

LexGrid



The information and links on this page are no longer being updated and are provided for reference purposes only.

What is the LexGrid?

LexGrid (Lexical Grid) provides support for a distributed network of lexical resources such as terminologies and ontologies via standards-based tools, storage formats, and access/update mechanisms.

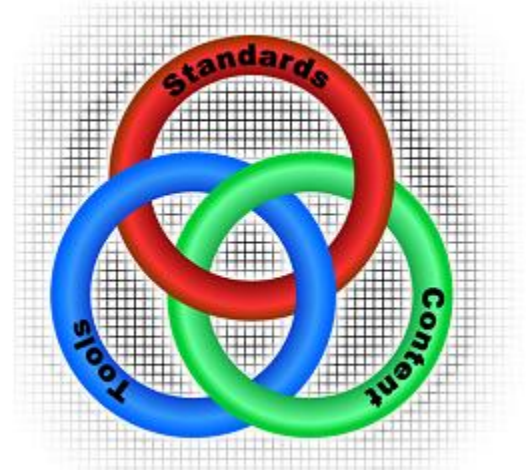
The *Lexical Grid Vision* is for a distributed network of terminological resources.

Currently, there are many terminologies and ontologies in existence. Just about every terminology has its own format, its own set of tools, and its own update mechanisms. The only thing that most of these pieces have in common with each other is their incompatibility. This makes it very hard to use these resources to their full potential. We have designed the Lexical Grid as a way to bridge terminologies and ontologies with a common set of tools, formats and update mechanisms. The Lexical Grid is:

- accessible through a set of common APIs
- joined through shared indices
- online accessible
- downloadable
- loosely coupled
- locally extendable
- globally revised
- available in web-space on web-time
- cross-linked

The realization of this vision requires three interlocking components, which are:

- Standards - access methods and formats need to be published and openly available
- Tools - standards based tools must be readily available
- Content - commonly used terminologies have to be available for access and download



Documentation and Training

- [LexGrid Documentation](#)
- [LexGrid Presentations](#)

Common Terminology Services

The Common Terminology Services (CTS) specification was developed as an alternative to a common data structure. We have published the HL7 CTS specification and a reference implementation of the specification.

The HL7 Common Terminology Services (HL7 CTS) is an Application Programming Interface (API) specification that is intended to describe the basic functionality that will be needed by HL7 Version 3 software implementations to query and access terminological content. It is specified as an API rather than a set of data structures to enable a wide variety of terminological content to be integrated within the HL7 Version 3 messaging framework without the need for significant migration or rewrite.

Instead of specifying what an external terminology must look like, HL7 has chosen to identify the common functional characteristics that an external terminology must be able to provide. As an example, an HL7 compliant terminology service will need to be able to determine whether a given concept code is valid within the particular resource. Instead of describing a table keyed by the resource identifier and concept code, the CTS specification describes an Application Programming Interface (API) call that takes a resource identifier and concept code as input and returns a true/false value. Each terminology developer is free to implement this API call in whatever way is most appropriate for them.

There are two layers between HL7 Version 3 message processing applications and the target vocabularies. The upper layer, the Message API communicates with in terms of vocabulary domains, realms, coded attributes and other artifacts of the RIM and HL7 messaging model. The lower layer, the Vocabulary API communicates in terms of coding system, concept codes, designations, and other vocabulary related entities.

Lexicon Query Services

Original work on the LQS evolved into the LexGrid project. The implementation of the LQS has fallen behind on changes that have made to the ldap database as LexGrid evolved, so there is no longer a download for it. In the future, there is the hope to provide an implementation of the LQS here that runs on LexGrid databases.

UMLS

The UMLS is a distribution of terminologies by the NLM. There is a well-documented format in which you can download the terminologies. The download for LexGrid provides tooling that allows you to import the UMLS formats directly into the LexGrid formats, so that you can use all of the tooling on any terminology from the UMLS.

Content and Storage

Many terminology storage formats, both standard and non-standard, are currently in use. The LexGrid Model and supporting software provide the ability to convert information from several common file formats, thereby making the content accessible through standardized API implementations and tooling.

Some formats are OWL (Web Ontology Language), RRF (Rich Release Format), OBO (Open Biomedical Ontologies) and LexGrid XML.



LexGrid conversion software is capable of importing these supported vocabulary formats into relational databases (SQL-based access)

LexGrid-based projects can now access an SQL database schema consistent with the LexGrid model. This database schema is currently available in two different versions, LexGrid SQL and SQL Lite. The "Lite" version is designed for small scale experimentation on an Access based database. The regular SQL schema can be fully scaled to large vocabularies on a variety of commercial and open source database management systems.

Information Models

The starting point for the LexGrid Data Model was the [Object Management Group \(OMG\)](#)  Terminology Query Services specification, with some revisions and many simplifications.

The OMG Model

- Lexicon Query Service Specification from OMG - [A pdf file of the model](#) 
- Lexicon Query Service Specification IDL from OMG - [The idl of the model](#) 

Supported Standards

Many terminology standards have been created by various organizations.

- CTS (Common Terminology Services) and LQS (Lexicon Query Services) are both APIs for accessing content that is stored in the LexGrid Model.
- UMLS (Unified Medical Language System) is a standard distribution format from the National Library of Medicine. RRF or Rich Release Format is an updated file format used in recent releases of the NCI MetaThesaurus.
- Terminology Query Services specification, with some revisions and many simplifications



Note

The 2005 Model of LexGrid is no longer supported by LexBIG.

- [LexGrid Model and Schema](#)

Links

- [EVS Technical guide](#)
- For more information on LexEVS, including access to the latest LexEVS documentation and files, see the [LexEVS](#) wiki.