The Expression Constraint Language (ECL)

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.

The Expression Constraint Language - Specification and Guide has full details on ECL, including use cases, requirements, logical model, syntax specification, examples, and implementation considerations. Appendix D - ECL Quick Reference is reproduced below as a starting point for familiarisation, showing a syntax overview and some examples of each of the key syntax features. Other useful appendices for finding out more about its use include Appendix A – Examples Of Valid Expressions and Appendix B – Examples Of Invalid Expressions.

This section provides a quick reference to the key syntax features of the Expression Constraint Language.

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.

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 *excluding* the concept itself
<<	Descendant or self of	1.0	Retrieves all descendants (subtypes) of the specified concept *including* the concept itself
<!	Child of	1.1	Retrieves all children (immediate subtypes) of the specified concept *excluding* the concept itself
<<!	Child or self of	1.4	Retrieves all children (immediate subtypes) of the specified concept *including* the concept itself
>	Ancestor of	1.0	Retrieves all ancestors (supertypes) of the specified concept *excluding* the concept itself
>>	Ancestor or self of	1.0	Retrieves all ancestors (supertypes) of the specified concept *including* the concept itself
>!	Parent of	1.1	Retrieves all parents (immediate supertypes) of the specified concept *excluding* the concept itself
>>!	Parent or self of	1.4	Retrieves all parents (immediate supertypes) of the specified concept *including* 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
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
/* */	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

Examples

The following table provides some examples of each of the key syntax features of the Expression Constraint Language.

Notes:

1. 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 set| ,
   - '**a**' and '**b**' each represent either a single concept or a set of concepts that are a subtype of 410662002 | Concept model attribute| , 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|	128477000 |Abscess|
*****	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|	181216001 |Entire lung| 65784005 |Structure of fundus of eye|
< x	The set of all descendants (both direct and indirect) of x	< 73211009 |Diabetes mellitus| < 73211009 |Diabetes mellitus)|	46635009 |Diabetes mellitus type 1| 8801005 |Secondary diabetes mellitus|
<< x	The set of all descendants (both direct and indirect) of x, plus x itself	<< 73211009 |Diabetes mellitus|	73211009 |Diabetes mellitus| 46635009 |Diabetes mellitus type 1| 8801005 |Secondary diabetes mellitus|
<! x	The set of all immediate children of x	<! 362965005 |Disorder of body system|	49601007 |Disorder of cardiovascular system| 362969004 |Disorder of endocrine system|
<<! x	The set of all immediate children of x, plus x itself	<<! 362965005 |Disorder of body system|	362965005 |Disorder of body system| 49601007 |Disorder of cardiovascular system| 362969004 |Disorder of endocrine system|

> x	The set of all ancestors (both direct and indirect) of x	> 279420009 |Hematoma of skin|	106076001 |Skin finding| 297968009 |Bleeding skin|
>> x	The set of all ancestors (both direct and indirect) of x, plus x itself	>> 279420009 |Hematoma of skin|	106076001 |Skin finding| 297968009 |Bleeding skin| 279420009 |Hematoma of skin|
>! x	The set of all immediate parents of x	>! 22298006 |Myocardial infarction|	57809008 |Myocardial disease| 251061000 |Myocardial necrosis|
>>! x	The set of all immediate parents of x, plus x itself	>>! 22298006 |Myocardial infarction|	22298006 |Myocardial infarction| 57809008 |Myocardial disease| 251061000 |Myocardial necrosis|

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| AND < 87628006 |Bacterial infectious disease|	430395005 |Pneumonia caused by Gram negative bacteria| 154283005 |Pulmonary tuberculosis|
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| OR < 275504005 |Cyst of lung|	446543007 |Tuberculous abscess of lung| 87119009 |Congenital cystic lung|
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| MINUS < 75444003 |Fetal electrocardiogram|	447114004 |12 lead electrocardiogram during exercise| 252417001 |24 Hour electrocardiogram|

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| : << 370135005 |Pathological process| = << 441862004 |Infectious process|	698573001 |Infected hematoma| 444109008 |Infection of wound hematoma|

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| : << 363704007 |Procedure site| = << 69695003 |Stomach structure| , << 405815000 |Procedure device| = << 86174004 |Laparoscope|	708987006 |Laparoscopic total gastrectomy| 57922004 |Laparoscopic pyloromyotomy|
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)| :{ 405813007 |Procedure site - Direct| = << 10200004 |Liver structure| , 260686004 |Method| = << 129433002 |Inspection - action| }	773252007 |Diagnostic laparoscopy of liver| 20933000 |Endoscopy of liver|

Cardinality

Syntax	Evaluation Notes	Example	Example Expansion Concepts
x **: [min .. max]** a **=** y	The set of concepts in **x**, which have between **min** and **max** necessary relationships with an attribute in **a** and a value in **y**	< 373873005 |Pharmaceutical / biologic product| : [3..*] 127489000 |Has active ingredient| = < 105590001 |Substance|	786732006 |Product containing only brompheniramine and codeine and phenylpropanolamine| 787979009 |Product containing cyanocobalamin and folic acid and pyridoxine|
x **: [min .. max] {** a **=** y **}**	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**	< 404684003 |Clinical finding| : [2..3]{ 363698007 |Finding site| = *, 116676008 |Associated morphology| = 72704001 |Fracture| }	271577005 |Fracture of shaft of tibia and fibula| 75857000 |Fracture of radius AND ulna|

Reversed Attributes

Syntax	Evaluation Notes	Example	Example Expansion Concepts
y **: R** a **=** x	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**	< 91723000 |Anatomical structure| : R 363698007 |Finding site| = < 445945000 |Infectious disease associated with acquired immune deficiency syndrome|	280369009 |Brain tissue structure| 39607008 |Lung structure| 395939008 |Clavulanic acid (substance)|
x **.** a	The set of attribute values (ie destination concepts) of all necessary relationships on a source concept in **x** with an attribute in **a**	< 27658006 |Product containing amoxicillin| . 127489000 |Has active ingredient|	372687004 |Amoxicillin| 395939008 |Clavulanic acid|