116676008 Associated morphology ¹⁷⁰⁷ = *	195967001 Asthma ¹⁷⁰⁸	404684003 Clinical finding ¹⁷⁰⁹ : 363698007 Finding site ¹⁷¹⁰ = 80891009 Heart structure ¹⁷¹¹
	73211009 Diabetes mellitus ¹⁷¹²	404684003 Clinical finding ¹⁷¹³ : 246075003 Causative agent ¹⁷¹⁴ = 372687004 Amoxicillin ¹⁷¹⁵

¹[\(see page 168\)](#) Please note that some of these examples are based on a hypothetical drug concept model.

²[\(see page 168\)](#) The SNOMED CT identifiers created with the '9999999' namespace are for example only, and should not be used in a production environment.

9.3 B.3 Cardinality - Invalid Expressions

Expression Constraint	INVALID Expression ¹ (see page 0)
-----------------------	--

- 1698 <http://snomed.info/id/387517004>
- 1699 <http://snomed.info/id/404684003>
- 1700 <http://snomed.info/id/79654002>
- 1701 <http://snomed.info/id/263225007>
- 1702 <http://snomed.info/id/404684003>
- 1703 <http://snomed.info/id/116676008>
- 1704 <http://snomed.info/id/72704001>
- 1705 <http://snomed.info/id/385933006>
- 1706 <http://snomed.info/id/404684003>
- 1707 <http://snomed.info/id/116676008>
- 1708 <http://snomed.info/id/195967001>
- 1709 <http://snomed.info/id/404684003>
- 1710 <http://snomed.info/id/363698007>
- 1711 <http://snomed.info/id/80891009>
- 1712 <http://snomed.info/id/73211009>
- 1713 <http://snomed.info/id/404684003>
- 1714 <http://snomed.info/id/246075003>
- 1715 <http://snomed.info/id/372687004>

	Precoordinated	Postcoordinated
<p>< 373873005 Pharmaceutical / biologic product¹⁷¹⁶ :</p> <p>[1..3] 127489000 Has active ingredient¹⁷¹⁷ =</p> <p>< 105590001 Substance¹⁷¹⁸</p>	<p>279999999108 Inert tablet¹⁷¹⁹</p>	<p>373873005 Pharmaceutical / biologic product¹⁷²⁰ :</p> <p>{ 127489000 Has active ingredient¹⁷²¹ =</p> <p>412031009 Paracetamol or derivative¹⁷²² },</p> <p>{ 127489000 Has active ingredient¹⁷²³ =</p> <p>387494007 Codeine¹⁷²⁴ },</p> <p>{ 127489000 Has active ingredient¹⁷²⁵ =</p> <p>255641001 Caffeine¹⁷²⁶ },</p> <p>{ 127489000 Has active ingredient¹⁷²⁷ =</p> <p>44068004 Doxylamine¹⁷²⁸ }</p>
	<p>437867004 Chlorphenamine + dextromethorphan + paracetamol + pseudoephedrine¹⁷²⁹</p>	
<p>< 373873005 Pharmaceutical / biologic product¹⁷³⁰ :</p> <p>[1..1] 127489000 Has active ingredient¹⁷³¹ =</p> <p>< 105590001 Substance¹⁷³²</p>	<p>279999999108 Inert tablet¹⁷³³</p>	<p>373873005 Pharmaceutical / biologic product¹⁷³⁴ :</p> <p>{ 127489000 Has active ingredient¹⁷³⁵ =</p> <p>412031009 Paracetamol or derivative¹⁷³⁶ },</p> <p>{ 127489000 Has active ingredient¹⁷³⁷ =</p> <p>387494007 Codeine¹⁷³⁸ }</p>
	<p>412556009 Paracetamol + codeine¹⁷³⁹</p>	

1716 <http://snomed.info/id/373873005>
 1717 <http://snomed.info/id/127489000>
 1718 <http://snomed.info/id/105590001>
 1719 <http://snomed.org/fictid#279999999108>
 1720 <http://snomed.info/id/373873005>
 1721 <http://snomed.info/id/127489000>
 1722 <http://snomed.info/id/412031009>
 1723 <http://snomed.info/id/127489000>
 1724 <http://snomed.info/id/387494007>
 1725 <http://snomed.info/id/127489000>
 1726 <http://snomed.info/id/255641001>
 1727 <http://snomed.info/id/127489000>
 1728 <http://snomed.info/id/44068004>
 1729 <http://snomed.info/id/437867004>
 1730 <http://snomed.info/id/373873005>
 1731 <http://snomed.info/id/127489000>
 1732 <http://snomed.info/id/105590001>
 1733 <http://snomed.org/fictid#279999999108>
 1734 <http://snomed.info/id/373873005>
 1735 <http://snomed.info/id/127489000>
 1736 <http://snomed.info/id/412031009>
 1737 <http://snomed.info/id/127489000>
 1738 <http://snomed.info/id/387494007>
 1739 <http://snomed.info/id/412556009>

<p>< 373873005 Pharmaceutical / biologic product¹⁷⁴⁰ : [0..1] 127489000 Has active ingredient¹⁷⁴¹ = < 105590001 Substance¹⁷⁴²</p>	<p>412556009 Paracetamol + codeine¹⁷⁴³</p>	<p>373873005 Pharmaceutical / biologic product¹⁷⁴⁴ : { 127489000 Has active ingredient¹⁷⁴⁵ = 412031009 Paracetamol or derivative¹⁷⁴⁶ }, { 127489000 Has active ingredient¹⁷⁴⁷ = 387494007 Codeine¹⁷⁴⁸ }</p>
<p>< 373873005 Pharmaceutical / biologic product¹⁷⁴⁹ : [1..*] 127489000 Has active ingredient¹⁷⁵⁰ = < 105590001 Substance¹⁷⁵¹</p>	<p>279999999108 Inert tablet¹⁷⁵²</p>	<p>373873005 Pharmaceutical / biologic product¹⁷⁵³ : 411116001 Has dose form¹⁷⁵⁴ = 385055001 Tablet¹⁷⁵⁵</p>
<p>< 404684003 Clinical finding¹⁷⁵⁶ : [1..1] 363698007 Finding site¹⁷⁵⁷ = < 91723000 Anatomical structure¹⁷⁵⁸</p>	<p>75857000 Fracture of radius and ulna¹⁷⁵⁹ 40733004 Infectious disease¹⁷⁶⁷</p>	<p>404684003 Clinical finding¹⁷⁶⁰ : { 116676008 Associated morphology¹⁷⁶¹ = 72704001 Fracture¹⁷⁶² , 363698007 Finding site¹⁷⁶³ = 62413002 Bone structure of radius¹⁷⁶⁴ , 363698007 Finding site¹⁷⁶⁵ = 23416004 Bone structure of ulna¹⁷⁶⁶ }</p>

1740 <http://snomed.info/id/373873005>
 1741 <http://snomed.info/id/127489000>
 1742 <http://snomed.info/id/105590001>
 1743 <http://snomed.info/id/412556009>
 1744 <http://snomed.info/id/373873005>
 1745 <http://snomed.info/id/127489000>
 1746 <http://snomed.info/id/412031009>
 1747 <http://snomed.info/id/127489000>
 1748 <http://snomed.info/id/387494007>
 1749 <http://snomed.info/id/373873005>
 1750 <http://snomed.info/id/127489000>
 1751 <http://snomed.info/id/105590001>
 1752 <http://snomed.org/fictid#279999999108>
 1753 <http://snomed.info/id/373873005>
 1754 <http://snomed.info/id/411116001>
 1755 <http://snomed.info/id/385055001>
 1756 <http://snomed.info/id/404684003>
 1757 <http://snomed.info/id/363698007>
 1758 <http://snomed.info/id/91723000>
 1759 <http://snomed.info/id/75857000>
 1760 <http://snomed.info/id/404684003>
 1761 <http://snomed.info/id/116676008>
 1762 <http://snomed.info/id/72704001>
 1763 <http://snomed.info/id/363698007>
 1764 <http://snomed.info/id/62413002>
 1765 <http://snomed.info/id/363698007>
 1766 <http://snomed.info/id/23416004>
 1767 <http://snomed.info/id/40733004>

<pre>< 404684003 Clinical finding¹⁷⁶⁸ : [2..*] 363698007 Finding site¹⁷⁶⁹ = < 91723000 Anatomical structure¹⁷⁷⁰</pre>	<pre>23406007 Arm fracture¹⁷⁷¹ 40733004 Infectious disease¹⁷⁷⁷</pre>	<pre>404684003 Clinical finding¹⁷⁷² : { 116676008 Associated morphology¹⁷⁷³ = 72704001 Fracture¹⁷⁷⁴ , 363698007 Finding site¹⁷⁷⁵ = 702468001 Bone structure of lower leg¹⁷⁷⁶ }</pre>
<pre>< 404684003 Clinical finding¹⁷⁷⁸ : { [2..*] 363698007 Finding site¹⁷⁷⁹ = < 91723000 Anatomical structure¹⁷⁸⁰ }</pre>	<pre>75857000 Fracture of radius and ulna¹⁷⁸¹</pre>	<pre>64572001 Disease¹⁷⁸² : { 116676008 Associated morphology¹⁷⁸³ = 396351009 Congenital septal defect¹⁷⁸⁴ , 363698007 Finding site¹⁷⁸⁵ = 113262008 Thoracic aorta structure¹⁷⁸⁶ } { 116676008 Associated morphology¹⁷⁸⁷ = 90141005 Congenital hypertrophy¹⁷⁸⁸ , 363698007 Finding site¹⁷⁸⁹ = 244384009 Entire right ventricle¹⁷⁹⁰ }</pre>

1768 <http://snomed.info/id/404684003>
1769 <http://snomed.info/id/363698007>
1770 <http://snomed.info/id/91723000>
1771 <http://snomed.info/id/23406007>
1772 <http://snomed.info/id/404684003>
1773 <http://snomed.info/id/116676008>
1774 <http://snomed.info/id/72704001>
1775 <http://snomed.info/id/363698007>
1776 <http://snomed.info/id/702468001>
1777 <http://snomed.info/id/40733004>
1778 <http://snomed.info/id/404684003>
1779 <http://snomed.info/id/363698007>
1780 <http://snomed.info/id/91723000>
1781 <http://snomed.info/id/75857000>
1782 <http://snomed.info/id/64572001>
1783 <http://snomed.info/id/116676008>
1784 <http://snomed.info/id/396351009>
1785 <http://snomed.info/id/363698007>
1786 <http://snomed.info/id/113262008>
1787 <http://snomed.info/id/116676008>
1788 <http://snomed.info/id/90141005>
1789 <http://snomed.info/id/363698007>
1790 <http://snomed.info/id/244384009>

<p>< 373873005 Pharmaceutical / biologic product¹⁷⁹¹ : [1..3] { [1..*] 127489000 Has active ingredient¹⁷⁹² = < 105590001 Substance¹⁷⁹³ }</p>	<p>279999999108 Inert tablet¹⁷⁹⁴</p> <p>437867004 Chlorphenamine + dextromethorphan + paracetamol + pseudoephedrine¹⁸⁰⁴</p>	<p>373873005 Pharmaceutical / biologic product¹⁷⁹⁵ : { 127489000 Has active ingredient¹⁷⁹⁶ = 412031009 Paracetamol or derivative¹⁷⁹⁷ }, { 127489000 Has active ingredient¹⁷⁹⁸ = 387494007 Codeine¹⁷⁹⁹ }, { 127489000 Has active ingredient¹⁸⁰⁰ = 255641001 Caffeine¹⁸⁰¹ }, { 127489000 Has active ingredient¹⁸⁰² = 44068004 Doxylamine¹⁸⁰³ }</p>
<p>< 373873005 Pharmaceutical / biologic product¹⁸⁰⁵ : [0..1] { 127489000 Has active ingredient¹⁸⁰⁶ = < 105590001 Substance¹⁸⁰⁷ }</p>	<p>412556009 Paracetamol + codeine¹⁸⁰⁸</p>	<p>373873005 Pharmaceutical / biologic product¹⁸⁰⁹ : { 127489000 Has active ingredient¹⁸¹⁰ = 412031009 Paracetamol or derivative¹⁸¹¹ }, { 127489000 Has active ingredient¹⁸¹² = 387494007 Codeine¹⁸¹³ }</p>

1791 <http://snomed.info/id/373873005>
 1792 <http://snomed.info/id/127489000>
 1793 <http://snomed.info/id/105590001>
 1794 <http://snomed.org/fictid#279999999108>
 1795 <http://snomed.info/id/373873005>
 1796 <http://snomed.info/id/127489000>
 1797 <http://snomed.info/id/412031009>
 1798 <http://snomed.info/id/127489000>
 1799 <http://snomed.info/id/387494007>
 1800 <http://snomed.info/id/127489000>
 1801 <http://snomed.info/id/255641001>
 1802 <http://snomed.info/id/127489000>
 1803 <http://snomed.info/id/44068004>
 1804 <http://snomed.info/id/437867004>
 1805 <http://snomed.info/id/373873005>
 1806 <http://snomed.info/id/127489000>
 1807 <http://snomed.info/id/105590001>
 1808 <http://snomed.info/id/412556009>
 1809 <http://snomed.info/id/373873005>
 1810 <http://snomed.info/id/127489000>
 1811 <http://snomed.info/id/412031009>
 1812 <http://snomed.info/id/127489000>
 1813 <http://snomed.info/id/387494007>

<pre>< 373873005 Pharmaceutical / biologic product¹⁸¹⁴ : [1..*]{ 127489000 Has active ingredient¹⁸¹⁵ = < 105590001 Substance¹⁸¹⁶ }</pre>	<pre>279999999108 Inert tablet¹⁸¹⁷</pre>	<pre>373873005 Pharmaceutical / biologic product¹⁸¹⁸ : 411116001 Has dose form¹⁸¹⁹ = 385055001 Tablet¹⁸²⁰</pre>
<pre>< 404684003 Clinical finding¹⁸²¹ : [1..1]{ 363698007 Finding site¹⁸²² = < 91723000 Anatomical structure¹⁸²³ }</pre>	<pre>75857000 Fracture of radius and ulna¹⁸²⁴ 40733004 Infectious disease¹⁸³⁴</pre>	<pre>404684003 Clinical finding¹⁸²⁵ : { 116676008 Associated morphology¹⁸²⁶ = 72704001 Fracture¹⁸²⁷ , 363698007 Finding site¹⁸²⁸ = 62413002 Bone structure of radius¹⁸²⁹ }, { 116676008 Associated morphology¹⁸³⁰ = 72704001 Fracture¹⁸³¹ , 363698007 Finding site¹⁸³² = 23416004 Bone structure of ulna¹⁸³³ }</pre>
<pre>< 404684003 Clinical finding¹⁸³⁵ : [0..0]{ [2..*] 363698007 Finding site¹⁸³⁶ = < 91723000 Anatomical structure¹⁸³⁷ }</pre>	<pre>-</pre>	<pre>64572001 Disease¹⁸³⁸ : { 116676008 Associated morphology¹⁸³⁹ =</pre>

1814 <http://snomed.info/id/373873005>
1815 <http://snomed.info/id/127489000>
1816 <http://snomed.info/id/105590001>
1817 <http://snomed.org/fictid#279999999108>
1818 <http://snomed.info/id/373873005>
1819 <http://snomed.info/id/411116001>
1820 <http://snomed.info/id/385055001>
1821 <http://snomed.info/id/404684003>
1822 <http://snomed.info/id/363698007>
1823 <http://snomed.info/id/91723000>
1824 <http://snomed.info/id/75857000>
1825 <http://snomed.info/id/404684003>
1826 <http://snomed.info/id/116676008>
1827 <http://snomed.info/id/72704001>
1828 <http://snomed.info/id/363698007>
1829 <http://snomed.info/id/62413002>
1830 <http://snomed.info/id/116676008>
1831 <http://snomed.info/id/72704001>
1832 <http://snomed.info/id/363698007>
1833 <http://snomed.info/id/23416004>
1834 <http://snomed.info/id/40733004>
1835 <http://snomed.info/id/404684003>
1836 <http://snomed.info/id/363698007>
1837 <http://snomed.info/id/91723000>
1838 <http://snomed.info/id/64572001>
1839 <http://snomed.info/id/116676008>

	<p>396351009 Congenital septal defect¹⁸⁴⁰ , 363698007 Finding site¹⁸⁴¹ = 25943004 Structure of atrioventricular node¹⁸⁴² , 363698007 Finding site¹⁸⁴³ = 113262008 Thoracic aorta structure¹⁸⁴⁴ } { 116676008 Associated morphology¹⁸⁴⁵ = 90141005 Congenital hypertrophy¹⁸⁴⁶ , 363698007 Finding site¹⁸⁴⁷ = 244384009 entire right ventricle¹⁸⁴⁸ }</p>
--	---

[\[see page 177\]](#) The SNOMED CT identifiers created with the '9999999' namespace are for example only, and should not be used in a production environment.

9.4 B.4 Conjunction and Disjunction - Invalid Expressions

Expression Constraint	INVALID Expression [see page 0]	
	Precoordinated	Postcoordinated
<p>< 19829001 Disorder of lung¹⁸⁴⁹ AND < 301867009 Edema of trunk¹⁸⁵⁰</p>	<p>73452002 Abscess of lung¹⁸⁵¹</p> <p>248508001 Abdominal wall edema¹⁸⁵⁵</p>	<p>248508001 Abdominal wall edema¹⁸⁵² : 116676008 Associated morphology¹⁸⁵³ = 40829002 Acute edema¹⁸⁵⁴</p>

1840 <http://snomed.info/id/396351009>
 1841 <http://snomed.info/id/363698007>
 1842 <http://snomed.info/id/25943004>
 1843 <http://snomed.info/id/363698007>
 1844 <http://snomed.info/id/113262008>
 1845 <http://snomed.info/id/116676008>
 1846 <http://snomed.info/id/90141005>
 1847 <http://snomed.info/id/363698007>
 1848 <http://snomed.info/id/244384009>
 1849 <http://snomed.info/id/19829001>
 1850 <http://snomed.info/id/301867009>
 1851 <http://snomed.info/id/73452002>
 1852 <http://snomed.info/id/248508001>
 1853 <http://snomed.info/id/116676008>
 1854 <http://snomed.info/id/40829002>
 1855 <http://snomed.info/id/248508001>

<p>< 19829001 Disorder of lung¹⁸⁵⁶ OR < 301867009 Edema of trunk¹⁸⁵⁷</p>	<p>19829001 Disorder of lung¹⁸⁵⁸</p> <hr/> <p>301867009 Edema of trunk¹⁸⁶²</p> <hr/> <p>128121009 Disorder of trunk¹⁸⁶³</p>	<p>128121009 Disorder of trunk¹⁸⁵⁹ :</p> <p>116676008 Associated morphology¹⁸⁶⁰ =</p> <p>44132006 Abscess¹⁸⁶¹</p>
<p>< 19829001 Disorder of lung¹⁸⁶⁴ AND ^ 700043003 Example problem list concepts reference set¹⁸⁶⁵</p>	<p>73452002 Abscess of lung¹⁸⁶⁶</p>	<p>19829001 Disorder of lung¹⁸⁶⁷ :</p> <p>116676008 Associated morphology¹⁸⁶⁸ =</p> <p>44132006 Abscess¹⁸⁶⁹</p>
<p>< 404684003 Clinical finding¹⁸⁷⁰ : 363698007 Finding site¹⁸⁷¹ = << 39057004 Pulmonary valve structure¹⁸⁷² AND 116676008 Associated morphology¹⁸⁷³ = << 415582006 Stenosis¹⁸⁷⁴</p>	<p>301104003 Pulmonary valve finding¹⁸⁷⁵</p> <hr/> <p>60573004 Aortic valve stenosis¹⁸⁷⁹</p>	<p>404684003 Clinical finding¹⁸⁷⁶ :</p> <p>116676008 Associated morphology¹⁸⁷⁷ =</p> <p>88015002 Partial stenosis¹⁸⁷⁸</p>
<p>< 404684003 Clinical finding¹⁸⁸⁰ : 116676008 Associated morphology¹⁸⁸¹ = << 55641003 Infarct¹⁸⁸² OR</p>	<p>368009 Heart valve disorder¹⁸⁸⁵</p>	<p>95281009 Sudden cardiac death¹⁸⁸⁶ :</p> <p>42752001 Due to¹⁸⁸⁷ =</p>

- 1856 <http://snomed.info/id/19829001>
- 1857 <http://snomed.info/id/301867009>
- 1858 <http://snomed.info/id/19829001>
- 1859 <http://snomed.info/id/128121009>
- 1860 <http://snomed.info/id/116676008>
- 1861 <http://snomed.info/id/44132006>
- 1862 <http://snomed.info/id/301867009>
- 1863 <http://snomed.info/id/128121009>
- 1864 <http://snomed.info/id/19829001>
- 1865 <http://snomed.info/id/700043003>
- 1866 <http://snomed.info/id/73452002>
- 1867 <http://snomed.info/id/19829001>
- 1868 <http://snomed.info/id/116676008>
- 1869 <http://snomed.info/id/44132006>
- 1870 <http://snomed.info/id/404684003>
- 1871 <http://snomed.info/id/363698007>
- 1872 <http://snomed.info/id/39057004>
- 1873 <http://snomed.info/id/116676008>
- 1874 <http://snomed.info/id/415582006>
- 1875 <http://snomed.info/id/301104003>
- 1876 <http://snomed.info/id/404684003>
- 1877 <http://snomed.info/id/116676008>
- 1878 <http://snomed.info/id/88015002>
- 1879 <http://snomed.info/id/60573004>
- 1880 <http://snomed.info/id/404684003>
- 1881 <http://snomed.info/id/116676008>
- 1882 <http://snomed.info/id/55641003>
- 1885 <http://snomed.info/id/368009>
- 1886 <http://snomed.info/id/95281009>
- 1887 <http://snomed.info/id/42752001>

<p>42752001 Due to¹⁸⁸³ = << 22298006 Myocardial infarction¹⁸⁸⁴</p>	<p>461089003 Cardiac abnormality due to heart abscess¹⁸⁸⁹</p>	<p>10633002 Acute congestive heart failure¹⁸⁸⁸</p>
<p>< 404684003 Clinical finding¹⁸⁹⁰ : { 363698007 Finding site¹⁸⁹¹ = << 39057004 Pulmonary valve structure¹⁸⁹² , 116676008 Associated morphology¹⁸⁹³ = << 415582006 Stenosis¹⁸⁹⁴ } OR { 363698007 Finding site¹⁸⁹⁵ = << 53085002 Right ventricular structure¹⁸⁹⁶ , 116676008 Associated morphology¹⁸⁹⁷ = << 56246009 Hypertrophy¹⁸⁹⁸ }</p>	<p>93075009 Congenital hypertrophy of pulmonary valve¹⁸⁹⁹</p>	<p>404684003 Clinical finding¹⁹⁰⁰ : 363698007 Finding site¹⁹⁰¹ = 39057004 Pulmonary valve structure¹⁹⁰² ,</p>
<p>^ 450990004 Adverse drug reactions reference set for GP/FP health issue¹⁹⁰⁶ : 246075003 Causative agent¹⁹⁰⁷ = (< 373873005 Pharmaceutical / biologic product¹⁹⁰⁸ / OR < 105590001 Substance¹⁹⁰⁹)</p>	<p>204370002 Stenosis of infundibulum of right ventricle¹⁹⁰⁵</p>	<p>116676008 Associated morphology¹⁹⁰³ = 56246009 Hypertrophy¹⁹⁰⁴</p>
<p>87628006 Bacterial infectious disease¹⁹¹⁰</p>	<p>609328004 Allergic disposition¹⁹¹⁴</p>	<p>609328004 Allergic disposition¹⁹¹¹ : 246075003 Causative agent¹⁹¹² = 84489001 Mold¹⁹¹³</p>

1883 <http://snomed.info/id/42752001>
1884 <http://snomed.info/id/22298006>
1889 <http://snomed.info/id/461089003>
1888 <http://snomed.info/id/10633002>
1890 <http://snomed.info/id/404684003>
1891 <http://snomed.info/id/363698007>
1892 <http://snomed.info/id/39057004>
1893 <http://snomed.info/id/116676008>
1894 <http://snomed.info/id/415582006>
1895 <http://snomed.info/id/363698007>
1896 <http://snomed.info/id/53085002>
1897 <http://snomed.info/id/116676008>
1898 <http://snomed.info/id/56246009>
1899 <http://snomed.info/id/93075009>
1900 <http://snomed.info/id/404684003>
1901 <http://snomed.info/id/363698007>
1902 <http://snomed.info/id/39057004>
1903 <http://snomed.info/id/116676008>
1904 <http://snomed.info/id/56246009>
1905 <http://snomed.info/id/204370002>
1906 <http://snomed.info/id/450990004>
1907 <http://snomed.info/id/246075003>
1908 <http://snomed.info/id/373873005>
1909 <http://snomed.info/id/105590001>
1910 <http://snomed.info/id/87628006>
1911 <http://snomed.info/id/609328004>
1912 <http://snomed.info/id/246075003>
1913 <http://snomed.info/id/84489001>
1914 <http://snomed.info/id/609328004>

	10629471000119106 Allergic rhinitis caused by mould ¹⁹¹⁵	
< 404684003 Clinical finding ¹⁹¹⁶ : 116676008 Associated morphology ¹⁹¹⁷ = (<< 56208002 Ulcer ¹⁹¹⁸ AND << 50960005 Hemorrhage ¹⁹¹⁹)	196652006 Acute duodenal ulcer ¹⁹²⁰	64572001 Disease ¹⁹²¹ : 116676008 Associated morphology ¹⁹²² = 405719001 Chronic ulcer ¹⁹²³
	74474003 Gastrointestinal haemorrhage ¹⁹²⁴	

[\[see page 183\]](#) Where necessary, these examples make some assumptions about the membership of the example reference sets.

9.5 B.5 Exclusion and Not Equals - Invalid Expressions

Expression Constraint	INVALID Expression	
	Precoordinated	Postcoordinated
<< 19829001 Disorder of lung ¹⁹²⁵ MINUS << 301867009 Edema of trunk ¹⁹²⁶	27719009 Acute gastrointestinal hemorrhage ¹⁹²⁷	19829001 Disorder of lung ¹⁹²⁸ : { 116676008 Associated morphology ¹⁹²⁹ = 40829002 Acute edema ¹⁹³⁰ , 363698007 Finding site ¹⁹³¹ = 22943007 Trunk structure ¹⁹³² }
	19242006 Pulmonary edema ¹⁹³³	

1915 <http://snomed.info/id/10629471000119106>
 1916 <http://snomed.info/id/404684003>
 1917 <http://snomed.info/id/116676008>
 1918 <http://snomed.info/id/56208002>
 1919 <http://snomed.info/id/50960005>
 1920 <http://snomed.info/id/196652006>
 1921 <http://snomed.info/id/64572001>
 1922 <http://snomed.info/id/116676008>
 1923 <http://snomed.info/id/405719001>
 1924 <http://snomed.info/id/74474003>
 1925 <http://snomed.info/id/19829001>
 1926 <http://snomed.info/id/301867009>
 1927 <http://snomed.info/id/27719009>
 1928 <http://snomed.info/id/19829001>
 1929 <http://snomed.info/id/116676008>
 1930 <http://snomed.info/id/40829002>
 1931 <http://snomed.info/id/363698007>
 1932 <http://snomed.info/id/22943007>
 1933 <http://snomed.info/id/19242006>

<p><< 19829001 Disorder of lung¹⁹³⁴ MINUS ^ 700043003 Example problem list concepts reference set¹⁹³⁵</p>	<p>67599009 Pulmonary congestion¹⁹³⁶</p>	<p>67599009 Pulmonary congestion¹⁹³⁷ : 363698007 Finding site¹⁹³⁸ = 3341006 Right lung structure¹⁹³⁹</p>
<p>< 404684003 Clinical finding¹⁹⁴⁰ : 116676008 Associated morphology¹⁹⁴¹ = ((<< 56208002 Ulcer¹⁹⁴² AND << 50960005 Hemorrhage¹⁹⁴³) MINUS << 26036001 Obstruction¹⁹⁴⁴)</p>	<p>397825006 Gastric ulcer¹⁹⁴⁵ 235670001 Gastric stomal obstruction¹⁹⁴⁹</p>	<p>64572001 Disease¹⁹⁴⁶ : 116676008 Associated morphology¹⁹⁴⁷ = 26036001 Obstruction¹⁹⁴⁸</p>
<p>< 404684003 Clinical finding¹⁹⁵⁰ : 116676008 Associated morphology¹⁹⁵¹ != << 26036001 Obstruction¹⁹⁵²</p>	<p>81060008 Intestinal obstruction¹⁹⁵³ 56265001 Heart disease¹⁹⁵⁹</p>	<p>64572001 Disease¹⁹⁵⁴ : 116676008 Associated morphology¹⁹⁵⁵ = 26036001 Obstruction¹⁹⁵⁶ , 363698007 Finding site¹⁹⁵⁷ = 422897007 Vascular structure of stomach¹⁹⁵⁸</p>

1934 <http://snomed.info/id/19829001>
 1935 <http://snomed.info/id/700043003>
 1936 <http://snomed.info/id/67599009>
 1937 <http://snomed.info/id/67599009>
 1938 <http://snomed.info/id/363698007>
 1939 <http://snomed.info/id/3341006>
 1940 <http://snomed.info/id/404684003>
 1941 <http://snomed.info/id/116676008>
 1942 <http://snomed.info/id/56208002>
 1943 <http://snomed.info/id/50960005>
 1944 <http://snomed.info/id/26036001>
 1945 <http://snomed.info/id/397825006>
 1946 <http://snomed.info/id/64572001>
 1947 <http://snomed.info/id/116676008>
 1948 <http://snomed.info/id/26036001>
 1949 <http://snomed.info/id/235670001>
 1950 <http://snomed.info/id/404684003>
 1951 <http://snomed.info/id/116676008>
 1952 <http://snomed.info/id/26036001>
 1953 <http://snomed.info/id/81060008>
 1954 <http://snomed.info/id/64572001>
 1955 <http://snomed.info/id/116676008>
 1956 <http://snomed.info/id/26036001>
 1957 <http://snomed.info/id/363698007>
 1958 <http://snomed.info/id/422897007>
 1959 <http://snomed.info/id/56265001>

<p>< 404684003 Clinical finding¹⁹⁶⁰ : [0..0] 116676008 Associated morphology¹⁹⁶¹ = << 26036001 Obstruction¹⁹⁶²</p>	<p>81060008 Intestinal obstruction¹⁹⁶³</p> <hr/> <p>234059001 Venous stenosis¹⁹⁷³</p>	<p>64572001 Disease¹⁹⁶⁴ : { 116676008 Associated morphology¹⁹⁶⁵ = 26036001 Obstruction¹⁹⁶⁶ , 363698007 Finding site¹⁹⁶⁷ = 422897007 Vascular structure of stomach¹⁹⁶⁸ } { 116676008 Associated morphology¹⁹⁶⁹ = 45771005 Acute bleeding ulcer¹⁹⁷⁰ , 363698007 Finding site¹⁹⁷¹ = 422897007 Vascular structure of stomach¹⁹⁷² }</p>
<p>< 404684003 Clinical finding¹⁹⁷⁴ : [0..0] 116676008 Associated morphology¹⁹⁷⁵ != << 26036001 Obstruction¹⁹⁷⁶</p>	<p>196652006 Acute duodenal ulcer¹⁹⁷⁷</p>	<p>64572001 Disease¹⁹⁷⁸ : { 116676008 Associated morphology¹⁹⁷⁹ = 26036001 Obstruction¹⁹⁸⁰ , 363698007 Finding site¹⁹⁸¹ = 422897007 Vascular structure of stomach¹⁹⁸² } { 116676008 Associated morphology¹⁹⁸³ = 45771005 Acute bleeding ulcer¹⁹⁸⁴ , 363698007 Finding site¹⁹⁸⁵ = 422897007 Vascular structure of stomach¹⁹⁸⁶ }</p>

1960 <http://snomed.info/id/404684003>
 1961 <http://snomed.info/id/116676008>
 1962 <http://snomed.info/id/26036001>
 1963 <http://snomed.info/id/81060008>
 1964 <http://snomed.info/id/64572001>
 1965 <http://snomed.info/id/116676008>
 1966 <http://snomed.info/id/26036001>
 1967 <http://snomed.info/id/363698007>
 1968 <http://snomed.info/id/422897007>
 1969 <http://snomed.info/id/116676008>
 1970 <http://snomed.info/id/45771005>
 1971 <http://snomed.info/id/363698007>
 1972 <http://snomed.info/id/422897007>
 1973 <http://snomed.info/id/234059001>
 1974 <http://snomed.info/id/404684003>
 1975 <http://snomed.info/id/116676008>
 1976 <http://snomed.info/id/26036001>
 1977 <http://snomed.info/id/196652006>
 1978 <http://snomed.info/id/64572001>
 1979 <http://snomed.info/id/116676008>
 1980 <http://snomed.info/id/26036001>
 1981 <http://snomed.info/id/363698007>
 1982 <http://snomed.info/id/422897007>
 1983 <http://snomed.info/id/116676008>
 1984 <http://snomed.info/id/45771005>
 1985 <http://snomed.info/id/363698007>
 1986 <http://snomed.info/id/422897007>

	8377001 Hernia, with obstruction 1987	
<p>< 404684003 Clinical finding ¹⁹⁸⁸ : [0..0] 116676008 Associated morphology ¹⁹⁸⁹ != << 26036001 Obstruction ¹⁹⁹⁰ AND [1..*] 116676008 Associated morphology ¹⁹⁹¹ = << 26036001 Obstruction ¹⁹⁹²</p>	196652006 Acute duodenal ulcer 1993	64572001 Disease ¹⁹⁹⁴ : { 116676008 Associated morphology ¹⁹⁹⁵ = 26036001 Obstruction ¹⁹⁹⁶ , 363698007 Finding site ¹⁹⁹⁷ = 422897007 Vascular structure of stomach ¹⁹⁹⁸ } { 116676008 Associated morphology ¹⁹⁹⁹ = 45771005 Acute bleeding ulcer ²⁰⁰⁰ , 363698007 Finding site ²⁰⁰¹ = 422897007 vascular structure of stomach ²⁰⁰² }
	8377001 Hernia, with obstruction 2003	
	56265001 Heart disease ²⁰⁰⁹	64572001 Disease ²⁰⁰⁴ : { 116676008 Associated morphology ²⁰⁰⁵ = 45771005 Acute bleeding ulcer ²⁰⁰⁶ , 363698007 Finding site ²⁰⁰⁷ = 422897007 Vascular structure of stomach ²⁰⁰⁸ }

1987 <http://snomed.info/id/8377001>
1988 <http://snomed.info/id/404684003>
1989 <http://snomed.info/id/116676008>
1990 <http://snomed.info/id/26036001>
1991 <http://snomed.info/id/116676008>
1992 <http://snomed.info/id/26036001>
1993 <http://snomed.info/id/196652006>
1994 <http://snomed.info/id/64572001>
1995 <http://snomed.info/id/116676008>
1996 <http://snomed.info/id/26036001>
1997 <http://snomed.info/id/363698007>
1998 <http://snomed.info/id/422897007>
1999 <http://snomed.info/id/116676008>
2000 <http://snomed.info/id/45771005>
2001 <http://snomed.info/id/363698007>
2002 <http://snomed.info/id/422897007>
2003 <http://snomed.info/id/8377001>
2004 <http://snomed.info/id/64572001>
2005 <http://snomed.info/id/116676008>
2006 <http://snomed.info/id/45771005>
2007 <http://snomed.info/id/363698007>
2008 <http://snomed.info/id/422897007>
2009 <http://snomed.info/id/56265001>

9.6 B.6 Nested Expression Constraints - Invalid Expressions

Expression Constraint	Valid Expression [see page 0]	
	Precoordinated	Postcoordinated
<< (^ 700043003 Example problem list concepts reference set ²⁰¹⁰)	6143009 Diabetic education ²⁰¹¹	71388002 Procedure ²⁰¹² : 405813007 Procedure site - Direct ²⁰¹³ = 80891009 Heart structure ²⁰¹⁴
	75367002 Blood pressure ²⁰¹⁵	
^ (< 450973005 GP/FP health issue reference set ²⁰¹⁶)	80146002 Appendectomy ²⁰¹⁷	-
	305342007 Admission to ward ²⁰¹⁸	
(< 404684003 Clinical finding ²⁰¹⁹ : 363698007 Finding site ²⁰²⁰ = << 39057004 Pulmonary valve structure ²⁰²¹) AND ^ 700043003 Example problem list concepts reference set ²⁰²²	125605004 Fracture of bone ²⁰²³	404684003 Clinical finding ²⁰²⁴ : 363698007 Finding site ²⁰²⁵ = 17401000 Cardiac valve structure ²⁰²⁶
	195967001 Asthma ²⁰²⁷	

2010 <http://snomed.info/id/700043003>
 2011 <http://snomed.info/id/6143009>
 2012 <http://snomed.info/id/71388002>
 2013 <http://snomed.info/id/405813007>
 2014 <http://snomed.info/id/80891009>
 2015 <http://snomed.info/id/75367002>
 2016 <http://snomed.info/id/450973005>
 2017 <http://snomed.info/id/80146002>
 2018 <http://snomed.info/id/305342007>
 2019 <http://snomed.info/id/404684003>
 2020 <http://snomed.info/id/363698007>
 2021 <http://snomed.info/id/39057004>
 2022 <http://snomed.info/id/700043003>
 2023 <http://snomed.info/id/125605004>
 2024 <http://snomed.info/id/404684003>
 2025 <http://snomed.info/id/363698007>
 2026 <http://snomed.info/id/17401000>
 2027 <http://snomed.info/id/195967001>

<p>(< 404684003 Clinical finding²⁰²⁸ : 363698007 Finding site²⁰²⁹ = << 39057004 Pulmonary valve structure²⁰³⁰) AND (< 64572001 Disease²⁰³¹ : 116676008 Associated morphology²⁰³² = << 415582006 Stenosis²⁰³³)</p>	<p>301104003 Pulmonary valve finding²⁰³⁴</p>	<p>404684003 Clinical finding²⁰³⁵ : 363698007 Finding site²⁰³⁶ = 39057004 Pulmonary valve structure²⁰³⁷</p>
<p>(<< 17636008 Specimen collection²⁰⁴² : 424226004 Using device²⁰⁴³ = << 19923001 Catheter²⁰⁴⁴) . 363701004 Direct substance²⁰⁴⁵</p>	<p>57617002 Urine specimen collection²⁰⁴⁶</p>	<p>17636008 Specimen collection²⁰⁴⁷ : 424226004 Using device²⁰⁴⁸ = 19923001 Catheter²⁰⁴⁹</p>
<p>(<< 404684003 Clinical finding (finding)²⁰⁵¹ OR << 272379006 Event (event)²⁰⁵²) : 255234002 After²⁰⁵³ = << 71388002 Procedure (procedure)²⁰⁵⁴</p>	<p>293690005 Peppermint oil allergy²⁰⁵⁵</p>	<p>404684003 Clinical finding²⁰⁵⁶ : 255234002 After²⁰⁵⁷ = 417163006 Injury²⁰⁵⁸</p>

2028 <http://snomed.info/id/404684003>
 2029 <http://snomed.info/id/363698007>
 2030 <http://snomed.info/id/39057004>
 2031 <http://snomed.info/id/64572001>
 2032 <http://snomed.info/id/116676008>
 2033 <http://snomed.info/id/415582006>
 2034 <http://snomed.info/id/301104003>
 2035 <http://snomed.info/id/404684003>
 2036 <http://snomed.info/id/363698007>
 2037 <http://snomed.info/id/39057004>
 2038 <http://snomed.info/id/76107001>
 2039 <http://snomed.info/id/64572001>
 2040 <http://snomed.info/id/116676008>
 2041 <http://snomed.info/id/415582006>
 2042 <http://snomed.info/id/17636008>
 2043 <http://snomed.info/id/424226004>
 2044 <http://snomed.info/id/19923001>
 2045 <http://snomed.info/id/363701004>
 2046 <http://snomed.info/id/57617002>
 2047 <http://snomed.info/id/17636008>
 2048 <http://snomed.info/id/424226004>
 2049 <http://snomed.info/id/19923001>
 2050 <http://snomed.info/id/122575003>
 2051 <http://snomed.info/id/404684003>
 2052 <http://snomed.info/id/272379006>
 2053 <http://snomed.info/id/255234002>
 2054 <http://snomed.info/id/71388002>
 2055 <http://snomed.info/id/293690005>
 2056 <http://snomed.info/id/404684003>
 2057 <http://snomed.info/id/255234002>
 2058 <http://snomed.info/id/417163006>

	82510005 Posttraumatic vertigo ²⁰⁵⁹	
<< 125605004 Fracture of bone ²⁰⁶⁰ : [0..0] ((<< 410662002 Concept model attribute ²⁰⁶¹ MINUS 363698007 Finding site ²⁰⁶²) MINUS 116676008 Associated morphology ²⁰⁶³) = *	704333004 Pathological fracture of hand due to osteoporosis ²⁰⁶⁴	125605004 Fracture of bone ²⁰⁶⁵ : 42752001 Due to ²⁰⁶⁶ = 417163006 Injury ²⁰⁶⁷
	722571004 Linear fracture of skull due to birth trauma ²⁰⁶⁸	
< 404684003 Clinical finding ²⁰⁶⁹ : 47429007 Associated with ²⁰⁷⁰ = (< 404684003 Clinical finding ²⁰⁷¹ : 116676008 Associated morphology ²⁰⁷² = << 55641003 Infarct ²⁰⁷³)	3238004 Pericarditis ²⁰⁷⁴	64572001 Disease ²⁰⁷⁵ : 47429007 Associated with ²⁰⁷⁶ = (404684003 Clinical finding ²⁰⁷⁷ : 363698007 Finding site ²⁰⁷⁸) = 277712000 Cardiac internal structure ²⁰⁷⁹)

[\[see page 190\]](#) Where necessary, these examples make some assumptions about the membership of the example reference sets.

2059 <http://snomed.info/id/82510005>
 2060 <http://snomed.info/id/125605004>
 2061 <http://snomed.info/id/410662002>
 2062 <http://snomed.info/id/363698007>
 2063 <http://snomed.info/id/116676008>
 2064 <http://snomed.info/id/704333004>
 2065 <http://snomed.info/id/125605004>
 2066 <http://snomed.info/id/42752001>
 2067 <http://snomed.info/id/417163006>
 2068 <http://snomed.info/id/722571004>
 2069 <http://snomed.info/id/404684003>
 2070 <http://snomed.info/id/47429007>
 2071 <http://snomed.info/id/404684003>
 2072 <http://snomed.info/id/116676008>
 2073 <http://snomed.info/id/55641003>
 2074 <http://snomed.info/id/3238004>
 2075 <http://snomed.info/id/64572001>
 2076 <http://snomed.info/id/47429007>
 2077 <http://snomed.info/id/404684003>
 2078 <http://snomed.info/id/363698007>
 2079 <http://snomed.info/id/277712000>

10 Appendix C - Dialect Aliases

This appendix provides a list of example aliases that may be used to specify a particular dialect in an ECL filter constraint. Please refer to the 'Dialect Filter' section on [6.8 Description Filters](#) (see page 105) for more information on how these dialect aliases are used in ECL.

All dialect aliases should follow the ABNF syntax shown below. This format is designed to be compatible with [BCP-47 \(Internet Best Current Practice Specification\)](#)²⁰⁸⁰, which ensures alignment with a range of other specifications - e.g. HTTP "accept-language" headers, and the HL7 FHIR "designation.language" data element.

dialectAlias = (language ["-" script] ["-" region] ["-" privateuse]) / privateuse

language = alpha alpha ; ISO 639-1 code ([List of codes](#)²⁰⁸¹)

script = alpha alpha alpha alpha ; ISO 15924 code ([List of codes](#)²⁰⁸²)

region = alpha alpha ; ISO 3166-1 code ([List of codes](#)²⁰⁸³)

privateuse = "x" 1*("-" 1*(alpha / digit)) ; the clinical scope or context of use

The table below lists the valid 'dialect' filter values and their equivalent 'dialectId' filter values, for a selection of known language reference sets. To request the addition of a new dialect alias, please use the 'Feedback' button on the bottom of this page.

dialect	dialectId
da-dk	554461000005103 Danish language reference set
en-au	32570271000036106 Australian English language reference set
en-ca	19491000087109 Canada English language reference set
en-gb	900000000000508004 Great Britain English language reference set
en-ie	21000220103 Irish language reference set
en-nz	271000210107 New Zealand English language reference set
en-nz-x-pat	281000210109 New Zealand English patient friendly terms language reference set
en-us	900000000000509007 United States of America English language reference set
en-x-gmdn	608771002 GMDN language reference set

2080 <https://www.rfc-editor.org/rfc/rfc5646.html>

2081 https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes

2082 https://en.wikipedia.org/wiki/ISO_15924#List_of_codes

2083 https://en.wikipedia.org/wiki/ISO_3166-1#Current_codes

dialect	dialectId
en-x-nhs-clinical	999001261000001100 National Health Service realm language reference set (clinical part)
en-x-nhs-dmd	999000671000001103 National Health Service dictionary of medicines and devices realm language reference set
en-x-nhs-pharmacy	999000691000001104 National Health Service realm language reference set (pharmacy part)
en-gb-x-drug	999000681000001101 United Kingdom Drug Extension Great Britain English language reference set
en-gb-x-ext	999001251000001103 United Kingdom Extension Great Britain English language reference set
es	450828004 Conjunto de referencias de lenguaje castellano para América Latina
es-uy	5641000179103 Conjunto de referencias de lenguaje castellano para Uruguay
et-ee	71000181105 Estonian language reference set
de	722130004 German language reference set
fr	722131000 French language reference set
fr-be	21000172104 Belgian French language reference set
fr-ca	20581000087109 Canada French language reference set
ja	722129009 Japanese language reference set
mi	291000210106 Maori language reference set
nl-be	31000172101 Belgian Dutch language reference set
nl-nl	31000146106 Netherlands Dutch language reference set
nb-no	61000202103 Norwegian Bokmål language reference set

dialect	dialectId
nn-no	91000202106 Norwegian Nynorsk language reference set
sv-se	46011000052107 Swedish language reference set
zh	722128001 Chinese language reference set

11 Appendix D - ECL Quick Reference

This section provides a quick reference to the key syntax features of the Expression Constraint Language.

11.1 Syntax Overview

The following table summarises the key symbols used in the Expression Constraint Language's brief syntax, with the ECL version in which each symbol was introduced. For more information about the version history of ECL, please refer to the 'History' section in [1. Introduction](#)(see page 8).

Symbol	Name	Version	Notes
	Pipe	1.0	Used on either side of a concept's term for human readability
*	Any	1.0	Retrieves all concepts in the substrate
^	Member of	1.0	Retrieves the referencedComponentId of all (active) members of a reference set (or set of reference sets)
^ [A, B]	Member of (with field selection)	2.0	Retrieves the values of fields A and B of all (active) members of a reference set (or set of reference sets) that match the included Member filters (if applicable)
<	Descendant of	1.0	Retrieves all descendants (subtypes) of the specified concept <i>excluding</i> the concept itself
<<	Descendant or self of	1.0	Retrieves all descendants (subtypes) of the specified concept <i>including</i> the concept itself
<!	Child of	1.1	Retrieves all children (immediate subtypes) of the specified concept <i>excluding</i> the concept itself
<<!	Child or self of	1.4	Retrieves all children (immediate subtypes) of the specified concept <i>including</i> the concept itself
>	Ancestor of	1.0	Retrieves all ancestors (supertypes) of the specified concept <i>excluding</i> the concept itself

Symbol	Name	Version	Notes
>>	Ancestor or self of	1.0	Retrieves all ancestors (supertypes) of the specified concept <i>including</i> the concept itself
>!	Parent of	1.1	Retrieves all parents (immediate supertypes) of the specified concept <i>excluding</i> the concept itself
>>!	Parent or self of	1.4	Retrieves all parents (immediate supertypes) of the specified concept <i>including</i> the concept itself
!!>	Top of set	2.2	Filters the results set, by matching only on concepts that have no ancestors within the set
!!<	Bottom of set	2.2	Filters the results set, by matching only on concepts that have no descendants within the set
A#B	Alternate identifier	2.2	Retrieves a single concept based on an alternate identifier, where A is the identifier scheme alias and B is the identifier code
AND	Conjunction	1.0	Retrieves the intersection of the results of each sub-expressions
OR	Disjunction	1.0	Retrieves the union of the results of each sub-expressions
MINUS	Exclusion	1.0	Retrieves the members of the first expression and excludes the members returned by the second expression
:	Refinement	1.0	Used before one or more attribute-value pairs to refine the set of concepts retrieved
[1..3]	Cardinality	1.0	Used to indicate the minimum and maximum number of occurrences of attributes or relationship groups

Symbol	Name	Version	Notes
R	Reverse flag	1.0	Retrieves the set of attribute values (i.e. destination concepts) of a specified attribute for a specified set of concepts
.	Dot notation	1.2	Retrieves the set of attribute values (i.e. destination concepts) of a specified attribute for a specified set of concepts
<i>/* */</i>	Comment	1.1	Allows comments to be added within the text of an expression constraint
{{ }}	Description filter	1.5	Filters the result set, by matching only on concepts which have a description with a matching term, language, type, dialect and/or acceptability
{{ D }}	Description filter	1.6	Filters the result set, by matching only on concepts which have a description with a matching term, language, type, dialect and/or acceptability
{{ C }}	Concept filter	1.6	Filters the result set based on the definition status, module, effectiveTime and active status of each concept
{{ M }}	Member filter	2.0	Filters the result set based on the value of specific fields in a reference set.
{{+ HISTORY}}	History supplement	2.0	Supplements the results with relevant inactive concepts

11.2 Examples

The following table provides some examples of each of the key syntax features of the Expression Constraint Language.

Notes:

- In the table above:
 - '**id**' represents a single SNOMED CT concept identifier,
 - '**term**' represents a term associated with the concept identified by '**id**',
 - '**x**', '**y**' and '**v**' each represent either a single concept or a set of concepts defined using an expression constraint,

- 'z' represents either a single concept or a set of concepts that are a subtype of 900000000000455006 | Reference set²⁰⁸⁴ ,
 - 'a' and 'b' each represent either a single concept or a set of concepts that are a subtype of 410662002 | Concept model attribute²⁰⁸⁵ , and
 - 'min' and 'max' are two numeric values that represent the minimum and maximum cardinality allowed.
2. The default substrate, to which expression constraints are applied, includes all concepts, active relationships, active descriptions and active reference set members of a chosen SNOMED CT versioned edition.

Simple expression constraints			
Syntax	Evaluation Notes	Example	Example Expansion Concepts
id term	Only the concept with the identifier 'id'	128477000 Abscess ²⁰⁸⁶	128477000 Abscess ²⁰⁸⁷
*	All concepts in the given substrate	*	Any concept in the given substrate
^ z	The set of concepts which are members of the reference sets in z	^ 723264001 Lateralizable body structure reference set ²⁰⁸⁸	181216001 Entire lung ²⁰⁸⁹ 65784005 Structure of fundus of eye ²⁰⁹⁰
< x	The set of all descendants (both direct and indirect) of x	< 73211009 Diabetes mellitus ²⁰⁹¹ < 73211009 Diabetes mellitus)	46635009 Diabetes mellitus type 1 ²⁰⁹² 8801005 Secondary diabetes mellitus ²⁰⁹³

2084 <http://snomed.info/id/900000000000455006>

2085 <http://snomed.info/id/410662002>

2086 <http://snomed.info/id/128477000>

2087 <http://snomed.info/id/128477000>

2088 <http://snomed.info/id/723264001>

2089 <http://snomed.info/id/181216001>

2090 <http://snomed.info/id/65784005>

2091 <http://snomed.info/id/73211009>

2092 <http://snomed.info/id/46635009>

2093 <http://snomed.info/id/8801005>

<p><< x</p>	<p>The set of all descendants (both direct and indirect) of x, plus x itself</p>	<p><< 73211009 Diabetes mellitus²⁰⁹⁴</p>	<p>73211009 Diabetes mellitus²⁰⁹⁵</p> <p>46635009 Diabetes mellitus type 1²⁰⁹⁶</p> <p>8801005 Secondary diabetes mellitus²⁰⁹⁷</p>
<p><! x</p>	<p>The set of all immediate children of x</p>	<p><! 362965005 Disorder of body system²⁰⁹⁸</p>	<p>49601007 Disorder of cardiovascular system²⁰⁹⁹</p> <p>362969004 Disorder of endocrine system²¹⁰⁰</p>
<p><<! x</p>	<p>The set of all immediate children of x, plus x itself</p>	<p><<! 362965005 Disorder of body system²¹⁰¹</p>	<p>362965005 Disorder of body system²¹⁰²</p> <p>49601007 Disorder of cardiovascular system²¹⁰³</p> <p>362969004 Disorder of endocrine system²¹⁰⁴</p>
<p>> x</p>	<p>The set of all ancestors (both direct and indirect) of x</p>	<p>> 279420009 Hematoma of skin²¹⁰⁵</p>	<p>106076001 Skin finding²¹⁰⁶</p> <p>297968009 Bleeding skin²¹⁰⁷</p>

2094 <http://snomed.info/id/73211009>
 2095 <http://snomed.info/id/73211009>
 2096 <http://snomed.info/id/46635009>
 2097 <http://snomed.info/id/8801005>
 2098 <http://snomed.info/id/362965005>
 2099 <http://snomed.info/id/49601007>
 2100 <http://snomed.info/id/362969004>
 2101 <http://snomed.info/id/362965005>
 2102 <http://snomed.info/id/362965005>
 2103 <http://snomed.info/id/49601007>
 2104 <http://snomed.info/id/362969004>
 2105 <http://snomed.info/id/279420009>
 2106 <http://snomed.info/id/106076001>
 2107 <http://snomed.info/id/297968009>

>> x	The set of all ancestors (both direct and indirect) of x, plus x itself	>> 279420009 Hematoma of skin 2108	106076001 Skin finding ²¹⁰⁹ 297968009 Bleeding skin ²¹¹⁰ 279420009 Hematoma of skin ²¹¹¹
>! x	The set of all immediate parents of x	>! 22298006 Myocardial infarction 2112	57809008 Myocardial disease 2113 251061000 Myocardial necrosis ²¹¹⁴
>>! x	The set of all immediate parents of x, plus x itself	>>! 22298006 Myocardial infarction 2115	22298006 Myocardial infarction ²¹¹⁶ 57809008 Myocardial disease 2117 251061000 Myocardial necrosis ²¹¹⁸

Conjunction, Disjunction and Exclusion

Syntax	Evaluation Notes	Example	Example Expansion Concepts
x AND y	The set of concepts that are both in x and in y (i.e. the intersection of x and y)	< 19829001 Disorder of lung ²¹¹⁹ AND < 87628006 Bacterial infectious disease ²¹²⁰	430395005 Pneumonia caused by Gram negative bacteria ²¹²¹

2108 <http://snomed.info/id/279420009>
 2109 <http://snomed.info/id/106076001>
 2110 <http://snomed.info/id/297968009>
 2111 <http://snomed.info/id/279420009>
 2112 <http://snomed.info/id/22298006>
 2113 <http://snomed.info/id/57809008>
 2114 <http://snomed.info/id/251061000>
 2115 <http://snomed.info/id/22298006>
 2116 <http://snomed.info/id/22298006>
 2117 <http://snomed.info/id/57809008>
 2118 <http://snomed.info/id/251061000>
 2119 <http://snomed.info/id/19829001>
 2120 <http://snomed.info/id/87628006>
 2121 <http://snomed.info/id/430395005>

			154283005 Pulmonary tuberculosis ²¹²²
x OR y	The set of concepts that are either in x or in y (i.e. the union of x and y)	< 73452002 Abscess of lung ²¹²³ OR < 275504005 Cyst of lung ²¹²⁴	446543007 Tuberculous abscess of lung ²¹²⁵ 87119009 Congenital cystic lung ²¹²⁶
x MINUS y	The set of concepts that are in x but are not in y (i.e. x excluding concepts in y)	< 29303009 Electrocardiographic procedure ²¹²⁷ MINUS < 75444003 Fetal electrocardiogram ²¹²⁸	447114004 12 lead electrocardiogram during exercise ²¹²⁹ 252417001 24 Hour electrocardiogram ²¹³⁰
Refinement			
Syntax	Evaluation Notes	Example	Example Expansion Concepts
x : a = y	The set of concepts in x , which have a necessary relationship with an attribute in a and a value in y	< 385494008 Hematoma ²¹³¹ : << 370135005 Pathological process ²¹³² = << 441862004 Infectious process ²¹³³	698573001 Infected hematoma ²¹³⁴ 444109008 Infection of wound hematoma ²¹³⁵

2122 <http://snomed.info/id/154283005>

2123 <http://snomed.info/id/73452002>

2124 <http://snomed.info/id/275504005>

2125 <http://snomed.info/id/446543007>

2126 <http://snomed.info/id/87119009>

2127 <http://snomed.info/id/29303009>

2128 <http://snomed.info/id/75444003>

2129 <http://snomed.info/id/447114004>

2130 <http://snomed.info/id/252417001>

2131 <http://snomed.info/id/385494008>

2132 <http://snomed.info/id/370135005>

2133 <http://snomed.info/id/441862004>

2134 <http://snomed.info/id/698573001>

2135 <http://snomed.info/id/444109008>

$x : a = y, b = v$	The set of concepts in x , which have both a necessary relationship with an attribute in a and a value in y , and also have a necessary relationship (either the same one or a different one) with an attribute in b and a value in v	$< 71388002 Procedure^{2136} :$ $<< 363704007 Procedure\ site^{2137}$ $= << 69695003 Stomach\ structure^{2138} ,$ $<< 405815000 Procedure\ device^{2139}$ $= << 86174004 Laparoscope^{2140}$	<p>708987006 Laparoscopic total gastrectomy²¹⁴¹</p> <p>57922004 Laparoscopic pyloromyotomy²¹⁴²</p>
$x : \{ a = y, b = v \}$	The set of concepts in x , which have a role group that contains both a necessary relationship with an attribute in a and a value in y , and also have a necessary relationship (either the same one or a different one) with an attribute in b and a value in v	$< 71388002 Procedure\ (procedure)^{2143} : \{ 405813007 Procedure\ site - Direct^{2144} = << 10200004 Liver\ structure^{2145} , 260686004 Method^{2146} = << 129433002 Inspection - action^{2147} \}$	<p>773252007 Diagnostic laparoscopy of liver²¹⁴⁸</p> <p>20933000 Endoscopy of liver²¹⁴⁹</p>
Cardinality			
Syntax	Evaluation Notes	Example	Example Expansion Concepts

2136 <http://snomed.info/id/71388002>
 2137 <http://snomed.info/id/363704007>
 2138 <http://snomed.info/id/69695003>
 2139 <http://snomed.info/id/405815000>
 2140 <http://snomed.info/id/86174004>
 2141 <http://snomed.info/id/708987006>
 2142 <http://snomed.info/id/57922004>
 2143 <http://snomed.info/id/71388002>
 2144 <http://snomed.info/id/405813007>
 2145 <http://snomed.info/id/10200004>
 2146 <http://snomed.info/id/260686004>
 2147 <http://snomed.info/id/129433002>
 2148 <http://snomed.info/id/773252007>
 2149 <http://snomed.info/id/20933000>

<p>x : [min .. max] a = y</p>	<p>The set of concepts in x, which have between min and max necessary relationships with an attribute in a and a value in y</p>	<p>< 373873005 Pharmaceutical / biologic product²¹⁵⁰ : [3..*] 127489000 Has active ingredient²¹⁵¹ =< 105590001 Substance²¹⁵²</p>	<p>786732006 Product containing only brompheniramine and codeine and phenylpropanolamine²¹⁵³</p> <p>787979009 Product containing cyanocobalamin and folic acid and pyridoxine²¹⁵⁴</p>
<p>x : [min .. max] { a = y }</p>	<p>The set of concepts in x, which have between min and max role groups that contain a necessary relationship with an attribute in a and a value in y</p>	<p>< 404684003 Clinical finding²¹⁵⁵ : [2..3]{ 363698007 Finding site²¹⁵⁶ = * , 116676008 Associated morphology²¹⁵⁷ = 72704001 Fracture²¹⁵⁸ }</p>	<p>271577005 Fracture of shaft of tibia and fibula²¹⁵⁹</p> <p>75857000 Fracture of radius AND ulna²¹⁶⁰</p>
<p>Reversed Attributes</p>			
<p>Syntax</p>	<p>Evaluation Notes</p>	<p>Example</p>	<p>Example Expansion Concepts</p>

2150 <http://snomed.info/id/373873005>
 2151 <http://snomed.info/id/127489000>
 2152 <http://snomed.info/id/105590001>
 2153 <http://snomed.info/id/786732006>
 2154 <http://snomed.info/id/787979009>
 2155 <http://snomed.info/id/404684003>
 2156 <http://snomed.info/id/363698007>
 2157 <http://snomed.info/id/116676008>
 2158 <http://snomed.info/id/72704001>
 2159 <http://snomed.info/id/271577005>
 2160 <http://snomed.info/id/75857000>

<p>y : R a = x</p>	<p>The set of concepts in y, which are the destination (ie attribute value) of a necessary relationship on a source concept in x with an attribute in a</p>	<p>< 91723000 Anatomical structure ²¹⁶¹ : R 363698007 Finding site ²¹⁶² = < 445945000 Infectious disease associated with acquired immune deficiency syndrome ²¹⁶³</p>	<p>280369009 Brain tissue structure ²¹⁶⁴ 39607008 Lung structure ²¹⁶⁵ 395939008 Clavulanic acid (substance) </p>
<p>x . a</p>	<p>The set of attribute values (ie destination concepts) of all necessary relationships on a source concept in x with an attribute in a</p>	<p>< 27658006 Product containing amoxicillin ²¹⁶⁶ . 127489000 Has active ingredient ²¹⁶⁷</p>	<p>372687004 Amoxicillin ²¹⁶⁸ 395939008 Clavulanic acid ²¹⁶⁹</p>

²¹⁶¹ <http://snomed.info/id/91723000>
²¹⁶² <http://snomed.info/id/363698007>
²¹⁶³ <http://snomed.info/id/445945000>
²¹⁶⁴ <http://snomed.info/id/280369009>
²¹⁶⁵ <http://snomed.info/id/39607008>
²¹⁶⁶ <http://snomed.info/id/27658006>
²¹⁶⁷ <http://snomed.info/id/127489000>
²¹⁶⁸ <http://snomed.info/id/372687004>
²¹⁶⁹ <http://snomed.info/id/395939008>

12 Appendix E - Reference Set Fields

In the SNOMED CT Release File Specification (<http://snomed.org/rfs>), SNOMED International specifies a set of [reference set types](#)²¹⁷⁰ with their own specific properties (e.g. an attribute value type reference set). Each reference set that is developed to conform to a specified type is defined as a subtype of the associated reference set type concept (e.g. [900000000000480006 | Attribute value type reference set](#)²¹⁷¹). All reference sets of a given type are populated with members using the same data structure - with the same set of field names in the same order. SNOMED International uses these reference set type data structures (as defined in the [Release File Specification](#)²¹⁷²) as the release file format for all reference sets of that type.

All [reference set type](#)²¹⁷³ concepts are a subtype of [900000000000455006 | Reference set](#)²¹⁷⁴, and have an associated set of reference set descriptors in the [Reference set descriptor reference set](#)²¹⁷⁵. Some reference set type concepts are organised under one or more reference set groups (e.g. [723564002 | MRCM reference set](#)²¹⁷⁶), which represent a group of reference set types (often with different data structures).

In the Expression Constraint Language (v2.0+) reference set field names are used to indicate which field values to return, and to filter reference set members based on specific field criteria. The first (non-metadata) field in every reference set (in order '0') must always be 'referencedComponentId'. For reference sets, which are a subtype of an international reference set type, the additional field names defined in the [SNOMED CT Release File Specification](#)²¹⁷⁷ must be used. In all other cases, the additional field names may use any latin-script alphabetic character (a-z or A-Z) defined by the owner of the corresponding reference set type concept. Owners of a reference set type are encouraged to explicitly document these field names, keep them unchanged and publish a machine readable representation of these (following the format used below). In the absence of this, the column name from the corresponding RF2 file (with all whitespace removed) will be used.

The international reference set types and their corresponding list of field names to be used in ECL v2.0+ are shown in the table below (for information only). A normative, computable representation of this table is attached below the table. Please note that this file may be extended by implementers with national or local reference set types.

Content Reference Set Types	
Reference Set Type	Field Names
446609009 Simple type reference set ²¹⁷⁸	referencedComponentId
733619002 Ordered component type reference set ²¹⁷⁹	referencedComponentId,order
900000000000480006 Attribute value type reference set ²¹⁸⁰	referencedComponentId,valueId

2170 <https://confluence.ihtsdotools.org/display/DOCRELFMT/5.2+Reference+Set+Types>

2171 <http://snomed.info/id/900000000000480006>

2172 <http://snomed.org/rfs>

2173 <https://confluence.ihtsdotools.org/display/DOCRELFMT/5.2+Reference+Set+Types>

2174 <http://snomed.info/id/900000000000455006>

2175 <http://snomed.info/id/900000000000456007>

2176 <http://snomed.info/id/723564002>

2177 <http://snomed.org/rfs>

2178 <http://snomed.info/id/446609009>

2179 <http://snomed.info/id/733619002>

2180 <http://snomed.info/id/900000000000480006>

900000000000521006 Association type reference set ²¹⁸¹	referencedComponentId,targetComponentId
733618005 Ordered association type reference set ²¹⁸²	referencedComponentId,targetComponentId,order
900000000000516008 Annotation type reference set ²¹⁸³	referencedComponentId,annotation
900000000000512005 Query specification type reference set ²¹⁸⁴	referencedComponentId,query
447258008 Ordered type reference set ²¹⁸⁵	referencedComponentId,order,linkedToId
762676003 OWL expression type reference set ²¹⁸⁶	referencedComponentId,owlExpression
1119417006 Postcoordinated expression type reference set ²¹⁸⁷	referencedComponentId,expression,substrate
Language Reference Set Types	
Reference Set Type	Field Names
900000000000506000 Language type reference set ²¹⁸⁸	referencedComponentId,acceptabilityId
Map Reference Set Types	
Reference Set Type	Field Names
900000000000496009 Simple map from SNOMED CT type reference set ²¹⁸⁹	referencedComponentId,mapTarget

2181 <http://snomed.info/id/900000000000521006>

2182 <http://snomed.info/id/733618005>

2183 <http://snomed.info/id/900000000000516008>

2184 <http://snomed.info/id/900000000000512005>

2185 <http://snomed.info/id/447258008>

2186 <http://snomed.info/id/762676003>

2187 <http://snomed.info/id/1119417006>

2188 <http://snomed.info/id/900000000000506000>

2189 <http://snomed.info/id/900000000000496009>

1187636009 Simple map to SNOMED CT type reference set ²¹⁹⁰	referencedComponentId,mapSource
447250001 Complex map from SNOMED CT type reference set ²¹⁹¹	referencedComponentId,mapGroup,mapPriority,mapRule,mapAdvice,mapTarget,correlationId
609331003 Extended map from SNOMED CT type reference set ²¹⁹²	referencedComponentId,mapGroup,mapPriority,mapRule,mapAdvice,mapTarget,correlationId,mapCategoryId
705111002 Map to SNOMED CT with correlation and origin type reference set ²¹⁹³	referencedComponentId,mapSource,attributeId,correlationId,contentOriginId
705109006 Code to expression type reference set type reference set ²¹⁹⁴	referencedComponentId,mapSource,expression,definitionStatusId,correlationId,contentOriginId
1193542003 Simple map with correlation from SNOMED CT type reference set ²¹⁹⁵	referencedComponentId,mapTarget,correlationId
1193543008 Simple map with correlation to SNOMED CT type reference set ²¹⁹⁶	referencedComponentId,mapSource,correlationId
1193544002 Simple map with correlation from SNOMED CT to SNOMED CT type reference set ²¹⁹⁷	referencedComponentId,mapTarget,correlationId
Metadata Reference Set Types	
Reference Set Type	Field Names
900000000000456007 Reference set descriptor type reference set ²¹⁹⁸	referencedComponentId,attributeDescription,attributeType,attributeOrder

2190 <http://snomed.info/id/1187636009>
 2191 <http://snomed.info/id/447250001>
 2192 <http://snomed.info/id/609331003>
 2193 <http://snomed.info/id/705111002>
 2194 <http://snomed.info/id/705109006>
 2195 <http://snomed.info/id/1193542003>
 2196 <http://snomed.info/id/1193543008>
 2197 <http://snomed.info/id/1193544002>
 2198 <http://snomed.info/id/900000000000456007>

900000000000534007 Module dependency type reference set ²¹⁹⁹	referencedComponentId,sourceEffectiveTime,targetEffectiveTime
900000000000538005 Description format type reference set ²²⁰⁰	referencedComponentId,descriptionFormat,descriptionLength
723589008 MRCM domain type reference set ²²⁰¹	referencedComponentId,domainConstraint,parentDomain,proximalPrimitiveConstraint,proximalPrimitiveRefinement,domainTemplateForPrecoordination,domainTemplateForPostcoordination,guideURL
723604009 MRCM attribute domain type reference set ²²⁰²	referencedComponentId,domainId,grouped,attributeCardinality,attributeInGroupCardinality,ruleStrengthId,contentTypeId
723592007 MRCM attribute range type reference set ²²⁰³	referencedComponentId,rangeConstraint,attributeRule,ruleStrengthId,contentTypeId
723563008 MRCM module scope type reference set ²²⁰⁴	referencedComponentId,mrcmRuleRefsetId




Field Nam...L 2.0.tsv

2199 <http://snomed.info/id/900000000000534007>
 2200 <http://snomed.info/id/900000000000538005>
 2201 <http://snomed.info/id/723589008>
 2202 <http://snomed.info/id/723604009>
 2203 <http://snomed.info/id/723592007>
 2204 <http://snomed.info/id/723563008>

13 References

1. *HL7 Version 3 Implementation Guide: Terminology – Using SNOMED CT in CDA R2 Models, Release 1*, HL7 5th DSTU Ballot, January 2014, http://wiki.hl7.org/index.php?title=File:V3_IG_SNOMED_R1_D5_2014JAN.docx
2. *SNOMED International APG Syntax Parsers*, IHTSDO, 2016, <http://apg.ihtsdotools.org/>
3. *NHS Logical Record Architecture for Health and Social Care*, UK Terminology Centre, November 2013, <https://isd.hscic.gov.uk/trud3/user/guest/group/0/pack/12>
4. *SNOMED CT Compositional Grammar – Specification and Guide*, IHTSDO, July 2015, <http://snomed.org/compgrammar>
5. *SNOMED International Glossary, Draft version July 2014*, <http://snomed.org/gl>
6. *SNOMED CT Languages Github Repository*, <https://github.com/IHTSDO/SNOMEDCT-Languages>
7. *SNOMED CT Starter Guide*, IHTSDO, February 2014, <http://snomed.org/sg>
8. *SNOMED CT Technical Implementation Guide*, IHTSDO, July 2014, <http://snomed.org/tig>

14 Previous Versions

 This page provides access to downloadable PDF copies of the current and previous versions of this document.
The most recent version should also be available via a link on the front page of the document.

15 Recent Updates

15.1 The most recently updated pages in this document are listed below

- [☰ Expression Constraint Language - Specification and Guide](#)(see page 7)
2 minutes ago • updated by Kai Kewley²²⁰⁵ • view change²²⁰⁶
- [☰ 1. Introduction](#)(see page 8)
20 minutes ago • updated by Kai Kewley²²⁰⁷ • view change²²⁰⁸
- [☰ 5.1 Brief Syntax \(Normative\)](#)(see page 21)
27 minutes ago • updated by Kai Kewley²²⁰⁹ • view change²²¹⁰
- [☰ 5.2 Long Syntax \(Informative\)](#)(see page 26)
29 minutes ago • updated by Kai Kewley²²¹¹ • view change²²¹²
- [☰ 6.1 Simple Expression Constraints](#)(see page 64)
2023-Aug-24 • updated by Kai Kewley²²¹³ • view change²²¹⁴
- [☰ 6.12 Top and Bottom](#)(see page 129)
2023-Jul-17 • updated by Kai Kewley²²¹⁵ • view change²²¹⁶
- [☰ Appendix D - ECL Quick Reference](#)(see page 196)
2023-Jul-12 • updated by Kai Kewley²²¹⁷ • view change²²¹⁸
- [☰ 1. Introduction](#)(see page 8)
2022-Sep-01 • updated by Linda Bird²²¹⁹ • view change²²²⁰
- [☰ 6.11 History Supplements](#)(see page 124)
2022-Aug-31 • updated by Anne Randorff Højen²²²¹ • view change²²²²
- [☰ Expression Constraint Language - Specification and Guide](#)(see page 7)
2022-Aug-24 • updated by Linda Bird²²²³ • view change²²²⁴
- [☰ Appendix C - Dialect Aliases](#)(see page 193)
2022-Aug-11 • updated by Linda Bird²²²⁵ • view change²²²⁶
- [☰ 6.8 Description Filters](#)(see page 105)

2205 <https://confluence.ihtsdotools.org/display/~kkewley>

2206 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=26840518&selectedPageVersions=50&selectedPageVersions=51>

2207 <https://confluence.ihtsdotools.org/display/~kkewley>

2208 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739382&selectedPageVersions=17&selectedPageVersions=18>

2209 <https://confluence.ihtsdotools.org/display/~kkewley>

2210 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739405&selectedPageVersions=17&selectedPageVersions=18>

2211 <https://confluence.ihtsdotools.org/display/~kkewley>

2212 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739406&selectedPageVersions=15&selectedPageVersions=16>

2213 <https://confluence.ihtsdotools.org/display/~kkewley>

2214 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739409&selectedPageVersions=13&selectedPageVersions=14>

2215 <https://confluence.ihtsdotools.org/display/~kkewley>

2216 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=212338972&selectedPageVersions=1&selectedPageVersions=2>

2217 <https://confluence.ihtsdotools.org/display/~kkewley>

2218 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=115883413&selectedPageVersions=5&selectedPageVersions=6>

2219 <https://confluence.ihtsdotools.org/display/~lbird>

2220 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739382&selectedPageVersions=16&selectedPageVersions=17>

2221 <https://confluence.ihtsdotools.org/display/~ahojen>



2222 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=142144136&selectedPageVersions=1&selectedPageVersions=2>

2223 <https://confluence.ihtsdotools.org/display/~lbird>

2224 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=26840518&selectedPageVersions=49&selectedPageVersions=50>

2225 <https://confluence.ihtsdotools.org/display/~lbird>

2226 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=115872393&selectedPageVersions=2&selectedPageVersions=3>

- 2022-Aug-11 • updated by Linda Bird²²²⁷ • view change²²²⁸
-  [6.10 Member Filters](#)(see page 121)
2022-Jun-27 • updated by Linda Bird²²²⁹ • view change²²³⁰
-  [5.3 Informative Comments](#)(see page 32)
2022-Jun-27 • updated by Linda Bird²²³¹ • view change²²³²
-  [5.2 Long Syntax \(Informative\)](#)(see page 26)
2022-Jun-27 • updated by Linda Bird²²³³ • view change²²³⁴
-  [5.1 Brief Syntax \(Normative\)](#)(see page 21)
2022-Jun-27 • updated by Linda Bird²²³⁵ • view change²²³⁶
-  [4.1 Details](#)(see page 19)
2022-Jun-07 • updated by Linda Bird²²³⁷ • view change²²³⁸
-  [4. Logical Model](#)(see page 18)
2022-Jun-07 • updated by Linda Bird²²³⁹ • view change²²⁴⁰
-  [3.2 Expression Constraint and Query Requirements](#)(see page 14)
2022-Jun-07 • updated by Linda Bird²²⁴¹ • view change²²⁴²
-  [6.9 Concept Filters](#)(see page 116)
2022-Apr-04 • updated by Linda Bird²²⁴³ • view change²²⁴⁴

2227 <https://confluence.ihtsdotools.org/display/~lbird>

2228 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=115872392&selectedPageVersions=4&selectedPageVersions=5>

2229 <https://confluence.ihtsdotools.org/display/~lbird>

2230 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=142144135&selectedPageVersions=3&selectedPageVersions=4>

2231 <https://confluence.ihtsdotools.org/display/~lbird>

2232 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739407&selectedPageVersions=18&selectedPageVersions=19>

2233 <https://confluence.ihtsdotools.org/display/~lbird>

2234 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739406&selectedPageVersions=14&selectedPageVersions=15>

2235 <https://confluence.ihtsdotools.org/display/~lbird>

2236 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739405&selectedPageVersions=16&selectedPageVersions=17>

2237 <https://confluence.ihtsdotools.org/display/~lbird>

2238 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739402&selectedPageVersions=12&selectedPageVersions=13>

2239 <https://confluence.ihtsdotools.org/display/~lbird>

2240 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739399&selectedPageVersions=10&selectedPageVersions=11>

2241 <https://confluence.ihtsdotools.org/display/~lbird>

2242 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=28739397&selectedPageVersions=12&selectedPageVersions=13>

2243 <https://confluence.ihtsdotools.org/display/~lbird>

2244 <https://confluence.ihtsdotools.org/pages/diffpagesbyversion.action?pagelId=134520492&selectedPageVersions=2&selectedPageVersions=3>