# MVF Operations

Assume:

Term/Label is a string that may be the same as a MVF VocabularyEntry.term

Lookup( MVFEntryId ) : MVFEntry  
Returns a representation of the MVFEntry with the given Id, usually including terms/labels, a definition and relationships to other “neighbour” MVFEntrys.

Resolve( Term ) : List<MVFEntry>  
Returns a (prioritized) list of MVFEntrys that are evoked by the provided term – usually performed by string matching on the MVFEntry’s label, but can be more sophisticated and leverage e.g. NLP

## Extension for: user stories – Using Synonyms, Finding a model element

Resolve( Term ) : List<MVFEntry, PrecisionOfMatch, PreferredTerm>

PrecisionOfMatch tells the user how the source Term argument relates to the term(s) for the VocabularyEntry. The list of potential values to return for this should include: PreferredTerm, NonPreferredTerm, SubstringOfPreferredTerm, SubstringOfNonPreferredTerm, ContainsPreferredTerm, ContainsNonPreferredTerm, InDefinitionOfPreferredTerm, InDefinitionOfNonPreferredTerm, …

PreferredTerm is the term attribute in the VocabularyEntry in the active Vocabulary for the MVFEntry with the isPreferred attribute = True, if there are any such VocabularyEntries.

*We assume that the modeling tool will have a way to dereference the MVFEntry to an Element of the model that it corresponds with (aka reverse lookup of the Element->currentMVFEntry).*

GetTerm( MVFEntryId, Vocabulary\* ) : Map<String,Term>  
Retrieves the terms(s) in VocabularyEntries for a given MVFEntry in the given vocabularies.

GetCloud( MVFEntryId, Depth?, RelationshipType ) : Graph  
Returns the neighbors of a given MVFEntry, possibly filtered by the type of relationship between MVFEntrys, and up to a certain number of steps in the graph

[Not applicable ?

Translation( Model, TargetLanguage ) : Model  
Given a model expressed in a modelling language and using one or more lexicon (aka MVFDictionary), returns a new model that carries the same information content, but is expressed using a different modelling language and/or lexicon

???-translation A translation that preserves the modelling language (grammar) but changes the underlying lexicon

]

Localization( Model, TargetVocabulary ) : Model  
Given a model expressed in a modelling language and using one or more lexicon (aka MVFDictionary), returns a model expressed in the same language and lexicon, but using a different vocabulary for that lexicon

[added 26 April 2019]

SimpleCreateEntry (Term, Language): MVFEntryID

FullCreateEntry (Term, Language, optional externalReference, optional Definition): MVFEntryID

[added 10 May 2019]

GetEntries (Vocabulary[optional]) : List<VocabularyEntry, MVFEntryID>

Returns a list of all the Vocabulary entries (along with the ID for the corresponding MVFEntry) for the active Vocabulary or Vocabulary argument provided.