KNOWLEDGE REPRESENTATION AND ONTOLOGIES FOR LIPIDS AND LIPIDOMICS

LOW HONG SANG

NATIONAL UNIVERSITY OF SINGAPORE

2009

Knowledge representation and ontologies for lipids and lipidomics

Low Hong Sang
(B.sc.(Hons), NUS)

Thesis
Submitted for the degree of Master of Science

Department of Biochemistry
Yong Loo Lin School of Medicine
National University of Singapore

Acknowledgements

First of all, I would like to thank the National University of Singapore and the Ministry of Education, Singapore for providing me with the opportunity as well as the financial support to pursue my aspiration for a post-graduate study in scientific research.

My deepest gratitude goes to my supervisors, Associate Professor Markus R. Wenk and Professor Wong Limsoon for their guidance and the invaluable advice that they provided me during the course of my graduate study. I am particularly thankful of the patience, graciousness and affirmation that they have shown to me.

I would also like to extend my sincere gratitude to our collaborator, namely Dr. Christopher James Oliver Baker from the Institute of Infocomm Research, the Agency for Science, Technology and Research (A*STAR) for his guidance and support. He has been instrumental in providing guidance and the necessary IT resources to enable the translation of my research work into sound application that can been applied in the field of lipidomics. I am particularly thankful to him for his patience with my shortcomings and for many of his constructive suggestions throughout the duration of my research.

I also like thank my friends from the lab for their support and friendship during the course of my research, specifically during certain critical juncture of my work.

Lastly, I would like to thank my family, especially my parents. They have always been there for me. I like to thank my church too for their prayers and for upholding me in matters of faith. Together, they have been the greatest source of strength and support in my work and my life.

Table of Contents

List of Publications

Baker CJO, Kanagasabai R, Ang WT, Veeramani A, **Low H-S**, Wenk MR: Towards ontology-driven navigation of the lipid *bibliosphere. BMC Bioinformatics*. 2008, 9(Suppl 1):S5.

Oral Presentation

Low H-S., Baker CJO., Garcia A., Wenk MR.
An OWL-DL Ontology for Classification of Lipids.
International Conference on Biomedical Ontology(ICBO2009), Buffalo, New York, USA, July 24-26 2009.

Kanagasabai R., Narasimhan K., **Low H-S.**, Ang WT., Wenk MR., Choolani MA., Baker CJO. Mining the Lipidome of Ovarian Cancer. AMIA Summit on Translational Bioinformatics, Annual Medical Informatics Association, San Francisco, United States of America. March 15-17 2009.

Kanagasabai R., **Low H-S.**, Ang WT., Wenk MR., Baker CJO.
Ontology-Centric Navigation of Pathway Information Mined from Text.
The 11th Annual Bio-Ontologies Meeting, co-located with ISMB 2008, Toronto Canada, July 20th 2008.

Kanagasabai R*., **Low H-S*.**, Ang WT., Veeramani A., Wenk MR., Baker CJO. Literature-driven, Ontology-centric Knowledge Navigation for Lipidomics. In Nixon, L., Cuel, R., Bergamini C., eds.: *CEUR Workshop* Proceedings of the Workshop on First Industrial Results of Semantic Technologies (FIRST 07), Busan, Korea, November 11th 2007.

Baker CJO., Kanagasabai R., Ang WT., Veeramani A., **Low H-S.**, Wenk MR. Towards Ontology-Driven Navigation of the Lipid *Bibliosphere.*
International Conference on Bioinfomatics 2007 (InCoB 2007), HKUST, Hong Kong SAR, People Republic of China, August 28th 2007.

Summary

In this thesis, semantic web technologies such as OWL ontology are explored for the purpose of representing knowledge from the field of lipid research.

The first chapter provides a concise background for the field of lipid research, including the emerging area of lipidomics and some of the challenges faced by lipid scientists. The same chapter also provides background on the development of the specific semantic web technologies, followed by a discussion of how these technologies can address some of the challenges identified in lipid research.

In the second chapter, the methodology employed to develop ontologies is described. Since there are no standardized methodologies for development of ontologies, the general development life cycle and broad principles that are adhered during the development of ontologies for lipids are discussed extensively in this chapter.

The third chapter begins with the description of the first Lipid Ontology, namely Lipid Ontology 1.0. Lipid Ontology 1.0 is a baseline ontology developed to support navigation of information through Knowlegator. Knowlegator is a knowledge visualization tool developed by I2R, A*STAR that enables visualization, navigation and query of knowledge captured in OWL-DL ontologies. This is followed the description of Lipid Ontology Reference and Lipid Ontology Ov.

The fourth chapter deals with the description of the Lipid Classification Ontology (LiCO) and Lipid Entity Representation Ontology (LERO). These ontologies are domain oriented ontologies that are built for the purpose of representing knowledge formally in OWL-DL and sharing the knowledge with the wider community-the OBO Foundry.

The fifth chapter describes an application scenario where the Lipid Ontology is employed in conjunction with a prototype ontology centric content delivery platform(Knowlegator) developed by Institute of Infocomm Research, A*STAR to facilitate knowledge discovery for lipidomics scientists. A preliminary performance analysis of the platform is conducted and the platform is subsequently used to facilitate navigation of pathways. Lastly, the prototype platform is employed to assess the lipidome of ovarian cancer in the literature.

The final chapter contains the concluding remarks for this thesis. A brief summary of the ontologies built during the course of the research is given. The adequacy of OWL-DL ontologies as medium of knowledge representation for biological knowledge is re-iterated, specifically for the use case in the knowledge domain of lipids and lipidomics and can be developed into an effective ontology centric application under a platform that is tightly integrated to other technological components of semantic web.

List of Tables

List of Figures

Chapter I: Background

1) Lipid

Lipids are naturally occurring, hydrophobic compounds that are readily soluble in organic solvents such as hydrocarbons, chloroform, benzene, ethers and alcohols. A more scientific definition classifies lipids as fatty acids and their derivatives, and substances related biosynthetically or functionally to these compounds [1]. This definition enables scientist to include compounds that are related closely to fatty acid derivatives such as prostanoids, aliphatic ethers, alcohols or cholesterols through biosynthetic pathways or by their biochemical or functional properties.

LIPID MAPS consortium introduced a new systematic nomenclature for lipids in 2004. The consortium defined lipids as hydrophobic or amphipathic small molecules that may originate entirely or in part by carbanion-based condensations of thioesters and/or by carbocation-based condensations of isoprene units [2]. Under this new nomenclature, lipids are divided into 8 major categories, namely the fatty acyls, glycerophospholipids, glycerolipids, sphingolipids, sacharrolipids, sterol lipids, prenol lipids and the polyketides.

1.1) Importance of Lipids in Biology or Lipid Biochemistry, Functions in Biology

Lipids and their metabolites play very important biological and cellular functions in living organisms. Lipids are known to be a source of stored metabolic energy and an important component in the formation of structural elements such as membranes, lipid bodies, transport vesicles in a cell. These structural elements enable subcellular partitioning necessary for cellular function and create barriers for diffusion of ions and

metabolites so that membrane potentials needed for basic cellular electrophysiological function can be maintained. In addition to that, lipid-based structural elements such as cell membranes or lipid bodies provide a liquid crystal bilayer medium that facilitates the assembly of supramolecular protein complexes required for the transmission of electrical and chemical signals in a cellular system. [3]

Lipids play important roles in signaling events of the cell. Lipids are synthesized, transported and recognized through coordinated events involving numerous enzymes, proteins and receptors. Moreover, lipids are important precursor molecules that act as endogenous reservoirs for the biosynthesis of lipid secondary messenger and other biologically relevant molecules. Many lipids are bio-active molecules. These lipids, such as menaquinones, vitamin E, prostaglandins, phosphatidylinositol phosphate function as important coenzymes, antioxidants, intra- and extra-cellular messengers in cellular processes. [4]

1.2) Lipid and Important Diseases

Since lipids are crucial to the biological function of cells and tissues, it is without surprise that many diseases such as artherosclerosis, cancer, Alzheimer's syndrome, tuberculosis and dengue viral infection are found associated to abnormality in the lipid metabolism. However, the mechanisms through which lipids affect these diseases are still not known. Assessment of the lipidome is the first step towards understanding the mechanism of these diseases and we have applied the bioinformatics approach described in this thesis to assess the lipidome of cancer, specifically ovarian cancer.

1.2.1) Cancer

Cancer is a multi factorial disease caused by genetic mutations of oncogenes or tumor suppressor genes that alter downstream signaling transduction pathways, protein interaction networks and metabolic processes in such a way that it produces apoptotic suppressing, rapid proliferating and invasive metastatic cell phenotype in the affected cells. It is increasing evident that lipid metabolites play important roles in cancer pathogenesis.

One of the lipids implicated in cancer is cardiolipin. A recent publication had shown that abnormal cardiolipin levels are behind the irreversible respiratory injury in tumors and link mitochondrial lipid defects to Warburg theory of cancer [5]. The Warburg effect is the first metabolic cause established by Otto Warburg as the primary cause of cancer [5, 6]. The Warburg effect suggests that cancer is caused by irreversible injury to cellular respiration where the affected cells become dependent on fermentation or glycolytic energy in order to compensate for lost energy from respiration. In a similar light, evidence had shown that increased de novo fatty acid synthesis, a metabolic pathway functionally related to glycolytic pathway also accompanies cancer pathogenesis [7].

Other examples of lipid implicated in cancer are sphingosine 1- phosphate (S1P) and ether lipid. The level of sphingosine 1- phosphate can determine whether a cell would undergo apoptosis or proliferation. The accumulation of S1P and subsequent activation of S1P receptors cause cells to develop cancerous phenotypes such as cell migration, cell proliferation, inhibition of apoptosis, upregulation of adhesion molecules [8].

Ether lipids such as 2 acetyl monoalkylglycerols are intermediates that can be hydrolyzed by KIAA1363, an uncharacterized enzyme highly elevated in aggressive cancer cells in an ether lipid signaling network. Inactivation of KIAA1363 disrupts the ether lipid metabolism required by the cancer cells to undergo cell migration and tumor growth [9].

1.3) Lipidomics

Lipidomics is a system level analysis that involves full characterization of lipid molecular species and their biological roles with respect to the expression of proteins involved in lipid metabolism and function, including gene regulation [10]. In Lipidomics, levels and dynamic changes of lipids and lipid-derived mediators in cells or subcellular compartments are identified and measured quantitatively in the form of lipid profiles. These lipid profiles are readouts from mass spectrometer and could be further analyzed to yield biological insights.

A mass spectrometer is an instrument capable of measuring the mass of molecules that have an electrical charge. A typical mass spectrometric analysis consists of 3 separate events: analyte ionization, mass-dependent ion separation and ion detection.

A major limitation of mass spectrometry used for lipidomics is the phenomena of suppression of ionization. This limitation can be overcome with the use of chromatographic techniques such as liquid chromatography (LC), thin-layer chromatography (TLC), gas chromatography (GC) or high-performance liquid chromatography (HPLC). Lipid mixtures can be separated by chromatography first

before being fed into the mass spectrometer for analysis. MS analyses apply to lipidomics are often conducted in conjunction with an upfront chromatography. An example of such application is Multiple Reaction Monitoring (MRM) analysis.

1.3.1) Lipidomics and System Biology

To study the functions of lipids, profiling of lipids using a combination of chromatographic and spectrometric techniques is not sufficient. Other techniques such as immobilized lipid assays, lipid-protein complex antibody assays, florescence imaging techniques have been applied in tandem with lipidomic experiments to study lipid-lipid, lipid-protein interactions as well the localisation of lipids. As such, lipidomics generates a large volume of heterogeneous experimental data. The analysis of lipidomics data would require a scientifically consistent integration of chemical and biochemical data from different technologies, with different formats and at various levels of granularity.

System biology is the computational integration of genomic, transcriptomic, proteomic and metabolomic data with the purpose of understanding the molecular mechanisms that undergirds a cell or a living organism [11]. Lipidomics studies the lipidome, which is a sub-fraction of the complete metabolome of a living being and complements other approaches in system biology.

Advances in lipidomics methods, coupled with improved data processing software solutions, demand the development of comprehensive lipid libraries to allow integration

of data from other approaches of system biology in addition to system-level identification, discovery and study of lipids [12].

In this light, Yetukuri *et al*. highlighted 3 challenges; a database system is needed to efficiently link the high volume of data from high throughput lipidomics experiments generated from the analytical platform [12]. Secondly, there is not one database that covers all possible lipids found in the diversity of organisms, tissue types and cell types. A mechanism is needed to integrate all lipid databases together in order to facilitate identification as well as discovery of new lipid species from all available data [12]. Lastly, the lipid information needs to be connected to other areas of biological organization at the correct level of granularity as most biological databases that describe proteins or pathways are often limited to the level of generic lipid classes instead the level of details produced from lipid MS experiments [12].

1.4) Lipid Databases

An interesting area of development is the emergence of many lipid databases (see Table 1). 2 types of databases are relevant to lipids. The first type is database that acts as repository of data for chemical compounds (including non-lipid data). Notable examples for this group of databases are PubChem, CHEBI and KEGG COMPOUND. The second type of databases is the lipid-dedicated databases. They include databases such as LIPIDAT, Lipid Bank and LIPID MAPS's LMSD. With the exception of LMSD, most of them are just repositories of lipid information. While each of these databases has lipids that are unique to their collections, large subsets of lipid information in these databases

overlap. In addition to that, none of these databases uses the same classification for lipids (with the exceptions of KEGG COMPOUND and LMSD). A lipid has many types of heterogenous information associated to it. However, most of these databases are not designed to handle all the heterogeneous information of lipids and are at most compatible to represent some but not all types of data. Lastly, some lipid databases do not make distinction between representations of lipid at different level of granularity. For example, LMSD has many lipid records that refer to a class of lipid rather than a single individual lipid molecule at the same taxonomic level whereas LipidBank and LIPIDAT have records for lipid mixtures at the same level as records of lipid.

Database	Brief description
LIPID MAPS Structure Database (LMSD)	10,789 lipid records; dedicated to lipidomics; provides lipid informatics tools and systematic nomenclature for lipids
	http://www.lipidmaps.org/
Lipid Bank	7009 lipid records; provides literature references for every lipid records; provides lipid profiles for some lipids; contain records for lipoproteins and glycolipids
	http://lipidbank.jp/
LIPIDAT	20,784 lipid records; provides physical and chemical properties of lipids
	http://www.lipidat.ul.ie/
KEGG COMPOUND	metabolome informatics resource; 1298 lipid records; provides connectivity to other KEGG databases
	http://www.genome.jp/kegg/compound/
ChEBI	Chemical database; provides ontological support, InCHiKey and SMILES
	http://www.ebi.ac.uk/chebi/
PubChem	Chemical database combining all records from all known chemical databases inclusive of lipid databases
	http://pubchem.ncbi.nlm.nih.gov/

Table 1: URL and description of services provided in known publicly accessible lipid and chemical databases

1.4.1) Pubchem, An Integrative Knowledgebase?

PubChem is an attempt by NCBI to set up a central repository for all chemical compounds, inclusive of lipids. It collates lipid records from all known lipid databases. It is organized as three linked databases within the NCBI's Entrez information retrieval system and provides a fast chemical structure similarity search tool. Unfortunately, it does not have a unified classification that could integrate all lipid records in a scientifically sensible manner; neither does it provide a universal syntactic format that could integrate the heterogeneous lipid data in a comprehensive manner. As a result of that, PubChem is filled with many redundant records of the same lipid.

1.5) Importance of Nomenclature/Systematic Classification for Lipidomics/Lipid System Biology

The collection of lipid data via a "system biology" approach requires the development of a comprehensive classification, nomenclature and chemical representation system capable of representing diverse classes of lipids that exist in nature.

Lipids, unlike their protein counterparts, do not have a systematic classification and nomenclature that is widely adopted by biomedical research community.

To address this problem, IUPAC-IUBMB proposed a systematic nomenclature for lipids in 1976 [14]. However, the proposed classification system is unwieldy, complicated and had often been applied erroneously by scientists [2]. This led to the generation of many unscientific lipid names. In addition to that, due to the lack of adoption, the IUPAC naming scheme was not extended and consequently could not adequately represent the

large number of novel lipid classes that have been discovered in the last 3 decades and because of that, this classification has become obsolete with respect to the current state of the arts in lipid research such as lipidomics.

The lack of a consistent nomenclature that is universally accepted led different lipid research groups to develop classification systems of lipids that are usually very narrow and only sound for a restricted category of lipid. As a result, a lipid molecule can be classified in many different ways, and be placed under different types of classification hierarchy. These classification systems are not mutually consistent and hence, create a lot of problems for systematic analysis of lipids. For example, Prostaglandin A1 is a lipid that can be found in 2 lipid databases, namely LipidBank and LMSD (see Table 2). Both databases name lipids differently. The lipid is given the systematic name of 9-oxo-15S-hydroxy-10Z,13E-prostadienoic acid by LMSD while 2 other systematic names can be found in LipidBank(7-[2(R)-(3(S)-Hydroxy-1(E)-octenyl)-5-oxo-3-cyclopenten-1(R)-yl]heptanoic acid & (8R,12S,13E,15S)-15-Hydroxy-9-oxo-10,13-prostadienoic acid). In addition to that, the same lipid is associated to 3 more different names in KEGG COMPOUND database, namely (13E)-(15S)-15-Hydroxy-9-oxoprosta-10,13-dienoate, Prostaglandin A1, PGA1. In short, a single lipid can be associated with a plethora of synonyms. This especially also true for the legacy literature resources as scientific publications are filled with broad synonyms, trivial names and instances of synonyms not linked to any systematic nomenclature or any chemically sound classification.

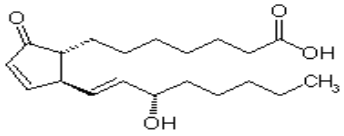
Prostaglandin A1 	Database	Identifiers
	LMSD	LMFA03010005
	LipidBank	XPR1000
	KEGG Compound	C04685

Table 2: Structure of Prostaglandin A1 and corresponding records in LMSD, LipidBank and KEGG COMPOUND database

LIPID MAPS consortium attempted to resolve this problem by developing a scientifically sound and comprehensive classification, nomenclature, and chemical representation system that incorporates a consistent nomenclature that followed the IUPAC nomenclature closely and yet is able to include new lipids that have yet to be systematically named by IUPAC [2]. This classification scheme organizes lipids into well-defined categories that cover the major domains of living creatures, namely, the archaea, eukaryotes and prokaryotes as well as the synthetic domain. This is a significant contribution to lipid research. Despite that, the uptake by the scientific community has been gradual. Many research groups are still using synonyms or old names that they are familiar with despite the introduction of a new nomenclature. Furthermore, literature resources on lipid research are steeped with instances of lipid synonyms that do not follow the new nomenclature. While the nomenclature is scientifically robust, it is still based on a cumbersome naming scheme. Under LIPIDMAPS scheme, for example, a derivative of vitamin D2 was given a systematic but very bulky and un-intuitive name of (5Z,7E,22E)-(3S)-26,26,26,27,27,27-hexafluoro-9,10-seco-5,7,10(19),22-ergostatetraene-3,25-diol.

Therefore, the naming of new lipids requires trained experts; and subsequent acceptance of new names by members of the lipid community is slow. In parallel, lipidomics technology has enabled the discovery of many novel lipids in a rate that is many folds

faster than the acceptance of new lipid names into the nomenclature. Consequently, many novel lipids such as mycolic acids do not have a LIPID MAPS systematic name.

1.5.1) Description Logics Based Definition of Lipids

While LIPID MAPS's effort contributes to the lipid research community by providing a central repository of lipids, where lipid classes are categorized extensively by is-a relationships [15], definitions for classes of lipids in LMSD are still implicit and are often dependent on a chemical diagram in the form a molecular graphic file that can only be accurately classified by a trained lipid expert. There is no rigorous definition for a specific lipid class that is independent of a graphical diagram. In addition to that, classes of lipids define in LIPID MAPS also suffer from several inadequacies. They are as follows:

a) Lack of explicit textual definitions
b) Lack of representative instance of lipid for a specific class of lipid(an empty class without data records) and hence, not even a graphical definition is available.
   An example of this is the sphingolipid class "Other Acidic glycosphingolipids" (SP0600)
c) The use of arbitrarily named lipid class to contain non-conventional lipid instances.
   An example is "Sphingoid base homologs and variants" and "Sphingoid base analogs"

d) Class name is not compatible with the lipid instances assigned to it where the class name is too generic or the class name do not adequately describe the lipid instances assigned to the class
e) Instances of lipid under a class share very little structural similarities

A rigorous definition would involve a minimal necessary and sufficient declaration in description logics that could adequately describe a lipid without a molecular structure diagram. With description logics, we could define a lipid such as an epoxy fatty acid as a molecule that must at least have a carboxylic acid group and an epoxy group. Taking this further, we define an epoxy fatty acid as a lipid that can only have epoxy group and carboxylic acid group. As a consequence, any molecules that have functional groups other than epoxy group and carboxylic acid group cannot be considered as an epoxy fatty acid. A graphical definition is not flexible, nor is it extensible. Changes in such a definition would mean redrawing a completely new chemical diagram. Subsequently, communicating, storing and transferring of such structural definition in the current format are inefficient as this system places a lot of emphasis on trained or domain expert of the field.

There is therefore a need for lipids to be defined in a manner that is systematic (following LIPID MAPS hierarchical structure) and semantically explicit.

2) Knowledge Representation in Semantic Web

Semantic web is an extension of the current WWW where information is given well-defined meaning so that it provides a computer with structured collections of information and sets of inference rules to do automated reasoning. While computers can parse web pages for layout and routine processing effectively, computers cannot reliably understand the semantics of a web page. With semantic web, computers are supplied with additional metadata associated to every web page so that computers can comprehend semantic documents and understand the meanings of terminology used in every document within its supposed frame of context [16]. Knowledge representation in semantic web often takes the form of an inter-connected network where pieces of structured and unstructured information are linked into commonly shared description logics ontologies.

2.1) 3 Major Components of Semantic Web Technology

Semantic Web knowledge representation is composed of 3 technological components. They are eXtensible Markup Language (XML), Resource Description Framework (RDF) and Web Ontology Language (OWL) [16]. XML allows users to create custom tags to annotate web pages or sections of text in a page. In short, XML allows users to add arbitrary structure into a web document. RDF expresses meaning by encoding semantics into sets of triples. A triple is similar to the subject, verb and object of an elementary sentence and can be written using XML tags. An RDF document makes assertion that a particular thing (subject) has properties (object). Every subject, verb and object expressed in RDF has a Universal Resource Identifier (URI). The use of URI ensures that concepts

(subject, object, verb) are not just words in a documents but are associated to the unique definition or contextual meaning on the web. This allows a computer to resolve the meaning of a word that means differently in different contexts. RDF uses XML to define a foundation for processing metadata and to provide a standard metadata structure for both the web and the enterprise. In addition to XML and RDF, semantic web technology also depends a lot on collections of information called ontologies. An ontology differs from an XML schema in that it is a knowledge representation, instead of being a message format. Ontology can be encoded using OWL. OWL is a semantic markup language for publishing and sharing of ontologies on the web that builds upon RDF by assigning a specific meaning to a certain RDF triples. (see Table 3)

Components of semantic web	Description	Compatible query language
XML	Structured Documents	XPath, XQuery
RDF	Data models for objects	RDQL, RQL, Versa, Squish
OWL	Semantic data models with complex relationships	nRQL, OWL-QL, JENA

Table 3: Basic components of semantic web and compatible query languages

2.2) Ontology

The word "Ontology" is a term used in the study of philosophy. It describes a theory about the nature of existence [17]. The term has since been co-opted by computer scientist as a technical term to describe an engineering artifact designed for a purpose, which is to enable the modeling and representation of knowledge of a specific domain for an information system or application.

2.2.1) Ontology in Computer Science/Information Science

In the field of computer science, an ontology is defined as a formal specification of shared conceptualization of a certain field of knowledge and provides a common vocabulary for an area of interest where the meaning of the terms and the relations between them are defined with different levels of formality [18]. Simply put, an ontology is a document or file that formally defines the relationships (verbs) among the terms (object and subject) required for an application or a knowledge domain. It defines a set of representational primitives with which to model a domain of knowledge. An ontology is a semantic level data model as it is implemented by languages such as OWL that are closer in expressive power to logical formalisms such as First-Order Logic. This allows the ontology designer to state semantic constraints.

2.2.2) Ontology as a Scientific Discipline

Science is characterized by the existence of a consensus core of established results being repeatedly challenge by multiple hypotheses that are less mature and grows cumulatively as the consensus core of the discipline absorbs hypotheses that were immature at first but could withstood attempts to refute them empirically [19]. Ontology provides a coherent and interoperable suite of controlled structured representations of entities and relations to describe, at any given stage, the consensus core knowledge of a scientific discipline. In addition to that, it also provides a basis for accumulation of scientific data that would lead to development of mature, if not new scientific theory [19]. Secondly, similarly to empirical science, ontology is required to be tested empirically and possess the identical progressive maturation pattern seen in the development of scientific theories [19]. This is

achieved when biologists use ontologies to aggressively annotate experimental results, including those already reported in literature [19]. Inversely, the annotation process generates corrections as well as new content to be added to these ontologies. This process is typical of an empirical scientific growth and generates improved annotation resource for future work. [19]

2.2.3) Uses of Ontologies

- Ontology can be treated as a source of words, synonyms, annotation of terms and terminologies. This resource allows a knowledge domain to be modeled for a logical consistent system such as a database system or a web service.
- Ontology provides a syntactic and semantic consistent representation for multiple data resources. Therefore, it can be used to integrate heterogenous data from multiple databases or resources and enables interoperability among these disparate systems.
- Ontology can also be considered as a specifying interface to independent, knowledge-based services, where the specification takes the form of definitions of representational vocabulary that provides meanings for the vocabulary and formal constraint on its coherent use. In short, Ontology specifies a vocabulary with which to make assertions, which may be inputs or outputs of knowledge agents, and provides a language for communicating with a query agent.
- Ontology provides a representational mechanism that can be used to instantiate domain models in knowledge bases, make queries to knowledge-based services and represent the results of calling such services. In this context, ontology is used in semantic web to specify standard conceptual vocabularies in order to exchange data

among systems, provide services for answering queries, publish reusable knowledge bases and offer services to facilitate interoperability across multiple, heterogenous systems , ontologies and databases.

2.3) Web Ontology Language (OWL)

OWL is a standard ontology language developed from World Wide Web Consortium (W3C) [20, 51]. OWL is derived from DAML+OIL Web Ontology language and has a rich sets of operators such as and, or, negation. OWL can be used to describe and define concepts, including defining complex concepts based on the simpler concepts.

Furthermore, an OWL ontology is based on a logical model that allows a reasoner to check whether or not all the statements and definitions in the ontology are mutually consistent and can also recognize which concepts fit under which definitions.

OWL ontology can be divided into 3 classes of sub language, namely, OWL-Lite, OWL-DL and OWL-Full. These sub languages differ from one another in the degree of their expressiveness.

- OWL-Lite is the least expressive language of the OWL family. It is intended to be used in situations where only a simple class hierarchy and simple constraints are needed [20].
- OWL-DL is an extension from OWL-Lite. It is more expressive because it is based on description logics. Description logics are a mathematical theory that describes a decidable fragment of First-Order Logic and are therefore amenable to automated reasoning [20].

- OWL-Full is the most expressive language of the OWL family. It is used in situation where the need for high level of expressiveness is more important than the need for decidability or computational completeness. An OWL-Full ontology cannot be reasoned over [20].

2.3.1) Components of OWL

OWL ontologies are composed of 3 components (see Figure 1). They are individuals, classes and properties. Individuals or instances represent objects in the domain of interests. Individuals are encapsulated in OWL classes. OWL classes or concepts are sets that contain individuals. They are described using formal descriptions that state precisely the requirements for the membership of the class. There are 2 types of classes, namely primitive class or defined class. A primitive class is a class with necessary conditions as its membership requirement, whereas a defined class is a class with necessary and sufficient conditions as its membership requirement. Properties are roles or attributes assign to individuals. There are 3 types of properties, namely object properties, datatype properties and annotation properties. Object properties are relationships that connect 2 individuals together. Within the framework of OWL-DL, object properties can be asserted in 4 ways, namely inverse, transitive, symmetric and functional properties.

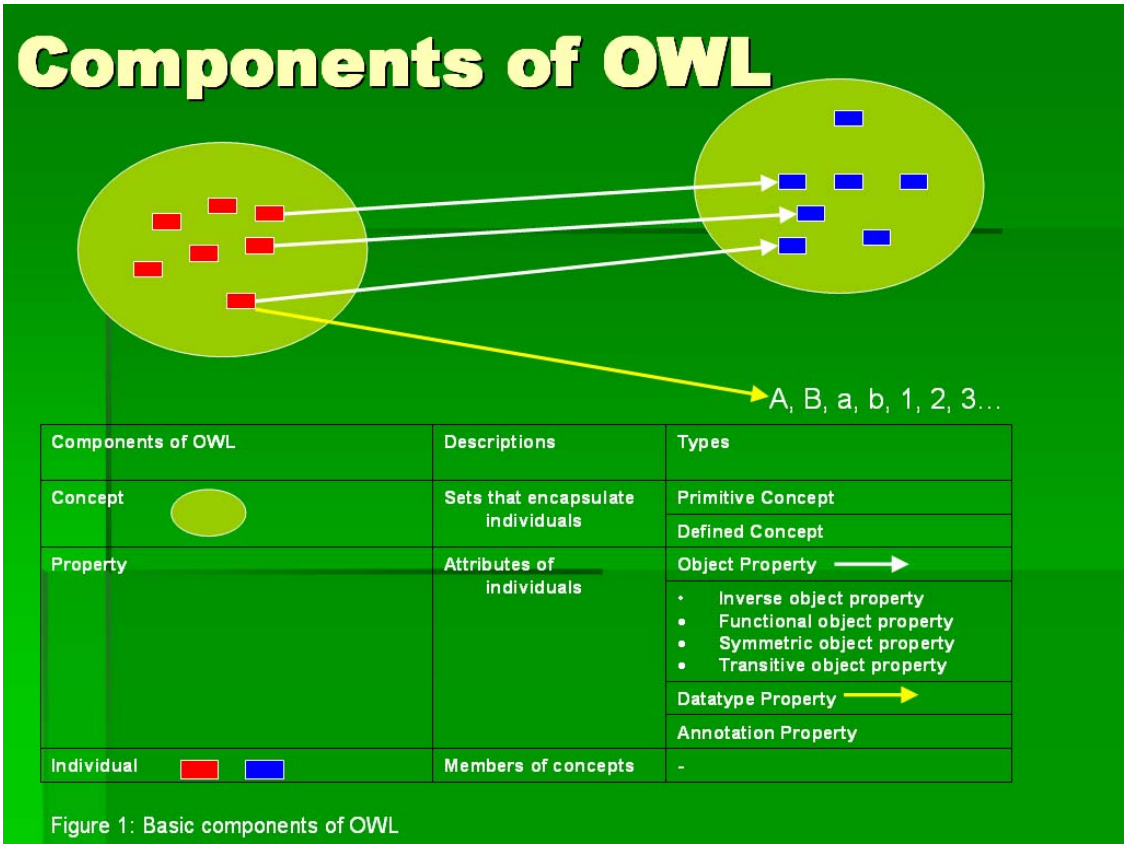


Figure 1: Basic components of OWL

2.4) Overview of Bio-Ontologies (see Table 4)

2.4.1) Open Biomedical Ontologies (OBO)

OBO repository is a large library of ontologies from the biomedical domain hosted by the National Center for Biomedical Ontology (NCBO) [21]. It was first created as a means of providing convenient access to the GO and its sister ontologies at a time where a resource like NCBO was not available. OBO has since evolved into a wide-base collaborative effort within the bio-ontologies community to enhance the quality and interoperability of ontologies in life sciences from the point of view of biological content and logical structure. Most of the ontologies in OBO are written in OBO flat file format, a simple textual syntax designed to be compact, readable by human and easy to parse. In this light, OBO foundry provides ontology design principles concerning syntax, unique identifiers,

content and documentation to the ontologies as a common agreement between users/editors.

2.4.2) OBO Foundry Principles:

The pricinples of the foundry can be summarized as follows [19, 23]:

1. The ontology must use a common and shared syntax(OBO or OWL format)
2. The ontology possesses a unique identifier namespace and has procedures for identifiying distinct successive versions
3. Terms or concepts must be provided with textual definition and, to a certain degree, formal definition such DL definitions
4. Every terms or concepts in the ontology should be provided with a unique identifier
5. Relationships or properties defined in the ontology must be compatible to the pattern set forth in the OBO relation ontology(RO) [24]
6. The ontology must embrace the principle of orthogonality where a specific ontology is expected to converge unto a single (upper) ontology that is recommended by the OBO community
7. The ontology should be open and be made available to be used by all without any limitations and be subjected to collaborative developmental process involving other ontology developers covering the neighboring biology domain
8. Other informal principles:
    a. The ontology should make distinction between plural concepts and singular concepts

b. The ontology should be grammatically consistent

c. The use of "or" and "and" is highly discouraged as it generates unnecessary ambiguity in the concepts

2.4.3) Formalized Bio-Ontologies:

An OBO formatted ontology is made up of a collection of stanzas that describes elements of the ontology. These stanzas describe a term that is equivalent to a concept, a relationship type or an instance. The OBO formatted syntax also consists of tag values associated to the stanza. The tag values have a structure that depends on the tag type. The tag type is described in the OBO specification using natural language [21]. This type of description is informal and does not make the conceptual structure of the OBO language clear [21]. Similarly, the semantics used to describe the natural language description for different types of tag-value pairs are also informally defined [21]. As a result, a description in OBO can be rather ambiguous and unclear. The DL family of ontology languages was developed precisely to address the problem as OWL can unambiguously specify the semantic properties of all ontology constructs. OWL-DL provides OBO with the much needed formal semantics.

Ontology	Uses
Gene Ontology	provides terminologies for annotation of results of biological experiments such as gene expression experiments and bioinfomatics resources
Disease Ontology	provides the controlled vocabulary for the mapping of diseases and associated conditions to particular medical codes such as ICD9CM, SNOMED
FungalWeb Ontology	integrates information relevant to industrial applications of fungal enzymes
ChEBI Ontology	provides structured controlled vocabulary to support interoperability between ChEBI and other

	knowledgebases
Chemical Ontology	provides semantic support for querying chemical databases
Tambis Ontology	describes and enable query of bioinformatics databases
OpenGalen	use in medical information management
EcoCyc	describes the whole metabolism of *E.coli*
BioPAX	describes biological pathways in OWL

Table 4: Examples of bio-ontologies and their respective uses

2.5) Semantic Technologies Applied to Chemical Nomenclature

There have been other significant developments where semantic technologies were used in the domain of chemistry and lipid analysis including of reports of ontologies built specifically to describe biologically relevant chemical entities, organic compounds and organic reactions [18, 25, 26]. Here we briefly summarize relevant work in the context of lipid classification.

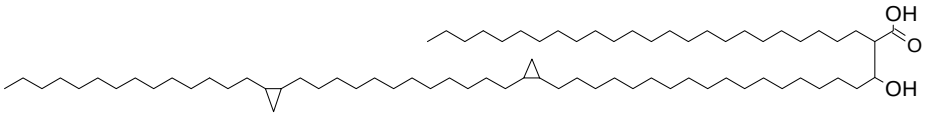
2.5.1) ChEBI

ChEBI (Chemical Entities of Biological Interest) is a project initiated by EBI to provide a high-quality controlled vocabulary to promote the correct and consistent use of unambiguous biochemical terminology throughout the molecular database in EBI [27]. ChEBI is now a database with 14,757 annotated entries of small molecules with an ontological structure integrated into it. The ChEBI ontology organizes all terms in the database under 4 sub-ontologies (Molecular Structure, Biological Role, Application, Subatomic Particle) and uses relationship definitions standardized by the OBO [22] community in order to support interoperability between ChEBI and other

knowledgebases (inclusive of databases and other biomedical ontologies). As of October 2007 ChEBI currently has 14 lipid sub-classes.

2.5.2) InChI

InChI [28] and InCHiKey [29] are non-proprietary identifiers for chemical substances that can be used in printed or electronic data sources, thus enabling easier linking of diverse data compilations. They encode chemical structures of molecules in a string of machine-readable characters unique to the respective molecule (see Figure 2). Preliminary work involving InChI in web searches had been very encouraging, given that there was 100 % recall and precision [28]. In addition several algorithms had been developed to facilitate sub-structure or even textual substring searches of chemical molecule information on the web [30, 31]. While chemical structures for individual lipids have been published in InChI format there has been, to our knowledge, no hierarchical formulation of lipid class definitions described in InChI.



InChI=1/C76H148O3/c1-3-5-7-9-11-13-15-17-18-19-20-21-22-23-24-27-30-34-42-48-54-60-66-74(76(78)79)75(77)67-61-55-49-43-35-31-28-25-26-29-33-39-45-51-57-63-71-69-73(71)65-59-53-47-41-37-36-40-46-52-58-64-72-68-70(72)62-56-50-44-38-32-16-14-12-10-8-6-4-2/h70-75,77H,3-69H2,1-2H3,(H,78,79)

Figure 2: Structure and InChI of an alpha mycolic acid

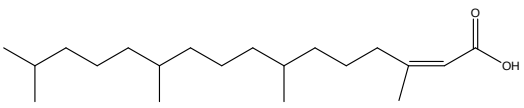
2.5.3) Chemical Ontology

The Chemical Ontology [25], CO, is a small molecule ontology that describes organic compound on the basis of chemical functional groups. It was initially developed to

describe chemical functional groups for the classification of chemical compounds and coded in OWL-DL [26] formalism. In the ontology an organic compound is defined explicitly by the presence or absence certain functional groups. This classification method, specifically with the use of explicit DL semantics, can be applied to lipids because functional groups describe the chemical reactivity in terms of atoms and their connectivity, and reflect the chemical behavior of a lipid in a biological context. Furthermore, current lipid database records often lack such annotations and classification often has to be done manually. Therefore, use of the chemical ontology presents a viable alternative to address the lack of clarity in lipid nomenclature, not just in providing an ontological framework where lipids terminology can be gathered in a single resource but it also provides an avenue to describe lipids nomenclature in an open and explicit semantics. However, the OWL version of Chemical Ontology is limited as it only provides 35 functional groups and that is not sufficient to describe the lipids as classified under LIPIDMAPS. At present, the Chemical Ontology had been used to classify only 28 classes of organic compound. Lipids are more complex biomolecules that can have multiple and distinct functional groups in one molecule. For example, Figure 2 shows an alpha mycolic acid that has a hydroxyl group and a carboxylic acid group. According to the Chemical Ontology, it is both an alcohol and a carboxylic acid. Such a definition is semantically ambiguous. In addition the molecule has a functional group that is not defined in Chemical Ontology, cyclopropane group.

Consequently, in order to accurately describe lipids, we need more functional groups, many of which have not been described in the Chemical Ontology. Moreover, the

Chemical Ontology classifies each class of organic compounds with just one functional group and it is solely based on the structural aspect of chemical compounds. Such a scheme cannot accurately classify lipids as it does not necessarily describe or represent the biochemistry of lipids and it is not adequate for the task of classifying lipids based on other criteria such as the biological origin of individual molecule. In contrast to Chemical Ontology, LIPID MAPS grouped lipids together based on at least the following criteria, namely structural similarity, biosynthetic origin and function. Table 5 shows examples of lipids taken from the LMSD to illustrate how different lipids classes are classified by LIPID MAPS. In Table 5a, LC_Fatty_Acids_and_Conjugates, are classified together as lipids that are characterized by a series of methylene groups and would terminate with a terminal carboxylic group [2]. In Table 5b, LC_Eicosanoids, are classified as lipids that derived from the same biosynthetic precursor Arachidonic acid and are known as bioactive molecules that play important role in signaling and inflammatory processes [10]. In Table 5c, LC_Octadecanoids, are classified as lipids that derived from the same biosynthetic precursor 12 oxo-phytodienoic acid while LC_Docosanoids are lipid that derived from the same biosynthetic precursor docosahexaenoic acid [2]. This is a lipid biology centric classification and it reflects the way in which lipid scientists classify lipids accurately.

a.Classification based on structure

3,7,11,15-tetramethyl-2Z-hexadecenoic acid , a methyl fatty acid under LC_Fatty_Acids_and_Conjugates.

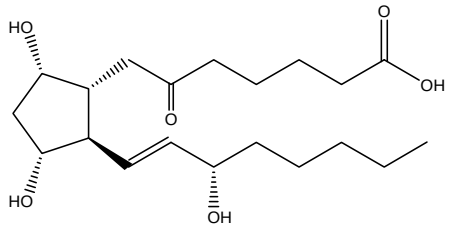
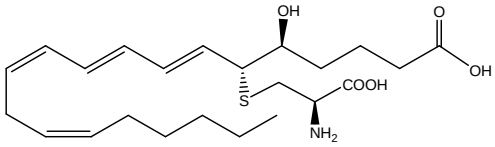
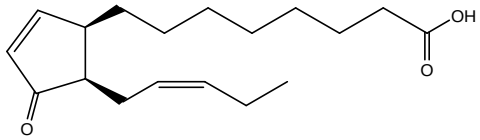
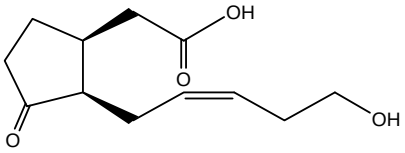
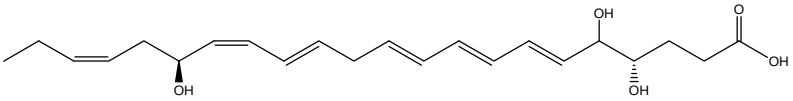
b.Classification based on functional role

6-oxo-9S,11R,15S-trihydroxy-13E-prostenoic acid or 6-keto-PGF1α, a prostaglandins under LC_Eicosanoids

5S-hydroxy,6R-(S-cysteinyl),7E,9E,11Z,14Z-eicosatetraenoic acid or LTE4, a leukotriene under LC_Eicosanoids.
c.Classification based on biosynthetic origin
A.LC_Octadecanoids

(9R,13R)-12-oxo-phytodienoic acid, a 12 oxophytodienoic acid under LC_Octadecanoids.

(1S,2R)-3-oxo-2-(5'-hydroxy-2'Z-pentenyl)-cyclopentaneacetic acid or Tuberonic acid, a jasmonic acid under LC_Octadecanoids.
B.LC_Dosocanoids

4S,5,17S-trihydroxy-docosa-6E,8E,10E,13E,15Z,19Z-hexaenoic acid or Resolvin 4, a dosocanoids.

Table 5: Structure, systematic name and class of some lipids classify by LIPID MAPS using criteria such structure, function and biosynthetic origin

2.5.4) Ontology and Text Mining

Alexopoulou *et al.* reported the use of automated text mining algorithm to assemble domain specific terminologies. These terms were then use to develop the Lipoprotein Metabolism Ontology (LMO) in a semi automated way for the purpose of conducting text mining in the field of lipoprotein metabolism [22]. Similarly, Baker *et al.* reported the use of Lipid Ontology to mine for textual information of lipid and lipid biology from literature sources and to subsequently make available these to the scientist in a dynamic display of knowledge map [32].

3.) Ontologies and Lipids

Lipids have many features and, likewise, there are many aspects in lipid biology. This is a lot of information and complex relationships. Ontology can capture this information-rich content and represent them meaningfully in classes/concepts, properties/relations, values/instances. Lipids do not have a universally accepted nomenclature. Ontology provides a place where a systematic nomenclature can be described and shared with everyone in the field so that a consensus can be arrived at. In addition to being able to represent a systematic classification of lipid, representation in OWL-DL ontology structure forces the chosen lipid nomenclature, that is mostly un-intuitive, to become an explicitly defined knowledge. This brings clarity to the knowledge and removes ambiguity from the meaning of many lipid terms, especially those from the bibliographic domain, that are saturated with many synonyms that are neither a standard nor clearly defined.

Lastly, due to the lack of a unified classification system and the heterogenous nature of data from lipidomics (due to different data formats associated to a wide range of technology platforms and granularity of data), integration of lipid data is difficult [12]. Here, OWL ontology acts as a standard where lipid knowledge can be made available through a common technology platform so that seamless integration of data and recycling of metadata can be achieved.

Chapter II: Ontology Development Methodology

Due to the vast and complex nature of biological knowledge, bio-ontologies are especially hard to engineer. This is further complicated by the volatility of the knowledge in the specific knowledge domain as the biologist's understanding of a domain is constantly changing.

1) Goal and Purpose

In an ontology development process, the purpose of the ontology is especially important. Depending on the intended use of the ontology, the cost and complexity of building a bio-ontology would vary. Naturally, an ontology designed to provide basic understanding of a knowledge domain would be less costly to build than ontology meant for complex semantic web applications such as complex query or automated reasoning. Therefore, the purpose of a bio-ontology must be decided as it would determine the complexity and subsequently the approach to be adopted for ontology development. The purpose of a bio-ontology can be easily narrowed down by identifying the required scope, possible use case scenarios or the type of competency questions that the ontology is meant to answer. Our competency questions are as follows:

Can the ontology be used to tell a story at various degree of granularity?

Can the ontology represent knowledge more explicitly, more detailed than what a database could do?

Can the ontology represent definition of lipid entity and lipid-centric data?

Can the ontology substitute or even supersede a database schema driven query model?

Can the ontology make implicit knowledge explicit?

Ultimately, the choice of methodology depends on the function of the ontology. Generally, bio-ontologies can be categorized into 3 major functions.

Task-oriented ontologies- Ontologies designed to perform concrete tasks such as data mining, resource integration and semantic reasoning. Task-oriented ontologies specify information of a knowledge domain necessary for a task and are designed for use in a specific application only. In its extreme form, task-oriented ontologies are highly specific and are purely engineering artifacts of specific applications in the industrial environment.

Domain-oriented ontologies- Ontologies that capture knowledge of a field of interest. Domain-oriented ontologies are formalized knowledge encoded in a knowledge representation language with the purpose to share knowledge with others in the field.

Generic ontologies- Ontologies with very general concepts whose only purpose is to integrate different ontologies.

2) Methodology

There is no standard methodology for building ontology. A methodology would include the ontology development life cycle that occurs during the development process, guidelines, principles that influence each stage of the life cycle. Castro *et al.* reviewed some of the methodologies used in industrial environment to build ontologies [33].

Among them are TOVE (Toronto Virtual Enterprise), Methontology, Diligent, Enterprise Methodology, Unified Methodology. These methodologies were assessed and were found to be very application specific. Most of them had been applied and deployed in highly controlled industrial environment in a one-off basis. Furthermore, none of these methodologies had been standardized out of their original industrial context long enough to impact wider ontology building community, including the bioinformatics or bio-ontologies community.

3) Ontology Development Lifecycle

While there is no standard methodology to develop ontologies, the development life cycles are common for most ontologies (see Figure 3).

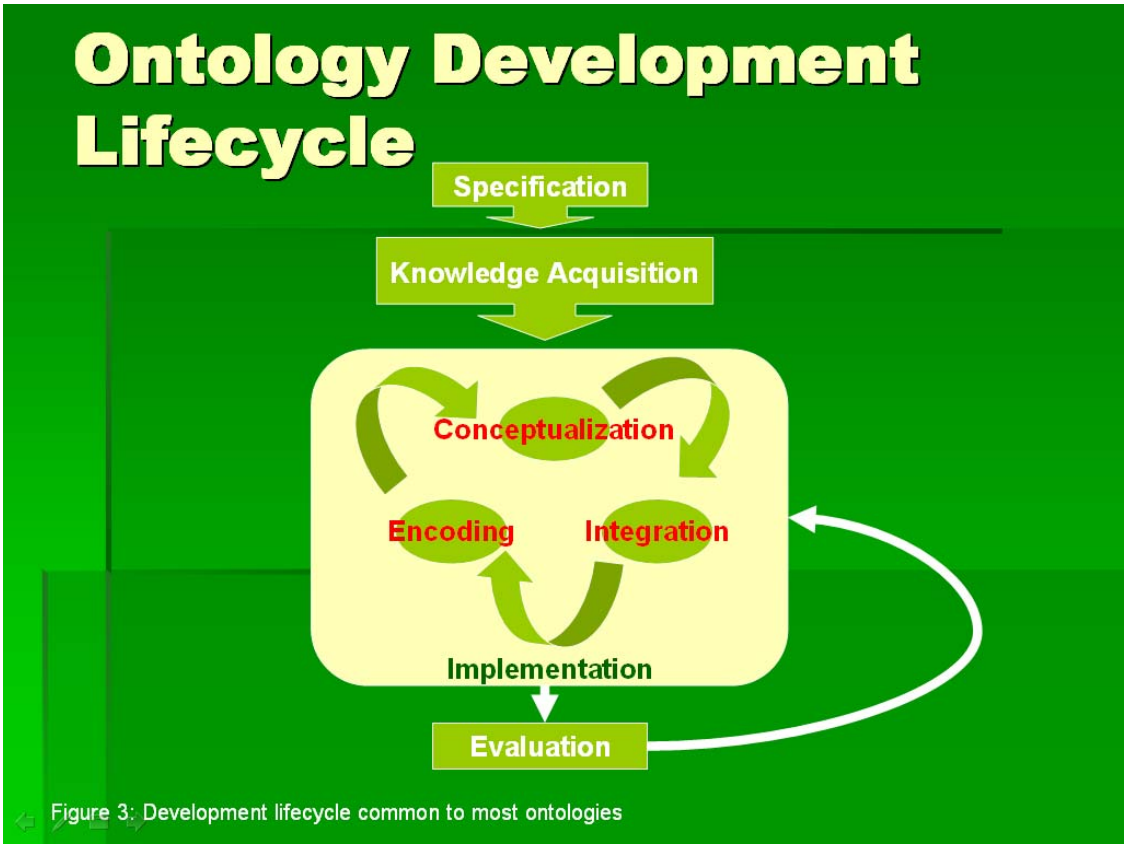


Figure 3: Development lifecycle common to most ontologies

3.1) Specification

A phase where the purpose, scope and granularity of an ontology is determined. This phase determines the type and coverage of data sources (databases, bibliographic information and reusable ontologies) needed to build an ontology that supports a specific purpose, application or task.

The Lipid Ontology is conceived to conceptualize and capture knowledge in the domain of lipids through the use of concepts, relations, instances and constraints on concepts. This ontology is a resource that provides a common terminology for the lipid domain and a basis for interoperability between information systems. It provides a consistent semantic and syntactic representation to integrate data from databases as well as other ontologies.

Other equally important motivations for Lipid Ontology can be summarized as follows:
(i) to provide, in a standardized OWL-DL format, a formal framework for the organization, processing and description of information in the emerging fields of lipidomics and lipid biology; (ii) to specify a data model to manage information on lipid molecules, define features and declare appropriate relations to other biochemical entities i.e. proteins, diseases, pathways; (iii) to enable the connection of the pre-existing or legacy 'lipid synonyms' found in literature or other databases to the LIPID MAPS classification system; (iv) to serve as an integration and query model for one or more data warehouses of lipid information; (v) to serve as a flexible and accessible format for building consensus on a current systematic classification of lipids and lipid nomenclature, which is particularly relevant to the discovery of new lipids and lipid classes that have yet

to be systematically named; (vi) to define lipid classification explicitly with respect to LIPID MAPS nomenclature using description logics in OWL-DL language and to establish a systematic classification of lipids that supports reasoning tasks such as checking ontology consistency, computing inference and realization.

The Lipid Ontology family of ontologies is built on a combination of task-oriented, domain-oriented and generic ontologies design principle. This family of ontologies consists of a combination of modules that supports reusing other concepts from other ontologies. It started of as a baseline ontology with a very specific semantic application to support. The baseline ontology was further developed into a reference ontology. Specialized ontology was then be modified from the reference ontology to perform a function for specific application (Figure 4). Depending on the purpose or application, the ontology can be made more comprehensive to support annotation or made simpler just to support a specialized computational task.

The first Lipid Ontology (Lipid Ontology 1.0) is specified by a database schema and it aims to provide a DL-based knowledge representation to represent and to integrate information from multiple databases. In addition to that, the ontology can integrate bibliographic information and is build with upper-level concepts to integrate other ontologies. In short, Lipid Ontology 1.0 is built to unify diverse bioinformatics data sources and literature databases in a consistent semantic and syntactic representation using semantic web technologies. Being a vehicle of knowledge representation, it has been used to map and represent knowledge in order facilitate intuitive knowledge

navigation and discovery by the end user through a visual query application. The integration of other bio-ontologies is not carried out until the deployment of Lipid Ontology Reference. The Lipid Ontology Reference is the result of integrating databases, bibliographic information and other ontologies into a single ontology. It is a reference ontology where other more task-oriented ontologies with specific application or domain oriented ontologies can be derived from. LiCO and LERO are specialized domain oriented ontologies designed to be OBO compliant so that the semantic richness and knowledge in LiCO and LERO can be accessed by the wider biomedical research community, especially the OBO community. Lastly, Lipid Ontology Ov is an application ontology extended from the Lipid Ontology Reference to enable pathway exploration on top of the original visual query paradigm applied to Lipid Ontology 1.0.

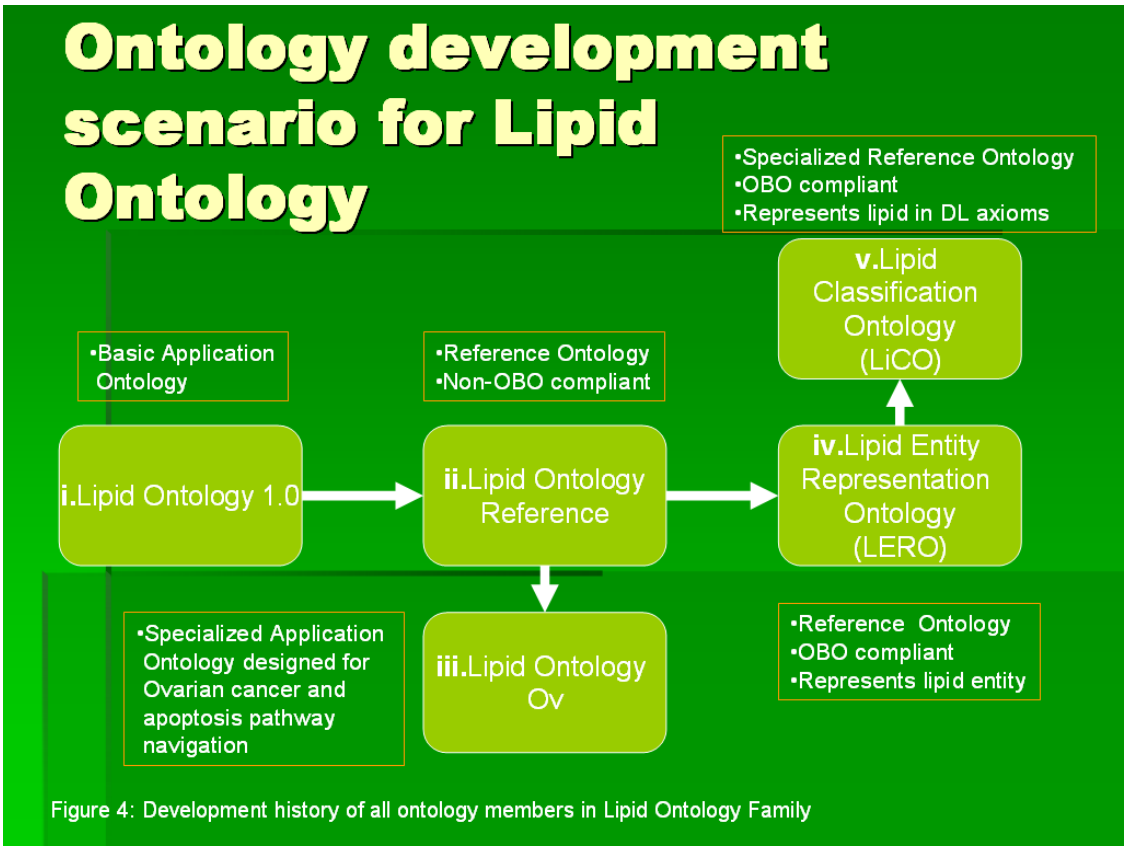


Figure 4: Development history of all ontology members in Lipid Ontology Family

3.2) Knowledge Acquisition

In the knowledge acquisition phase, domain knowledge is acquired from domain experts, database metadata, other ontologies and other re-usable information such text book information and research papers. Information can be used in 2 ways. Firstly, they are models or examples where the model of knowledge domain of lipids could be based on. Secondly, they provide actual data that could be incorporated into the ontology.

Data relevant to biologists such as pathways, chemical compound entries, annotations, structures as well as associated disease phenotype, protein information are often stored in multiple databases with distinct and incompatible data formats. Other sources of information are found in various text, papers and literature resources. A typical knowledge acquisition begins with the selection of suitable resources from which data can be retrieved. The choice of appropriate resources depends on factors such as the quality, accuracy, the speed of update, consistency and reliability of the data. Once the resource has been identified, extraction of terms and associated data can be achieved manually or with perl script automation. Depending on the quality of the data, manual curation may be needed to remove any inconsistency, ambiguity, contradiction or error.

3.2.1) Knowledge Resources

During the development of Lipid Ontology, we integrate the schema from an existing lipid database, LipidDW, together with the lipid content in the form of database annotations from entries found in several distributed biological databases, namely LMSD,

LipidBank, KEGG COMPOUND databases. In addition to that, other online resources relevant to lipids such as Lipid Library and Wikipedia are consulted too.

LipidDW is an in house relational data warehouse system designed to integrate lipid data from LMSD, LipidBank, KEGG COMPOUND databases as well as associating them with other data such as disease phenotype from OMIM, enzyme from BRENDA [52], protein from Swiss-Prot [53] and pathway from KEGG PATHWAY [34].

The LIPID MAPS STRUCTURE DATABASE (LMSD) is the official database of LIPID MAPS consortia [15]. To date, the database contains a total of 10,789 entries, including 2688 Fatty acyls (FA), 3009 Glycerolipids (GL), 1971 Glycerphospholipids (GP), 621 Sphingolipids (SP), 1745 Sterol lipids (ST), 609 Prenol lipids (PR), 10 Saccharolipids (SL), and 136 Polyketides (PK). Lipid entries from the database are connected to Wikipedia, LipidBank, KEGG COMPOUND database and PubChem via hyperlinks where identical entries are available.

LipidBank is the official database of the Japanese Conference on the Biochemistry of Lipids (JCBL) [35]. The database contains 7009 unique molecular structures, their lipid names (common name, IUPAC), spectral information (mass, UV, IR, NMR and others), and most importantly, literature information. The database lists natural lipids only and is annotated with information that is manually curated and approved by experts in lipid research.

KEGG COMPOUND is a chemical structure database for metabolic compounds and other chemical substances that are relevant to biological systems [36]. The compounds represented in KEGG COMPOUND include Lipids, Peptides, Polyketides, non-ribosomal peptides and plant secondary metabolites. It is tightly integrated with KEGG BRITE (a collection of hierachical classification to biological entities and systems) and KEGG PATHWAY (a collection of pathway maps built from known molecular interactions and reaction networks) to enable the inference of higher-order functions for the compounds.

Lipid Library is an ISI-recommended online resource for lipids produced by Dr William W. Christie, a consultant to Mylnefield Lipid Analysis and is hosted by Scottish Crop Research Institute (and MRS Lipid Analysis Unit), Invergowrie, Dundee, Scotland. [1]

Wikipedia is a multilingual, web-based, free-content encyclopedia project [37]. Wikipedia's articles provide links to guide the user to related pages with additional information. While largely an informal resource, Wikipedia does provide reliable basic knowledge in the domain of chemistry and chemical nomenclature.

In addition to that, we consulted published scientific literatures on nomenclature of lipids extensively. In particularly, we based our lipid entity hierarchy on the LIPID MAPS classification hierarchy recommended by the LIPID MAPS consortium [2]. In addition to that, we also consulted literatures published by the IUPAC society on the nomenclature of various classes of lipids [14].

Other OWL-based ontologies that are openly available through the internet are additional information-rich resources that we relied on to build our ontology. Similar to the case with databases, ontological resources had been used in 2 ways, firstly as references to model our knowledge domain and secondly, as modules where we literally re-use or incorporate into our ontology.

BFO, also known as Basic Formal Ontology is a multi-categorical ontology that provides very high level upper-ontology framework to help in the organization and integration of biomedical information [38]. It is a formal upper ontology and promotes the development of orthogonal ontologies that would eventually converge onto its upper ontology. It is available in OWL format.

BioTop is a top domain ontology that provides definitions for the most important basic entities necessary to describe the phenomena in the domain of biomedical sciences [39]. The BioTop ontology provides an upper ontology necessary for low level biomedical ontology to connect with BFO (see Figure 5). It is available in OWL format.

ChemTop is an ontology that inherits large amount of definitions from BioTop and aims to play the role of BioTop for the domains not cover by BioTop, specifically the chemical domain (see Figure 5). It is available in OWL format.

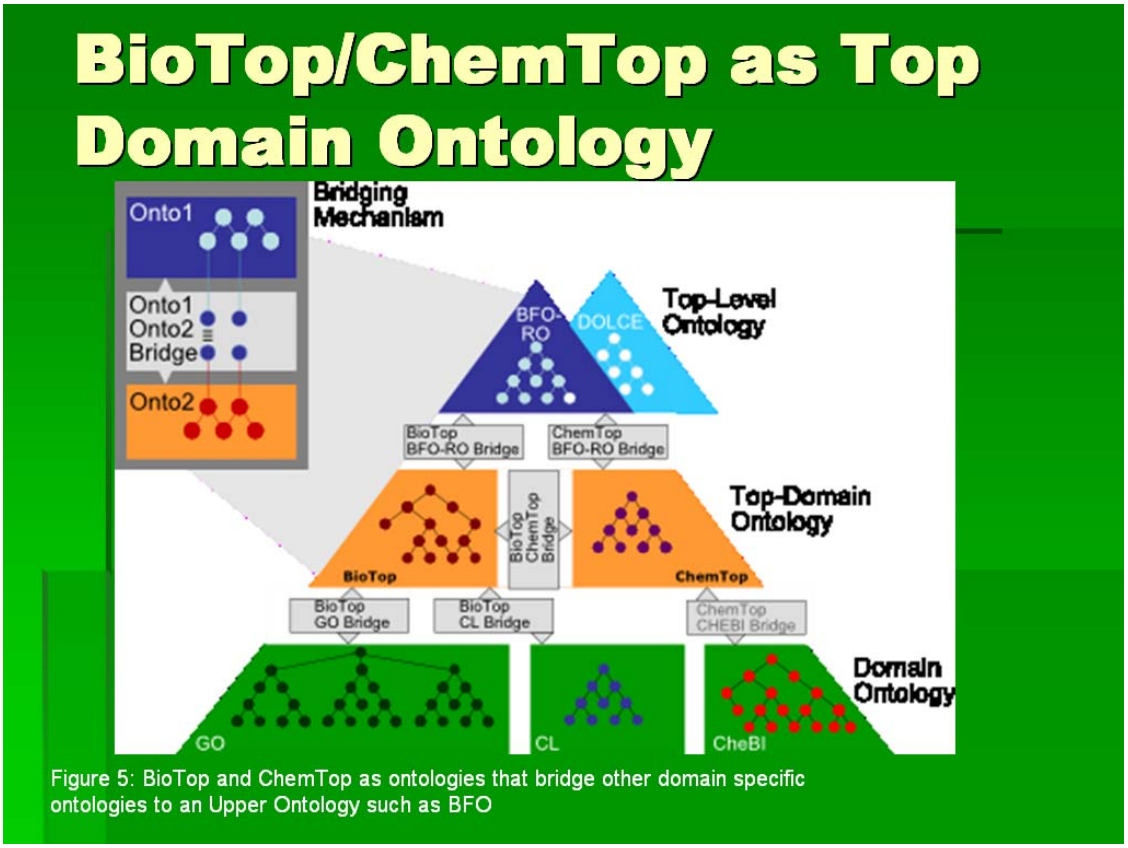


Figure 5: BioTop and ChemTop as ontologies that bridge other domain specific ontologies to an Upper Ontology such as BFO

FungalWeb Ontology is a large-scale integrated bio-ontology in the field of fungal genomics [40]. It provides an integrated accessibility to distributed information across multiple databases and ontologies and is the core of a semantic web system. It is available in OWL-DL format.

Disease Ontology is a controlled medical vocabulary developed at the Bioinformatics Core Facility in collaboration with the NuGene Project at the Center for Genetic Medicine [41]. It was designed to facilitate the mapping of diseases and associated conditions to particular medical codes such as ICD9CM, SNOMED and others. Disease Ontology is implemented as a directed acyclic graph (DAG) and it is stored in the form of OBO format.

The NCI Thésaurus is a public domain description logic-based terminology produced by the National Cancer Institute to facilitate translational research and to support the bioinformatics infrastructure of the Institute [42]. It is deep and complex compared to most broad clinical vocabularies and implements rich semantic interrelationships between the nodes of its taxonomies. It is available in OWL format.

The Gene Ontology project provides a controlled vocabulary to describe gene and gene product attributes in any organism [43]. Gene Ontology can be organized into 3 sub ontologies, namely cellular component, biological process and molecular function. Gene Ontology terms are used extensively by biologist to annotate gene products. The ontology often acts as a semantic integrating system and is one of the most widely used ontology in the biomedical research domain. It is available in both OBO format and OWL format.

The Pathway Ontology is a controlled vocabulary for pathways that captures various kinds of biological networks, relationships between them and alterations or malfunctioning of such networks within a hierarchical structure [44]. The Pathway Ontology is developed at Rat Genome Database. It is available in OWL format.

Chemical Ontology is a novel ontology based on chemical functional groups that was developed to identify, categorize and make semantic comparison of small molecules [25]. This is an application ontology and has been encoded in OBO. A smaller and simpler version of the Chemical Ontology is available in OWL-DL format.

Molecule Role Ontology is a structured controlled vocabulary of concrete protein names and generic protein names built to annotate signal transduction pathway molecules in the scientific literature [45]. It is available in OWL format

Lastly, informal interviews with laboratory scientist, lipid experts and text mining experts are also a key part of the knowledge acquisition cycle.

3.3) Implementation

The implementation phase consists of 3 sub phases, namely conceptualization, integration and encoding phase. It is a step where the information is built into an ontology via an iterative cycle of conceptualization, integration and encoding.

3.3.1) Conceptualization

Conceptualization is a phase where key concepts with properties associated to other concepts as well as properties between the concepts for the knowledge domain are identified. The concepts and properties are assigned their natural language terms and subsequently organized into an explicit conceptual model such as an is-a subsumption hierarchy. We take a DL based conceptualization approach. With DL conceptualization, we specify frames or classes as collections of instances where each frame can have a collection of slots or attributes that are values or other frames without the problems of unclear semantics common to all frame based representation. Unlike frame based representation, DL uses clear semantics and defines concepts in terms of descriptions using other roles and concepts in such a way that it could be used to derive classification

taxonomies. Below is a description of various attributes of the DL conceptualization that we have implemented into the Lipid Ontology.

Concepts are sets that contain instances. Concepts describe accurately the requirements for membership of the class using formal descriptions. There are 2 types of concepts.

- Defined concepts are concepts with at least one necessary and sufficient condition. It means that when an individual has properties that satisfy the membership requirement of a defined class, it can be inferred to be a member of the class.

- Primitive concepts are concepts with necessary condition. It means that when an individual is assigned to a specific primitive concept. The individual must have properties that satisfy the membership requirement of the class. The same cannot be inferred from the reverse direction.

Relationships are links that exist between 2 concepts or 2 instances. There are 2 types of relationships.

- Subsumption relationship organizes concepts into a superclass-subclass hierarchy.

- Associative relationship relates individuals of concepts. The object property in OWL describes this relationship; an object property links 2 instances together. Theoretically, we can also define an associative relationship between 2 concepts specifying all instances of a concept are related to at least one instance of another concept.

Upper Ontology:

An upper ontology consists of top-level concepts in an ontology that are defined in very generic term and act as superconcepts that subsume other concepts from other ontologies. Concepts from other ontologies need be integrated into the hierarchical structure of the upper ontology without violating any of the semantic correctness. By maintaining an upper ontology in the Lipid Ontology, we enable specific concepts from other ontologies to be added into the Lipid Ontology as an independent module. The upper ontology is maintained in Lipid Ontology 1.0 and has expedited the development process of Lipid Ontology Reference. For LiCO and LERO, we incorporate an upper ontology that is compliant to OBO specification because we want to use the ontology to share domain knowledge with the wider bio-ontologies community. The same OBO compliance has not been applied to Lipid Ontology 1.0, Lipid Ontology Reference and Lipid Ontology Ov as these ontologies are application-centric ontologies that need to adhere to a specification that is compatible for their intended applications.

Axiomatic Restriction:

Also known as property constraint and consists of rules for membership requirement of classes. Property constraints were applied heavily to define lipid entity in LiCO and LERO.

Closure Axioms:

When a closure axiom is applied for a concept, it means that a property constraint can only be achieved with the use of members of a specific class only. Closure axiom is applied heavily to define lipid entity in LiCO and LERO.

3.2.2) Integration

Integration is a phase where data and information acquired from existing databases, ontologies and other informal resources are put together into a consistent ontology. Information collected from databases, other ontologies as well as the hand-crafted baseline ontology are merged into a new ontology. Alternatively, knowledge can be integrated without merging ontologies and this can be done by imports.

The Lipid Ontology was integrated at 2 levels, the data level and the semantic level. A typical data integration exercise involves identifying overlapping or identical database entries and annotations. These entries are subsequently linked up with a series of hyperlinks. Integration for ontology differs from database integration in that it emphasizes semantic integration on top of the usual data integration.

Data Integration:

During data integration, data with heterogenous granularities and formats are normalized into a consistent syntactic representation. For the Lipid Ontology development scenario, data integration occurs when the Lipid Ontology is instantiated into a knowledge base or when ontologies are merged together or when ontologies are imported into the Lipid Ontology 1.0.

Semantic Integration:

Semantic integration is done to enable an accurate and consistent mix of data from different sources. It involves identifying identical, similar, or overlapping data elements

from various resources as well as their semantic relationships with one another so that these heterogenous data elements can be mapped into a common frame of reference.

- Principle of Orthogonality
    - The principle of orthogonality asserts that ontologies from every knowledge domain should eventually converge upon a single upper ontology [19]. Subsequently, ontologies that are orthogonal are build as interoperable modules that could be combined together to give rise to an incrementally evolving knowledge network. The principle of orthogonality brings several benefits. It ensures that the ontology that was build has been validated, used and maintained by the domain experts and that it would work well with other ontologies. Ontologies, being orthogonal, would reduce the need to map or align ontologies. This is because ontology alignment is very difficult, costly, error prone. Moreover, orthogonality ensures mutual consistency of ontologies, thereby allowing ontologies to be combined with one another, resulting in the accumulation of scientific knowledge. Lastly, orthogonality eliminates redundancy as every domain expert can just focus on his area of expertise without the need to worry about related fields of knowledge.
- Challenges in Semantic Integration
    - Language mismatches due to ontologies being written in different ontology languages, syntaxes, logical notations, language expressivity and semantics of primitives (same name, different meaning).
    - Model-level mismatches due to conceptualization mismatches (differences in the way a domain is interpreted, different ontological concepts, different

relationships between concepts) and explication mismatches (differences in the way the conceptualization is specified) between ontologies.

- Lack of clear semantics due to inconsistency in the use of certain terms within the same ontology, unnecessary proliferation of terms, different levels of granularity that are used in the ontology are not explicitly stated, mixed levels of granularity and overloading of relationship/property in an ontology.

- Choice of Reusable Ontologies
  - Reusing ontologies is not just about selecting a section of the source ontology and incorporating it into the target ontology. A knowledge engineer needs to extrapolate the context from the source ontology to the target ontology. By doing so, a knowledge engineer transfers the meaning convey by the concepts and semantics from the source ontology to the target. Therefore, exact linguistic matches are not crucial and this criteria itself is not sufficient to justify reusability of concepts in the source ontology. When identifying reusable ontologies, a knowledge engineer needs to focus on what the concepts in mind have been use for, how these concepts relate to other concepts, how these concepts are incorporated in the relevant processes as well as how a domain expert understands them.

In the development of Lipid Ontology, we design our ontology to be as orthogonal as possible with other ontologies. We do not embrace the notion of absolute orthogonality and we accept that there are many ways to design and build ontologies. Therefore, our ontologies are a cross between pragmatism and absolute orthogonality. The Lipid

Ontology family of ontologies are designed to be as orthogonal as possible without sacrificing functional purposes. Where possible, we provide modified versions of Lipid Ontology that are orthogonal to other ontologies in the wider community, specifically the OBO community. To this end, Lipid Ontology 1.0, Lipid Ontology Reference and Lipid Ontology Ov remain application specific and do not adhere to the general OBO design principle. However, smaller, specialized ontologies such as LiCO, LERO that are orthogonal to OBO can be crafted out of the Lipid Ontology Reference to provide accessibility of formalized knowledge to the wider bio-ontology community.

Methods of Semantic Integration:

- Syntactic Parsing –Applicable when concept terms in an ontology are made up of terms or combination of terms from other ontologies. It is achieved by syntactically parsing terms in one ontology in search for terms from another ontology. However, syntactic parsing is limited in its applicability as it is not scalable and it does not really semantically integrate multiple ontologies [40,46].
- Use of a formal knowledge representation language that supports imports from other ontologies –An example would be OWL-DL where OWL-DL ontologies can import other OWL ontologies, either locally or via HTTP. With this, semantic integration and reuse of ontologies are achieved without parsing.
- Upper level ontologies –Different ontologies are presented as independent modules that can be connected via a top level ontology that provide concepts with upper level semantics as such that these ontologies can be subsumed under the concepts provided by the upper ontology.

- Ontology alignment -Alignment is also known as mapping and it involves identifying semantically similar concepts between ontologies and relating them via equivalence and subsumption properties. It is very costly and difficult as it is largely dependent on manual human effort.

The semantic integration is implemented in Lipid Ontology 1.0 to give rise to Lipid Ontology Reference. Because Lipid Ontology 1.0 is built with upper ontology concepts and is based on OWL-DL language, integration of ontologies is achieved by importing parts of other ontologies as independent modules that could be subsumed by the upper level concepts in Lipid Ontology 1.0. In addition to that, parts of other ontologies are aligned and subsequently made to relate with Lipid Ontology 1.0 via subsumption property. The ontology alignment procedure differs from standard alignment procedure in that concept terms are transferred without the relationships that these concepts had participated in the source ontologies.

3.3.3) Encoding

Encoding is a phase where the results of conceptualization and integration are represented in a formal knowledge representation language.

The Lipid Ontology family of ontologies is encoded in OWL-DL with Protégé 3.4 beta.

The choice of knowledge representation language is simple. We are looking for a knowledge representation language that could express complex relationship in a way that is both intuitive to human and machine. In addition to that, we want the ontology to be able to undergo semantic reasoning. OWL-DL is a knowledge representation language

that has a high level of expressivity, semantic richness as well as a logical structure that supports computational decidability. Another reason for using OWL-DL is because there are quite a number of ontologies out there written in OWL-DL. By using OWL-DL, we designed the Lipid Ontology family of ontologies to be at least syntactically compatible with other OWL ontologies and, as a result of that, we could re-use these ontologies easily. In addition to that, it is a W3C-endorsed knowledge representation language for semantic web application and we expect widespread adoption of OWL-DL by semantic web application developer as well as knowledge representation specialist alike in the near future. The use of OWL-DL will ensure that the Lipid Ontology family of ontologies to remain compatible and reusable with respect to any future development in semantic web technologies.

Protégé 3.4 beta:

Protégé is an ontology editor and a knowledge-base editor developed at Stanford University to allow domain experts to build knowledge-based systems by creating and modifying reusable ontologies (Figure 6) [47]. We use Protégé system because it allows us to build a frame-based ontology that is capable of executing DL-based reasoning. The latest version of Protégé editor is Protégé 4.0. It is still in the early development stage and is not necessarily stable. Furthermore, being a new version of Protégé editor, it does not have all the plug-ins integrated into it. Protégé 3.4 beta, on the other hand, is an established version of protégé editor that is stable and integrated with a full suite of plug-ins to enhance its functionalities.

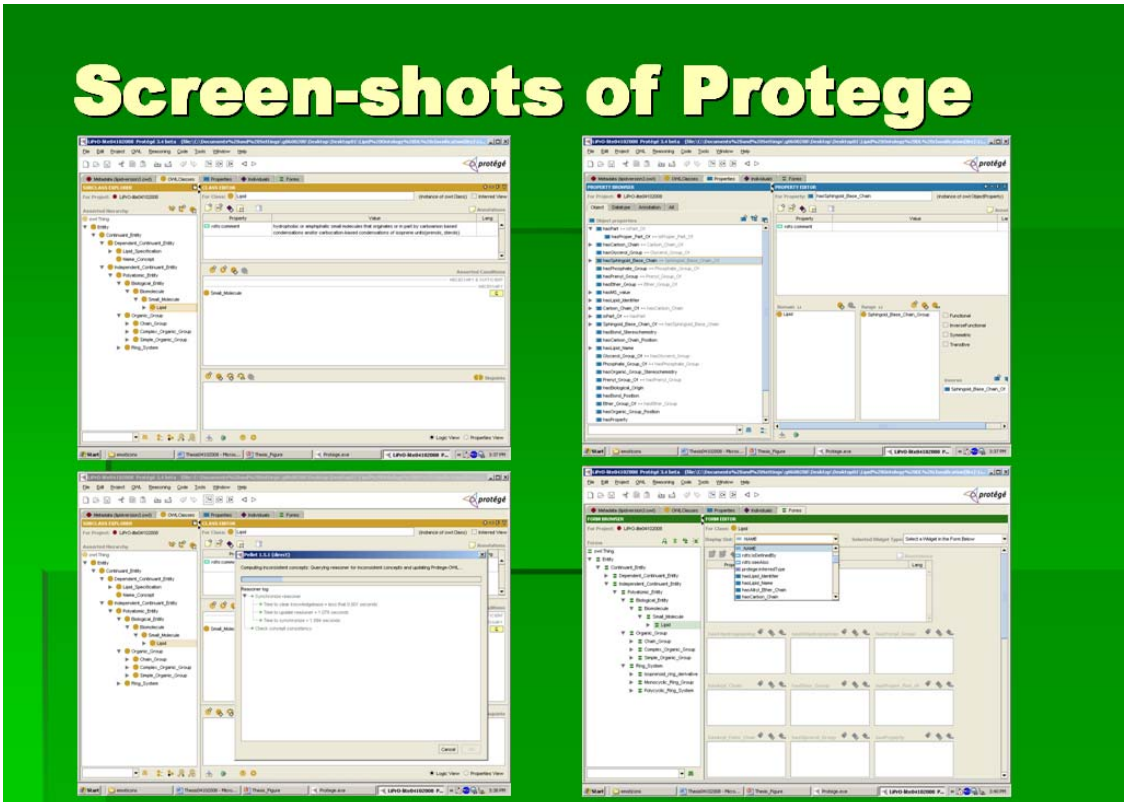


Figure 6: Various screenshots of the user interface provided by OWL editor, Protégé 3.4 beta

Protégé Plug-in use in the Lipid Ontology development process:

PROMPT:

The PROMPT plug-in (see Figure 7) is integrated into the Protégé editor to enable the management of multiple ontologies in Protégé environment, the PROMPT knowledge framework extends the capability of the Protégé editor in the following ways [48]:

- compare different versions of the same ontology
- map one ontology to another
- merge two ontologies into one
- extract a part of an ontology and add it into another ontology

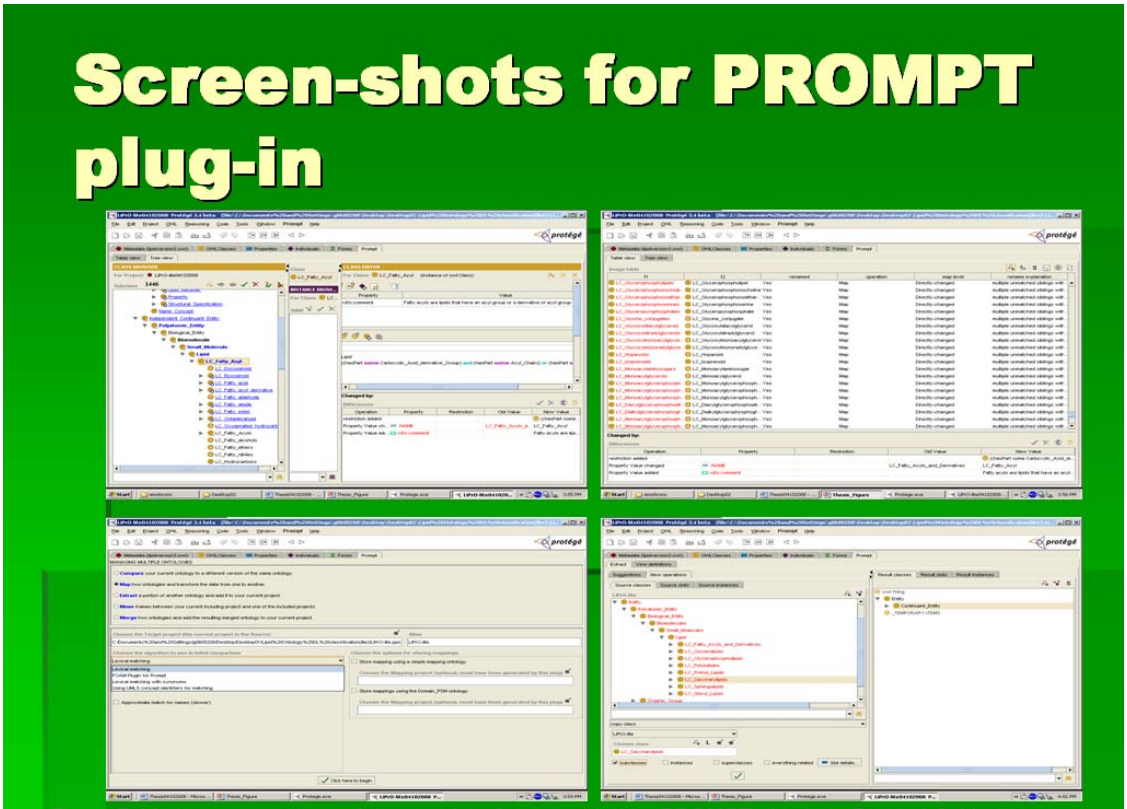


Figure 7: Various screenshots of the user interface provided by PROMPT plug-in in Protégé 3.4 beta

OWL-Viz:

OWLViz (see Figure 8) is a plug-in built to be used in conjunction with Protégé editor. It enables class hierarchies in an OWL Ontology to be viewed and incrementally navigated, allowing comparison of the asserted class hierarchy and the inferred class hierarchy [49].

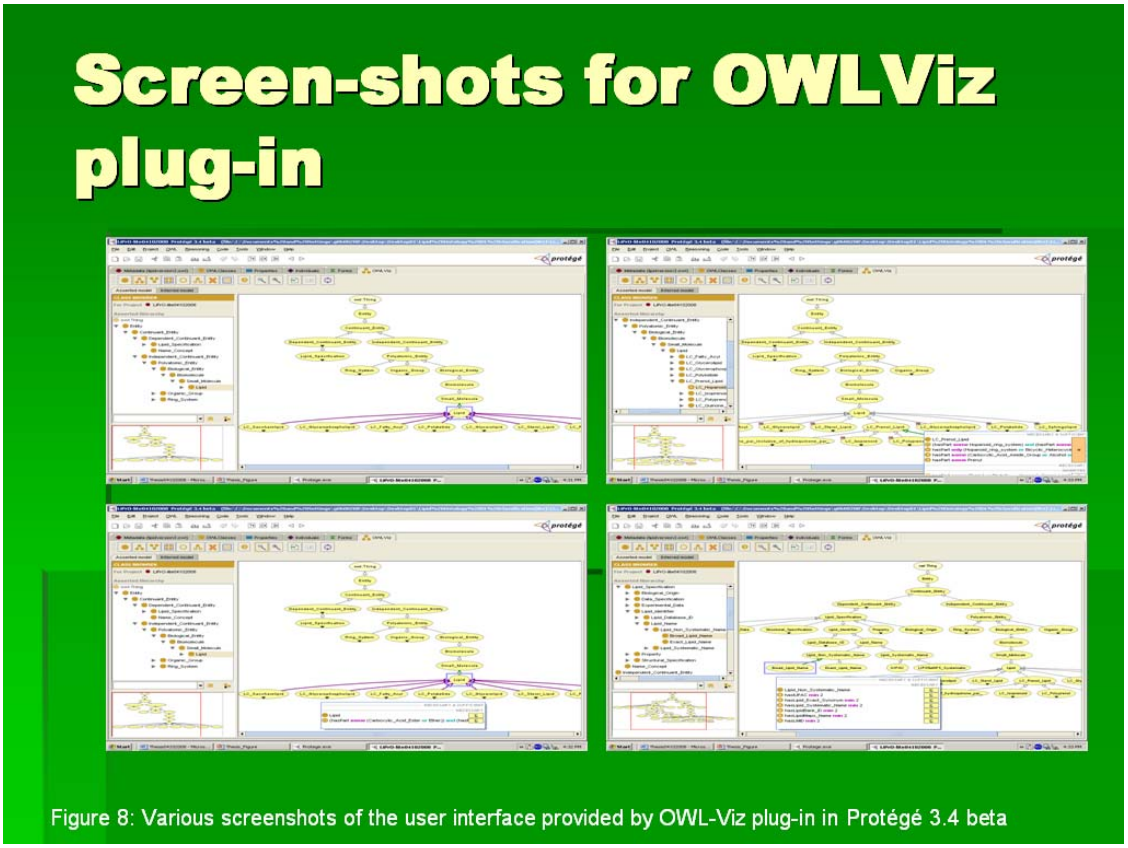


Figure 8: Various screenshots of the user interface provided by OWL-Viz plug-in in Protégé 3.4 beta

Jambalaya:

Jambalaya (see Figure 9) is a plug-in created for Protégé editor and it provides an integrated environment that utilize SHriMP(Simple Hierarchical Multiple Perspective) to visualize the knowledge bases created by the user [50]. SHriMP enables an end user to better browse, explore and interact with complex information spaces of an ontology.

Screen-shots for Jambalaya plug-in

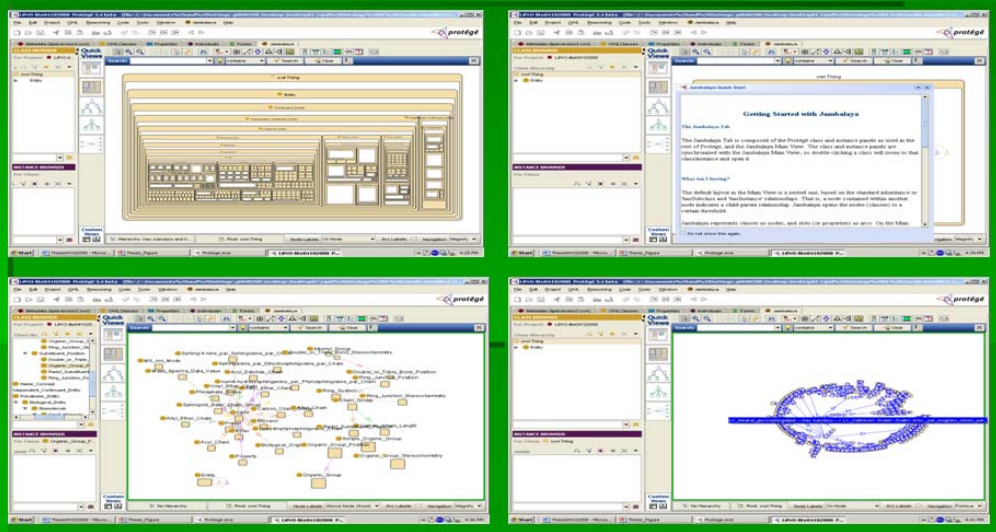


Figure 9: Various screenshots of the user interface provided by Jambalaya plug-in in Protégé 3.4 beta

Chapter III: Representing the World of Lipids, Lipid Biochemistry, Lipidomics and Biology in an Integrative Knowledge Framework

Our goal is to take advantage of the combination of the OWL [20, 51] framework with expressive Description Logics (DL) without losing computational completeness and decidability of reasoning systems. We use Protégé 3.4 beta [47] as a knowledge representation editor. The Ontology is designed with a high level of granularity and is implemented in the OWL-DL language. During the knowledge acquisition and data integration phase of ontology development, we have consulted lipid content in the form of database annotations, texts from the scientific literature, and entries within distributed biological databases.

1) Lipid Ontology 1.0

The Lipid Ontology 1.0 is developed to integrate lipid database entries and the bibliographic information associated to it. The ontology is partially specified by the data schema of an in-house lipid data-warehouse system, LipidDW [34]. LipidDW is a data warehouse system that sought to provide a simple platform where an end user can view related information (pathway, enzyme, protein, disease) about a specific lipid entity.

Lipid Ontology 1.0 is an application ontology designed to work together with a full-text literature acquisition pipeline and knowledge visualization platform (Knowlegator) to integrate bibliographic information with the existing data from lipid databases and to provide an intuitive visual query and navigation of lipid-centric information to end users. Knowlegator(Knowledge naviGator) is a tool that allows navigation of A-box instances

through an intuitive interface capable of converting a visual query built by a naïve end user into the query language syntax that communicates with the knowledgebase (instantiated ontology) for relevant information [32]. When fully instantiated, this ontology accounts for 10,789 lipids instances from LIPID MAPS (inclusive of 749 overlapping lipids from KEGG and 2897 overlapping lipids from LipidBank).

1.2) Ontology Description

1.2.1) Upper Ontology Concepts

We have incorporated top level, generic concepts into the upper ontology of Lipid Ontology 1.0(Figure 10). These concepts enable Lipid Ontology 1.0 to accept ontologies from other knowledge domain as orthogonal modules. These are generic concepts relevant to lipidomics or lipid biology, namely Diseases, Functional_Category, Processes, Isomer, Experimental_Protocol, Specification, Pathways, Biological_Entity(inclusive of Cell, Suborganellar_Component, Subcellular_Organelle, Biomolecules)(Table 6). The choice of upper ontology concepts enables Lipid Ontology to be built with a high level of modularity so as to provide a seamless integration of other biologically relevant knowledge domain into Lipid Ontology 1.0.

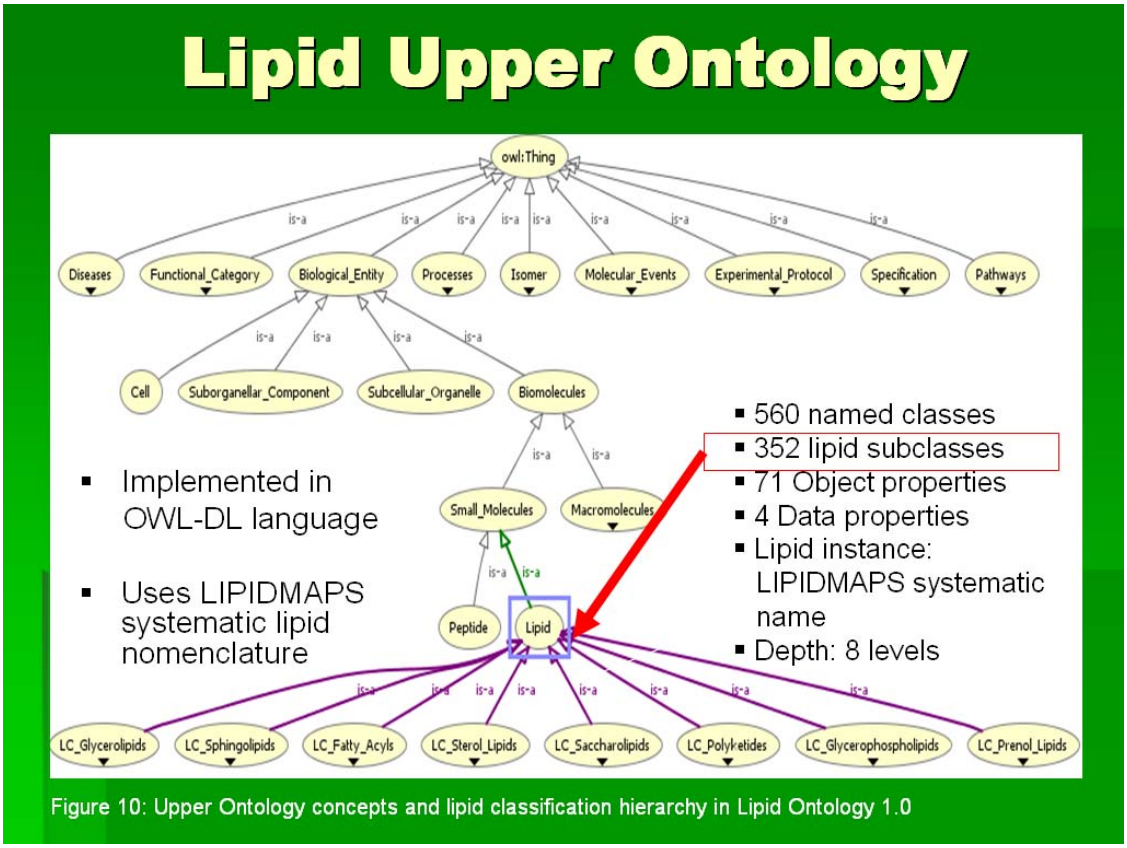


Figure 10: Upper Ontology concepts and lipid classification hierarchy in Lipid Ontology 1.0

Concept name	No. of Concepts
Biological entity	387
Data Source	1
Diseases	28
Experimental Protocol	41
Functional category	75
Isomer	20
Molecular events	2
Pathways	3
Processes	3
Specification	112
Total number of Concepts	672

Table 6: Current numbers of concepts in Lipid Ontology 1.0 divided across 10 sub-concepts

1.2.2) Lipid Concepts

Information about individual lipid molecules is modeled in the Lipid and Lipid Specification concepts. The Lipid concept is a sub-concept of Small_Molecules subsumed by the super-concept of Biomolecules. We have included the LIPID MAPS systematic classification hierarchy under the Lipid concept (Figure 10). The hierarchy consists of 8 major lipid categories and in total has about 352 lipid subclasses. The LIPID MAPS systematic name is modeled as an instance of a lipid. This instantiation of lipids is further extended to include lipids that are not classified in LIPID MAPS by instantiating these lipids with InChI. The use of the LIPID MAPS systematic name connects the LIPID MAPS classification system to other lipid associated information found in the Lipid_Specification concept and the rest of the ontology. The Lipid_Specification is a super-concept representing information about individual lipids (Table 7). The Lipid_Specification concept entails the following sub-concepts; Biological_Origin, Data_Specification (with a focus on high throughput data from Lipidomics), Experimental_Data (mainly mass spectrometry data values of lipids), Properties, Structural_Specification and Lipid_Identifier (that carries within it 2 other sub-concepts; Lipid_Database_ID and Lipid_Name) (Figure 11).

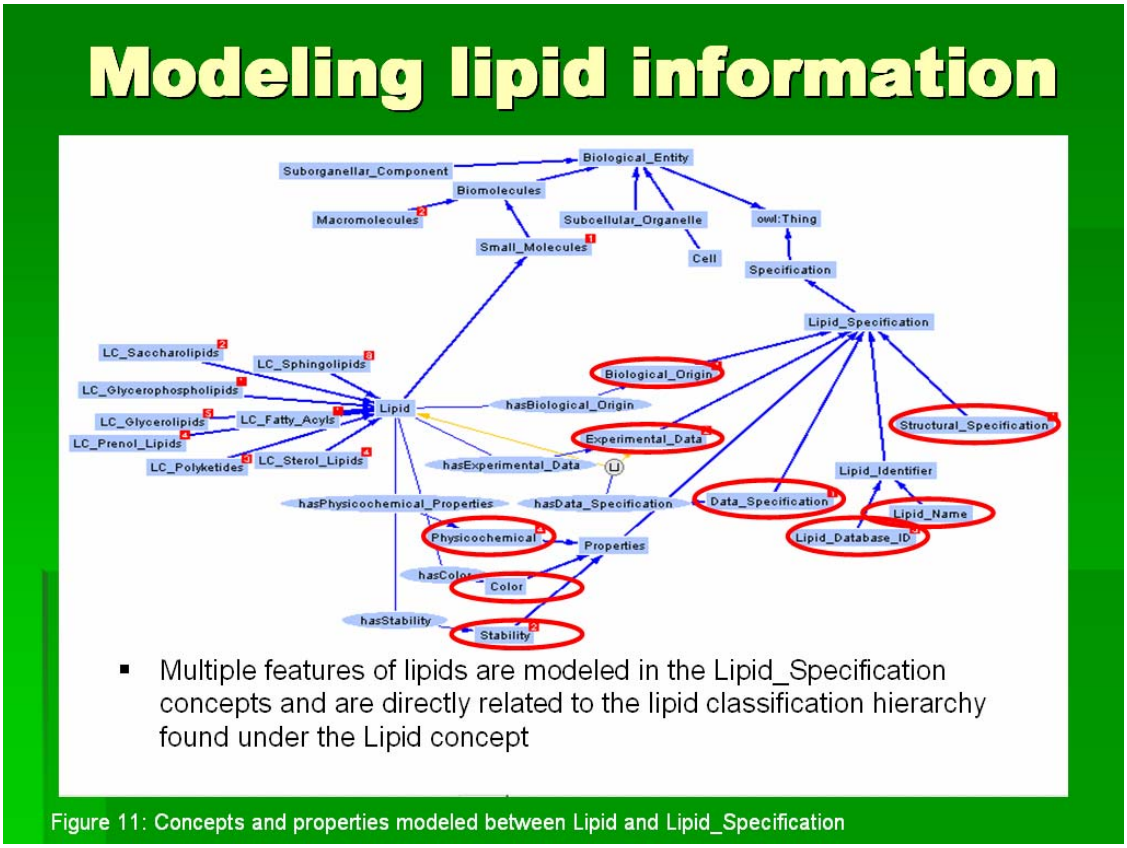


Figure 11: Concepts and properties modeled between Lipid and Lipid_Specification

Domain	Property	Range
Lipid	hasBiological_Origin	Biological_Origin
Lipid	hasData_Specification	Data_Specification
Lipid	hasExperimental_Data	Experimental_Data
Lipid	hasLipid_Identifier	Lipid_Identifier
Lipid	hasProperties	Properties
Lipid	hasStructural_Specification	Structural_Specification

Table 7: Relationship (domain, property and range) between Lipid sub-concept and other sub-concepts under Lipid_Specification

Provision for Database Integration

To facilitate data integration each Lipid instance is related to other databases with the hasDatabaseIdentifier property (Table 8). The object property hasDatabaseIdentifier connects a lipid instance to a database identifier. Specifically, our lipid ontology is designed to capture database information from the following databases: Swiss-prot, NCBI OMIM and PubMed, BRENDA and KEGG. Moreover, we have also made provisions in the ontology for it to store information from NCBI taxonomy database. The database record identifiers from each database are considered as instances of the respective database record. Identifier concepts are subsumed by a database specific superclass. For example, the Swiss-Prot_ID concept is subsumed by the Protein_Identifier super-concept which is in turn subsumed by the Protein_Specification super-concept. The presence of a Protein_Specification super-concept is provisional, should we decide to enrich the ontology with protein related information.

Domain	Property	Range	Database source
Lipid	hasSwiss-Prot_ID	Swiss-Prot_ID	Swiss-Prot
Lipid	hasOMIM_ID	OMIM_ID	OMIM
Lipid	hasEC_num	EC_num	BRENDA
Lipid	hasKEGG_ID	KEGG_ID	KEGG
Lipid	hasPMID	PMID	PUBMED

Table 8: Relationships (domain, property and range) between Lipid sub-concept and other sub-concepts that relates to external databases

1.2.4) Lipid-Protein Interactions

The inclusion of lipid-protein interactions in the ontology, necessitates the existence of the concept Protein which is subsumed by Macromolecule and Biomolecule concepts. The systematic name of a protein in the Swiss-Prot database serves as an instance of the Protein concept. Lipid instance is related to a protein instance by the object property InteractsWith_Protein (see Figure 12).

1.2.5) Lipids and Diseases

Information about lipids implicated in disease can also be modeled. We have added a primitive concept of Disease in the ontology. A disease name is considered as a disease instance which is related to a lipid instance by the object property hasRole_in_Disease property (see Figure 12).

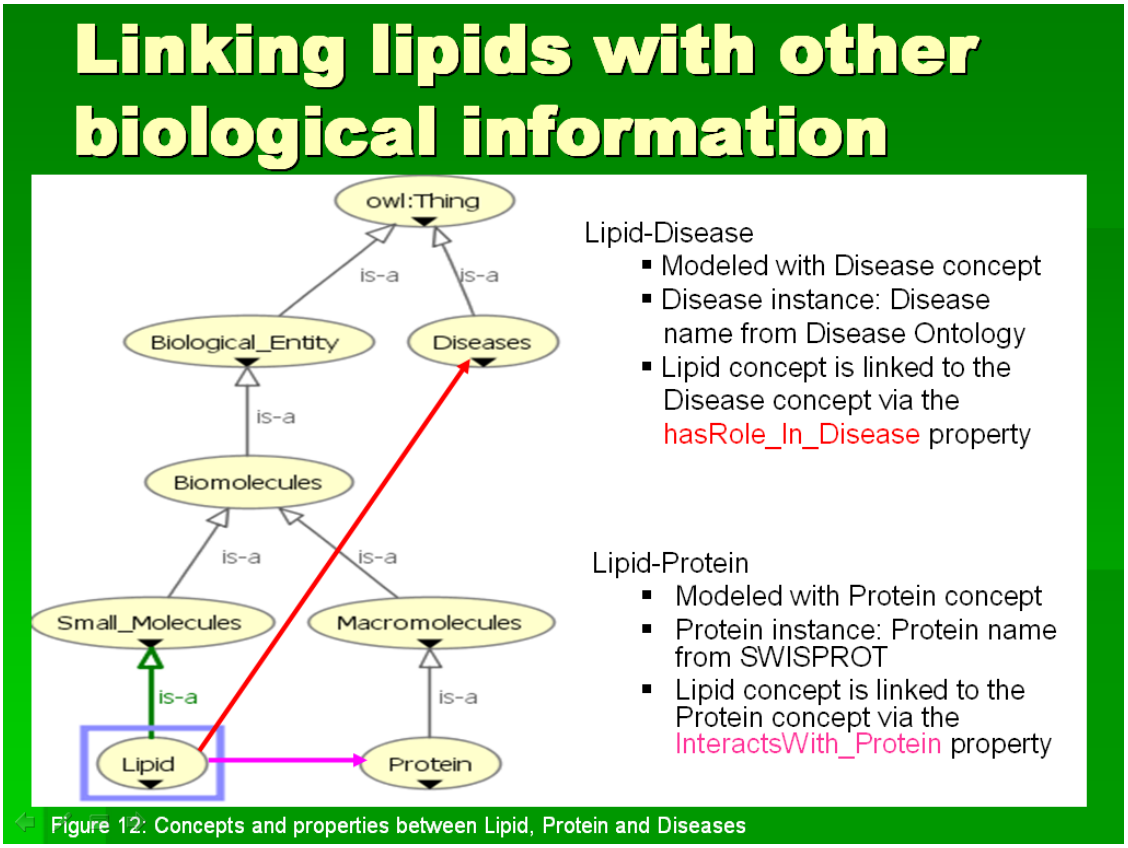


Figure 12: Concepts and properties between Lipid, Protein and Diseases

1.2.6) Modelling Lipid Synonyms

Due to the inattentive use of systematic lipid classifications, a lipid molecule can have many synonyms which need to be modeled into the ontology. In our Lipid Ontology, a lipid instance is a LIPID MAPS systematic name or an InChI and synonyms include the IUPAC names, lipid symbols and other commonly used lipid names (both scientific and un-scientific ones). We address the multiple name issue by introducing two sub-concepts, Lipid_Systematic_Name and Lipid_Non_Systematic_Name (see Figure 13). These two concepts are sub-concepts of Lipid_Identifier, which is subsumed by the super-concept Lipid_Specification. For every LIPID MAPS systematic name, there is typically one

IUPAC systematic name and one or more non systematic names. Every LIPID MAPS systematic name can be related to an IUPAC systematic name via hasIUPAC property and to non-systematic names via hasLipid_non-Systematic_Name property. A non-systematic name is related to an IUPAC name via a hasIUPAC_synonym property. In the same way, the IUPAC name is related to non-systematic name via hasBroad_Lipid_Synonym and hasExact_Lipid_Synonym properties. Lastly, the non-systematic name and IUPAC name are related to the LIPID MAPS systematic name via a hasLIPIDMAPS_synonym property. The current ontology model does not account for a non-systematic name that has other non-systematic names as its synonyms, i.e a direct synonym relationship between 2 non-systematic names. In order to identify this type of relation we have to deduce such relationship in an indirect manner. Where a non-systematic name is related to a systematic name, the systematic name can be examined for other non-systematic names. As long as there is more than one non-systematic name found linked to the systematic name, we can be certain that these non-systematic names are synonyms of one another.

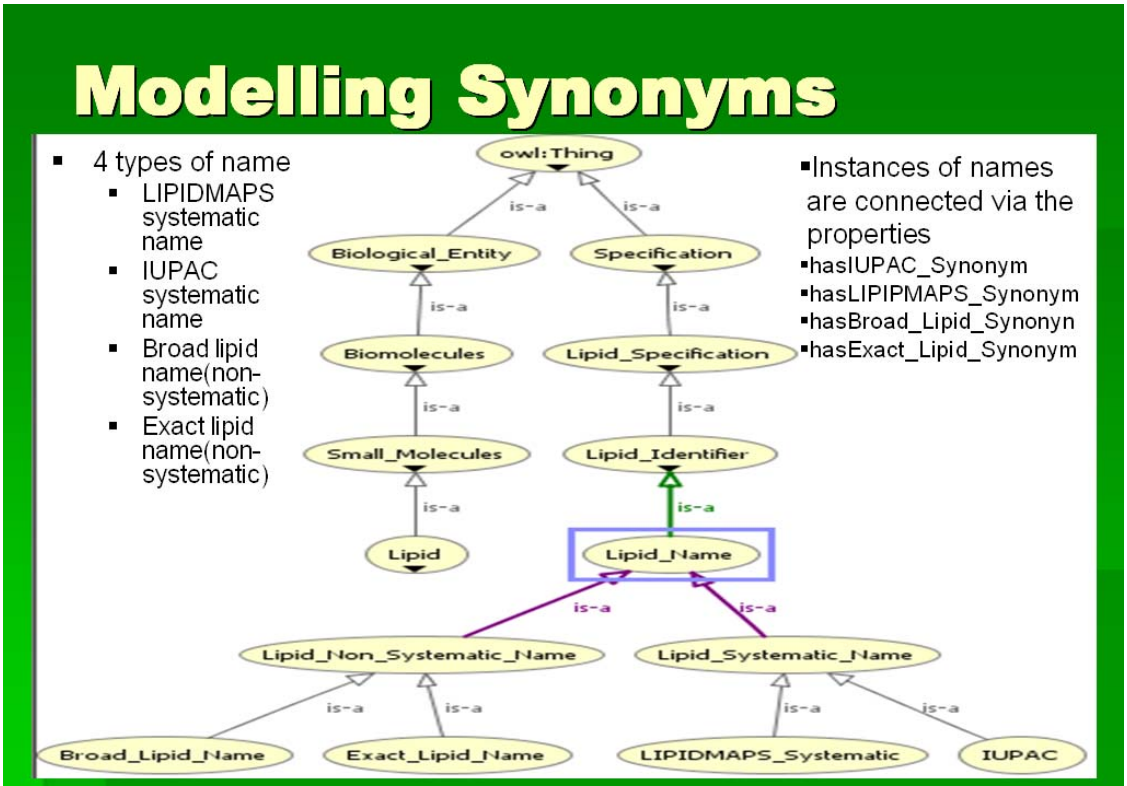


Figure 13: Concepts and properties used to model lipid synonyms

1.2.6.1) Extending Synonym Modeling

A broad lipid name is a broad synonym that describes several lipid molecules in one go. In our ontology, it is related to the Lipid concept and other name concepts such as IUPAC, Exact_Lipid_Name via a hasBroad_Lipid_Synonym property (see Figure 14). This means that if a non-systematic name has one or more, IUPAC names/LIPID MAPS systematic names/LIPID MAPS identifiers/KEGG compound identifiers/LipidBank identifiers, it is actually a broad lipid synonym. On the other hand, an exact lipid name is a non-systematic name that describe exactly 1 lipid molecule.

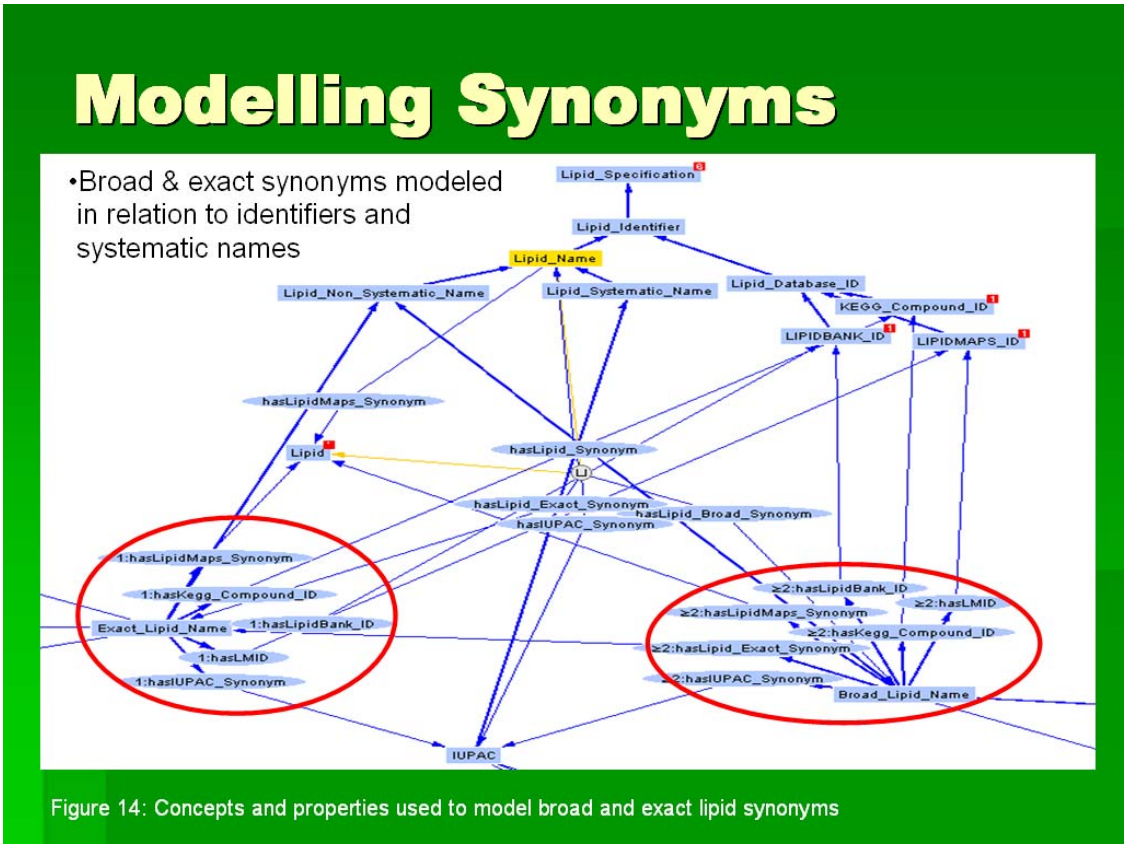


Figure 14: Concepts and properties used to model broad and exact lipid synonyms

1.2.7) Literature Specification

One of the main applications of Lipid Ontology 1.0 is to provide a knowledge framework where effective text-mining of lipid-related information can be carried out. To achieve this, we introduce a top level Literature_Specification super concept into the ontology so that non-biological units of information can be instantiated. The Literature_Specification comprises 10 sub-concepts, namely Author, Document, Issue, Journal, Literature_Identifier (with a sub-concept PMID, the PubMedIDentifier), Sentence, Title, Volume, Year (see Figure 15). The Document concept captures details of documents selected by the end user for subsequent text mining. It is related to multiple concepts

within the Literature_Specification hierarchy via several object properties. The Document concept also has 3 datatype properties; author_of_Document, journal_of_Document, title_of_Document that become instantiated with the author name, journal name and title of the article in the form of text strings. In future version we intend to adopt full Dublin Core units of document metadata by importing the OWL-DL version of this ontology and extend it to include our Sentence concept which is related to the concept Document via the occursIn_Document property. Sentence also has a datatype property, 'text_of_Sentence' that is instantiated by a text string from the documents that were found to have a lipid name and a protein name occurring in the same sentence. Sentence is related to Lipid and Protein concepts via the hasLipid and hasProtein object properties.

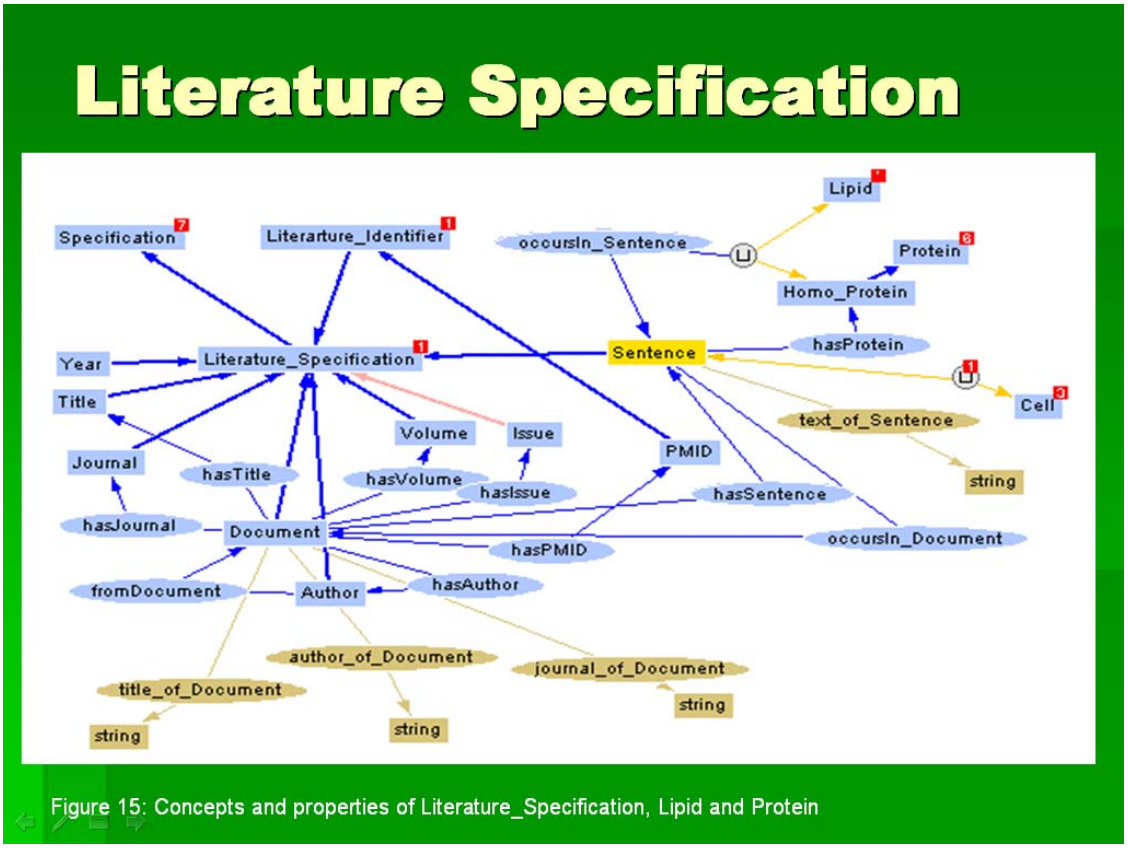


Figure 15: Concepts and properties of Literature_Specification, Lipid and Protein

2) Lipid Ontology Reference

A key purpose in lipidomics research is to understand the role of individual lipids or lipid classes in the onset and progression of diseases. Therefore, a knowledge representation framework capable to representing diseases are crucial to advancing knowledge in the study of diseases and is only sufficient if lipids are represented with respect to other biological entities such as enzymes, pathways, proteins and cells. In other words, the Lipid Ontology needs to make provision so that it can be connected to other ontological formalizations that describe concepts such as pathways, cell types, tissue types and disease classes. When connecting these ontologies, care must be taken to ensure ontologies incorporated are contextually consistent to the main ontology component, which in this case, would be Lipid Ontology 1.0.

The Lipid Ontology Reference is an integrative, comprehensive and reusable knowledge representation for the knowledge domain of lipids, lipid biology and lipidomics. It integrates as much conceptual information from other biological knowledge domain as possible and acts as a reference ontology where simpler, specialized application ontologies can be built from. At present, it integrates 5 ontologies to represent knowledge and relationships for the following knowledge domains, Disease, Pathway, Protein, Cellular Component, Cell and Tissue. Although it is a reference ontology, Lipid Ontology Reference is not OBO compliant because it needs to support application in the Knowlegator [32] visual query application. It is necessary that the ontology's semantic format do not differ too much from Lipid Ontology 1.0 so that application ontologies built from it remains compatible to the Knowlegator platform.

2.1) Ontology Description

2.1.1) Concept Alignment and Integration of Ontologies

We expect Lipid Ontology Reference to adequately describe the multifaceted information of a lipid instance, especially its relationships to other biochemical and biomedical related entities such as proteins, diseases, enzymes and pathways. Therefore, sufficient knowledge domain components needed to describe the relevant cellular phenomena must be built into the ontology.

Several ontologies are examined for suitability and subsequently, selected parts of these ontologies are re-used in the building of Lipid Ontology Reference.

Ontologies are either integrated directly into Lipid Ontology Reference via PROMPT [48] or imported into Lipid Ontology Reference by as local repositories.

2.1.2) Evaluation of GO for Alignment and Integration into Lipid Ontology Reference

Gene Ontology is a large and widely used ontology in the biomedical research field. Its annotation is very valuable to biomedical research community [43]. GO describes 3 aspects of biological phenomena, Molecular Function, Biological Process and Cellular Component [43]. We include Molecular Function and Biological Process of GO for the purpose of annotating the various biological entities in Lipid Ontology while Cellular Component of GO is considered as one of the biological entity in Lipid Ontology Reference (see Figure 16). Molecular Function and Biological Process are placed under

the concept GO_Molecular_Function and GO_Biological_Process, whereas Cellular Component of GO is placed under Cellular_Component in Lipid Ontology Reference. In principle, they can be considered as orthogonal to the Molecular_Entity_Functional _Classification, Processes and Cellular_Component concepts in Lipid Ontology Reference respectively.

2.1.2.1) Processes

Lipid Ontology Reference adopts directly the definition of biological process found in NCI terminology for oncology [42], instead of GO's Biological Process. This is because NCI describes the granularity of biological processes with greater degree of resolution. NCI defines Biological Process as a super-concept that encapsulates processes at various levels of granularity and includes generic concepts such as Cellular, Multicellular, Organismal, Population, Pathologic, Subcellular Process and Viral Function. GO does not make such distinctions and merely organize the process by their functions.

For example, a cellular process "leukocyte migration"(GO:0050900) and a subcellular process "antigen processing and presentation"(GO:0019882) of GO are arranged as immediate subclasses of "immune system process"(GO:0002376). "immune system process"(GO:0002376) itself has an unclear level of granularity. Furthermore, this class is arranged at the same level with the term "cellular process"(GO:0009987) and "cell killing"(GO:0001906), another cellular process.(Table 9)

Top level concept	Sub-concept	Distinction by Lipid Ontology Reference
immune system process GO:0002376		Unclear
	leukocyte migration GO:0050900	cellular process
	antigen processing and presentation GO:0019882	subcellular process
cellular process GO:0009987		cellular process
cell killing GO:0001906		cellular process

Table 9: Examples of concepts from Biological Process of Gene Ontology with unclear granularity according to the formalization of Lipid Ontology Reference

Under Lipid Ontology Reference's definition, "leukocyte migration"(GO:0050900), "cell killing" (GO:0001906) should be placed under "cellular process"(GO:0009987) while "antigen processing and presentation"(GO:0019882) should be placed under subcellular process concept.

2.1.2.2) Cellular Component

Lipid Ontology Reference defines cellular component as components of a cell and it makes distinction between cellular components (golgi apparatus, mitochondria, a complete organelle found in a cell) and subcellular components (components of a complete organelle). Such distinction is described differently in the Cellular Component of GO.

In Cellular Component of GO, terms for subcellular component and cellular component are all grouped together under the super-concept Cellular Component. For example,

terms at different level of granularity in GO such as cell, apical plasma membrane, transport vesicle are all classified under the super-concept Cellular Component. In this case, the term cell should not be classified as a cellular component because it is not a part of a cell according to Lipid Ontology Reference's definition. Similarly, apical plasma membrane is a part of an organelle and should not be classified together with transport vesicle, a complete organelle. Apical plasma membrane should be classified as a subcellular component according to Lipid Ontology Reference's definition.

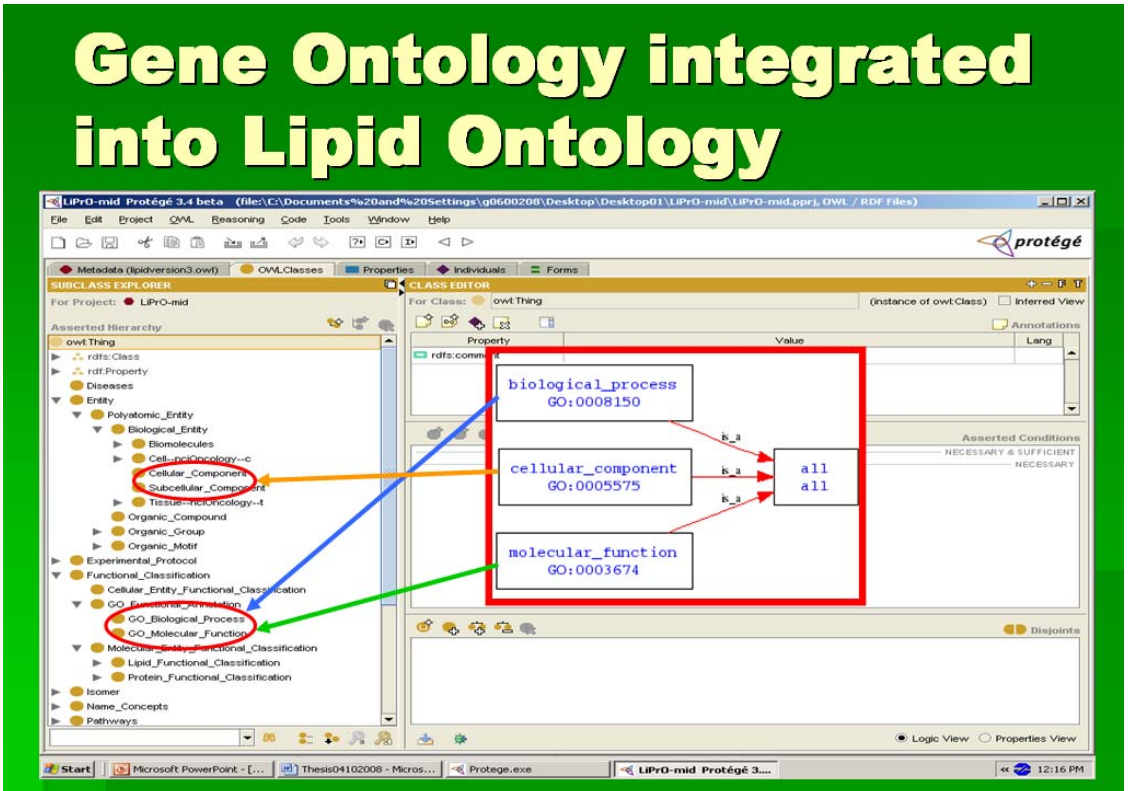


Figure 16: Concepts from Gene Ontology imported into Lipid Ontology Reference

GO handles part of an organelle by dividing a root concept with a term of cellular component with <cellular component term> concept and <cellular component term> part

concept. In this case, “plasma membrane” (GO: 0005886) would have a “part” counterpart of “plasma membrane part” (GO: 0044459). The term “apical plasma membrane” (GO: 0016324) is classified under “plasma membrane part” concept. All these terms are encapsulated within the upper class Cellular Component. In principle, all <cellular component term> part can be considered orthogonal to subcellular component in Lipid Ontology Reference (see Figure 17).

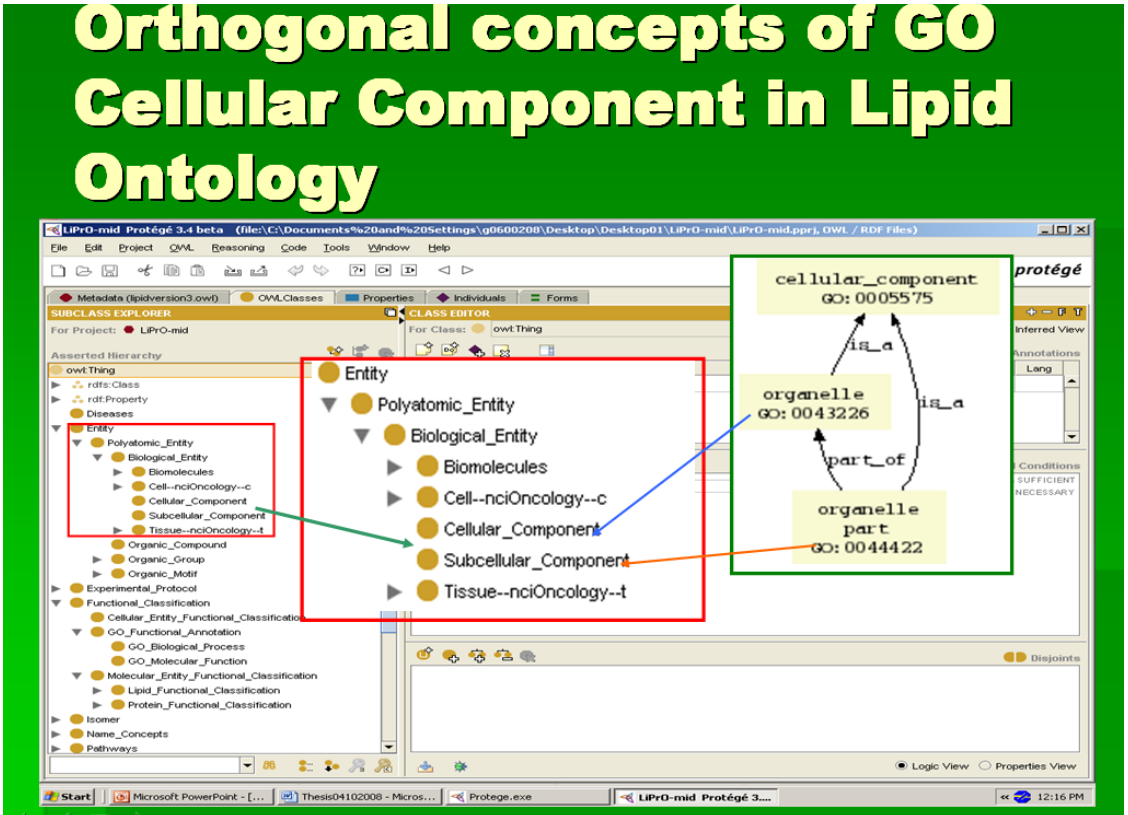


Figure 17: Concepts in Lipid Ontology Reference that are orthogonal to concepts of Cellular_Components in GO

In addition to that, GO also includes terms that are not suitable to define as part of an organelle such as “virion” (GO: 0019012), “extracellular matrix” (GO: 0031012), “synapse” (GO: 0045202), and “membrane-enclosed lumen”(GO: 0031974). As an

example, a membrane-enclosed lumen is a region of space between cells/tissues and is not necessary a part of an organelle (see Figure 18). It is clear that GO is ideally useful for annotation of gene product localization, rather than to describe cellular components as according to the formalization in Lipid Ontology Reference. For the time being, terms in Cellular Component of GO is placed under the Cellular_Component of Lipid Ontology Reference.

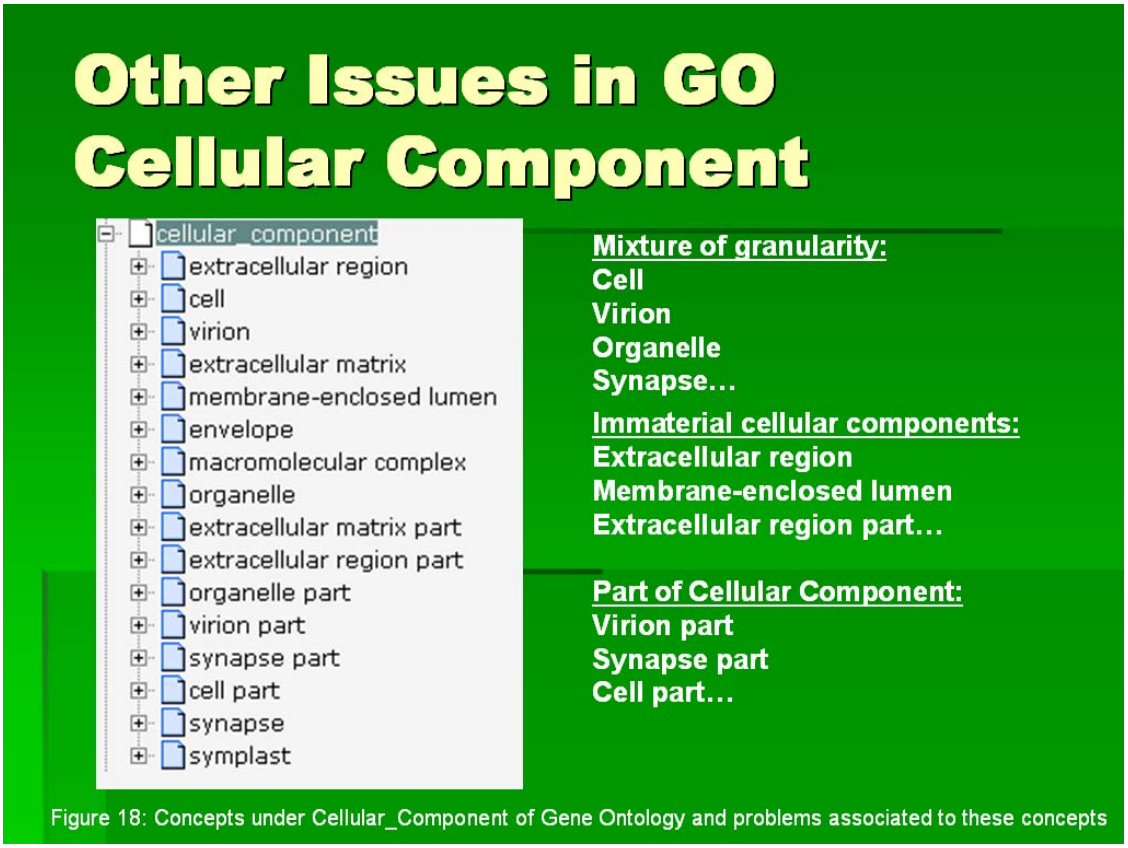


Figure 18: Concepts under Cellular_Component of Gene Ontology and problems associated to these concepts

2.1.3) Evaluation of Molecule Role Ontology for Alignment and Integration into Lipid Ontology Reference

The Protein concept is examined and is directly integrated into Lipid Ontology Reference under the Protein_Functional_Classification (see Figure 19). The Protein_Functional_Classification supplies concepts of functional role that a particular protein instance can play in a biological process.

The Chemical concept is examined and sub-concepts of molecule role irrelevant to lipids are removed from the Chemical concept before the Chemical concept was aligned and integrated into Lipid Ontology Reference. The Chemical concept is grouped together with Toxin and Enzyme_Chemistry (encapsulates enzyme reactants and effectors) under the Lipid_Functional_Classification concept where the Lipid_Functional_Classification supplies concepts of functional role that a particular lipid instance can play in a biological process (see Figure 19).

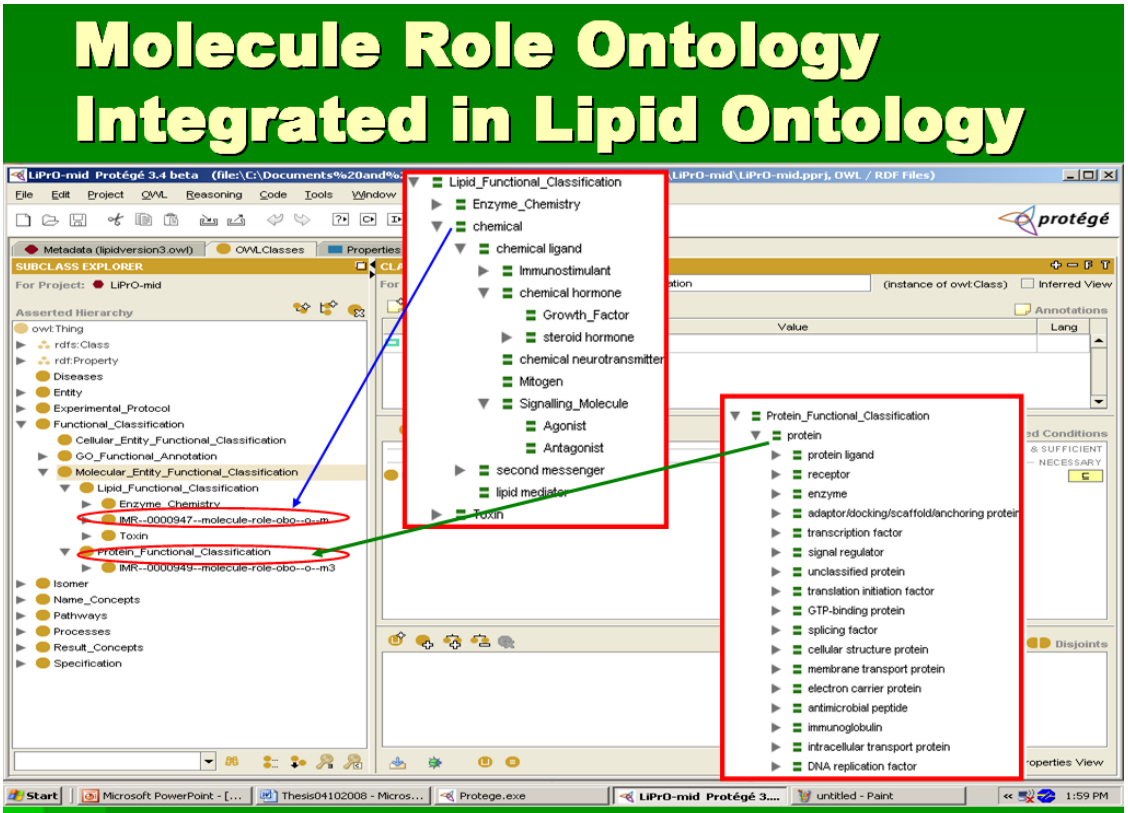


Figure 19: Concepts(Chemical & Protein) of Molecule Role Ontology incorporated into Lipid Ontology Reference

2.1.4) Evaluation of NCI Thesaurus for Alignment and Integration into Lipid Ontology Reference

Cell, Tissue, Organism, Biological Process concepts from NCI Thesaurus are examined and are integrated directly into Lipid Ontology Reference as orthogonal modules.

Disease_and_Disorder from NCI Thesaurus is placed under Diseases in Lipid Ontology Reference. It is an extensive list of disease terms. We have taken the initiative to simplify the list by removing redundant concepts, specifically for the Neoplasms section. NCI employs several means of classifying neoplasms, including using morphology, site of disease and tissue types. Identical terms are repeated several times due to different

approaches applied to classify neoplasms. We retain only the classification of Neoplasms by site in this iteration of Lipid Ontology Reference.

Concept aligned and integrated to LiPrO	Ontology	Equivalent Concepts in Lipid Ontology Reference	Integration Methodology
Biological_Process*	Gene Ontology[43]	GO_Biological_Process	OWL Import
Cellular_Component	Gene Ontology[43]	Cellular_Component	OWL Import
Molecular_Function*	Gene Ontology[43]	GO_Molecular_Function	OWL Import
Disease_and_Disorder	NCI Thesaurus[42]	Diseases	OWL Import
Cell*	NCI Thesaurus[42]	Cell	Ontology alignment
Tissue*	NCI Thesaurus[42]	Tissue	Ontology alignment
Organism*	NCI Thesaurus[42]	Organism	Ontology alignment
Biological_Process	NCI Thesaurus[42]	Processes	Ontology alignment
Pathway*	Pathway Ontology (http://purl.org/obo/owl/PW)	Pathways	Ontology alignment
Chemical	Molecule Role Ontology (http://purl.org/obo/owl/IMR)	Lipid_Functional_Classification	Ontology alignment
Protein	Molecule Role Ontology (http://purl.org/obo/owl/IMR)	Protein_Functional_Classification	Ontology alignment

* Concepts aligned and integrated into Lipid Ontology Reference with minimal modifications.

Table 10: All concepts aligned and integrated into Lipid Ontology Reference

3) Specialized Lipid Ontology for Apoptosis Pathway and Ovarian Cancer

As diseases are composed of multiple processes and interconnected pathways, visualization and subsequent guided exploration of pathways are crucial to the understanding of relevant medically important diseases. Lipid Ontology Ov is a specialized application ontology derived from the Lipid Ontology Reference to integrate bibliographic information and facilitate pathway exploration by the end user with the use of Knowlegator. Knowlegator provides an interactive query paradigm for pathway discovery from full-text scientific papers as well as navigation of annotations across

biological systems and data types. The ontology provides a query model to facilitate navigation of the pertinent sentences by researchers in specific fields of research, namely ovarian cancer, lipid-related pathways and acts as a knowledgebase when it is instantiated.

3.1) Ontology Description

To facilitate the navigation of pathway information we modify the existing Lipid Ontology Reference by incorporating Protein concepts under two newly defined superconcepts

(i) Monomeric_Protein_or_Protein_Complex_Subunit and

(ii) Multimeric_Protein_Complex.

Multimetic_Protein_Complex is a super-concepts that subsume other concepts polymeric protein complexes that are composed of more than one monomeric protein and they are asserted with necessary conditions where the membership requirement of these concepts is restricted by relevant cardinality and existential axioms.

For example, PP2A is a complex consisting of a common heterodimeric core enzyme, composed of a 36 kDa catalytic subunit (subunit C), and a 65 kDa constant regulatory subunit (PR65 or subunit A), that associates with a variety of regulatory subunits. Proteins that associate with the core dimer include three families of regulatory subunits B.

The concept of PP2A (complex) are defined the following necessary conditions.

“hasPart some PP2R” (subunit B)

“hasPart exactly 1 PR65” (subunit A)

“hasPart exactly 1 PP2C” (subunit C/catalytic subunit)

The incorporation of protein entities into the Protein concept are achieved either by importing protein entities found in Molecule Roles Ontology or by adding the names manually.

In total, we have incorporated 111 concepts of protein class under Multimetic_Protein_Complex and Monomeric_Protein_or_Protein_Complex _Subunit.

Similar to the scenario reported for lipids, every protein entity is related to instances found under concepts subsumed by Protein_Database_Identifier, namely GI_Accession, MGI_ID, Uniprot_ID and concepts subsumed by Protein_Name, specifically, Protein_Broad_Synonym and Protein_Exact_Synonym. The implementation of instances is similar to our previous use case applied to lipids.

The instantiation of these protein concepts brings to the ontology an additional layer of annotation that may be relevant to an end user, namely these instances can be interpreted as proteins with specific molecule role. Protein entities relate to one another via the property "hasProtein_Protein_Interaction_ with". Each protein entity then relates to a lipid entity via the property "interactsWith_ Lipid". These extensions facilitate query of protein-protein interactions derived from tuples found by the text mining of full text documents. In addition to that, a protein entity relates to a gene entity via the “isGene_Product” property.

Lastly, in the interest of connecting these biomolecules(protein and lipid) to relevant disease condition. We connect Protein and Lipid instances to instances of Disease via "participates_in_Disease-protein-" and "participates_in_Disease-lipid-" respectively. The property "participates_in_Disease-lipid-" is equivalent to "hasRole_in_Disease" in Lipid Ontology 1.0.

4) Conclusion

We describe 3 application ontologies, namely Lipid Ontology 1.0, Lipid Ontology Reference and Lipid Ontology Ov. These 3 ontologies are developed to support the knowledge visualization platform (Knowlegator) and provide an intuitive visual query and navigation of lipid centric information to end users. Lipid Ontology 1.0 is a basic application ontology that integrates bibliographic information with the existing data from lipid databases and provides a basic query model for the Knowlegator platform. Lipid Ontology Reference is built based on the content of Lipid Ontology 1.0 by integrating other OWL ontologies into Lipid Ontology 1.0. Lipid Ontology Reference provides a content rich reference from which other, simpler, specialized application ontologies can be developed. Lipid Ontology Ov is such an application ontology; and it has been applied to assess the lipidome of ovarian cancer with respect to apoptosis in the bibliosphere. For further discussion on the use of these application ontologies, please refer to Chapter V.

Chapter IV: Representing Lipid Entity

1) Lipid Classification Ontology (LiCO)

LiCO is a reference ontology created to share formalized definitions of lipid with the wider bio-ontology, bioinfomatics and lipidomics community. It is compliant to the requirement of OBO and is designed to be as orthogonal to OBO ontologies as possible. LiCO provides research communities with DL-based definition of lipids classified according the LIPID MAPS nomenclature. It describes lipid classes comprehensively with the use of DL axiomatic restriction and covers all 8 major categories of lipids classified by the LIPID MAPS consortium.

1.1) Ontology Description

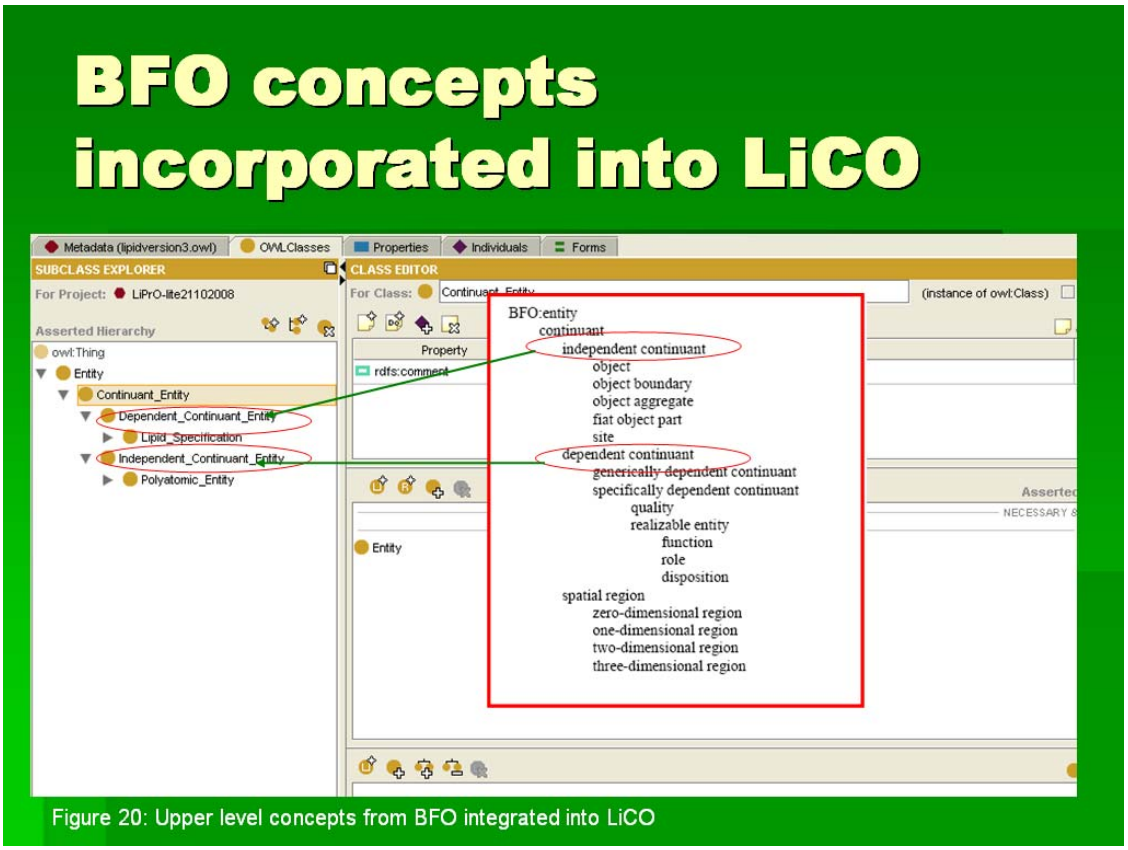
1.1.1) Upper Ontology Concepts

LiCO aims to share our knowledge of lipid definition with experts and scientists in the wider community. For this purpose, we re-design the Lipid Ontology 1.0 to be as orthogonal as possible to other ontologies. We achieve this by incorporating new upper ontology concepts, namely, the BFO upper ontology concepts and ChEBI upper ontology concepts.

1.1.1.1) BFO Upper Ontology Concepts

BFO upper ontology concepts are concepts compliant to the requirement of OBO. They represent the upper level categories common to domain ontologies developed by scientists in different domains and at different levels of granularity in a consistent fashion. We have re-used Continuant_Entity, Independent_Continuant_Entity from BFO (see

Figure 20). The use of these concepts enables the LiCO to be added on to BFO ontology as a module.



1.1.1.2) Upper Ontology Concepts from ChEBI

These are concepts used in ChEBI. We have re-used only 1 concept from ChEBI, namely the concept Polyatomic_Entities. Because ChEBI concepts are not necessarily OBO compliant and do not make distinction between the plural and singular forms, we modified the concept Polyatomic_Entities from the plural form to singular form, Polyatomic_Entity. The use of Polyatomic_Entity positions this concept as a concept that

is orthogonal to ChEBI without violating OBO or BFO compliance. This ensures that LiCO is orthogonal to ChEBI and can be added into ChEBI as a module.

1.1.2) OBO Compliance Assertion in Lipid Classification Ontology

The original Lipid Ontology 1.0 uses plural nouns to name lipid classes. This is because the lipid class in Lipid Ontology 1.0 is considered as a collection of lipid instances. Unfortunately, this representation of lipid is semantically and grammatical inconsistent due to how the subsumption hierarchy is specified in OWL-DL. The subsumption hierarchy in OWL-DL ontology is an "is_a" subsumption hierarchical relationship and the use of plural lipid classes is not compatible with the "is_a" subsumption relation. Similarly, the plural lipid classes are not compatible with most of the object properties use in the Lipid Ontology 1.0 because these properties were expressed as singular verb too. For example, to say that acylglycerols(plural subject) is_a(singular verb) lipids(plural predicate) is incorrect. Similarly, to say that acylglycerols(plural subject) has_LMID(singular verb and predicate) is also incorrect.

We correct this incorrect expression of English by changing all plurally named classes into the singular form. In addition to that, OBO criterion makes distinction between an object and a group of object. By re-expressing all classes in Lipid Ontology 1.0 as singular nouns, we are ensuring LiCO's classification is orthogonal to other OBO ontologies to a certain degree.

In addition to that, OBO community also discourages the inclusion of "and" and "or" in the name of a concept. Inclusion of "and" or "or" in a concept name suggests a plural subject and introduces unnecessary semantic ambiguities. We address this issue by simplifying concept names that carry "and" or "or" in them. Lipid classes such as Fatty_acids_and_conjugates are simplified to just Fatty_acid, the root chemical term of the original concept. In this case, we are saying that all subclasses and instances of Fatty_acids_and_conjugates are essentially Fatty_acid. Some lipid classes can not be simplified this way because the subclasses or instances are not the same as root chemical term of the original concept. An example of this is C22_bile_acids_alcohols_and_derivatives. It is re-expressed as C22 _bile_acid_structural_derivative and 3 subclasses, namely C22_bile_acid_derivative, C22_bile_acid_alcohol_derivative and C22_bile_acid are created under this newly named class. This is because C22_bile_acid_derivative, C22_bile_acid_alcohol_derivative and C22_bile_acid do shared structural similarity with the root chemical, C22_bile_acid but are not the same as the root chemical term.

1.1.3) Textual Definition

Another important principle that underlies an OBO compliant ontology is the provision of textual annotation for all terms in the ontology. In LiCO, it is our intention to provide textual annotation for all DL-defined lipid classes, except for Polyketide. We are currently in the process of supplying LiCO with textual definitions.

1.1.4) Concepts Re-used from Chemical Ontology

Prior to extending the ontology for classification tasks we have reviewed existing ontologies for reusable components. We have reviewed the Chemical Ontology for reuse of the Organic_Group concept hierarchy and have added 32 organic groups from Chemical Ontology into LiCO. This is done manually in the Protégé 3.4 beta editing environment. In addition to that, we create 63 new concepts under the Organic_Group super-concept. The Organic_Group concept hierarchy is reorganized and is asserted with new is-a relationship. In order to describe the lipids with complex chemical moieties, we rename the Organic_Group concept into Simple_Organic_Group and position it together with newly created Complex_Organic_Group and Chain_Group concepts under a new Organic_Group concept. The Simple_Organic_Group subsumes the chemical functional group concepts from the former Organic_Group while the Complex_Organic_Group subsumes concepts for complex chemical moieties such as Organic_Sugar_Group and Amino_Acid. In addition to that, we have also created the new Ring_System concept to describe lipids with ring structure.

1.1.5) Axiomatic and Relationship Constraints in LiCO

In Chemical Ontology, Organic_Compound are concepts with hasPart relationship to concepts under Organic_Group. The same property is used in LiCO to relate concepts subsumed by Lipid to concepts subsumed by Organic_Group. Inversely, an inverse property partOf is used to relate concepts subsumed by Organic_Group with concepts subsumed by Lipid.

Lipids are very complicated biomolecules and most lipids can only be adequately classified with more than one distinct functional group. Lipids are defined by multiple sets of organic groups and these definitions are used to restrict the membership of individual lipids to specific classes of lipids. Therefore, description logic rules with greater complexity than what is used in Chemical Ontology are needed to describe lipids. For Lipid Ontology, we use 2 types of concept to define the structure of lipids. They are Organic_Group and Ring_System.

The Organic_Group consists of Chain_Group, Simple_Organic_Group and Complex_ Organic_Group. Simple_Organic_Group consists of concepts that describe basic functional groups whereas complex organic group encapsulates glycans and amino acids. Glycans, in particular, are used to classify lipids such as sacharrolipid, and other sugar-linked lipids such as sphingolipids. These concepts are used to extensively to define lipids in all 8 categories of lipids in LiCO.

The Ring_System consists of Isoprenoid_ring_derivative, Monocyclic_Ring_Group and Polycyclic_Ring_System. These concepts are used to define lipids that have at least one or more rings. Specifically, they are used mainly for Sterol_Lipid, Prenol_Lipid and other lipids with rings.

The Chain_Group consists of Carbon_Chain_Group and Sphingoid_Base_Chain_Group. Sphingoid_Base_Chain_Group is used exclusively for Sphingolipid whereas Carbon_ Chain_Group is applied to other lipid classes accordingly.

These concepts play a very important role as they formed the necessary structural description to define the identity of the lipid-based compound.

1.1.6) Hierarchical Classification of Lipids

Classes of lipids are organized in a hierarchical basis. The classes at the top of the hierarchy are restricted by necessary conditions that are more generic in nature. As the lipid classification hierarchy becomes deeper, necessary conditions that are more specific are used to define the membership requirement for a particular class of lipid. At the end of hierarchy, lipid classes are restricted by necessary and sufficient conditions and closure axioms.

There are 2 ways to assert greater specificity as we go down hierarchy.

The first way involves specifying the subclass of the present class to restrict the definition of a lipid. Necessary conditions such as "hasPart some Carboxylic_Acid_derivative_Group" can be further specified by specifying the subclass of Carboxylic_Acid_derivative_Group, which is described in the example below as an Aldehyde.

For example, Fatty_Aldehyde is a Fatty_Acyl with at least one Aldehyde. It has the following necessary condition.

"hasPart some Carboxylic_Acid_derivative_Group(inherited from Fatty_Acyl)

hasPart some Aldehyde"

The second way involves the use of cardinality axiom (see Table 11).

The Cardinality axiom can be applied to concepts at any level. Once it is declared, the cardinality axiom restricts the number of a particular concept to be allowed in a restriction. When it is applied to Fatty_Aldehyde, we can declare "hasAldehyde_Group exactly 1" in the necessary and sufficient conditions. The same Cardinality axiom has been applied to members of Chain_Group as well. This is particularly useful when a lipid class can be defined by the number of certain organic group concept or Chain_Group concept.

For example, Triacylglycerol is an Acylglycerol with 3 acyl chains. It is restricted with the following necessary conditions

"hasAcyl_Chain exactly 3"

Concepts(Range)	Property
Carbon_Chain_Group	hasCarbon_Chain
Allyl_Ether_Chain	hasAllyl_Ether_Chain
Acyl_Chain	hasAcyl_Chain
Alkyl_Ether_Chain	hasAlkyl_Ether_Chain
Meromycolic_Chain	hasMeromycolic_Chain
Acyl_Ester_Chain	hasAcyl_Ester_Chain
Vinyl_Ether_Chain	hasVinyl_Ether_Chain
Alkyl_Chain	hasAlkyl_Chain
Glycerol	hasGlycerol_Group
Sphingoid_Base_Chain_Group	hasSphingoid_Base_Chain
Dehydrophytosphingosine_Chain	hasDehydrophytosphinganine_Chain
Sphing-4-nine_par_Sphingosine_par_Chain	hasSphing-4-enine_Chain
num4-hydroxysphinganine_par_Phytosphingosine_par_Chain	has4-hydroxysphinganine_Chain
Sphinganine_par_Dihydrosphingosine_par_Chain	hasSphinganine_Chain
Phosphate_Group	hasPhosphate_Group
Prenyl	hasPrenyl_Group
Ether	hasEther_Group
Phytyl	hasPhytyl_Group

*For list of lipid applied with cardinality group (see Appendix C)

Table 11: Concepts (range) and corresponding properties in LiCO that enable definitions of lipid with cardinality axioms

1.1.7) Closure Axioms

The closure axiom is applied to a defined concept at the end of a concept hierarchy. Superclasses and other primitive concepts are not closed by closure axiom to avoid inconsistency among disjointed sibling classes. Closure axioms restrict the type of relationship constraints allowed for a lipid class.

1.1.8) Definitions of Fatty_Acyl

The fatty acyls are a diverse group of molecules synthesized by chain-elongation of an acetyl-CoA primer with malonyl-CoA (or methylmalonyl-CoA) groups [2]. We define a Fatty_Acyl as a lipid that has at least one Carboxylic_Acid_derivative_Group and at least one Acyl_Chain.

An example of Fatty_Acyl is Docosanoid. Docosanoid is described as a subclass of Fatty_Acyl. It inherits from Fatty_Acyl, the Carboxylic_Acid_derivative_Group and the Acyl_Chain. This Carboxylic_Acid_derivative_Group is further specified to be a Carboxylic_Acid in Docosanoid, whereas the Acyl_Chain of Docosanoid was further specified with a cardinality axiom in conjuction with the property hasAcyl_Chain. Consequently, Docosanoid is defined to have only 1 Acyl_Chain. Moreover, Docosanoid has multiple and distinct functional groups such as Carboxylic_Acid, Alkenyl_Group, Alcohol and Cyclopentenone. These functional groups are made to relate with Dosocanoid via the property “hasPart” in conjuction with the existential axiom “some”. A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Docosanoids so that lipids of this class can only

have the following functional groups, namely, Carboxylic_Acid, Alkenyl_Group, Alcohol, Cyclopentenone and Acyl_Chain. (see Table 12)

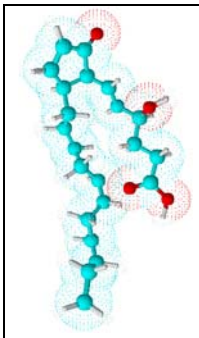
	Necessary and Sufficient Conditions
	LC_Fatty_Acyl (hasPart some Carboxylic_Acid) and (hasPart some Alcohol) and (hasPart some Alkenyl_Group) and (hasPart some Cyclopentenone) and (hasAcyl_Chain exactly 1) *hasPart only (Carboxylic_Acid or Alcohol or Alkenyl_Group or Cyclopentenone or Acyl_Chain)*
	Necessary Conditions inherited from LC_Fatty_Acyl
	((hasPart some Carboxylic_Acid_derivative_Group) and (hasPart some Acyl_Chain)) or (hasPart some Alkyl_Chain)

Table 12: DL definition for docosanoid (closue axiom in italics)

1.1.8.1) Axiomatic and Relationship Constraints for Exceptional Lipid Classes in Fatty_Acyl

Although most lipids can be classified by functional groups, certain lipids within the LIPID MAPS nomenclature are found in classes even though these lipids do not have the required functional groups. This is because the LIPID MAPS nomenclature classifies lipids based on their chemical structure or their biosynthetic origin. For example, lipids such as Fatty_alcohol, Fatty_Nitrille, Fatty_ether and Hydrocarbon are classified by LIPIDMAPS as a member of Fatty_Acyl although they do not have an Acyl_Group. In order to reconcile this contradicting decision, we expand the definition of Fatty_Acyl to include Alkyl_Chain, a characteristic structure of those exceptional Fatty_Acyl classes. A Fatty_alcohol inherits an Alkyl_Chain from Fatty_Acyl and is further defined to have only 1 Alkyl_Chain in the necessary and sufficient condition. This necessary and

sufficient condition also includes a "hasPart" property that connects Fatty_alcohol to an Alcohol concept. Such a definition enables us to include lipids without an Acyl_Group as a member of Fatty_Acyl (see Table 13). In addition to that, we create a new lipid class, namely Fatty_Acyl_derivative, a subclass of Fatty_Acyl where those exceptional lipids are classified as members.

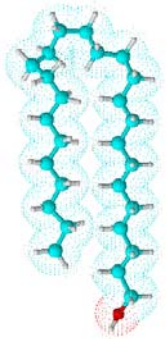
	Necessary and Sufficient Conditions
	LC_Fatty_acyl_derivative (hasPart some Alcohol) and (hasAlkyl_Chain exactly 1) hasPart only (Alcohol or Alkyl_Chain)
	Necessary Conditions inherited from LC_Fatty_acyl_derivative
	hasPart some Alkyl_Chain
	Necessary Conditions inherited from LC_Fatty_Acyl
	((hasPart some Carboxylic_Acid_derivative_Group) and (hasPart some Acyl_Chain)) or (hasPart some Alkyl_Chain)

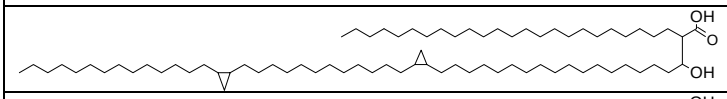
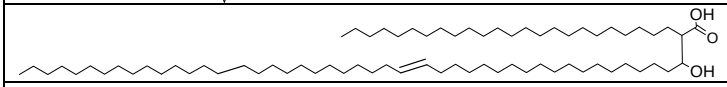
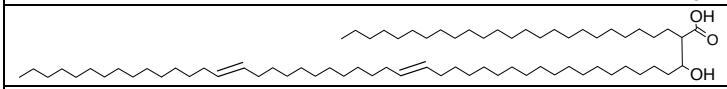
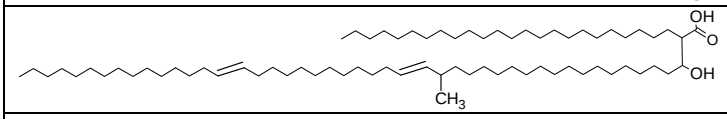
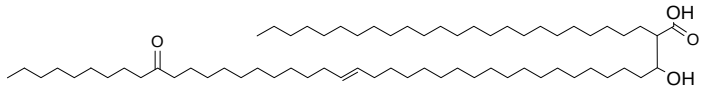
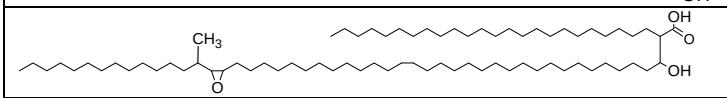
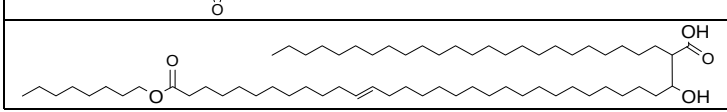
Table 13: DL definition for fatty alcohol

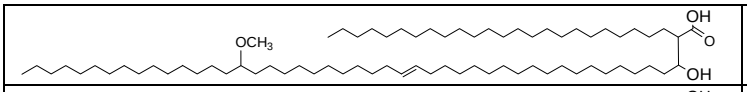
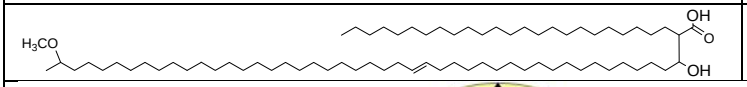
1.1.8.2) Extension of Mycolic Acid Class

Lipidomics primarily uses mass spectrometric analysis to characterize biologically important lipids and full structural characterization of lipids is elucidated with NMR. Mycolic acid is a family of structurally related lipids that constitute a major component of the cell wall of *Mycobacterium tubeculosis* and several other bacteria. They are medically important lipids which have been implicated in some of the most characteristic pathogenic features of mycobacterial disease. By 1998, there had been at least 500 known chemical structures of related mycolates [54]. By comparison, the LMSD currently contains only 3 mycolic acid records. There are therefore many mycolic acids with known structure that have yet to be systematically named or classified. Classification of these lipids is an important task needed for the system-level analysis of mycobacterial

pathogenesis and would contribute significantly to the molecular biology and lipidomics studies of mycolates from mycobacteria. Here we illustrate the extension of LiCO to include Mycolic_Acid class not found in LMSD and demonstrate the assignment of a real example of an alpha mycolate (see Figure 2) to the LiCO.

Based on LIPID MAPS nomenclature, we classify Mycolic_acid as a member of Fatty_Acid. We extend the classification of Mycolic_acid by adding 9 defined subclasses, Alpha_mycolic_acid, Alpha_prime_mycolic_acid, Alpha_1_mycolic_acid, Alpha_2_mycolic_acid, Keto_mycolic_acid, Epoxy_mycolic_acid, Wax_ester_mycolic_ acid, Methoxy_mycolic_acid and Omega-1_methoxy_mycolic_acid. These defined classes are distributed among 5 primitive classes, namely General_mycolic_acid, General_methylated_mycolic_acid, General_alpha_mycolic_acid, Oxygenated_mycolic_ acid, General_methoxy_mycolic_acid. (see Table 14)

Structure	Class type of Mycolic acid
	Alpha_mycolic_acid
	Alpha_prime_mycolic_acid
	Alpha_1_mycolic_acid
	Alpha_2_mycolic_acid
	Keto_mycolic_acid
	Epoxy_mycolic_acid
	Wax_ester_mycolic_acid

	Methoxy_mycolic_acid
	Omega-1_methoxy_mycolic_acid

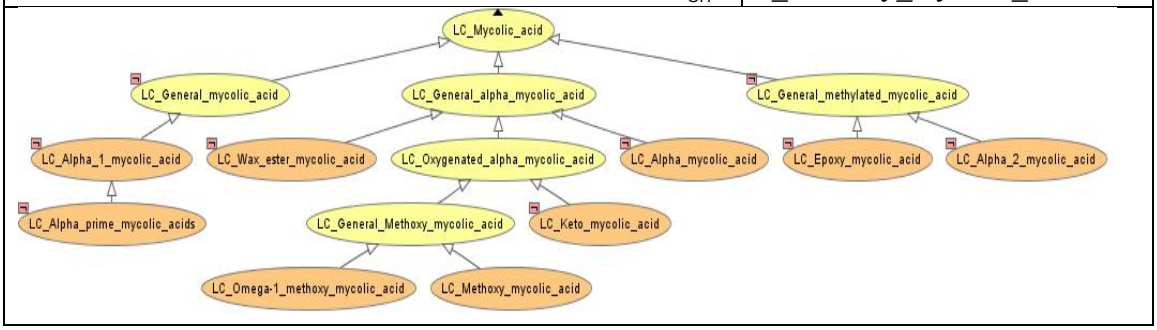


Table 14: Known classes of mycolic acid and their classification within LiCO

Alpha mycolic acid is a mycolic acid that has the following functional groups; carboxylic acid, cyclopropane and an alpha-hydroxyl acid group. The carboxylic acid group is a member of the acyl group and it is not an ester group. Therefore, according to the classification scheme below, alpha mycolic acid must be a member of Fatty_Acyl.

Among members of Fatty_Acyl, only Octadecanoid, Docosanoid, Eisocsanoid and Fatty_Acid have Carboxylic_Acid. Alpha_mycolic_acid does not have a cycloketone group and therefore, it cannot be Docosanoid, Eicosanoid or Octadecanoid. Therefore, it is a member of Fatty_Acid. Among members of Fatty_Acid, only Mycolic_acid has Alpha-Hydroxy_Acid_Group and a Meromycolic_Chain. Therefore, alpha mycolic acid is classified under this class of Fatty_Acid.

Because Alpha_mycolic_acid is the only class that accepts mycolic acid with Cyclopropane, the lipid example in Figure 2 is classified as a member of Alpha_mycolic_acid. (see Table 15)

	Necessary and Sufficient Conditions
	LC_General_alpha_mycolic_acid hasPart some Cyclopropane hasPart only (Alkenyl_Group or Alpha-Hydroxy_Acid_Group or Cyclopropane or Carboxylic_Acid or Meromycolic_Chain)
	Necessary Conditions inherited from LC_General_alpha_mycolic_acid
	hasPart some (Cyclopropane or Alkenyl_Group)
	Necessary Conditions inherited from LC_Mycolic_acid
	(hasPart some Alpha-Hydroxy_Acid_Group) and (hasMeromycolic_Chain exactly 1)
	Necessary Conditions inherited from LC_Fatty_acid
	(hasPart some Carboxylic_Acid) and (hasAcyl_Chain exactly 1)
	Necessary Conditions inherited from LC_Fatty_Acyl
	((hasPart some Carboxylic_Acid_derivative_Group) and (hasPart some Acyl_Chain)) or (hasPart some Alkyl_Chain)

Table 15: DL definition for alpha mycolic acid

1.1.9) Definitions of Glycerophospholipid

Glycerophospholipids are glycerol-containing lipids that also have at least one phosphate headgroup. Depending on the biological source, glycerophospholipids may be subdivided into distinct classes based on the nature of the polar headgroup at the *sn-3* or *sn-1* position of the glycerol backbone [2]. We define Glycerophospholipid as a lipid that has at least a Carboxylic_Acid_Ester or Ether, at least a Glycerophosphate_Group and at least a carbon chain from the Carbon_Chain_Group.

An example of Glycerophospholipid is Diacylglycerophosphocholine. Diacylglycerophosphocholine is a subclass of Glycerophosphocholine. Glycerophosphocholine is a subclass of Glycerophospholipid and has inherited Carbon_Chain_Group, Glycerophosphate_Group and either Carboxylic_Acid_Ester or Ether from Glycerophospholipid. The Glycerophosphate_Group is further specified to be

a Glycerophosphatidylcholine in Glycerophosphocholine. Following that, Diacylglycerophosphocholine inherits the functional group concepts from Glycerophosphocholine. In addition to that, the Carbon_Chain_Group of the Diacylglycerophosphocholine is furthered specified with a cardinality axiom "hasAcyl_Chain exactly 2". A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Diacylglycerophosphocholine so that lipids of this class can only have the following functional groups, namely, Carboxylic_Acid_Ester, Glycerophosphatidylcholine and 2 Acyl_Chains. (see Table 16)

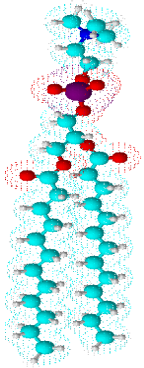
	Necessary and Sufficient Conditions
	LC_Glycerophosphocholine hasAcyl_Chain exactly 2 hasPart only (Glycerophosphatidylcholine or Acyl_Chain or Carboxylic_Acid_Ester)
	Necessary Conditions inherited from LC_Glycerophosphocholine
	hasPart some Glycerophosphatidylcholine
	Necessary Conditions inherited from LC_Glycerophospholipid
	(hasPart some (Carboxylic_Acid_Ester or Ether)) and (hasPart some Glycerophosphate_Group) and (hasPart some Carbon_Chain_Group)

Table 16: DL definition for diacylglycerophosphocholine

1.1.9.1) Use of the Term "phosphatidyl" and "phosphatidic acid"

Due to the overlap of identical terms use to name concepts use for Lipid classes and concepts of Organic_Group, we modify the names of Organic_Group concepts use to define Glycerophospholipid. The rationale of applying the modification to the Organic_Group concepts instead of Lipid class names is to ensure that the Lipid classification hierarchy will remain as identical as possible with LIPID MAPS

nomeclature. An example of such a lipid is Glycerophosphocholine (a lipid class), defined by Glycerophosphatidylcholine (organic group concept modified from Glycerophosphocholine organic group). In another example, Glycerophosphate (a lipid class) is defined by Glycerophophatidic_acid(an organic group concept).

1.1.10) Definitions of Glycerolipid

Glycerolipids encompass all glycerol-containing lipids, with the exception of glycerophospholipids. Glycerolipids are dominated by the mono-, di- and tri-substituted glycerols, the most well-known being the acylglycerols. Additional subclasses are represented by the glycerolglycans, which are characterized by the presence of one or more sugar residues attach to glycerol via a glycosidic linkage [2]. We define Glycerolipid as a lipid that has at least a Carboxylic_Acid_Ester or Ether, at least a Glycerol or Glyceroglycan and at least a carbon chain from the Carbon_Chain_Group.

An example of Glycerolipid is Triacylglycerol. Triacylglycerol is a subclass of Triradylglycerol. Triradylglycerol is a subclass of Glycerolipid and has inherited Carbon_Chain_Group, either Glycerol or Glyceroglycan and either Carboxylic_Acid_Ester or Ether from Glycerolipid. Triradylglycerol is defined to have only Glycerol, Carboxylic_Acid_Ester. In addition to that, Carbon_Chain_Group is specified with a cardinality axiom “hasCarbon_Chain exactly 3”. Following that, Triacylglycerol inherits all functional group concepts from Triradylglycerol and a cardinality axiom “hasAcyl_Chain exactly 3” is applied to Carbon_Chain_Group in Triacylglycerol. A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Triacylglycerol so that lipids of this

class can only have the following functional groups, namely, Carboxylic_Acid_Ester, Glycerol and 3 Acyl_Chains. (see Table 17)

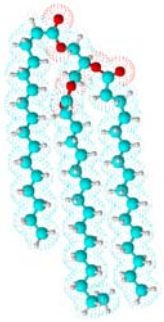
	Necessary and Sufficient Conditions
	LC_Triradylglycerol hasAcyl_Chain exactly 3 hasPart only (Glycerol or Acyl_Chain or Carboxylic_Acid_Ester)
	Necessary Conditions inherited from LC_Triradylglycerol
	(hasCarbon_Chain exactly 3) and (hasPart some Glycerol)
	Necessary Conditions inherited from LC_Glycerolipid
	(hasPart some (Carboxylic_Acid_Ester or Ether)) and (hasPart some Carbon_Chain_Group)

Table 17: DL definition of triacylglycerol

1.1.10.1) Differences Between Specifying Cardinality Axiom for Glycerolipid and Glycerophospholipid

LIPID MAPS organizes Glycerolipid by the number of acyl chains whereas Glycerophospholipid is organized according to head groups, regardless of the number of acyl chains. Cardinality axiom is applied differently to specify the Carbon_Chain_Group for these 2 categories of lipids.

Glycerolipid was divided by the number of chains first before the chains were specifically specified.

"hasPart some *Carbon_Chain_Group*"(inherited from *Glycerolipid*)

"hasCarbon_Chain_Group exactly 3" (inherited from *Triradylglycerol*)

"hasAcyl_Chain exactly 3" (for *Triacylglycerol*)"

Glycerophospholipid is divided by headgroups first regardless to the type of carbon chains or number of chains before the chain was specifically specified.

"hasPart some *Carbon_Chain_Group*" (inherited from *Glycerophopholipid*)

"hasPart some *Glycerophosphatidylcholine*" (no Cardinality axiom inherited from *Glycerophosphocholine*. Rather, a headgroup was specified)

"hasAcyl_Chain exactly 2" (for *Diacylglycerophosphocholine*)

The rationale behind this implementation is to ensure that the organization of ontology to be consistent with respect to the classification found in the LIPID MAPS nomenclature.

1.1.11) Definitions of Saccharolipid

Saccharolipids are compounds where fatty acids are linked directly to a sugar backbone [2]. We define Saccharolipid as a lipid that has at least a Glycan_Group and at least an Acyl_Chain.

An example of Saccharolipid is Triacylaminosugar. Triacylaminosugar is a subclass of Acylaminosugar. Acylaminosugar is a subclass of Saccharolipid and has inherited Acyl_Chain and Glycan_Group from Saccharolipid. Acylaminosugar is defined to have additional Phosphate_Group and Amino_Acid. Moreover, the Glycan_Group of Acylaminosugar is further specified to be either a Monomeric_Glycan_Group or a non Trehalose Dimeric_Glycan_Group. Following that, Triacylaminosugar inherits the functional group concepts from Acylaminosugar. Triacylaminosugar is further defined to have Carboxylic_Acid_Amide_Group and Carboxylic_Acid_Ester_Group. The Carbon_Chain_Group of Triacylaminosugar is specified by a cardinality axiom

"hasAcyl_Chain exactly 2". A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Triacylaminosugar so that lipids of this class can only have the following functional groups, namely, Carboxylic_Acid_Ester_Group, Carboxylic_Acid_Amide_Group, Glycan_Group, Phosphate_Group, Amino_Acid and 2 Acyl_Chains. (see Table 18)

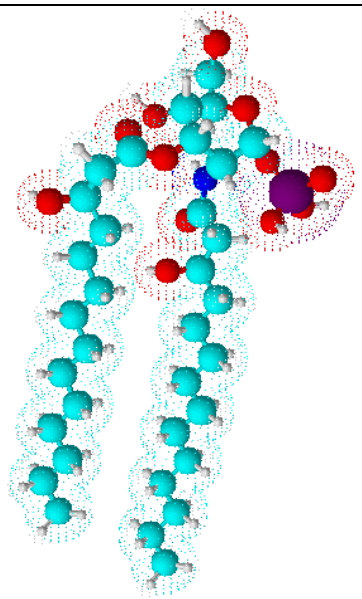
	Necessary and Sufficient Conditions
	LC_Acylaminosugar (hasAcyl_Chain exactly 2) and (hasPart some Carboxylic_Acid_Ester_Group) and (hasPart some Carboxylic_Acid_Amide_Group) and (hasPart some Amino_Acid) hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain)
	Necessary Conditions inherited from LC_Acylaminosugar
	(hasPart some (Monomeric_Glycan_Group or (Dimeric_Glycan_Group and not Trehalose))) and (hasPart some Phosphate_Group)
	Necessary Conditions inherited from LC_Saccharolipid
	(hasPart some Acyl_Chain) and (hasPart some Glycan_Group)

Table 18: DL definition of triacylaminosugar

1.1.12) Definitions of Sphingolipid

Sphingolipids are compounds that share a common structural feature, a sphingoid base backbone that is synthesized *de novo* from serine and a long-chain fatty acylcoenzyme A, that is further converted into ceramides, phosphosphingolipids, glycosphingolipids and other chemical species, including protein adducts [2]. We define Sphingolipid as a lipid that has at least a Primary_Amine or Carboxylic_Acid_Secondary_Amide, an Alcohol and at least a sphingoid base chain from Sphingoid_Base_Chain_Group.

An example of Sphingolipid is Acylceramide. Acylceramide is a subclass of Ceramide. Ceramide is a subclass of Sphingolipid and has inherited Sphingoid_Base_Chain_Group, Alcohol and either a Primary_Amine or Carboxylic_Acid_Secondary_Amide from Sphingolipid. The Carboxylic_Acid_Secondary_Amide is subsequently specified in Ceramide. Ceramide is further defined to have Carboxylic_Acid_Ester_Group and Acyl_Chain. In addition to that, the Sphingoid_Base_Chain_Group is specified with a cardinality axiom "hasSphingoid_Base_Chain exactly 1" in Ceramide.

Acylceramide inherits the functional group concepts from Ceramide. In addition to that, the Sphingoid_Base_Chain_Group is specified to be a Sphing-4-ene_Chain with a cardinality axiom "hasSphing-4-ene_Chain exactly 1" whereas the Acyl_Chain is specified to be an Acyl_Ester_Chain with a cardinality axiom "hasAcyl_Chain exactly 1" in Acylceramide. Following that, Acylceramide is further defined with additional Alkenyl_Group. A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Acylceramide so that lipids of this class can only have the following functional groups, namely, Carboxylic_Acid_Ester_Group, Carboxylic_Acid_Secondary_Amide, Alcohol, 1 Sphing-4-ene_Chain and 1 Acyl_Ester_Chain. (see Table 19)

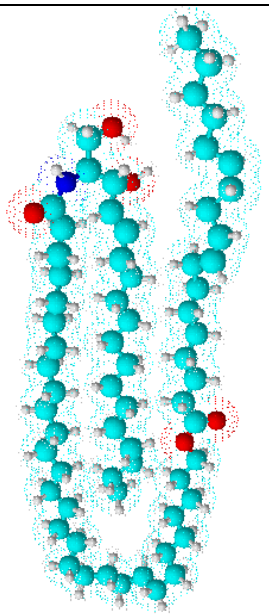
	Necessary and Sufficient Conditions
	LC_Ceramide (hasPart some Alkenyl_Group) and (hasSphing-4-enine_Chain exactly 1) and (hasAcyl_Ester_Chain exactly 1) hasPart only (Alkenyl_Group or Sphing-4-nine_par_Sphingosine_par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester_Group or Acyl_Chain or Alcohol)
	Necessary Conditions inherited from LC_Ceramide
	(hasSphingoid_Base_Chain exactly 1) and (hasPart some Acyl_Chain) and (hasPart some Carboxylic_Acid_Secondary_Amide) and (hasPart some Carboxylic_Acid_Ester_Group)
	Necessary Conditions inherited from LC_Sphingolipid
	(hasPart some Sphingoid_Base_Chain_Group) and (hasPart some (Primary_Amine or Carboxylic_Acid_Secondary_Amide)) and (hasPart some Alcohol)

Table 19: DL definition of acylceramide

1.1.12.1) Unclassified Sphingolipid

Some Sphingolipid classes are not defined with DL definitions due to the classification inadequacy found in LMSD. Some of these inadequacies are as follows:

f) Lack of explicit textual definitions in LMSD

g) Lack of representative instance of lipid for a specific class of lipid(an empty concept without data entries)

   An example of this is the sphingolipid class “Other Acidic glycosphingolipids” (SP0600).

h) The use of arbitrarily named lipid class to contain non-conventional lipid instances

   An example is “Sphingoid base homologs and variants” and “Sphingoid base analogs”.

Closer examination of the “Sphingoid base homolog and variants” indicates that most instances in the lipid class can be classified elsewhere as “Lysosphingomyelins” and “Sphingoid base 1- Phosphates” in the LIPID MAPS hierarchy. It is possible that our assumed lipid definition of “Lysosphingomyelins” and “Sphingoid base 1-Phosphate” may be broader that what LIPID MAPS had originally intended. The “Sphingoid base homolog and variants” may include more types of sphingolipids (inclusive of lysosphingomyelins and sphingoid base 1-phosphates) that are not covered by the present LIPID MAPS nomenclature. We make provision in LiCO for that by renaming “Sphingoid base homolog and variants” to Sphingoid_base_homolog_ structural_derivative and creating 2 empty subclasses under the concept, namely Sphingoid_base_homolog and Sphingoid_base_homolog_variant.

We handle the unclassified sphingolipids either by excluding the lipid class from the hierarchy in the Ontology or by creating an equivalent empty lipid class that is not equipped with any DL constraints.

1.1.13) Definitions of Prenol_Lipid

Prenol lipids are synthesized from the 5-carbon precursors isopentenyl diphosphate and dimethylallyl diphosphate that are produced mainly via the mevalonic acid (MVA) pathway [2]. Prenol_Lipid is defined as a lipid that has either Phytyl or Prenyl.

An example of Prenol Lipid is Ubiquinone. Ubiquinone is a subclass of Quinone. Quinone is a subclass of Prenol_Lipid and inherited either Prenyl or Phytyl from Prenol_Lipid. In addition to that, Quinone is defined with at least a

Quinone_Ring_System. Following that, Prenyl is specified in Ubiquinone with minimum cardinality axiom and maximum cardinality axiom that restrict Ubiquinone to have only 3 to 10 Prenyl("hasPrenyl_Group min 3 and hasPrenyl_Group max 10"). Ubiquinone is further defined with Ubiquinone_ring, Alkenyl_Group, Ketone, Ether and Isoprene_Chain. A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Ubiquinone so that lipids of this class can only have the following functional groups, namely, Ubiquinone_ring, Isoprene_Chain, Alkenyl_Group, Ketone, Ether and Prenyl. (see Table 20)

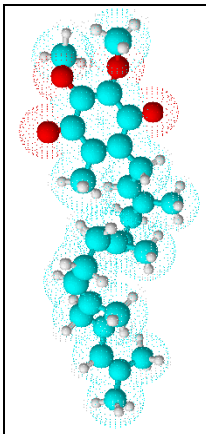
	Necessary and Sufficient Conditions
	LC_Quinonr_par_inclusive_of_hydroquinone_par_ (hasPart some Isoprene_Chain) and (hasPrenyl_Group min 3) and (hasPrenyl_Group max 10) (hasPart some Ubiquinone) and (hasPart some Alkenyl_Group) and (hasPart some Ketone) and (hasPart some Ether) hasPart only (Isoprene_Chain or Ubiquinone_ring or Prenyl or Alkenyl_Group or Ketone or Ether)
	Necessary Conditions inherited from LC_Quinone_par_inclusive_of_hydroquinone_par_
	hasPart some Quinone_ring_system
	Necessary Conditions inherited from LC_Prenol_Lipid
	hasPart some (Prenyl or Phytyl)

Table 20: DL definition of ubiquinone

1.1.14) Definitions of Sterol_Lipid

Sterol lipids share a common biosynthetic pathway via polymerization of dimethylallyl pyrophosphate/isopentenyl pyrophosphate with prenol lipids but have obvious differences in terms of their eventual structure and function [2]. Sterol_Lipid is defined as lipid that is composed of Cyclopenta-a-Phenanthrene_Ring_System.

An example of Sterol_Lipid is Cholesterol_structural_derivative. Cholesterol_structural_derivative is a subclass of Sterol, which in turns inherits Cyclopenta-a-Phenanthrene_Ring_System from Sterol_Lipid. The Cyclopenta-a-Phenanthrene_Ring_System is further specified as Cyclopenta-a-Phenanthrene_Ring in Sterol. Following that, this Cyclopenta-a-Phananthrene_Ring is further specified as Cholestane in Cholesterol_structural_derivative. Cholesterol_structural_derivative is further defined with an Iso-Octyl_Derivative and either Alcohol or Epoxy or Ketone or Alkenyl_Group. A closure axiom is needed to restrict the type of relationship constraints allowed for a lipid class. Closure axiom is applied to Cholesterol so that lipids of this class can only have the following functional groups, namely, Cholestane, Alcohol, Alkenyl_Group, Epoxy, Ketone and Iso-Octyl_Derivative. (see Table 21)

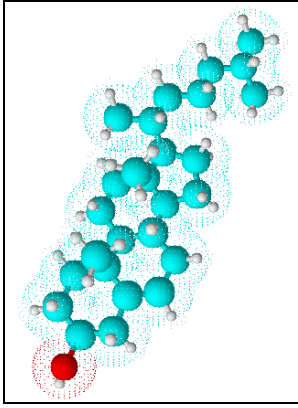
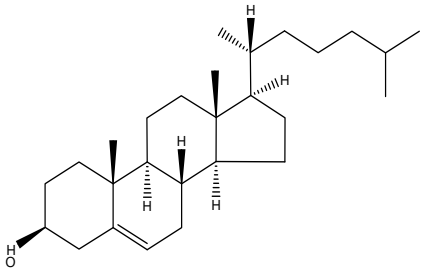
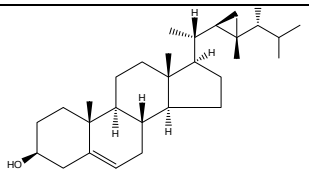
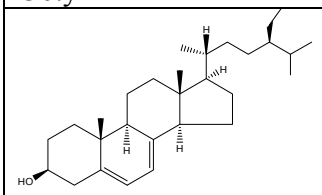
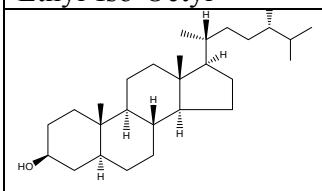
	Necessary and Sufficient Conditions
	LC_Sterol (hasPart some Cholestane) and (hasPart some Iso-Octyl_Derivative) and (hasPart some (Alcohol or Ketone or Alkenyl_Group or Epoxy)) hasPart only (Cholestane or Iso-Octyl_Derivative or Alcohol or Ketone or Alkenyl_Group or Epoxy)
	Necessary Conditions inherited from LC_Sterol
	hasPart some Cyclopenta-a-Phenanthrene_Ring
	Necessary Conditions inherited from LC_Sterol_Lipid
	hasPart some Cyclopenta-a-Phenanthrene_Ring_System

Table 21: DL definition of cholesterol structural derivative

1.1.14.1) The Use of Alkyl_derivative Chain and the Use of Fissile Variant

Most sterol lipids are lipids that have a tetracyclic nucleus that is a cyclopenta-a-phenanthrene structure. Sterol lipid such as Cholesterol is well known as a lipid that is composed of the tetracyclic nucleus with an Iso-Octyl Chain at carbon-17. However, as

we examine LIPID MAPS, we encounter many lipid instances under the “Cholesterol and derivatives” class that vary in the Iso-Octyl chain that protrude from the tetracyclic nucleus (see Table 22). Basically, these are lipid derivatives of cholesterol where the Iso-Octyl chain has been modified biochemically. Because there can be an almost unlimited possibility to the type and number of modification to the iso-octyl chain, we introduce a new class of carbon chain, namely, Iso-Octyl_Derivative. The generic form of Iso-Octyl_Derivative, Alkyl_Derivative_Chain specifies biochemically modified alkyl chain that are too numerous to be specify. Currently, we specify 14 Alkyl_Chain_Derivative in LiCO based on what is needed to define lipid classes from LMSD. Similar approach has been applied to Organic_Group concepts use to define prenol lipid, specifically the Isoprenoid_derivative.

Sterol with Iso-Octyl Chain	Sterols with Iso-Octyl derivative	Class type of Sterol
	 Cyclopropanoyl-Iso-Octyl	Gorgosterol_structural_derivative
	 Ethyl-Iso-Octyl	Stigmasterol_structural_derivative
	 Methyl-Iso-Octyl	Ergosterol_structural_derivative

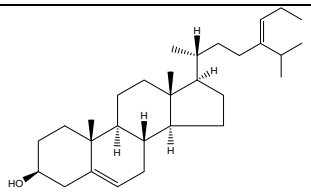
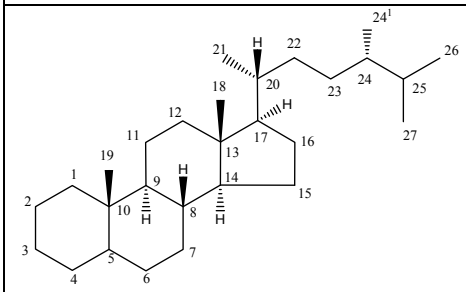
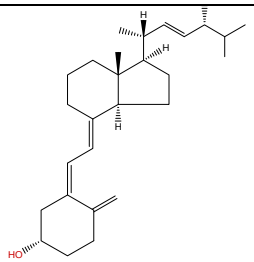
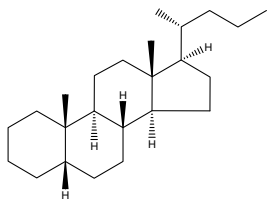
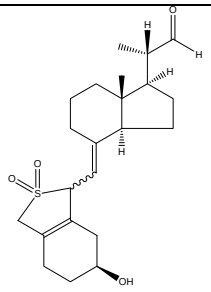
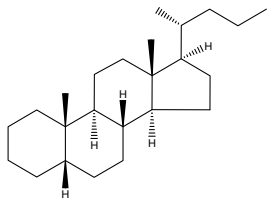
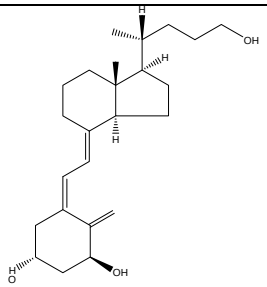
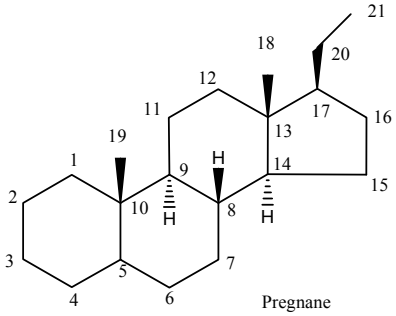
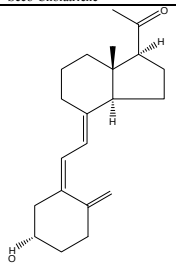
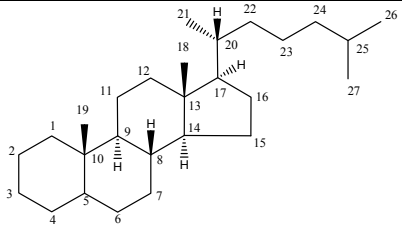
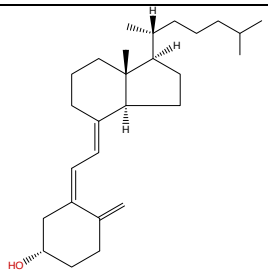
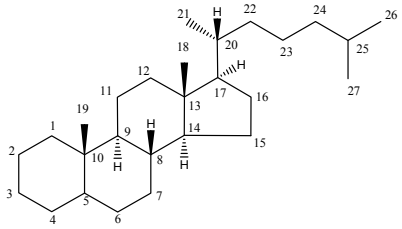
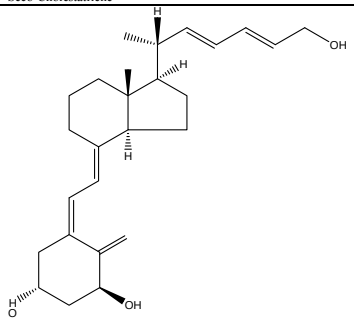
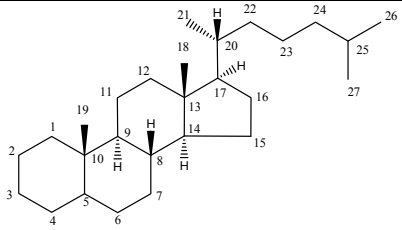
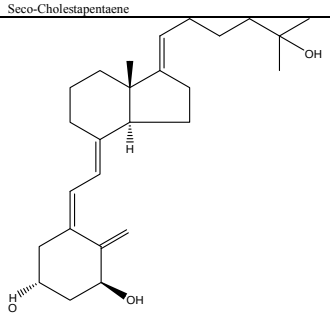
	 Propyl-Iso-Octyl	C24-propyl_sterol_structural_derivative

Table 22: Examples of sterols with iso-octyl chain derivative compare to sterol with iso-octyl chain

In addition to that, in order to define the non-conventional sterol lipid (basically lipids that do not have the Cyclopenta-a-Phenanthrene_Ring) such as the secosteroid, we introduce concepts of fissile variants of tetracyclic nucleus (Cyclopenta-a-Phenanthrene_fissile_variant) to define these lipids.(Table 23)

Sterols with cyclopenta-a-Phenanthrene ring structure	Sterols with cyclopenta-a-Phenanthrene ring fissile variant	Class type of Secosteroid
 Ergostane	 Ergocalciferol	Vitamin D2
	 Seco-Choladiene	Vitamin D3

	 Seco-Cholatriene	Vitamin D3
 Pregnane	 Seco-Pregnatriene	Vitamin D3
 Cholestane	 Seco-Cholestatriene	Vitamin D3
 Cholestane	 Seco-Cholestapentaene	Vitamin D3
 Cholestane	 Seco-Cholestatetraene	Vitamin D3

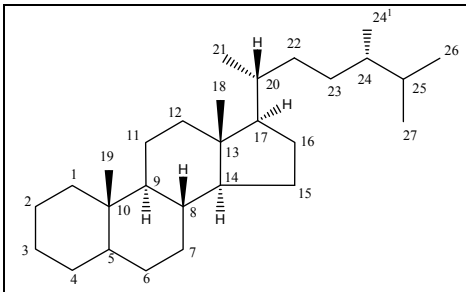
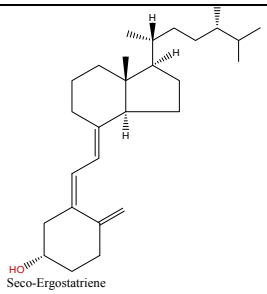
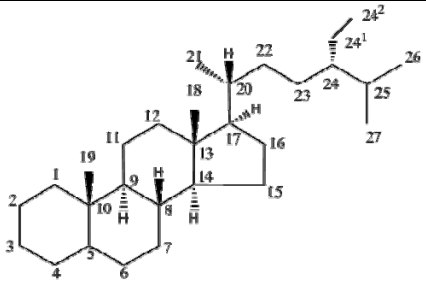
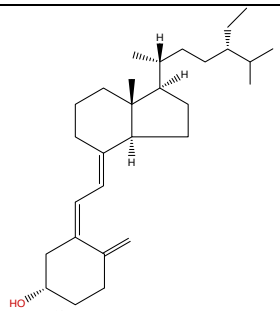
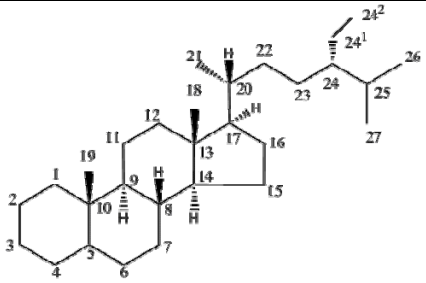
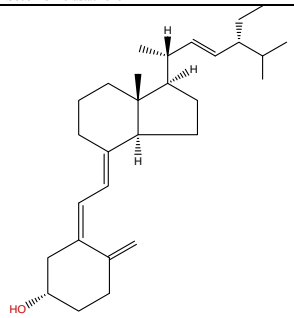
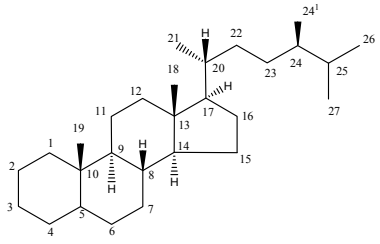
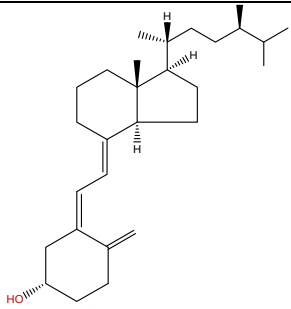
 Ergostane	 Seco-Ergostatriene	Vitamin D4
 Poriferastane	 Seco-Poriferastatriene	Vitamin D5
 Poriferastane	 Seco-Poriferastatetraene	Vitamin D6
 Campestane	 Seco-Campestatriene	Vitamin D7

Table 23: Examples of sterol with ring fissile variants with comparison to sterol with normal tetracyclic ring

1.1.14.2) Use of Taurine

In order to classify Steroid_conjugate, specifically Taurine_conjugate, we introduce the concept of organic group Taurine. Taurine or 2-aminoethanesulfonic acid, is an organic acid. It is a major constituent of bile and can be found in the lower intestine and in small amounts in the tissues of many animals and in humans as well [37]. Taurine is a derivative of the sulfur-containing (sulfhydryl) amino acid, cysteine. It is one of the few known naturally occurring sulfonic acids. In LiCO, we classify Taurine as a unique functional group that can be both classified as Organic Sulfur group and as well as amino acid.

2) Lipid Entity Representation Ontology

Lipid Entity Representation Ontology (LERO) is an OBO compliant application ontology created to represent and to address the nomenclature issues in lipids. Besides what has been described in LiCO, LERO includes additional concepts for lipid database identifiers, lipid synonyms, as well as other properties needed to further describe lipids. LERO is an ontology equivalent of a lipid database schema and can be used to provide semantic meaning and annotation for a lipid database.

2.1) Ontology Description:

The entities in LERO can be divided into 2 major types: they are either Independent_ Continuant_Entity or Dependent_Continuant_Entity. Lipid is a subclass of Independent_ Continuant_Entity. Similar to LiCO, lipids in LERO are defined by Organic_Group and Ring_System. Both Organic_Group and Ring_System are also sub-concepts of Independent_Continuant_Entity.

2.1.2) Lipid Specification

In LERO, we include concepts under the Lipid_Specification concept to specify other properties of Lipid. These properties are dependent on the identity of the lipid and are subsumed under the concept of Dependent_Continuant_Entity.

Information about individual lipid molecules is modeled in the Lipid and Lipid Specification concepts according to the method employed in Lipid Ontology 1.0. In addition to the 10 concepts modeled in Lipid Ontology 1.0, we expand on these concepts by adding new sub-concepts (see Figure 21).

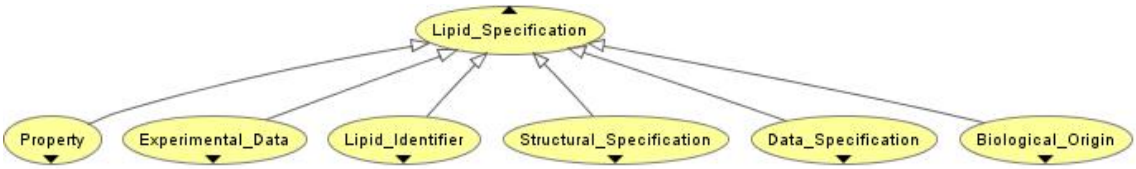


Figure 21- Immediate subclasses of Lipid_Specification concept

2.1.2.1) Biological Origin

We add Cellular_Product_Origin and Organismal_Origin under the concept Biological_Origin. Biological origin describes the biological source of a lipid molecule.

2.1.2.2) Data Specification

The Data_Specification is used to annotate the mass spectromentry data found under the Experimental_Data concept. It provides the Ion_Mode necessary to annotate the mass spectromentry data. The Ion_Mode is a concept that covers 13 instances that could be

used to annotate actual m/z values or the mass spectrometry readings from the instrument. (see Figure 22)

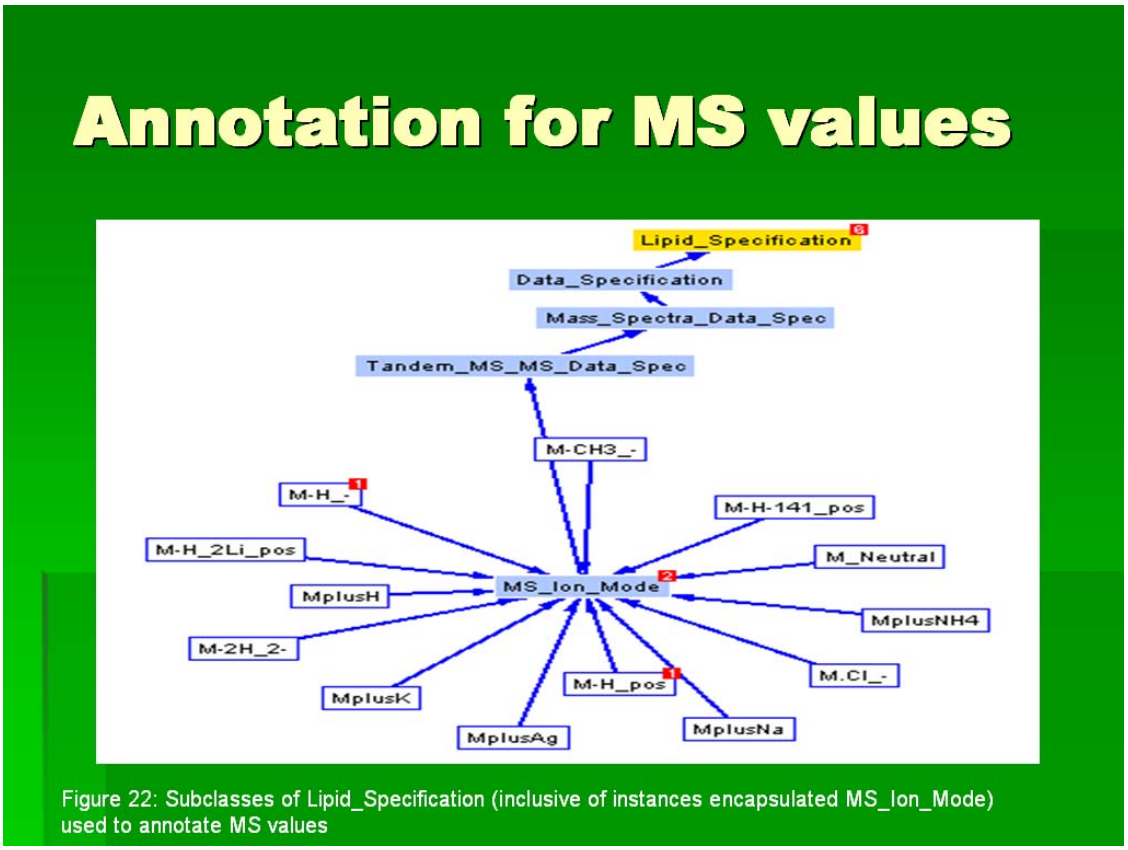


Figure 22: Subclasses of Lipid_Specification (inclusive of instances encapsulated MS_Ion_Mode) used to annotate MS values

2.1.2.3) Experimental Data

Experimental_Data is expanded to include concepts that specify mass spectrometry data of a lipidomics experiment, specifically the tandem MS MS values.

A mass spectrometry measurement for lipidomics comes in 2 forms; the Precursor/Parent Ion m/z value and the Product/Daughter Ion m/z values. The Daughter Ions can be further classified into Head m/z value(typically useful for lipids with distinct headgroups such as Glycerophopholipid, Sphingolipid) and Tail m/z value(relevant for lipids with acyl or

other types of tail/chain). The Others m/z value is meant for MS measurements of non-tail or non-headgroup fragment of lipids. (see Figure 23)

2.1.2.4) Lipid Identifier

Lipid_Identifier remains the same as Lipid Ontology 1.0 with 3 database sub-concepts, KEGG_Compound_ID, LIPIDBANK_ID, LIPIDMAPS_ID and the lipid name concepts. At this point of time, we make provisions in LERO to integrate lipid information from 3 databases only, namely KEGG COMPOUND database, LIPIDBANK and LMSD (see Figure 24). Please refer Figure 12 for description of name concepts. Future development of LERO will make provision to add LIPIDAT into the knowledgebase.

2.1.2.5) Property

Property is expanded from Color, Physicochemical properties and Stability properties to include specific concepts for biophysical properties such as pH, Boiling_Pt(point), Melting_Pt(point), (physical)State)_at_room_temp(temperature), Maximum_Stable_pH, Minimum_Stable_pH, Maximum_Temperature_Pt(point), Minimum_Temperature_Pt (point). (see Figure 23)

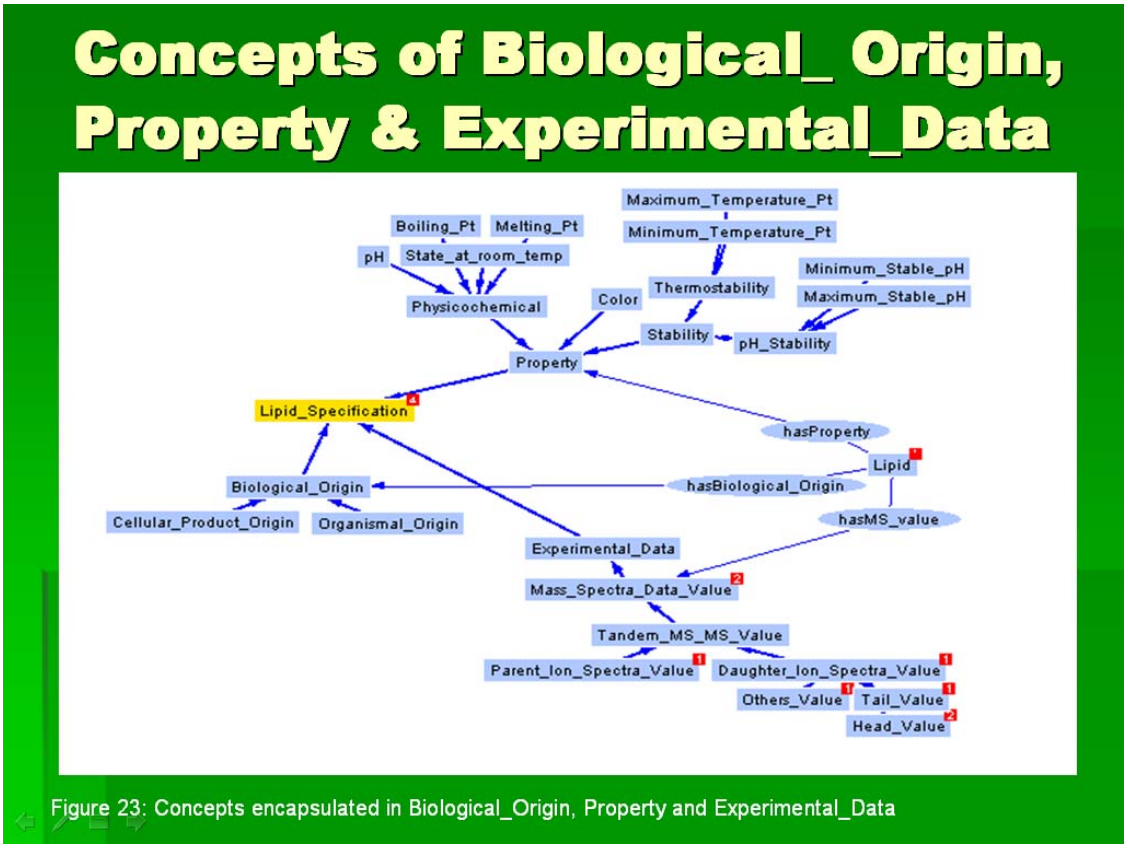


Figure 23: Concepts encapsulated in Biological_Origin, Property and Experimental_Data

The inclusion of relevant biophysical properties for lipids is important as we provide LERO with necessary concepts to adequately integrate and represent the data and knowledge from LIPIDAT, a high quality, hand curated database of lipid with a focus on the biophysical properties of lipids.

2.1.2.5) Structural Specification

Structural_Specification provides concepts needed to specify structural properties of lipids. With these concepts, we could specify the stereochemical state of the organic groups, the ring junctions and double bonds. In addition to that, we could specify the position of carbon chain, organic group and ring junction as well as the length of the carbon chain and its degree of unsaturation. (see Figure 24)

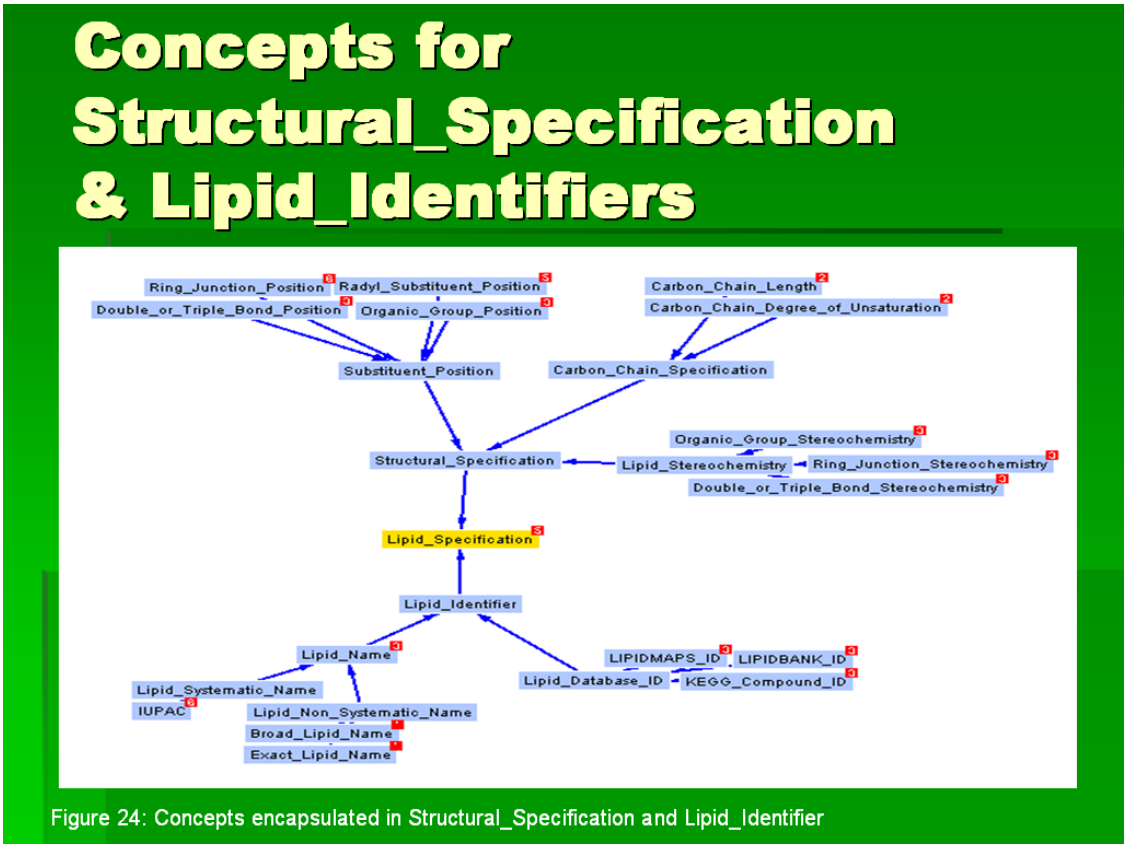


Figure 24: Concepts encapsulated in Structural_Specification and Lipid_Identifier

The inclusion of structural specification enables a lipid entity in LERO to be equipped with the necessary metadata to describe structural properties in greater chemical details. With the instantiation of a lipid entity along with the specification of organic group, ring system and associated structural specifications, a lipid entity can be easily translated into the LIPID MAPS abbreviated format that is widely use in LIPID MAPS consortia. Inversely, we could also convert the lipid information found LIPID MAPS abbreviated format into respective instances in LERO. (see Figure 25)

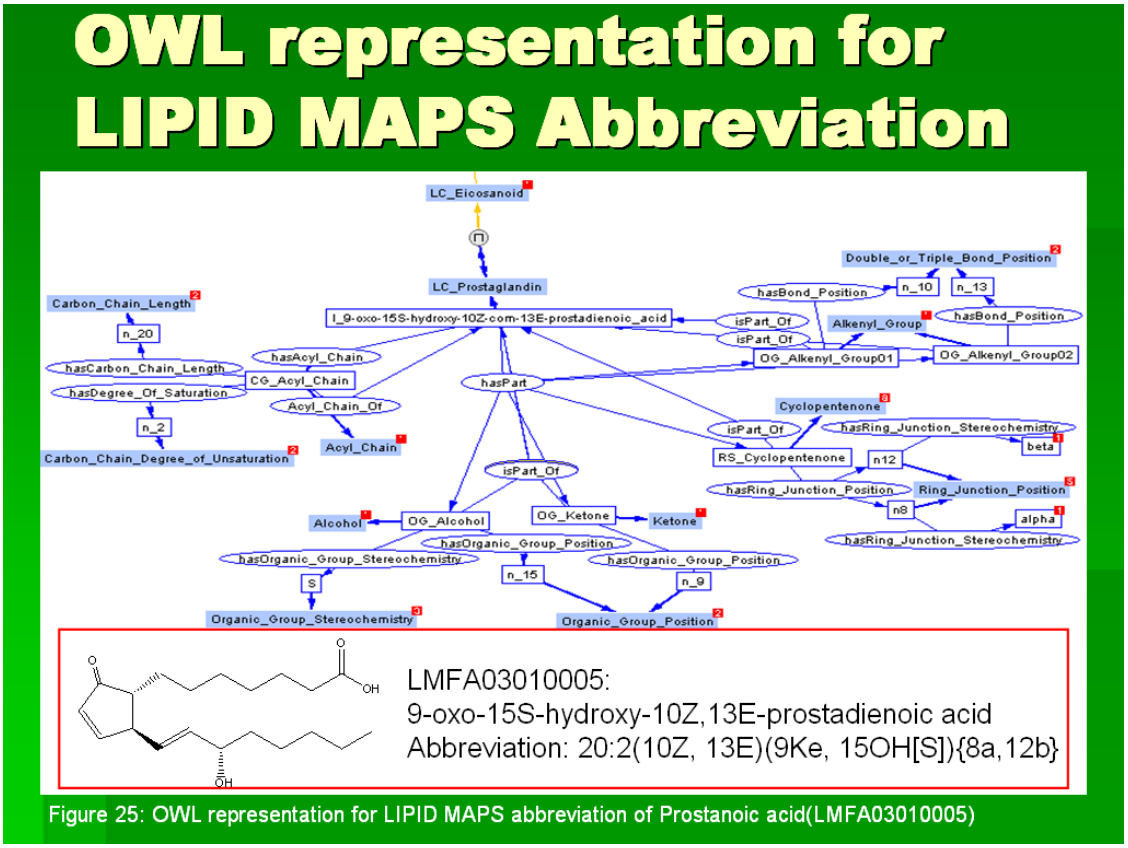


Figure 25: OWL representation for LIPID MAPS abbreviation of Prostanoic acid(LMFA03010005)

LIPID MAPS abbreviated format is a generalized lipid abbreviation format that was developed to enable structures, systematic names and relevant lipid ontological information (a form of standard controlled vocabularies) to be generated automatically from a single source format. The LIPID MAPS abbreviated format consists of 4 parts: i) carbon chain length with any degree of unsaturation ii)position and stereo-geometry of double and triple bond iii)position, type and stereochemistry of substituents iv)position of carbocyclic ring junction and stereochemistry.

An automated mechanism is available in LIPID MAPS database to generate lipid structures as well as their associated "ontological" information from just the LIPID MAPS abbreviation format. A populated LERO acts as a repository for lipidomics data

and associated lipid metadata and is a data source where LIPID MAPS compatible data format can be generated and be subsequently used to generate lipid structure automatically. The availability of lipid structure would allow us to generate unique InChI for every lipid entity instantiated in LERO.

3) Discussion

The current version of LiCO provides DL definitions for classification of lipid instances to 7 categories of Lipids in LIPID MAPS. Future versions of LiCO will extend the support for classification to the Polyketide category of LIPID MAPS.

3.1) Breadth of Classification

The definition of lipids can specify in 3 levels of coverage, specifically:

1) Class membership that satisfy strict, narrow adherence to the known nomenclature

2) Class membership to include lipids that are known to exist biologically or biosynthetically in the real world

3) Class membership to include hypothetical lipids

For example, Cholesterol is well known as lipid that is composed of a 4 rings or tetracyclic cyclopenta[a]phenanthrene structure. The four rings have trans-ring junctions, an Iso-Octyl side chain and two Methyl_Group. This is the strict definition of Cholesterol. Cholesterol is classified as Cholesterol and derivatives under LIPID MAPS nomenclature. It is renamed as Cholesterol_structural_derivative concept in LiCO.

Lipid instances under the Cholesterol_structural_derivative class vary due to different biochemical modifications in the Iso-Octyl chains and in the tetracyclic cyclopenta[a]phenanthrene structure. Examples of such cholesterol derivatives are cholest-(25R)-5-en-3β,26-diol, cholest-22E-en-3β-ol, Cucurbitacin B (see Table 24). As the result of that, the Cholesterol_structural_derivative class has a much broader definition than the strict nomenclature definition. A strict nomenclature definition is not sufficient but if we consider hypothetical lipids, there could be infinitely many more derivatives of cholesterol.

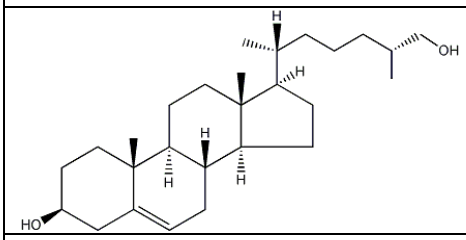
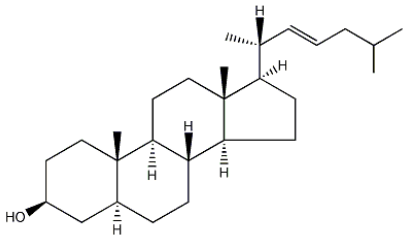
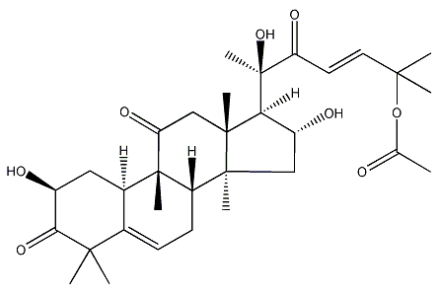
Structure	LIPID MAPS Identifier	LIPID MAPS Systematic name
	LMST01010088	cholest-(25R)-5-en-3β,26-diol
	LMST01010099	cholest-22E-en-3β-ol
	LMST01010104	Cucurbitacin B*
*a common name as no systematic name provided by LIPID MAPS		

Table 24: Examples of lipids from Cholesterol_structural_derivative

Fatty acid is another good example. A basic fatty acid consists of an Acyl_Chain and a Carboxylic_Acid. Theoretically, a fatty acid can have an Acyl_Chain of infinite carbon length. For each carbon length, there can be many permutations where an Alkenyl_Group can be inserted into the Acyl_Chain. In addition to that, the Acyl_Chain can also undergo many biosynthetic modifications where other chemical and functional groups are added into the Acyl_Chain. If we consider hypothetical lipids, there could be infinitely many more instances of Fatty_acid.

For our lipid classification exercise, we adopt the second option where a lipid class membership would include lipids that are known to exist biologically or biosynthetically in the real world. In this case, we define lipids based on the instances made available in LMSD. Our approach to this is one that is between pragmatism and absolute correctness. We do not support the use of strict, narrow adherence to the traditional nomenclature as that would exclude many real lipids whereas the option of considering definition for hypothetical lipids is too broad and is too unrealistic to be implemented in our case. Furthermore, adoption of definition for hypothetical lipids would make certain classes of lipids so generic such that a restrictive DL definition can not be applied to it.

3.2) Limitations of Present DL Definitions: Overlap of Ring_System, Chain_Group and Organic_Group

A lipid definition in LiCO includes members from Chain_Group, Complex_Organic_Group, Simple_Organic_Group and Ring_System. Unlike concepts of Lipid, DL and textual definitions are not implemented for them. A quick examination of

these concepts indicates that structurally, Monocyclic_Ring_Group, Chain_Group, Complex_Organic_Group and some members of Simple_Organic_Group such as Glycerol_derivative_Group are composed of several members of Simple_Organic_Group. Similar observation could be made of Polycyclic_Ring_System(composed of Monocyclic_Ring_Group). When a Chain_Group is specified in a DL definition of lipid, the concept would have also specified the functional group that is found in the Chain_Group. However, because DL definitions were not implemented for Chain_Group, Complex_Organic_Group, Simple_Organic_Group and Ring_System, we cannot make this assumption. As a result of that, in the current version of LiCO, when we specify Chain_Group, Complex_Organic_Group, and Ring_System, we still have to specify the Simple_Organic_Group found in these concepts in order to account for them.

For example, Fatty_Aldehyde has an Acyl_Chain. The Acyl_Chain of a Fatty_Aldehyde contains an Aldehyde_Group, a subclass of an Acyl_Group. Without assuming the structurally overlapping nature of the Acyl_Chain and Aldehyde_Group, the DL definition of a Fatty_Aldehyde is given as the following necessary and sufficient conditions:

(hasPart some Aldehyde) and (hasAcyl_Chain exactly 1)

hasPart only (Aldehyde or Acyl_Chain)

However, if we are to eliminate the overlapping Organic_Group, we only need to specify the Acyl_Chain in the necessary and sufficient conditions as Aldehyde, an acyl group that should have been accounted in the Acyl_Chain.

(hasAcyl_Chain exactly 1)

hasPart only (Acyl_Chain)

This simpler and more intuitively correct solution has not been implemented in LiCO as the provision of systematic DL definitions for Chain_Group, Complex_Organic_Group and Ring_System is beyond the research scope of this thesis.

3.3) Reclassification of Lipid Classes by Automatic Structural Inference

One of the benefits of using OWL-DL is to be able to automatically compute class hierarchy. The use of a reasoner to compute subclass-superclass relationships between classes is vital for the automatic maintenance of large ontology. In addition to that, automatic computation of subclass-superclass relationships could lead to inference of new relationships between the classes. Automatic inference could be used to infer new relationship between the different classes of lipid and to re-classify lipid nomenclature in a way that is logically consistent and computationally systematic. Currently, lipids are hand-classify in most databases and the use of automatic inference could minimize human errors that are inherent in maintaining and generating large, possibly multiple inheritance, classification hierarchy for lipid. A cursory examination of the current LIPID MAPS classification indicates that the following lipids may benefit from an automatic inference exercise.

Glycerolipids and Glycerophospholipids are essentially lipids that have at least a glycerol moiety. Glycerophospholipids are biosynthetically derived from glycerolipids [2].

Fatty acyl and Polyketide are lipids that are synthesized by enzymes that shared the same mechanistic features. Polyketides are synthesized by polyketide synthases, which are modular, multi-enzyme complexes that sequentially condense simple carboxylic acid derivative. Interestingly, many fatty acyls are either end products or derivation of the end products from the Polyketide pathway [2].

Prenol lipid and Sterol lipid share a common biosynthetic pathway via the polymerization of the dimethylallyl pyrophosphate/isopentenyl pyrophosphate [2].
At some point of the biosynthesis, these 3 groups of lipids have shared a common structural or precursor form and this may serve as basis for classifying them together.

Future work for LiCO could focus on developing fundamental structural definition for lipid classes that could account for the biosynthetic origin of the lipids. Automated classification using ontological reasoning had been successfully applied to protein classification [55] through the coordination of protein domain analysis of sequence data, ontology, an instance store, and DL reasoning. OWL-DL Ontology can drive technological development in automated classification for biological entities. With the addition of precisely defined DL-axioms to the LiCO, it is possible to apply this type of automated classification in our future work.

3.4) Lack of DL Definitions for Lipoproteins and Glycolipids

The current version of LiCO does not have DL definitions for lipoproteins and glycolipids. This is because the lipid classification hierarchy in LiCO is derived from

LIPID MAPS systematic nomenclature. LIPID MAPS systematic nomenclature does not consider lipoproteins as lipids and therefore, make not provisions for lipoproteins in the hierarchy. As for glycolipids, LIPID MAPS avoided the term "glycolipids" intentionally to maintain a focus on lipid structure. All eight categories of lipids in LIPID MAPS include important glycan derivatives, thus making an additional glycolipid class unnecessary and incompatible with the overall goal of lipid characterization.

3.5) The Choice of Using Object Property over Datatype Property

LERO build on LiCO's DL definition of lipids by adding additional concepts into the ontology to describe lipids in a more complete manner. This includes describing lipids with respect to their records in known lipid or chemical databases, their synonyms as well as their experimental properties such physicochemical properties and M/Z values from lipidomics experiments. Many of these attributes of lipids are numeric values. OWL-DL provides datatype properties where these numeric attributes can be assigned as range to relevant concepts in the ontology. However, as with the case of LERO, we do not use datatype property extensively. All properties in LERO are object properties.

An object property is a property that connects 2 objects to one another. It allows an attribute of an object to be specified through a relationship to another object. For an object property, both domain and range are classes or instances of classes. A datatype property is a property that connects an object to a value. For a datatype property, the domain is a class or an instance and the range is a value. The datatype property is used

for classes with numeric or string type attributes. It is a simpler way to representing values and is less resource consuming.

Despite this advantage, we do not use a datatype property in LERO. This is because many concepts that could have a datatype property such as Mass_Spectra_Data_Value need to be annotated by another object (see Figure 26).

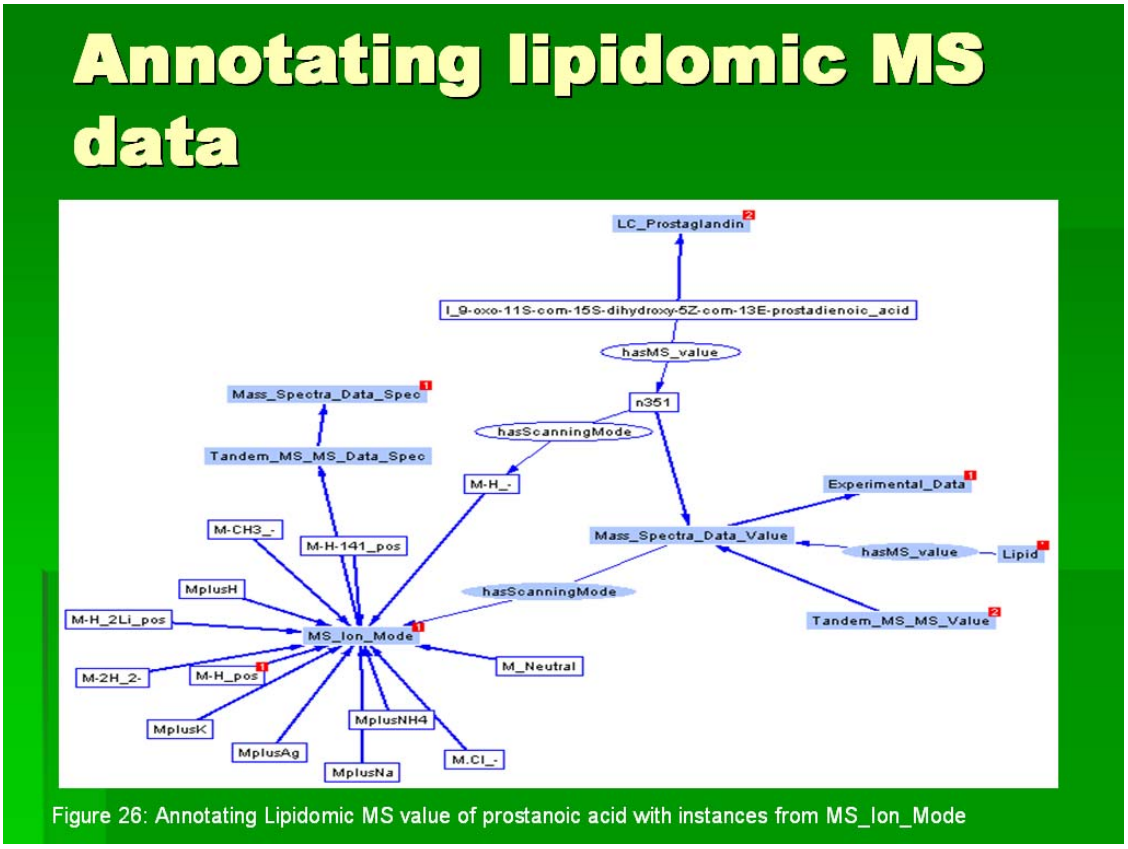


Figure 26: Annotating Lipidomic MS value of prostanoic acid with instances from MS_Ion_Mode

One of the advantages of OWL-DL knowledge representation is the ability to define a concept with complex, axiomatic constraints. The use of datatype property to define an attribute for objects greatly limits this advantage because complex axiomatic constraints cannot be specify for concepts whose range is a datatype, rather than an object.

3.6) Potential applications of LiCO and LERO

LiCO is a reference ontology that aims to share formalized DL definitions of lipids organized according to LIPID MAPS systematic classification with the wider bioinformatics and biological research community. It contains minimal definitions require to describe lipid entity formally. LERO extends the content of LiCO to describe lipid entity in a more comprehensive manner. While LERO can function as a reference ontology for complete representation of lipid entity, it is also capable of acting as application ontology for the purpose of integrating and uniting all lipid-related resources under a logically consistent, formalized knowledge representation framework for lipids. LERO provides a uniform, semantic web compliant, syntactic and semantic format to integrate lipid data from multiple databases, ontologies and other related resources. When lipid data is instantiated in LERO according to the formalized knowledge representation specify in the ontology, nomenclature inconsistencies found across multiple databases are resolved as every lipid records are normalized against the LIPID MAPS systematic classification hierarchy. LERO connects synonyms of lipids, experimental data and other data of lipids associated to the records from the databases to the systematic nomenclature proposed by LIPID MAPS. This unified, instantiated ontology then represents knowledge in a logical consistent manner to any information systems, inclusive of bioinformatics application as well as other semantic web related applications. One of these possible application of LERO is an integrative lipid knowledgebase that could connect large volume of experimental data generated from the analytical platform of lipidomics to a database system that contains information from all known resources of lipids in order to

facilitate rapid identification and discovery of new lipid species from the biological sample.

LERO is compliant to OBO specification and it provides an avenue for the LIPID MAPS classification system to be shared and to participate in the work of the wider bioinformatics and bio-ontology community (see Figure 27). In addition to that, LERO, written in OWL, a w3c-endorsed knowledge representation language to support interoperability of multiple, disparate information systems as well as sharing of formalized knowledge in the semantic web, is well placed as a lipid-centric ontology that can be combined with ontologies and knowledgebase from other biological domains in novel bioinformatics applications. These developments shall facilitate the uptake of the nomenclature by the biological research community and shall help establish the LIPID MAPS systematic nomenclature as a standard nomenclature for the lipid research community. There are already a number of databases, such as ChEBI and Uniprot, which are supported by OWL-DL-based semantic framework. As semantic web technologies mature, we should expect to see many of these knowledgebases from various biological domains converging unto a single knowledge representation information system and drive high-throughput, multi-dimensional, system-level bioinformatics analysis at various levels of granularities.

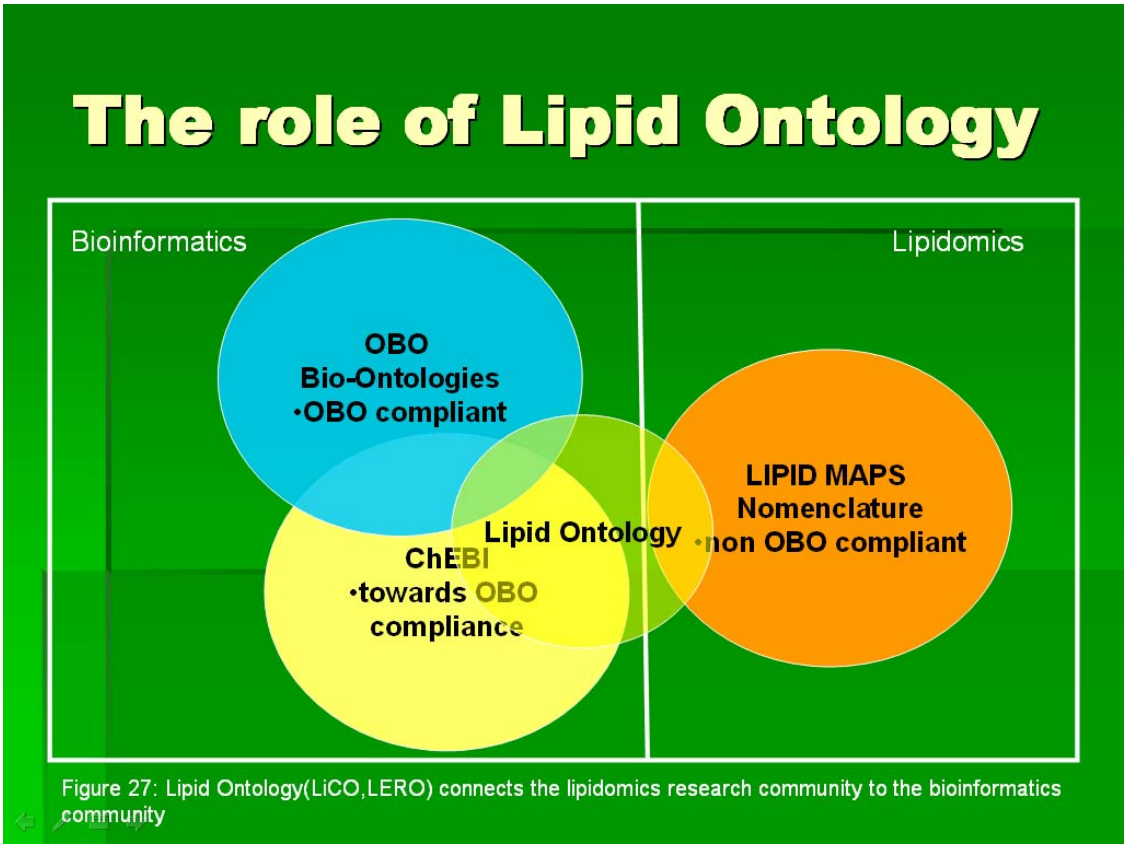


Figure 27: Lipid Ontology(LiCO,LERO) connects the lipidomics research community to the bioinformatics community

4) Conclusion

We describe 2 reference ontologies, namely Lipid Classification Ontology(LiCO) and Lipid Entity Representation Ontology(LERO). These ontologies are developed to share formalized knowledge with the wider biological research community. LiCO contains formalized DL definitions of lipids whereas LERO extends from LiCO to include other lipid-related informations such as synonyms and database identifiers. These 2 ontologies provide an avenue for establishing standardized lipid nomenclature and resolving nomenclature confusion that is prevalent in lipid research. In addition to that, LERO also provides a standard knowledge representation framework that supports interoperability between disparate information systems. The development of these ontologies will pave

the way for a bioinformatic analysis system capable of processing the large volume of heterogeneous data generated from the “system biology” approach.

Chapter V: Application scenario

A key motivation in developing the Lipid Ontology is to support an ontology-centric content delivery platform that provides unrestricted accessibility of lipid information in the scientific literature to a lipidomics researcher. A typical lipidomics researcher is interested in the identity of lipids found in his or her experimental work and wants find out all other informations associated to these lipids. In a post experiment analysis, a user needs to visit several databases, websites and read 5-6 papers to get the information that he wants. Even then, the information obtain may still be incomplete and fragmented. Here we describe a prototype ontology centric content delivery platform develop in conjunction with Institute of Infocomm Research, A*STAR to facilitate knowledge discovery for lipidomics scientists.

1) Literature Driven Ontology Centric Knowledge Navigation for Lipidomics

The platform comprises of a content acquisition engine that drives the delivery and conversion of literature (full text papers) to a custom format ready for text mining. A series of natural language processing algorithms that identifies target concepts or keywords and tags individual sentences according to the terms they contain. A custom-designed java program that instantiates sentences and relations to instances of each target concept found in the sentence into the ontology (specifically the Lipid_Specification and Lipid, Protein, Disease). A visual query and navigation interface, Knowlegator, facilitates query navigation over instantiated object properties and datatype properties in the

instantiated ontology through the reasoning engine RACER and the A-box query language nRQL. (see Figure 28)

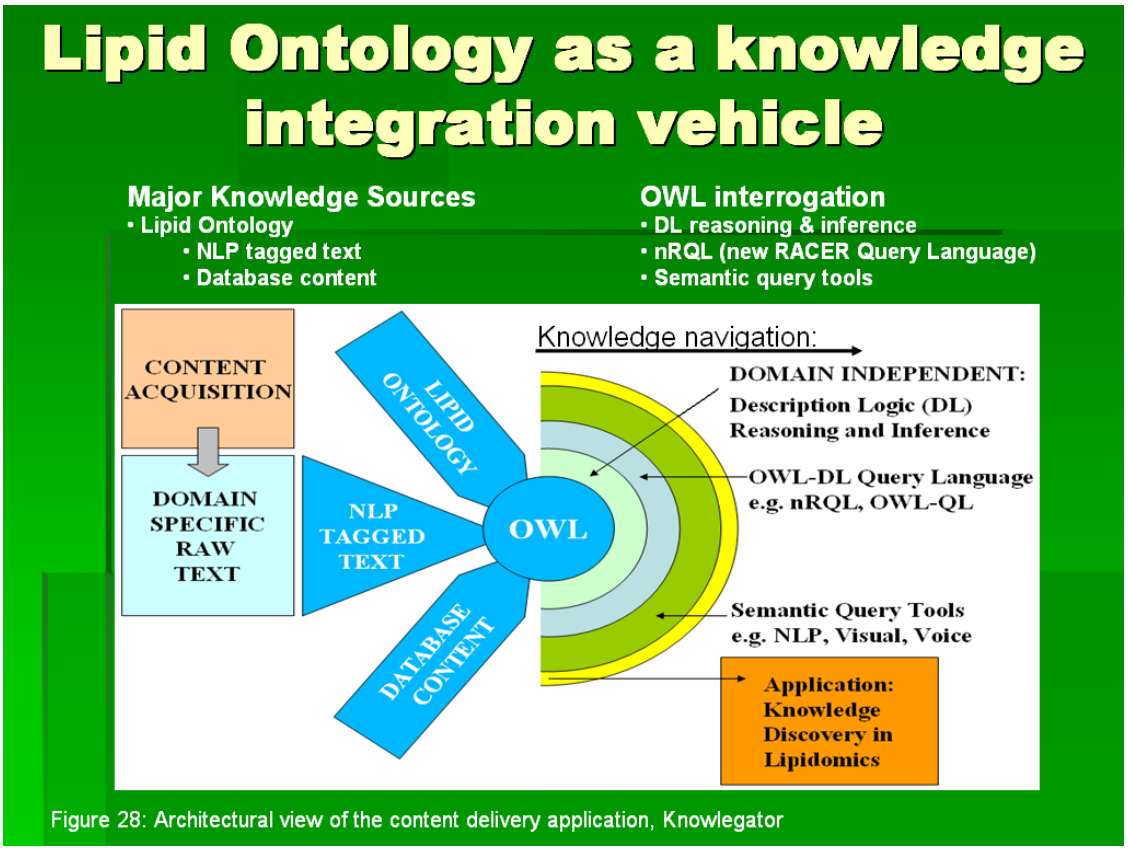


Figure 28: Architectural view of the content delivery application, Knowlegator

1.1) Knowledge Acquisition Pipeline

The knowledge acquisition pipeline consists of a custom perl script that takes keywords and acquires full-text documents from Pubmed search. The acquired full-text papers, in the form of pdfs are converted in ascii text format before being processed by NLP algorithms.

1.2) Natural Language Processing and Text-Mining

Text-mining and NLP are carried out using a text mining toolkit called BioText Suite that performs text processing tasks such as tokenization, part of speech tagging, named entity recognition, grounding and relation mining. See Figure 29 for detailed description of the text mining processes.

The text mining machinery uses a gazetteer that processes retrieved abstracts and full-text documents. It recognizes entities by matching term dictionaries against tokens of processed text. The lipid name dictionary is generated from Lipid DataWarehouse that contains lipid names from LipidBank, LMSD, KEGG, including associated IUPAC names, broad and exact synonyms. To resolve the problem of multiple synonyms in lipid nomenclature, we assemble a list of synonyms for lipids that can be found in the LMSD. These synonyms came from records of KEGG and LipidBank databases that have an equivalent record found in LMSD. Essentially, synonyms are taken from KEGG and LipidBank databases to enrich the lipid name list from LMSD. These synonyms are subsequently grounded to their equivalent name in LMSD and manually curated against any inconsistencies. At present, the list has 41,531 names, that covers 10,087 LIPID MAPS systematic names, 8,468 IUPAC names, 22,976 non-systematic names. The protein name dictionary comes from the manually curated UniProtKB database. The disease name list is created from the Disease Ontology of Centre for Genetic Medicine. Relationships between protein, lipid and disease are detected by a constraint-based association mining approach where the 2 entities are considered related if they co-occur in a sentence and satisfy a set of specified rules.

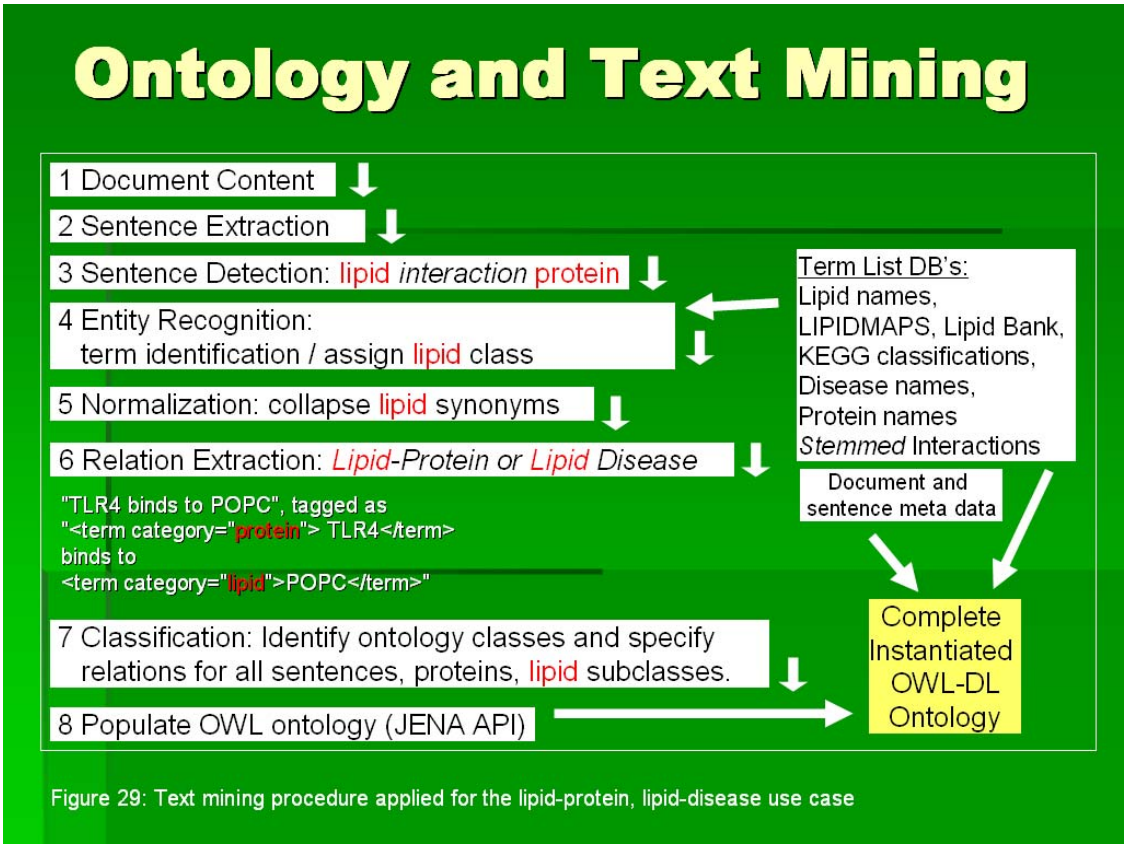


Figure 29: Text mining procedure applied for the lipid-protein, lipid-disease use case

Figure 29 shows the steps in the textmining procedure: At step 1, the downloaded document content is converted from its original format, mostly pdf into ascii text file. Following this step, each document is broken down to many distinct sentences. At step 3, sentences that have lipid terms, proteins term and an interaction term are identified. After that, lipid terms found in the sentence are identified and are assigned to an appropriate lipid class. At step 5, abbreviations of lipid name are normalized and lipid synonyms were grounded to LIPID MAPS systematic name. The relevant sentences are then tagged according to correct term categories (protein, lipid, disease, interaction). These tagged sentences are then classified according to formalized knowledge framework in the ontology. Once that is done, sentences are instantiated into the Lipid Ontology, along

with the corresponding relation between concepts (disease, protein, LIPID MAPS ID, document PMID).

1.3) Ontology Instantiation

A custom java based script written using the JENA API (http://jena.sourceforge.net/) carries out the instantiation of grounded entities as class instances into the respective ontology classes and the instantiation of relations detected as Object Property instances. Sentences and provenance information such as PMID are instantiated as Datatype property instances.

1.4) Visual Query and Reasoning through Knowlegator

Knowlegator(Knowledge naviGator) is a tool that allows navigation of A-box instances through an intuitive interface capable of converting a visual query built by a naïve end user into the query language syntax that communicates with the knowledgebase (instantiated ontology) for relevant information (see Figure 30). Knowlegator receives OWL-DL ontologies as inputs and passes them to RACER and issues a series of instructions to query the ontology for visual representation in the component panel. The component panel lays out the content of the ontology as tree structures of concepts, roles (property) and instances. This panel allows user to build visual query on the query canvas via a "drag and drop" feature. When an item is dropped into the query canvas, an associated nRQL query is automatically generated. The resulting nRQL syntax is used to query the knowledgebase for information. Information retrieves from the process will be presented in the results panel. As the numbers of object (concepts, property, instance)

drop into the query canvas increase, the complexity of the query also increases incrementally. With this tool, an end user can formulate deep and complex query to extract the relevant information from the knowledgebase.

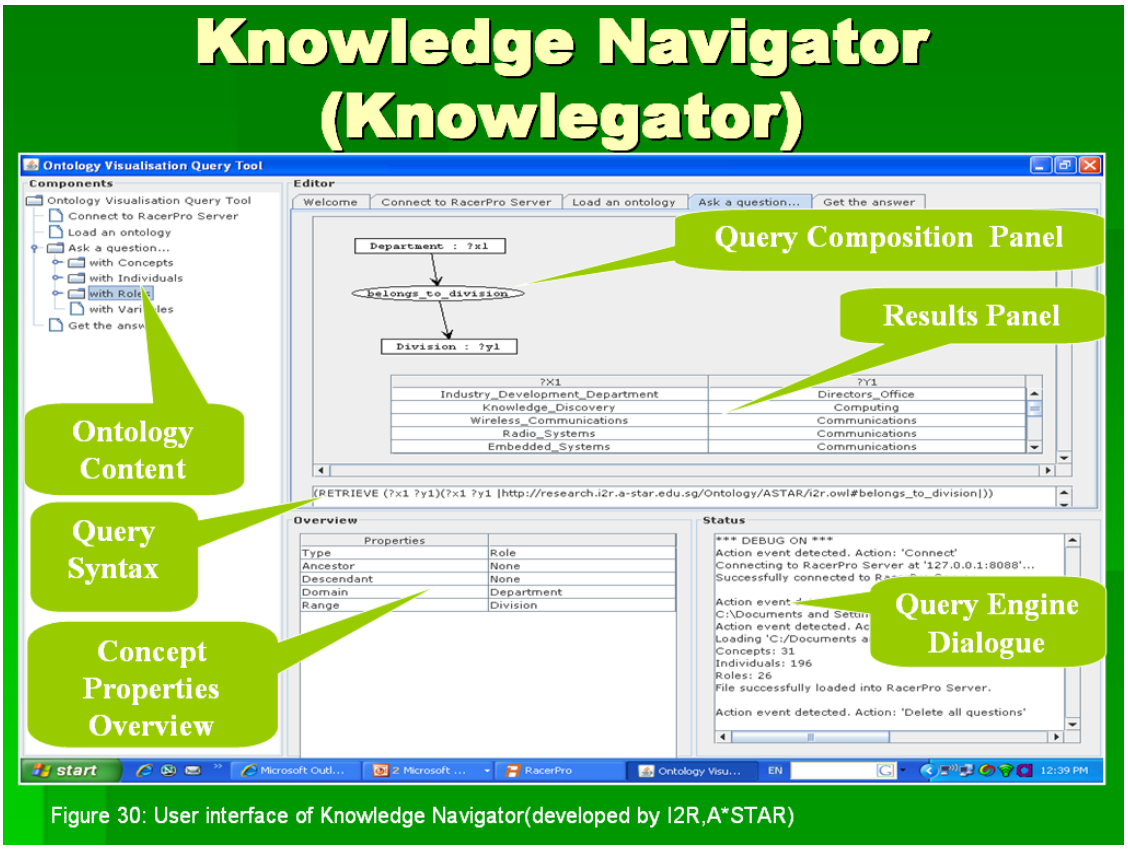


Figure 30: User interface of Knowledge Navigator(developed by I2R,A*STAR)

1.5) Preliminary Performance Analysis

Content acquisition engine identifies 495 search results for the time period July 2005 to April 2007 with search phrase “lipid interact* protein”. Of the 495 articles, 262 full-text papers are successfully downloaded. Named entity recognition and relation detection remove 121 documents that have no lipid-protein relations. Ontology instantiation is carried out with the remaining 141 documents. Initial named entity recognition (NER)

component detects 92 LIPIDMAPS systematic names, 52 IUPAC names, 412 exact synonyms, 6 broad synonyms, 319 protein names. 92 LIPIDMAPS names are instantiated into 35 unique classes under the Lipid name hierarchy, at an average of about 2.6 lipids per class. Cross-links to 59 Lipidbank entries and 41 KEGG entries are also established. Brute-force co-occurrence detection and subsequent relation word filtering yield over 683 sentences. The ontology instantiation process took 22 seconds overall. The experiments have been done on a 3.6 Ghz Xeon Linux workstation with 4 processors and 8GB RAM. (see Figure 31)

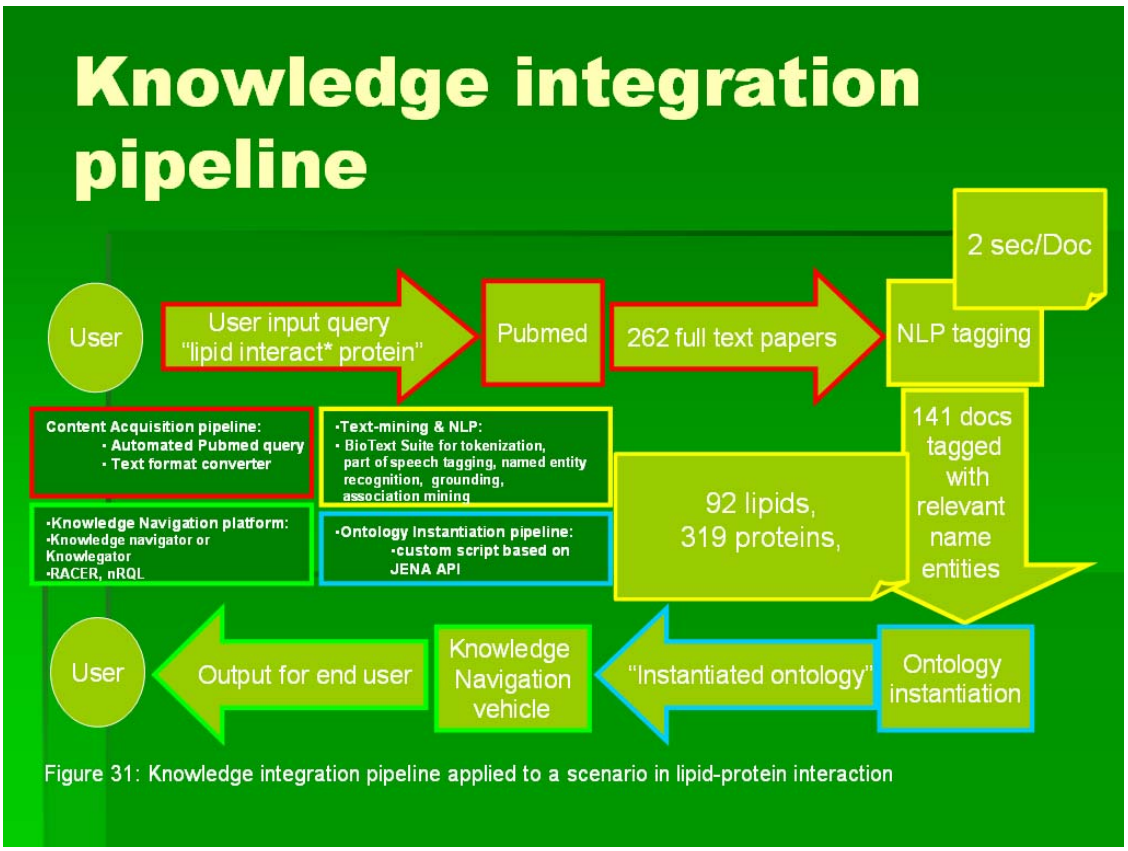


Figure 31: Knowledge integration pipeline applied to a scenario in lipid-protein interaction

2) Ontology Centric Navigation of Pathways

Disease processes such as cancer formation is a multi-step process caused by genetic alterations that change a normal cell to a cancerous cell. Molecular events such as genetic mutations, translocations, amplifications, deletions and viral gene insertions can affect signal transduction pathways critical to the prevention of the growth of malignant cell types. For example, inactivation of pro-apoptotic proteins or up-regulation of anti-apoptotic proteins lead to unchecked growth of cells and ultimately to cancer. Analysis of relevant biological pathways is key to understanding medically important diseases such as these.

The initial application of the content delivery platform is aimed at detecting binary relationship between concepts such as disease, protein and lipid. This is insufficient to provide useful analysis at a pathway level. Consequently, we extend from the system to enable the navigation of biological pathway.

Here, we extend the prior work with lipid-protein, lipid-disease interaction by adding a generic pathway discovery algorithm to the platform. The algorithm will support tacit knowledge discovery across biological systems such as proteins, lipids and diseases as well as mining for pathway segments that can interactively be re-annotated with relations to other biological entities that can be recognized in the full text documents.

2.1) Pathway Navigation Algorithm

A generic pathway discovery algorithm is implemented to mine all object properties in the ontology in order to discover transitive relationships between 2 entities(Figure 30). Given 2 concept instances Csource and Ctarget, the algorithm seeks to compute a pathway between them in the following steps:

1.The algorithm lists all object property instance triples in which the domain matches Csource.

2.Every listed instance is treated as the source concept instance and the related object property instances are explored. This process is repeated recursively until Ctarget is reached or if no object property instances are found.

3.All resulting transitive paths are output in the ascending order of path length.

We further restrict the generic pathways to protein-protein interaction pathway by adding 2 simple constraints to the generic algorithm:

1. the source and domain concepts are restricted to proteins

2. only object property instances of hasProtein-Protein_Interaction_With are included

To evaluate the performance of the named entity/concept recognition and the effectiveness of the pathway navigation algorithm, we extend the ontology by incorporating 48 protein class entities from a simplified apoptosis pathway into the Monomeric_Protein_or_Protein_Complex_Subunit and Multimeric_Protein_Complex either by importing it from Molecule Roles Ontology or by manually adding them. In addition to that, we construct a gold standard corpus of 10 full-texts papers related to

apoptosis pathway. Our text mining procedure is able to identify 119 sentences and tag these sentences with associated Protein name or Disease name (specifically cancer).

These sentences are re-annotated manually for all accurate mentions of the disease and protein concepts. The system is later evaluated in terms of precision and recall. Precision is defined as the fraction of correct concepts recognized over the total number of concepts output and recall is defined as the fraction of concepts recognized among all correct concepts. See Table 25 for evaluation results. Evaluation shows that the NER achieves performance comparable to state of the art dictionary based approaches.

Named Entities	Mentions		Precision	Recall
	Target	Returned		
Disease	32	37	0.54	0.62
Lipid	58	25	0.96	0.47
Protein	269	181	0.76	0.51
Micro Average			0.75	0.51

Table 25: Precision and recall of name entity recognition

2.2) Navigating Pathways with Knowlegator

Knowlegator permits user to drag 2 proteins into the query canvas and then invoke a search for relation between these 2 concepts (see Figure 32). The results are returned as a list of possible pathways that can be rendered as a chain of labeled concepts and instances illustrating the linkage between 2 starting entities. The path covers a variety of relationships and data types, namely, protein, lipid, disease and provenance data such as sentences or document identifiers. An end user only needs to select a desired path to be viewed on the query canvas (see Figure 32). In addition to that, consistent with our interest in lipids, an additional algorithm is introduced into the knowlegator so that user

can apply specific constraint on existing pathway to discover lipid-protein interaction relevant to the existing pathway. This method overlays new material on top of existing knowledge that is being displayed and it allows the user to control the amount of new knowledge that will be presented and increase it incrementally to facilitate knowledge discovery.

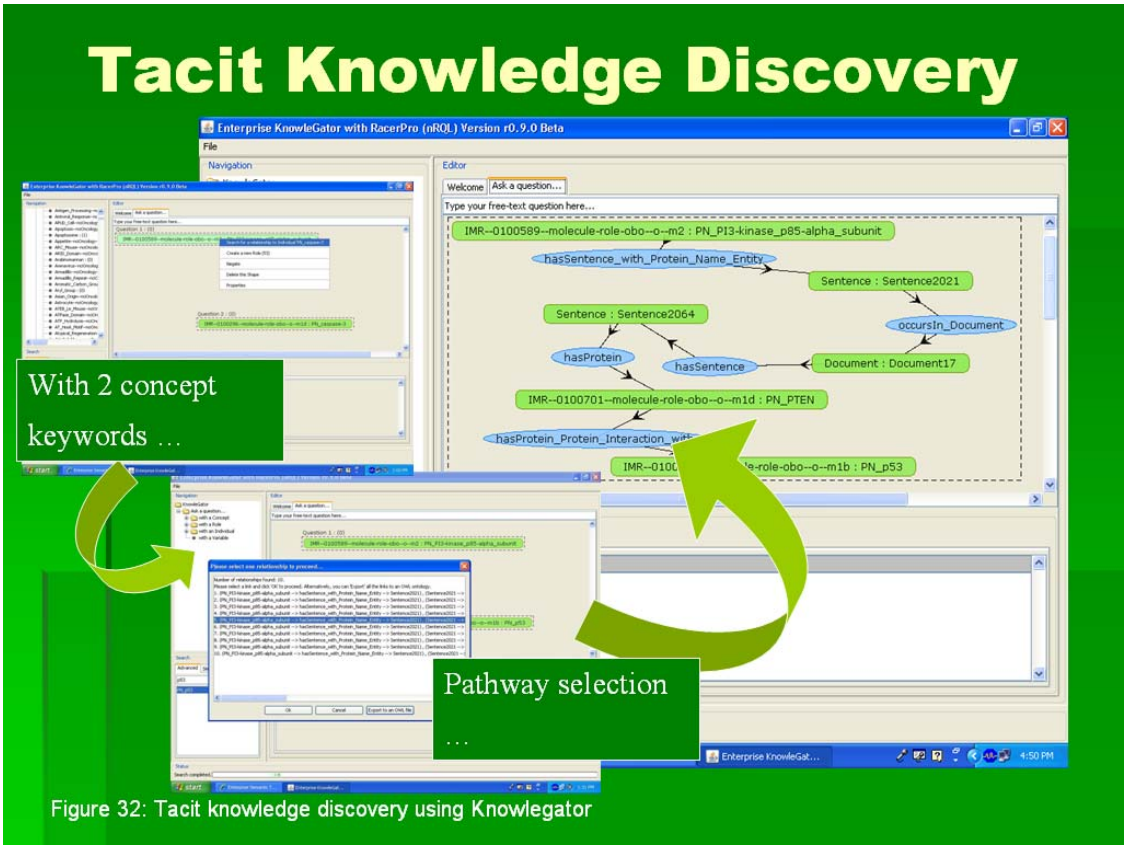


Figure 32: Tacit knowledge discovery using Knowlegator

3) Mining for the Lipidome of Ovarian Cancer

Ovarian cancer is one of the most common gynecological cancers in developed countries and is the fifth leading cause of all cancer-related death afflicting women. It is one of the least understood cancers. If it is detected early, the chances of a patient surviving death

due to ovarian cancer improve to 95%. Lipids are known to play an integral part in the genesis, progression and metastasis stages of the disease. Many researchers hope to discover an effective biomarker, be it lipid or lipid-related protein that is capable of diagnosing the disease at its onset.

Identification of diagnostic biomarkers depends on the understanding of the complex interplays of biomolecules (lipid and protein) that have been reported in the literature. A comprehensive assessment of the lipidome of ovarian cancer from the literature is yet to be available.

We apply ontology-centric knowledge integration platform to address the lack of explicit knowledge in the subject. As described earlier, the platform is a combination of several semantic web technologies such as text mining, OWL-DL ontology and knowledge representation, ontology population and visual query technologies designed to aggregate knowledge from the scientific bibliosphere. Here, we deploy the integrated text mining and semantic navigation infrastructure to explore the role of lipid-protein interactions in ovarian cancer processes with respect to the apoptosis pathway.

7498 PubMed abstracts are identified by manual curation to be relevant to the subject of ovarian cancer. Out of these, 683 abstracts are identified to contain lipid names. We manage to download 241 full text documents. These documents are then subjected to the text conversion and standard text mining procedure employed in our knowledge

integration platform; specifically they are mined for terms related to ovarian cancer, apoptosis, lipids, hormones and proteins.

3.1) Gold Standard Apoptosis Pathway

A gold standard apoptosis pathway is constructed by manual consultation from literature sources. The pathway consists of 71 proteins and is enriched with additional metadata such as Canonical Protein name, Alternative name, Gene name, Sequence Length, Uniprot ID, GO Component, GO Function and GO Process from corresponding Uniprot information.

3.2) Assembling of Additional Term Lists for Text Mining

In addition to the lipid, protein and disease dictionary, we assemble a hormone name list from UMLS. A list of proteins associated to ovarian cancer and apoptosis is manually created from PubMed abstracts. The proteins are provided along with provenance data such as Canonical Protein name, Alternative name, Gene name, Sequence Length, Uniprot ID, GO Component, GO Function and GO Process.

3.4) Mining Relationships

We seek to detect 10 types of relationship pairs. They are Protein(OC)-Protein(OC), Protein(OC)-Protein(Apoptosis), Protein(OC)-Protein(Apoptosis), Lipid-Protein (Apoptosis), Lipid-Protein(OC), Lipid-Lipid, Lipid-Hormone, Hormone-Hormone, Protein(OC)-Hormone and Protein(Apoptosis)-Hormone. As describe before, every relation pair is instantiated as Object Property instances whereas the exact interaction

sentences and relevant provenance information are instantiated as Datatype Propety instances.

3.5) Interaction in the Ovarian Cancer-Apoptosis-Lipidome

A cursory examination of the result indicates interaction among the proteins far out-numbered interaction of other entity pairs. Since our interest is in lipidome, we examine the result for Lipid-related interactions. For complete detail of the mining result, please refer to Table 26.

Interaction Type	Abstract (7498)	Full Paper (241)
OC-AP	505	195
AP-Lipid	10	8
Protein Hormone	9	2
OC-Lipid	11	14
OC-Hormone	8	1
Lipid Hormone	2	18
AP-AP	113	59
OC-OC	223	13
Lipid-Lipid	3	23
Hormone-Hormone	2	6

Table 26: Interactions mined from the ovarian cancer bibliome

Discussion of the biological significance of our finding is beyond the scope of this thesis, but in order to illustrate the effectiveness of knowledge integration platform, we will discuss briefly the lipidome revolving around one of the protein, Akt(Protein Kinase B). Akt is a protein that plays an important role in protein lipidome interaction in ovarian. It is known to affect 2 biological pathways in ovarian cancer, namely the anti-apoptosis and cell metastasis pathways. Our results are able to show that its interaction either directly or indirectly with several lipids. For instance, we identify LPA (lysophosphatidic acid) that

could bind to LPA receptors to initiate a signaling cascade that would end up with activation of Akt. In addition to that, we also discover that phosphatidic acid, a precursor to LPA and Phorbol, a known inhibitor of LPAR/LPA binding associates to the Akt on the graph depicting the text mining results. These lipid compounds may point to additional potential drug targets other than to conventionally presumed PI3K. For full details of the graphical network of the interactions, please see figure.

4) Discussion

Through the coordination of distributed literature resources, natural language processing, ontology development, automated ontology instantiation, visual query guided reasoning over OWL-DL A-boxes, we address the problem of navigating large volumes of complex biological knowledge or data in the field of Lipidomics, with a focus on knowledge found in legacy unstructured full text of scientific publications.

4.1) Role of Ontology in Query

The Lipid Ontology, a knowledge representation in OWL-DL, is both a data structure for a knowledgebase and a query model compatible to semantic web technologies such as nRQL and RACER reasoner. This, couple with an interface that is capable of bridging the ontology and the reasoning engine, we present to end user several query paradigms that greatly improve usability and effectiveness of knowledgebase system.

4.2) Query Paradigms of Knowlegator

An OWL-DL ontology models specific domain knowledge and represents the domain in a fashion that is consistent to the knowledge framework in mind of an end user. In addition to that, the ontology provides additional DL capability for reasoning purposes. When such an ontology is loaded into Knowlegator, the visual query interface presents a visual query model/system that is highly intuitive and interactive to end users. This ontology-centric visual query paradigm allows end users to build complex and deep query with minimal learning curve and without the need to understand query syntax of SQL or nRQL. The additional semantic richness of an OWL-DL ontology allows direct access to provenance information (such as sentences, identifiers, titles) related to the concepts that are being queried. Lastly, visual query paradigm provides ease of navigation for end user when navigating large graphs of pathways as demonstrated in our application scenario.

To further comment on the capability of the visual query paradigm, we compare the visual query model with using the same query, specifically "lipids that interact with proteins, which occur in a particular sentence of a particular document that are at the same time related to a particular disease" against a relational database (see Figure 33). The same query can be easily constructed from the relationships in the ontology via visual query compared to the relational database. For the database scenario, in order to process this query, each concept needs to be modeled into separate tables and each relationship needs to be modeled into additional connection tables to reduce redundancies. An SQL query statement for the query above would require 8 table joins. Such a SQL query is not intuitive to a user without prior knowledge of the database. Moreover, the

type of queries that a user can make is more or less restricted in a relational database. To enable new query, database query model and structure would need to change. This is not so for the ontology-centric visual query paradigm, as an OWL-DL ontology is built in with many relationships and concepts to formulate complex query with greater flexibility while remaining consistent to the knowledge in the mind of an end user.

The implementation knowledge navigation algorithm further improves then usability of the platform by enabling tacit knowledge discovery between 2 concepts (with or without constraint on the types of concept). This allows users to generate cross discipline paths or stepwise extensions to existing know paths by adding additional annotations or alternate paths such as overlaying lipids on top on an existing protein-protein interaction pathway.

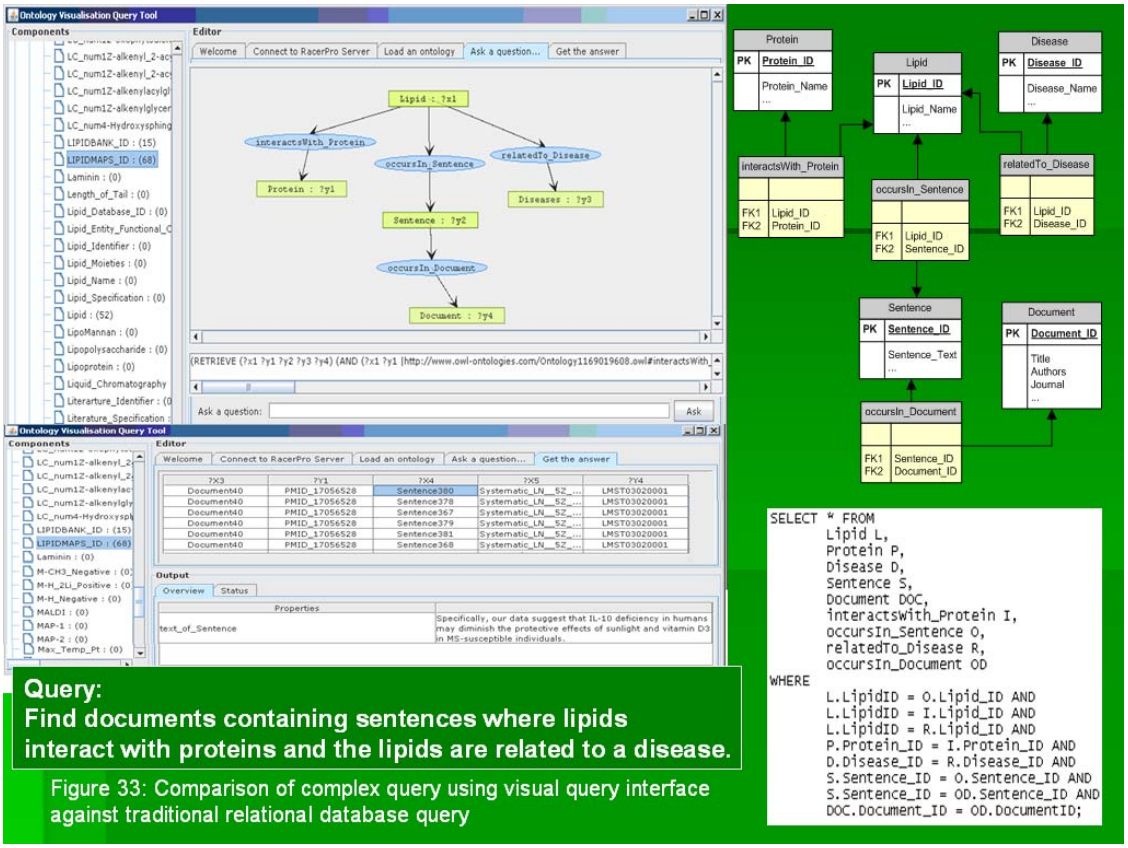


Figure 33: Comparison of complex query using visual query interface against traditional relational database query

5) Conclusion

We build a Lipid Ontology in the Web Ontology Language (OWL) to represent the knowledge of lipids and their relationship to other biological entities such as protein, pathway and disease. The ontology model resolves nomenclature inconsistencies by grounding lipid synonyms to individual lipid names. We report a document delivery system that in conjunction with a lipid specific text mining platform instantiates lipid sentences into the Lipid Ontology. Navigation of lipid literature is then facilitated using a drag 'n' drop visual query composer which poses description logic queries to the OWL-DL ontology. In addition to that, we also develop a pathway navigation algorithm that enable tacit knowledge discovery between 2 concepts. We apply this content delivery and knowledge navigation platform successfully to assess the lipidome of ovarian cancer with

respect to apoptosis pathway. Future direction of this work involves scaling up the coverage of this platform and employing more effective text mining techniques.

Chapter VI: Conclusion

We describe 5 ontologies, namely Lipid Ontology 1.0, Lipid Ontology Reference, Lipid Ontology Ov, Lipid Classification Ontology (LiCO) and Lipid Entity Representation Ontology (LERO). Lipid Ontology 1.0 is a basic application ontology that integrates bibliographic information with the existing data from lipid databases and provides a basic query model for the Knowlegator platform while Lipid Ontology Reference provides a content rich reference from which other, simpler, specialized application ontologies can be developed. Lipid Ontology Ov is a specific application ontology that has been applied to assess the lipidome of ovarian cancer with respect to apoptosis in the bibliosphere. LiCO contains formalized DL definitions of lipids whereas LERO extends from LiCO to include other lipid-related informations such as synonyms and database identifiers. Together, these ontologies have been used to represent knowledge of lipids for various purposes. These ontologies, while embryonic in their nature have demonstrated that OWL-DL ontologies are adequate for the task of representing knowledge from the biological domain and subsequent be applied in a way that would benefit scientific research through coordinated efforts involving other semantic web technologies.

We have demonstrated the usefulness of ontologies in a content acquiring, text-mining, NLP, intuitive query and information navigation application applied to the field of lipidomics. Future work in this area includes scaling up the coverage of this platform, employing more effective text mining techniques and using more rigorously defined ontologies.

References:

1. The Lipid Library: http://www.lipidlibrary.co.uk

2. Fahy E, Subramaniam S, Brown HA, Glass CK, Merrill AH Jr, Murphy RC, Raetz CR, Russell DW, Seyama Y, Shaw W, Shimizu T, Spener F, van Meer G, VanNieuwenhze MS, White SH, Witztum JL, Dennis EA: **A comprehensive classification system for lipids.** *J. Lipid Res.* 2005, 46: 839-862.

3. Gross RW, Jenkins CM, Yang J, Mancuso DJ, Han X: **Functional lipidomics: the roles of specialized lipids and lipid-protein interactions in modulating neuronal function.** *Prostaglandins & other Lipid Mediators.* 2005, 77: 52-64.

4. Fernandis AZ, Wenk MR: **Membrane lipids as signaling molecules.** *Curr. Opin. Lipidol.* 2007, 18: 121-128.

5. Kiebish MA, Han X, Cheng H, Chuang JH, Seyfried TN: **Cardiolipin and electron transport chain abnormalities in mouse brain tumor mitochondria: Lipidomic evidence supporting the Warburg Theory of Cancer.** *J. Lipid Res.* 2008, 49: 2545-2556.

6. Warburg O: **On the origin of cancer cells.** *Science.* 1956, 123: 309-314.

7. Menendez JA, Lupu R: **Fatty acid synthanse and lipogenic phenotype in cancer pathogenesis.** *Nat. Rev. Can.* 2007, 7: 763-777.

8. Huwiler A, Zangemeister-Wittke U: **Targeting the conversion of ceramide to sphingosine 1-phosphate as a novel strategy for cancer therapy.** *Crit. Rev. Oncology/Hematology.* 2007, 63: 150-159.

9. Chiang KP, Niessen S, Saghatellan A, Cravatt BF: **An enzyme that regulates ether lipid signaling pathways in cancer annotated by multidimensional profiling.** *Chem. & Biol.* 2006, 13: 1041-1050.

10. Wenk MR: **The emerging field of Lipidomics.** *Nat. Rev. Drug Discov.* 2005, 4: 594-610.

11. Watson AD: **Lipidomics: a global approach to lipid analysis in biological systems.** *J. Lipid Res.* 2006, 47: 2101-2111.

12. Yetukuri L, Katajamaa M, Medina-Gomez G, Seppanen-Laakso T, Vidal-Puig A, Oresic M: **Bioinformatics strategies for lipidomics analysis: characterization of obesity related hepatic steatosis.** *BMC Syst. Biol.* 2007, 1:12-27.

13. The PubChem Project: http://pubchem.ncbi.nlm.nih.gov/

14. IUPAC-IUB Commission on Biochemical Nomenclature (CBN): **The nomenclature of lipids (recommendations 1976).** *Eur. J. Biochem.* 1977, 79: 11–21.

15. Sud M, Fahy E, Cotter D, Brown A, Dennis EA, Glass CK, Merrill AH Jr, Murphy RC, Raetz CR, Russell DW, Subramaniam S: **LMSD: LIPID MAPS structure database.** *Nucleic Acids Res.* 2007, 35: D527-D532.

16. Berners-Lee T, Hendler J, Lassila O: **The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities.** *Scientific American.* 2001.

17. Gruber T. **Ontology.** In Liu L, Ozsu MT. (Eds): *Encyclopedia of Database System,* Springer-Verlag, 2008.

18. Sankar P, Aghila G: **Design and development of chemical ontologies for reaction representation.** *J. Chem. Inf. Model.* 2006, 46: 2355-2368.

19. Smith B: **Ontology (Science).** *Nature Preceedings.* 2008. (http://precedings.nature.com/documents/2027/version/2/html)

20. Horridge M, Knublauch H, Rector A, Stevens R, Wroe C: **A practical guide to building OWL ontologies using the Protégé plug-in and CO-ODE tools edition 1.0.** The University of Manchester. 2004.

21. Golbreich C, Horridge M, Horrocks I, Motik B, Shearer R: **OBO and OWL: Leveraging semantic web technologies for life sciences. .** In: Aberer K, Choi K-S, Noy N, Allemang D, Lee K-I, Nixon L, Golbeck J, Mika P, Maynard D, Mizoguchi R, Schreiber G, Cudré-Mauroux P (Eds). *The Semantic Web*: Springer 2008, pp. 169-182.

22. The Open Biomedical Ontologies: http://www.obofoundry.org/

23. Alexopoulou D, Wächter T, Pickersgill L, Eyre C, Schroeder M: **Terminologies for text-mining; an experiment in the lipoprotein metabolism domain.** *BMC Bioinformatics* 2008, 9(Suppl 4):S2.

24. Smith B, Ceusters W, Klagges B, Kohler J, Kumar A, Lomax J, Mungall C, Neuhaus F, Rector AL, Rosse C: **Relations in biomedical ontologies.** *Genome Biol.* 2005, 6:R46.

25. Feldman HJ, Dumontier M, Ling S, Hogue CWW: **CO: A Chemical Ontology for Identification of Functional Groups and Semantic Comparison of Small Molecules.** *FEBS Letters.* 2005, 579:4685-4691

26. Villanueva-Rosales N, Dumontier M: **Describing chemical functional groups in OWL-DL for the classification of chemical compounds.** 2007, OWL:

Experiences and Directions (OWLED 2007), colocated with European Semantic Web Confernece (ESWC2007), Innsbruck, Austria.

27. Degtyarenko K, de Matos P, Ennis M, Hastings J, Zbinden M, McNaught A, Alcántara R, Darsow M, Guedj M, Ashburner M: **ChEBI: a database and ontology for chemical entities of biological interest.** *Nucleic Acids Res.* 2008, 36: D344–D350.

28. Coles SJ, Day NE, Murray-Rust P, Rzepa HS, Zhang Y: **Enhancement of the chemical semantic web through the use of InChI identifiers.** *Org. Biomol. Chem.*, 2005, 3:1832-1834

29. The IUPAC International Chemical Identifier (InChI™) : http://old.iupac.org/inchi/release102.html

30. Prasanna MD, Vondrasek J, Wlodawer A, Rodriguez H, Bhat TN: **Chemical compound navigator: A web-based chem-BLAST, chemical taxonomy-based search engine for browsing compounds.** *Protein.* 2006, 63(14):907-917

31. Sun B, Mitra P, Giles CL: **Mining, Indexing, and Searching for Textual Chemical Molecule Information on the Web.** 2008, WWW2008: 17th World Wide Web Conference.

32. Baker CJO, Kanagasabai R, Ang WT, Veeramani A, Low H-S, Wenk MR: **Towards ontology-driven navigation of the lipid *bibliosphere.*** *BMC Bioinformatics*. 2008, 9(Suppl 1):S5.

33. Castro AG, Rocca-Serra P, Stevens R, Taylor C, Nashar K, Ragan MA, Sansone S-A: **The use of concept map during knowledge elicitation in ontology development processes – the nutrigenomics use case.** *BMC Bionformatics.* 2006, 7:267-281.

34. Koh J and Wenk MR: **Lipid Data Warehouse** *(Unpublished)*

35. Watanabe K, Yasugi E, and Oshima M: **"How to search the glycolipid data in LIPIDBANK for Web: the newly developed lipid database".** *Japan Trend Glycosci. and Glycotechnol.* 2000, 12:175-184.

36. Kanehisa M, Goto S, Kawashima S, Okuno Y, Hattori M: **The KEGG resource for deciphering the genome.** *Nucleic Acids Res.* 2004, 32: D277-280

37. The Wikipedia Project: http://en.wikipedia.org/wiki/Main_Page

38. Basic Formal Ontology(BFO): http://www.ifomis.org/bfo

39. BioTop: http://www.imbi.uni-freiburg.de/biotop/

40. Shaban-Nejad A, Baker CJO, Haarslev V, Butler G: **The FungalWeb Ontology: Semantic Web Challenges in Bioinformatics and Genomics.** In: Gil Y, Motta E, Benjamins VR, Musen MA (Eds). *The Semantic Web- ISWC 2005*: Springer 2005, pp. 1063-1066.

41. Disease Ontology: http://diseaseontology.sourceforge.net/

42. Sioutos N, de Coronado S, Haber MW, Hartel FW, Shaiu WL, Wright LW: **NCI Thesaurus: a semantic model integrating cancer-related clinical and molecular information.** *J Biomed Inform.* 2007, 40:30-43.

43. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G: **Gene ontology: tool for the unification of biology. The Gene Ontology Consortium.** *Nat Genet.* 2000, 25:25-9.

44. The Pathway Ontology: http://purl.org/obo/owl/PW

45. The Molecule Role Ontology: http://purl.org/obo/owl/IMR

46. Aranguren ME: Ontology design patterns for the formalization of biological ontologies. (M.Sc. Thesis, University of Manchester, 2005).

47. The Protégé project, Stanford University: http://protege.stanford.edu.

48. Fridman Noy N and Musen M: **PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment.** In Proceedings of AAAI-2000, Austin, Texas. MIT Press/AAAI Press, 2000.

49. OWLviz: http://www.co-ode.org/downloads/owlviz/

50. Jambalaya, Stanford University: http://www.thechiselgroup.org/jambalaya

51. Patel-Schneider PF, Hayes P, Horrocks I: **OWL Web Ontology Language Semantics and Abstract Syntax, W3C Recommendation,** 2004, http://www.w3.org/TR/owl-semantics/, last accessed 6 December 2005.

52. Schomburg I, Chang A, Schomburg D: **BRENDA, enzyme data and metabolic information.** *Nucleic Acids Res.* 2002, 30: 47-49.

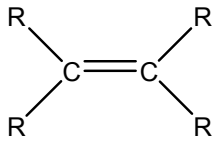
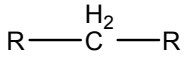
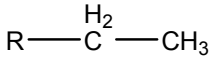
53. Boeckmann B, Bairoch A, Apweiler R, Blatter M-C, Estreicher A, Gasteiger E, Martin MJ, Michoud K, O'Donovan C, Phan I, Pilbout S, Schneider M: **The**

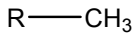
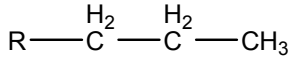
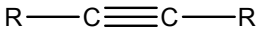
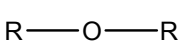
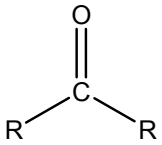
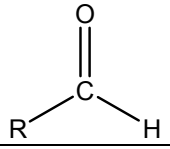
SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res.* 2003, 31: 365-370.

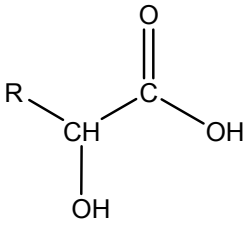
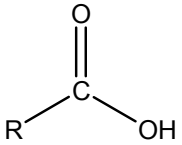
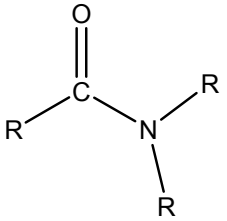
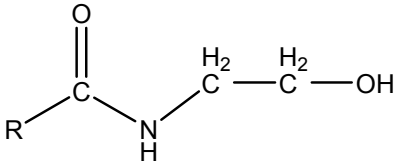
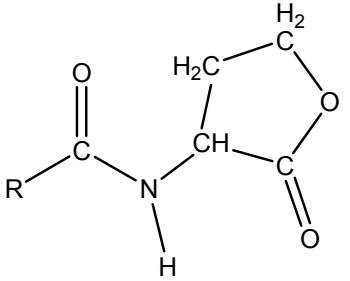
54. Barry CE 3rd, Lee RE, Mdluli K, Sampson AE, Schroeder BG, Slayden RA, Yuan Y: **Mycolic acids: structure, biosynthesis and physiological functions.** *Prog. Lip. Res.* 1998, 37:143-179

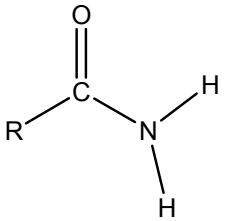
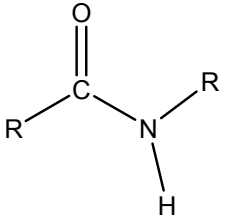
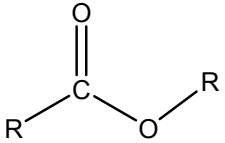
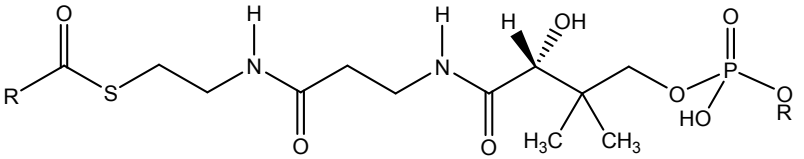
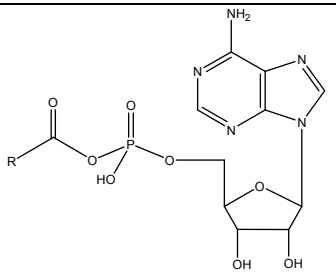
55. Wolstencroft K, Lord P, Tabernero L, Brass A, and Stevens R: **Protein classification using ontology classification.** *Bioinformatics.* 2006, 22: e530 - e538

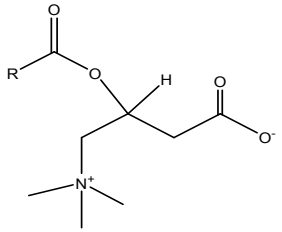
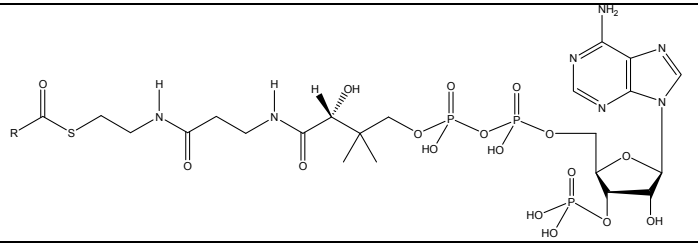
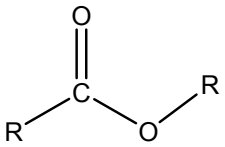
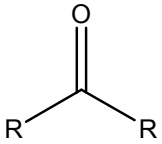
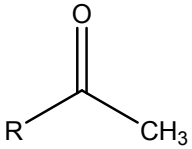
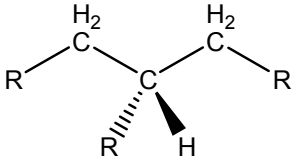
Appendix A: The Simple Organic Group made available in Lipid Ontology (LiCO & LERO)

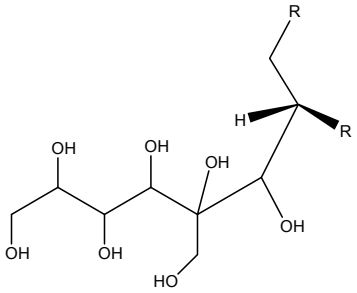
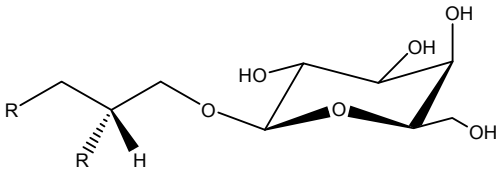
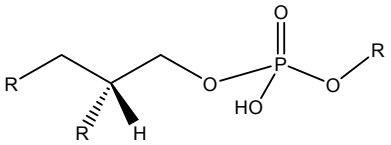
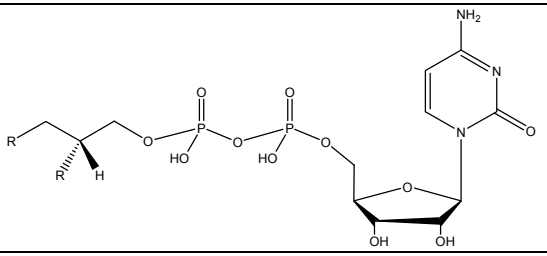
Simple Organic Group name		General structure
Superclass	Class	
Carbon Group	Aliphatic Carbon Group	--
Aliphatic Carbon Group	Alkenyl Group*	
Alkenyl Group	Distal Alkenyl	An alkenyl group that is located furthest from origin of a carbon chain and nearest to the terminal end of the chain. Also define as the more (or most) distant of two (or more) alkenyl groups along the same carbon chain.
Alkenyl Group	Proximal Alkenyl	An alkenyl group that is located nearest to the origin of a carbon chain and furthest from the terminal end of the chain. Opposite of distal alkenyl group along the same carbon chain.
Aliphatic Carbon Group	Alkyl Group	
Alkyl Group	Ethyl	

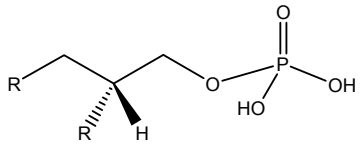
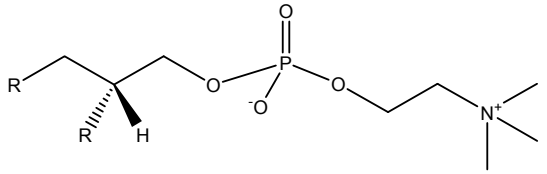
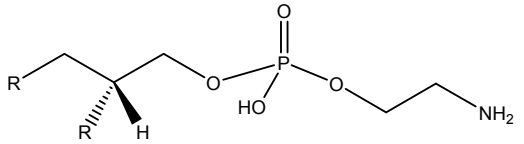
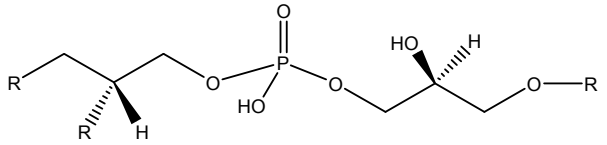
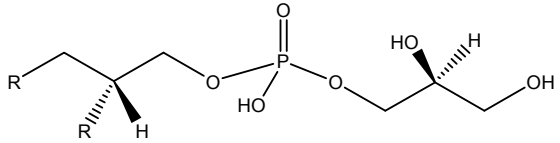
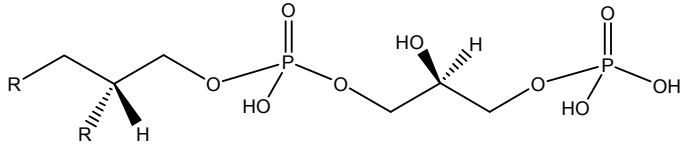
Alkyl Group	Methyl	
Alkyl Group	Propyl	
Aliphatic Carbon Group	Alkyl Halide Group*	 X = halide
Aliphatic Carbon Group	Alkynyl Group	
Carbon Group	Ether*	
Carbon Group	Hydroxy Compound*	
Hydroxy Compound	Alcohol*	
Carbonyl Compound Group*	Carboxylic Acid derivative Group*	
Carboxylic Acid derivative Group	Aldehyde*	

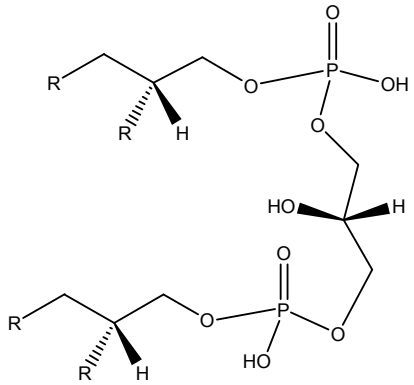
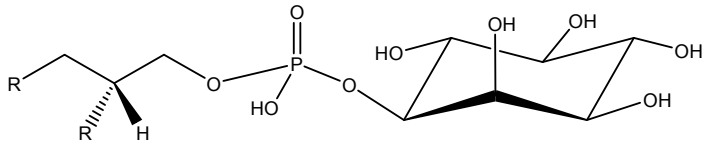
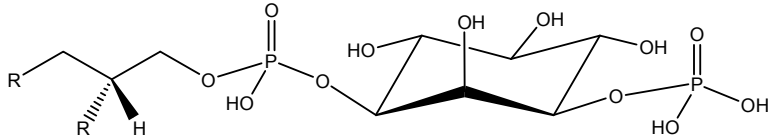
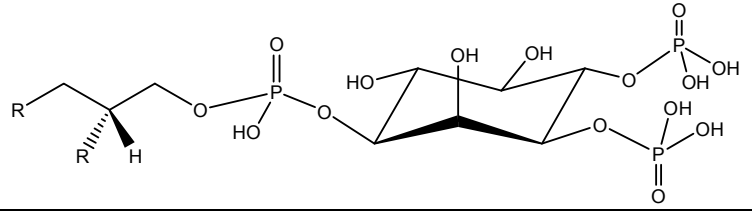
Carboxylic Acid derivative Group	Alpha-Hydroxy Acid Group*	
Carboxylic Acid derivative Group	Carboxylic Acid*	
Carboxylic Acid derivative Group	Carboxylic Acid Amide Group*	
Carboxylic Acid Amide Group	Carboxylic Acid Ethanolamine	
Carboxylic Acid Amide Group	Carboxylic Acid Homoserine Lactone	

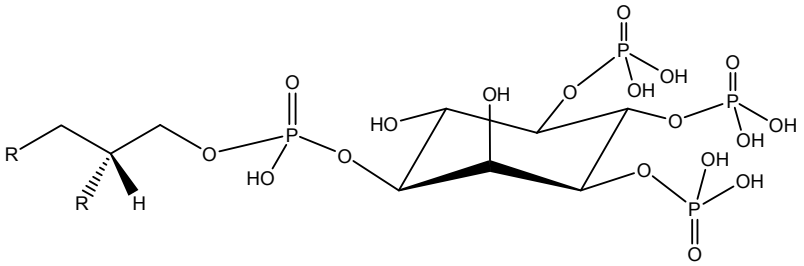
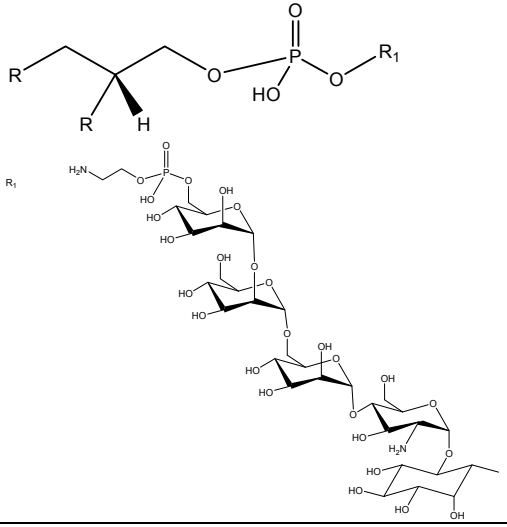
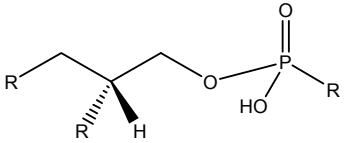
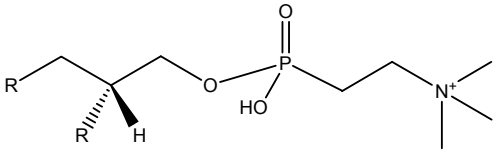
Carboxylic Acid Amide Group	Carboxylic Acid Primary Amide*	
Carboxylic Acid Amide Group	Carboxylic Acid Secondary Amide*	
Carboxylic Acid derivative Group	Carboxylic Acid Ester Group*	
Carboxylic Acid Ester Group	Carboxylic Acid-ACP	
Carboxylic Acid Ester Group	Carboxylic Acid-Adenosine Monophosphate	

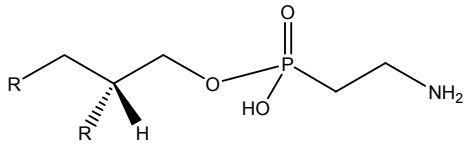
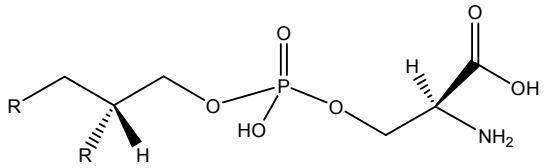
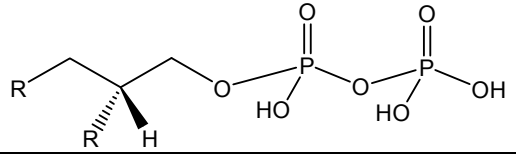
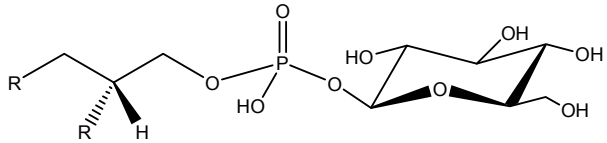
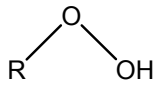
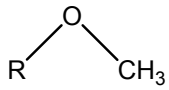
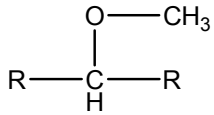
Carboxylic Acid Ester Group	Carboxylic Acid-Carnitine	
Carboxylic Acid Ester Group	Carboxylic Acid-CoA	
Carboxylic Acid Ester Group	Carboxylic Acid Ester*	
Carboxylic Acid derivative Group	Ketone*	
Carboxylic Acid derivative Group	Terminal Acyl Group	
Glycerol derivative Group	Glycerol	

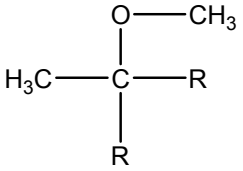
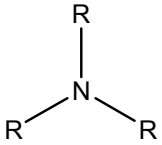
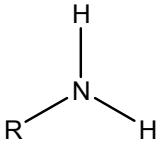
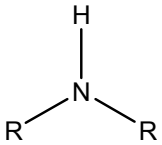
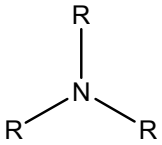
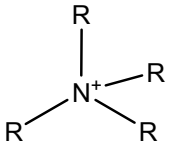
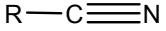
Glycerol derivative Group	Glycerol-nonitol	
Glycerol derivative Group	Glycerolglycan	
Glycerol derivative Group	Glycerophosphate Group	
Glycerophosphate Group	Cytidine Phosphate Glycerol Group	A generic structure similar to Cytidine Diphosphate Glycerol, with undefined number of phosphate group instead of 2.
Cytidine Phosphate Glycerol Group	Cytidine Diphosphate Glycerol	

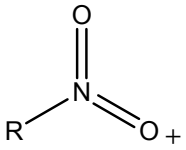
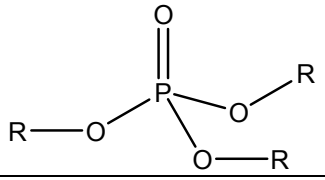
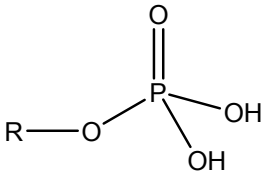
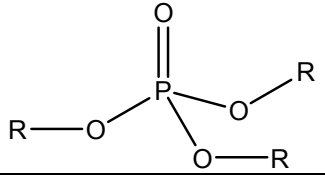
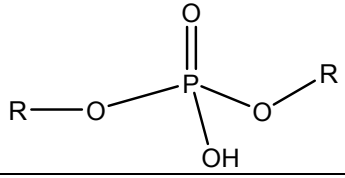
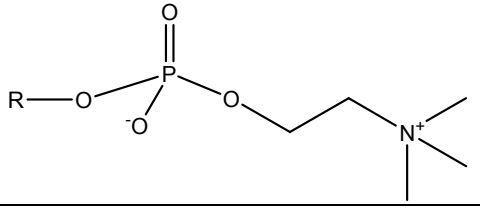
Glycerophosphate Group	Glycerophosphatidic acid	
Glycerophosphate Group	Glycerophosphatidylcholine	
Glycerophosphate Group	Glycerophosohatidylethanolamine	
Glycerophosphate Group	Glycerophosphatidylglycerol Group	
Glycerophosphoglycerol Group	Glycerophosphatidylglycerol	
Glycerophosphoglycerol Group	Glycerophosphatidylglycerophosphate	

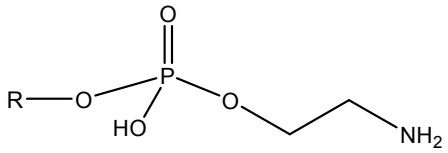
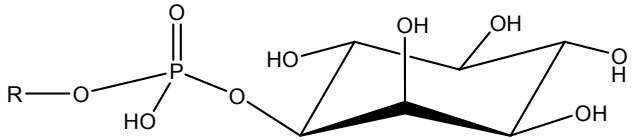
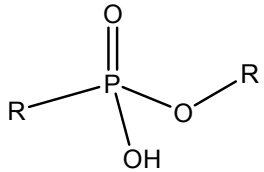
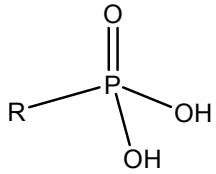
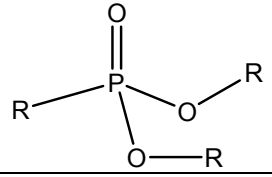
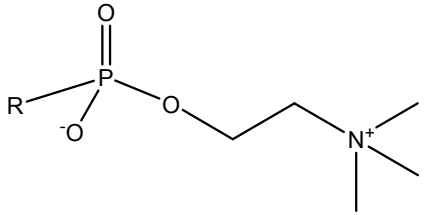
Glycerophosphoglycerol Group	Glycerophosphatidylglycerophosphoglycerol	
Glycerophosphate Group	Glycerophosphatidylinositol Group	A generic structure similar to Glycerophosphatidylinositol, with undefined number of phosphate group instead of 1.
Glycerophospholinositol Group	Glycerophosphatidylinositol	
Glycerophospholinositol Group	Glycerophosphatidylinositol Monophosphate	
Glycerophospholinositol Group	Glycerophosphatidylinositol Diphosphates	

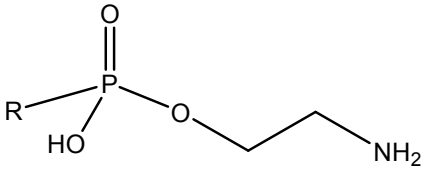
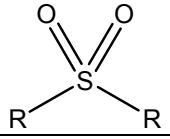
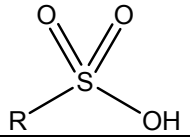
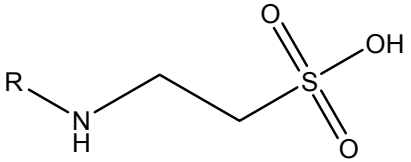
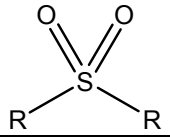
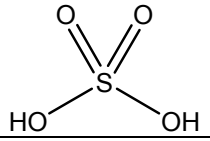
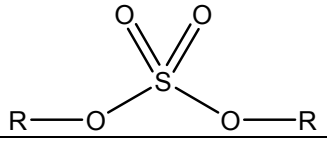
Glycerophospholinositol Group	Glycerophosphatidylinositol Triphosphates	
Glycerophospholinositol Group	Glycerophosphatidylinositol Glycans	
Glycerophosphate Group	Glycerophosphono Group	
Glycerophosphono_Group	Glycerophosphonotidylcholine	

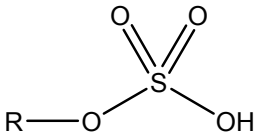
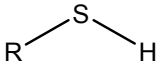
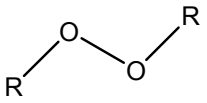
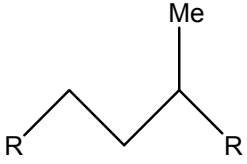
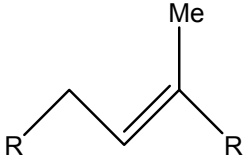
Glycerophosphono_Group	Glycerophosphonotidylethanolamine	
Glycerophosphate Group	Glycerophosphatidylserine	
Glycerophosphate Group	Glyceropyrophosphatidic acid	
Glycerophosphate Group	Glycosylglycerophosphate Group	
Simple Organic Group	Hydroperoxide	
Simple Organic Group	Methoxy Group	
Methoxy Group	non-terminal Methoxy	

Methoxy Group	Terminal Methoxy	
Organic Nitrogen Group	Amine Group*	
Amine Group	Primary Amine*	
Amine Group	Secondary Amine*	
Amine Group	Tertiary Amine*	
Amine Group	Quarternary Ammonium Salt*	
Organic Nitrogen Group	Nitrile Group*	

Organic Nitrogen Group	Nitro Group*	
Organic Phosphate Group	Phosphate Group	
Phosphate Group	Phosphoric Acid*	
Phosphate Group	Phosphoric Acid Diester	
Organic Phosphate Group	Phospho Group	
Phospho Group	Phosphocholine	

Phospho Group	Phosphoethanolamine	
Phospho Group	Phosphoinositol	
Organic Phosphate Group	Phosphono Group	
Phosphono Group	Phosphonic Acid*	
Phosphono Group	Phosphonic Acid Diester	
Phosphono Group	Phosphonocholine	

Phosphono Group	Phosphonoethanolamine	
Organic Sulfur Group	Sulfonic Acid Derivative Group *	
Sulfonic Acid Derivative Group	Sulfonic Acid Group*	
Sulfonic Acid Derivative Group	Taurine	
Organic Sulfur Group	Sulfuric Acid Derivative Group*	
Sulfuric Acid Derivative Group	Sulfuric Acid*	
Sulfuric Acid Derivative Group	Sulfuric Acid Diester*	

Sulfuric Acid Derivative Group	Sulfuric Acid Monoester*	
Organic Sulfur Group	Thiol*	
Simple Organic Group	Peroxide	
Simple Organic Group	Phytyl	 Me = methyl group
Simple Organic Group	Prenyl	 Me = methyl group

Simple organic groups that are re-used from the organic group definitions of Chemical Ontology (CO) [25]

Appendix B: LIPID MAPS Classification hierarchy and equivalent concepts in Lipid Ontology (LiCO & LERO)

- the list of lipid classes from the LIPID MAPS Classification hierarchy is taken from the nomenclature published on LIPID MAPS website dated since 2007 and may differs from the most recent nomenclature published by LIPID MAPS.

Lipid class name from LIPID MAPS classification hierarchy	Equivalent lipid class name/hierarchy in Lipid Ontology(LiCO & LERO)
Fatty Acyls[FA]	LC_Fatty_Acyl
Fatty Acids and Conjugates [FA01]	LC_Fatty_acid
Straight chain fatty acids [FA0101]	LC_Straight_chain_fatty_acid
Methyl branched fatty acids [FA0102]	LC_Methyl_branched_fatty_acid
Unsaturated fatty acids [FA0103]	LC_Unsaturated_fatty_acid
Hydroperoxy fatty acids [FA0104]	LC_Hydroperoxy_fatty_acid
Hydroxy fatty acids [FA0105]	LC_Hydroxy_fatty_acid
Oxo fatty acids [FA0106]	LC_Oxo_fatty_acid
Epoxy fatty acids [FA0107]	LC_Epoxy_fatty_acid
Methoxy fatty acids [FA0108]	LC_Methoxy_fatty_acid
Halogenated fatty acids [FA0109]	LC_Halogenated_fatty_acid
Amino fatty acids [FA0110]	LC_Amino_fatty_acid
Cyano fatty acids [FA0111]	LC_Cyano_fatty_acid
Nitro fatty acids [FA0112]	LC_Nitro_fatty_acid
Thia fatty acids [FA0113]	LC_Thia_fatty_acid
Carbocyclic fatty acids [FA0114]	LC_Carbocyclic_fatty_acid
Heterocyclic fatty acids [FA0115]	LC_Heterocyclic_fatty_acid
Mycolic acids [FA0116]	LC_Mycolic_acid LC_General_alpha_mycolic_acid LC_Alpha_mycolic_acid LC_Oxygenated_alpha_mycolic_acid LC_General_Methoxy_mycolic_acid LC_Methoxy_mycolic_acid LC_Omega-1_methoxy_mycolic_acid

	LC_Keto_mycolic_acid LC_Wax_ester_mycolic_acid LC_General_methylated_mycolic_acid LC_Alpha_2_mycolic_acid LC_Epoxy_mycolic_acid LC_General_mycolic_acid LC_Alpha_1_mycolic_acid LC_Alpha_prime_mycolic_acids
Dicarboxylic acids [FA0117]	LC_Dicarboxylic_acid
Octadecanoids [FA02]	LC_Octadecanoid
12oxophytodienoic acid metabolites [FA0201]	LC_num12-oxophytodienoic_acid_metabolite
Jasmonic acids [FA0202]	LC_Jasmonic_acid
Other Octadecanoids [FA0200]	-
Eicosanoids [FA03]	LC_Eicosanoid
Prostaglandins [FA0301]	LC_Prostaglandin
Leukotrienes [FA0302]	LC_Leukotriene
Thromboxanes [FA0303]	LC_Thromboxane
Lipoxins [FA0304]	LC_Lipoxin
Hydroxy/hydroperoxyeicosatrienoic acids [FA0305]	LC_Eicosatrienoic_acid_derivative LC_Hydroperoxyeicosatrienoic_acid LC_Hydroxyeicosatrienoic_acid
Hydroxy/hydroperoxyeicosatetraenoic acids [FA0306]	LC_Eicosatetraenoic_acid_derivative LC_Hydroperoxyeicosatetraenoic_acid LC_Hydroxyeicosatetraenoic_acid
Hydroxy/hydroperoxyeicosapentaenoic acids [FA0307]	LC_Eicosapentaenoic_acid_derivative LC_Hydroperoxyeicosapentaenoic_acid LC_Hydroxyeicosapentaenoic_acid
Epoxyeicosatrienoic acids [FA0308]	LC_Epoxyeicosatrienoic_acid
Hepoxilins [FA0309]	LC_Hepoxilin
Levuglandins [FA0310]	LC_Levuglandin
Isoprostanes [FA0311]	LC_Isoprostane
Clavulones and derivatives [FA0312]	LC_Clavulone_structural_derivative LC_Clavulone LC_Clavulone_derivative

Other Eicosanoids [FA0300]	-
Docosanoids [FA04]	LC_Docosanoid
Fatty alcohols [FA05]	LC_Fatty_alcohol
Fatty aldehydes [FA06]	LC_Fatty_aldehyde
Fatty esters [FA07]	LC_Fatty_ester
Wax monoesters [FA0701]	LC_Wax_monoester
Wax diesters [FA0702]	LC_Wax_diester
Cyano esters [FA0703]	LC_Cyano_ester
Lactones [FA0704]	LC_Lactone
Fatty acyl CoAs [FA0705]	LC_Fatty_acyl_CoA
Fatty acyl ACPs [FA0706]	LC_Fatty_acyl_ACP
Fatty acyl carnitines [FA0707]	LC_Fatty_acyl_carnitine
Fatty acyl adenylates [FA0708]	LC_Fatty_acyl_adenylate
Fatty amides [FA08]	LC_Fatty_amide
Primary amides [FA0801]	LC_Primary_amide
Nacyl amines [FA0802]	LC_N-acyl_amine
Fatty acyl homoserine lactones [FA0803]	LC_Fatty_acyl_homoserine_lactone
Nacyl ethanolamines (endocannabinoids) [FA0804]	LC_N-acyl_ethanolamine_par_endocannabinoid_par_
Fatty nitriles [FA09]	LC_Fatty_nitrile
Fatty ethers [FA10]	LC_Fatty_ether
Hydrocarbons [FA11]	LC_Hydrocarbon
Oxygenated hydrocarbons [FA12]	-
Fatty acyl glycosides [FA13]	-
Fatty acyl glycosides of mono and disaccharides [FA1301]	-
Sophorolipids [FA1302]	-
Rhamnolipids [FA1303]	-
Other Fatty acyl glycosides [FA1300]	-
Other Fatty Acyls [FA00]	-
Glycerolipids[GL]	LC_Glycerolipid
Monoradylglycerols [GL01]	LC_Monoradylglycerol
Monoacylglycerols [GL0101]	LC_Monoacylglycerol

Monoalkylglycerols [GL0102]	LC_Monoalkylglycerol
Mono(1Zalkenyl)glycerols [GL0103]	LC_Mono-par-1Z-alkenyl-par-glycerol
Diradylglycerols [GL02]	LC_Diradylglycerol
Diacylglycerols [GL0201]	LC_Diacylglycerol
1alkyl,2acylglycerols [GL0202]	LC_1-alkyl_2-acylglycerol
1acyl,2alkylglycerols [GL0207]	LC_1-acyl_2-alkylglycerol
Dialkylglycerols [GL0203]	LC_Dialkylglycerol
1Zalkenylacylglycerols [GL0204]	LC_num1Z-alkenylacylglycerol
Diglycerol tetraethers [GL0205]	LC_Di-glycerol_tetraether
Diglycerol tetraether glycans [GL0206]	LC_Di-glycerol_tetraether_glycan
Triradylglycerols [GL03]	LC_Triradylglycerol
Triacylglycerols [GL0301]	LC_Triacylglycerols
Alkyldiacylglycerols [GL0302]	LC_Alkyldiacylglycerols
Dialkylmonoacylglycerols [GL0303]	LC_Dialkylmonoacylglycerols
1Zalkenyldiacylglycerols [GL0304]	LC_num1Z-alkenyldiacylglycerols
Estolides [GL0305]	LC_Estolides
Glycosylmonoradylglycerols [GL04]	LC_Glycosylmonoradylglycerol
Glycosylmonoacylglycerols [GL0401]	LC_Glycosylmonoacylglycerol
Glycosylmonoalkylglycerols [GL0402]	LC_Glycosylmonoalkylglycerol
Glycosyldiradylglycerols [GL05]	LC_Glycosyldiradylglycerol
Glycosyldiacylglycerols [GL0501]	LC_Glycosyldiacylglycerol
Glycosylalkylacylglycerols [GL0502]	LC_Glycosylalkylacylglycerol
Glycosyldialkylglycerols [GL0503]	LC_Glycosyldialkylglycerol
Other Glycerolipids [GL00]	-
Glycerophospholipids[GP]	LC_Glycerophospholipid
Glycerophosphocholines [GP01]	LC_Glycerophosphocholine
Diacylglycerophosphocholines [GP0101]	LC_Diacylglycerophosphocholine
1alkyl,2acylglycerophosphocholines [GP0102]	LC_num1-alkyl_2-acylglycerophosphocholine
1acyl,2alkylglycerophosphocholines [GP0108]	LC_num1-acyl_2-alkylglycerophosphocholine
1Zalkenyl,2acylglycerophosphocholines [GP0103]	LC_num1Z-alkenyl_2-acylglycerophosphocholine
Dialkylglycerophosphocholines [GP0104]	LC_Dialkylglycerophosphocholine

Monoacylglycerophosphocholines [GP0105]	LC_Monoacylglycerophosphocholine
Monoalkylglycerophosphocholines [GP0106]	-
1Zalkenylglycerophosphocholines [GP0107]	LC_num1Z-alkenylglycerophosphocholine
Glycerophosphoethanolamines [GP02]	LC_Glycerophosphoethanolamine
Diacylglycerophosphoethanolamines [GP0201]	LC_Diacylglycerophosphoethanolamine
1alkyl,2acylglycerophosphoethanolamines [GP0202]	-
1acyl,2alkylglycerophosphoethanolamines [GP0208]	-
1Zalkenyl,2acylglycerophosphoethanolamines [GP0203]	-
Dialkylglycerophosphoethanolamines [GP0204]	LC_Dialkylglycerophosphoethanolamine
Monoacylglycerophosphoethanolamines [GP0205]	LC_Monoacylglycerophosphoethanolamine
Monoalkylglycerophosphoethanolamines [GP0206]	-
1Zalkenylglycerophosphoethanolamines [GP0207]	LC_num1Z-alkenylglycerophosphoethanolamine
Glycerophosphoserines [GP03]	-
Diacylglycerophosphoserines [GP0301]	-
1alkyl, 2acylglycerophosphoserines [GP0302]	-
1Zalkenyl,2acylglycerophosphoserines [GP0303]	-
Dialkylglycerophosphoserines [GP0304]	-
Monoacylglycerophosphoserines [GP0305]	-
Monoalkylglycerophosphoserines [GP0306]	-
1Zalkenylglycerophosphoserines[GP0307]	-
Glycerophosphoglycerols [GP04]	LC_Glycerophosphoglycerol
Diacylglycerophosphoglycerols [GP0401]	LC_Diacylglycerophosphoglycerol
1alkyl,2acylglycerophosphoglycerols[GP0402]	LC_num1-alkyl_2-acylglycerophosphoglycerol
1acyl,2alkylglycerophosphoglycerols[GP0411]	-
1Zalkenyl,2acylglycerophosphoglycerols[GP0403]	LC_num1Z-alkenyl_2-acylglycerophosphoglycerol
Dialkylglycerophosphoglycerols [GP0404]	LC_Dialkylglycerophosphoglycerol
Monoacylglycerophosphoglycerols [GP0405]	LC_Monoacylglycerophosphoglycerol
Monoalkylglycerophosphoglycerols [GP0406]	-
1Zalkenylglycerophosphoglycerols[GP0407]	LC_num1Z-alkenylglycerophosphoglycerol
Diacylglycerophosphodiradylglycerols [GP0408]	LC_Diacylglycerophosphodiradylglycerol
Diacylglycerophosphomonoradylglycerols [GP0409]	LC_Diacylglycerophosphomonoradylglycerol

Monoacylglycerophosphomonoradylglycerols [GP0410]	LC_Monoacylglycerophosphomonoradylglycerol
Glycerophosphoglycerophosphates [GP05]	LC_Glycerophosphoglycerophosphate
Diacylglycerophosphoglycerophosphates [GP0501]	LC_Diacylglycerophosphoglycerophosphate
1alkyl,2acylglycerophosphoglycerophosphates[GP0502]	LC_num1-alkyl_2-acylglycerophosphoglycerophosphate
1Zalkenyl,2acylglycerophosphoglycerophosphates[GP0503]	LC_num1Z-alkenyl_2-acylglycerophosphoglycerophosphate
Dialkylglycerophosphoglycerophosphates [GP0504]	LC_Dialkylglycerophosphoglycerophosphate
Monoacylglycerophosphoglycerophosphates [GP0505]	LC_Monoacylglycerophosphoglycerophosphate
Monoalkylglycerophosphoglycerophosphates [GP0506]	-
1Zalkenylglycerophosphoglycerophosphates[GP0507]	LC_num1Z-alkenylglycerophosphoglycerophosphate
Glycerophosphoinositols [GP06]	LC_Glycerophosphoinositol
Diacylglycerophosphoinositols [GP0601]	LC_Diacylglycerophosphoinositol
1alkyl,2acylglycerophosphoinositols[GP0602]	LC_num1-alkyl_2-acylglycerophosphoinositol
1Zalkenyl,2acylglycerophosphoinositols[GP0603]	LC_num1Z-alkenyl_2-acylglycerophosphoinositol
Dialkylglycerophosphoinositols [GP0604]	LC_Dialkylglycerophosphoinositol
Monoacylglycerophosphoinositols [GP0605]	LC_Monoacylglycerophosphoinositol
Monoalkylglycerophosphoinositols [GP0606]	-
1Zalkenylglycerophosphoinositols[GP0607]	LC_num1Z-alkenylglycerophosphoinositol
Glycerophosphoinositol monophosphates [GP07]	LC_Glycerophosphoinositol_monophosphate
Diacylglycerophosphoinositol monophosphates [GP0701]	LC_Diacylglycerophosphoinositol_monophosphate
1alkyl,2acylglycerophosphoinositol monophosphates [GP0702]	-
1Zalkenyl,2acylglycerophosphoinositol monophosphates [GP0703]	-
Dialkylglycerophosphoinositol monophosphates [GP0704]	-
Monoacylglycerophosphoinositol monophosphates [GP0705]	-
Monoalkylglycerophosphoinositol monophosphates [GP0706]	-
1Zalkenylglycerophosphoinositol monophosphates [GP0707]	-
Glycerophosphoinositol bisphosphates [GP08]	LC_Glycerophosphoinositol_bisphosphate
Diacylglycerophosphoinositol bisphosphates [GP0801]	LC_Diacylglycerophosphoinositol_bisphosphate
1alkyl,2acylglycerophosphoinositol bisphosphates [GP0802]	-
1Zalkenyl,2acylglycerophosphoinositol bisphosphates [GP0803]	-
Monoacylglycerophosphoinositol bisphosphates [GP0804]	-
Monoalkylglycerophosphoinositol bisphosphates [GP0805]	-

1Zalkenylglycerophosphoinositol bisphosphates [GP0806]	-
Glycerophosphoinositol trisphosphates [GP09]	LC_Glycerophosphoinositol_trisphosphate
Diacylglycerophosphoinositol trisphosphates [GP0901]	LC_Diacylglycerophosphoinositol_trisphosphate
1alkyl,2acylglycerophosphoinositol trisphosphates [GP0902]	-
1Zalkenyl,2acylglycerophosphoinositol trisphosphates [GP0903]	-
Monoacylglycerophosphoinositol trisphosphates [GP0904]	-
Monoalkylglycerophosphoinositol trisphosphates [GP0905]	-
1Zalkenylglycerophosphoinositol trisphosphates [GP0906]	-
Glycerophosphates [GP10]	LC_Glycerophosphate
Diacylglycerophosphates [GP1001]	LC_Diacylglycerophosphate
1alkyl,2acylglycerophosphates[GP1002]	LC_num1-alkyl_2-acylglycerophosphate
1Zalkenyl,2acylglycerophosphates[GP1003]	LC_num1Z-alkenyl_2-acylglycerophosphate
Dialkylglycerophosphates [GP1004]	LC_Dialkylglycerophosphate
Monoacylglycerophosphates [GP1005]	LC_Monoacylglycerophosphate
Monoalkylglycerophosphates [GP1006]	-
1Zalkenylglycerophosphates[GP1007]	LC_num1Z-alkenylglycerophosphate
Glyceropyrophosphates [GP11]	LC_Glyceropyrophosphate
Diacylglyceropyrophosphates [GP1101]	LC_Diacylglyceropyrophosphate
Monoacylglyceropyrophosphates [GP1102]	LC_Monoacylglyceropyrophosphate
Glycerophosphoglycerophosphoglycerols [GP12]	LC_Glycerophosphoglycerophosphoglycerol
Diacylglycerophosphoglycerophosphodiradylglycerols [GP1201]	LC_Diacylglycerophosphoglycerophosphodiradylglycerol
Diacylglycerophosphoglycerophosphomonoradylglycerols [GP1202]	LC_Diacylglycerophosphoglycerophosphomonoradylglycerol
1alkyl,2acylglycerophosphoglycerophosphodiradylglycerols[GP1203]	LC_num1-alkyl_2-acyl glycerophosphoglycerophosphodiradylglycerol
1alkyl,2acylglycerophosphoglycerophosphomonoradylglycerols[GP1204]	LC_num1-alkyl_2-acyl glycerophosphoglycerophosphomonoradylglycerol
1Zalkenyl,2acylglycerophosphoglycerophosphodiradylglycerols[GP1205]	LC_num1Z-alkenyl_2-acyl glycerophosphoglycerophosphodiradylglycerol
1Zalkenyl,2acylglycerophosphoglycerophosphomonoradylglycerols[GP1206]	LC_num1Z-alkenyl_2-acyl glycerophosphoglycerophosphomonoradylglycerol
Dialkylglycerophosphoglycerophosphodiradylglycerols [GP1212]	LC_Dialkylglycerophosphoglycerophosphodiradylglycerol
Dialkylglycerophosphoglycerophosphomonoradylglycerols [GP1213]	LC_Dialkylglycerophosphoglycerophosphomonoradylglycerol

Monoacylglycerophosphoglycerophosphomonoradylglycerols [GP1207]	LC_Monoacylglycerophosphoglycerophosphomonoradylglycerol
Monoalkylglycerophosphoglycerophosphodiradylglycerols [GP1208]	-
Monoalkylglycerophosphoglycerophosphomonoradylglycerols [GP1209]	-
1Zalkenylglycerophosphoglycerophosphodiradylglycerols [GP1210]	LC_num1Z-alkenyl glycerophosphoglycerophosphodiradylglycerol
1Zalkenylglycerophosphoglycerophosphomonoradylglycerols[GP1211]	LC_num1Z-alkenyl glycerophosphoglycerophosphomonoradylglycerol
CDPGlycerols[GP13]	LC_CDP-Glycerol
CDPdiacylglycerols[GP1301]	LC_CDP-diacylglycerol
CDP1alkyl,2acylglycerols[GP1302]	LC_CDP-1-alkyl_2-acylglycerol
CDP1Zalkenyl,2acylglycerols[GP1303]	LC_CDP-1Z-alkenyl_2-acylglycerol
CDPDialkylglycerols[GP1304]	LC_CDP-dialkylglycerol
CDPMonoacylglycerols[GP1305]	LC_CDP-monoacylglycerol
CDPMonoalkylglycerols[GP1306]	-
CDP1Zalkenylglycerols[GP1307]	LC_CDP-1Z-alkenylglycerol
Glycosylglycerophospholipids [GP14]	LC_Glycosylglycerophospholipid
Diacylglycosylglycerophospholipids [GP1401]	LC_Diacylglycosylglycerophospholipid
1alkyl,2acylglycosylglycerophospholipids[GP1402]	LC_num1-alkyl_2-acylglycosylglycerophospholipid
1Zalkenyl,2acylglycosylglycerophospholipids[GP1403]	LC_num1Z-alkenyl_2-acylglycosylglycerophospholipid
Dialkylglycosylglycerophospholipids [GP1407]	-
Monoacylglycosylglycerophospholipids [GP1404]	LC_Monoacylglycosylglycerophospholipid
Monoalkylglycosylglycerophospholipids [GP1405]	-
1Zalkenylglycosylglycerophospholipids[GP1406]	LC_num1Z-alkenylglycosylglycerophospholipid
Glycerophosphoinositolglycans [GP15]	LC_Glycerophosphoinositolglycan
Diacylglycerophosphoinositolglycans [GP1501]	LC_Diacylglycerophosphoinositolglycan
1alkyl,2acylglycerophosphoinositolglycans[GP1502]	LC_num1-alkyl_2-acylglycerophosphoinositolglycan
1Zalkenyl,2acylglycerophosphoinositolglycans[GP1503]	LC_num1Z-alkenyl_2-acylglycerophosphoinositolglycan
Monoacylglycerophosphoinositolglycans [GP1504]	-
Monoalkylglycerophosphoinositolglycans [GP1505]	-
1Zalkenylglycerophosphoinositolglycans[GP1506]	LC_num1Z-alkenylglycerophosphoinositolglycan
Glycerophosphonocholines [GP16]	LC_Glycerophosphonocholine
Diacylglycerophosphonocholines [GP1601]	LC_Diacylglycerophosphonocholine
1alkyl,2acylglycerophosphonocholines[GP1602]	-

1Zalkenyl,2acylglycerophosphonocholines[GP1603]	-
Dialkylglycerophosphonocholines [GP1604]	-
Monoacylglycerophosphonocholines [GP1605]	-
Monoalkylglycerophosphonocholines [GP1606]	-
1Zalkenylglycerophosphonocholines [GP1607]	-
Glycerophosphonoethanolamines [GP17]	LC_Glycerophosphonoethanolamine
Diacylglycerophosphonoethanolamines [GP1701]	LC_Diacylglycerophosphonoethanolamine
1alkyl,2acylglycerophosphonoethanolamines [GP1702]	-
1Zalkenyl,2acylglycerophosphonoethanolamines [GP1703]	-
Dialkylglycerophosphonoethanolamines [GP1704]	-
Monoacylglycerophosphonoethanolamines [GP1705]	-
Monoalkylglycerophosphonoethanolamines [GP1706]	-
1Zalkenylglycerophosphonoethanolamines [GP1707]	-
Diglycerol tetraether phospholipids (caldarchaeols) [GP18]	LC_Di-glycerol_tetraether_phospholipid_par_ caldarchaeol _par_
Glycerolnonitol tetraether phospholipids [GP19]	LC_Glycerol-nonitol_tetraether_phospholipid
Oxidized glycerophospholipids [GP20]	-
Other Glycerophospholipids [GP00]	-
Sphingolipids[SP]	LC_Sphingolipid
Sphingoid bases [SP01]	LC_Sphingoid_base
Sphing4enines (Sphingosines) [SP0101]	LC_Sphing-4-enine_par_Sphingosine_par_
Sphinganines [SP0102]	LC_Sphinganine
4Hydroxysphinganines(Phytosphingosines) [SP0103]	LC_num4-Hydroxysphinganine_par_Phytosphingosine_par_
Sphingoid base homologs and variants [SP0104]	LC_Sphingoid_base_homolog_structural_derivative LC_Sphingoid_base_homolog LC_Sphingoid_base_homolog_variant
Sphingoid base 1phosphates[SP0105]	LC_Sphingoid_base_1-phosphate
Lysosphingomyelins and lysoglycosphingolipids [SP0106]	LC_Lysosphingomyelin LC_Lysoglycosphingolipid
Nmethylated sphingoid bases [SP0107]	LC_N-methylated_sphingoid_base
Sphingoid base analogs [SP0108]	-
Ceramides [SP02]	LC_Ceramide
Nacylsphingosines(ceramides) [SP0201]	LC_N-acylsphingosine_par_ceramide_par_

Nacylsphinganines(dihydroceramides) [SP0202]	LC_N-acylsphinganine_par_dihydroceramide_par_
Nacyl4hydroxysphinganines(phytoceramides) [SP0203]	LC_N-acy-4-hydroxysphinganine_par_phytoceramide_par_
Acylceramides [SP0204]	LC_Acylceramide
Ceramide 1phosphates[SP0205]	LC_Ceramide_1-phosphate
Phosphosphingolipids [SP03]	LC_Phosphosphingolipid
Ceramide phosphocholines (sphingomyelins) [SP0301]	LC_Ceramide_phosphocholine_par_sphingomyelin_par_
Ceramide phosphoethanolamines [SP0302]	LC_Ceramide_phosphoethanolamine
Ceramide phosphoinositols [SP0303]	LC_Ceramide_phosphoinositol
Phosphonosphingolipids [SP04]	LC_Phosphonosphingolipid
Neutral glycosphingolipids [SP05]	LC_Neutral_glycosphingolipid
Simple Glc series [SP0501]	LC_Simple_Glc_series_par_GlcCer__LacCer__etc__par_
GalNAcβ13Galα14Galβ14Glc(Globo series) [SP0502]	LC_GalNAcb1-3Gala1-4Galb1-4Glc-_par_Globo_series_par_
GalNAcβ14Galβ14Glc(Ganglio series) [SP0503]	LC_GalNAcb1-4Galb1-4Glc-_par_Ganglio_series_par_
Galβ13GlcNAcβ13Galβ14Glc(Lacto series) [SP0504]	LC_Galb1-3GlcNAcb1-3Galb1-4Glc-_par_Lacto_series_par_
Galβ14GlcNAcβ13Galβ14Glc(Neolacto series) [SP0505]	LC_Galb1-4GlcNAcb1-3Galb1-4Glc-_par_Neolacto_series _par_
GalNAcβ13Galα13Galβ14Glc(Isoglobo series) [SP0506]	LC_GalNAcb1-3Gala1-3Galb1-4Glc-_par_Isoglobo_series_par_
GlcNAcβ12Manα13Manβ14Glc(Mollu series) [SP0507]	LC_GlcNAcb1-2Mana1-3Manb1-4Glc-_par_Mollu_series_par
GalNAcβ14GlcNAcβ13Manβ14Glc(Arthro series) [SP0508]	LC_GalNAcb1-4GlcNAcb1-3Manb1-4Glc-_par_Arthro_series _par_
Gal(Gala series) [SP0509]	LC_Gal-_par_Gala_series_par_
Other Neutral glycosphingolipids [SP0500]	-
Acidic glycosphingolipids [SP06]	LC_Acidic_glycosphingolipid
Gangliosides [SP0601]	LC_Ganglioside
Sulfoglycosphingolipids (sulfatides) [SP0602]	LC_Sulfoglycosphingolipid_par_sulfatide_par_
Glucuronosphingolipids [SP0603]	LC_Glucuronosphingolipid
Phosphoglycosphingolipids [SP0604]	LC_Phosphoglycosphingolipid
Other Acidic glycosphingolipids [SP0600]	-
Basic glycosphingolipids [SP07]	LC_Basic_glycosphingolipid
Amphoteric glycosphingolipids [SP08]	LC_Amphoteric_glycosphingolipid
Arsenosphingolipids [SP09]	-
Other Sphingolipids [SP00]	-
Sterol Lipids[ST]	LC_Sterol_Lipid

Sterols [ST01]	LC_Sterol
Cholesterol and derivatives [ST0101]	LC_Cholesterol_structural_derivative LC_Cholesterol LC_Cholesterol_derivative
Cholesteryl esters [ST0102]	LC_Cholesteryl_ester
Ergosterols and C24methyl derivatives [ST0103]	LC_Ergosterol_structural_derivative LC_C24-methyl_ergosterol_derivative LC_Ergosterol
Stigmasterols and C24ethyl derivatives [ST0104]	LC_Stigmasterol_structural_derivative LC_C24-ethyl_stigmasterol_derivative LC_Stigmasterol LC_Stigmasterol_derivative
C24propyl sterols and derivatives [ST0105]	LC_C24-propyl_sterol_structural_derivative LC_C24-propyl_sterol LC_C24-propyl_sterol_derivative
Gorgosterols and derivatives [ST0106]	LC_Gorgosterol_structural_derivative LC_Gorgosterol LC_Gorgosterol_derivative
Furostanols and derivatives [ST0107]	LC_Furostanol_structural_derivative LC_Furostanol LC_Furostanol_derivative
Spirostanols and derivatives [ST0108]	LC_Spirostanol_structural_derivative LC_Spirostanol LC_Spirostanol_derivative
Furospirostanols and derivatives [ST0109]	LC_Furospirostanol_structural_derivative LC_Furospirostanol LC_Furospirostanol_derivative
Cycloartanols and derivatives [ST0110]	LC_Cycloartanol_structural_derivative LC_Cycloartanol LC_Cycloartanol_derivative
Calysterols and derivatives [ST0111]	LC_Calysterol_structural_derivative LC_Calysterol LC_Calysterol_derivative
Cardanolides and derivatives [ST0112]	LC_Cardanolide_structural_derivative LC_Cardanolide LC_Cardanolide_derivative

Bufanolides and derivatives [ST0113]	LC_Bufanolide_structural_derivative LC_Bufanolide LC_Bufanolide_derivative
Brassinolides and derivatives [ST0114]	LC_Brassinolide_structural_derivative LC_Brassinolide LC_Brassinolide_derivative
Solanidines and alkaloid derivatives [ST0115]	LC_Solanidine_structural_derivative LC_Alkaloid_derivative LC_Solanidine LC_Solanidine_derivative
Steroids [ST02]	LC_Steroid
C18 steroids (estrogens) and derivatives [ST0201]	LC_C18_steroid_par_estrogen_par__structural_derivative
C19 steroids (androgens) and derivatives [ST0202]	LC_C19_steroid_par_androgen_par__structural_derivative
C21 steroids (gluco/mineralocorticoids, progestogins) and derivatives [ST0203]	LC_C21_steroid_structural_derivative
Secosteroids [ST03]	LC_Secosteroid
Vitamin D2 and derivatives [ST0301]	LC_Vitamin_D2_structural_derivative LC_Vitamin_D2 LC_Vitamin_D2_derivative
Vitamin D3 and derivatives [ST0302]	LC_Vitamin_D3_structural_derivative LC_Vitamin_D3 LC_Vitamin_D3_derivative
Vitamin D4 and derivatives [ST0303]	LC_Vitamin_D4_structural_derivative LC_Vitamin_D4 LC_Vitamin_D4_derivative
Vitamin D5 and derivatives [ST0304]	LC_Vitamin_D5_structural_derivative LC_Vitamin_D5 LC_Vitamin_D5_derivative
Vitamin D6 and derivatives [ST0305]	LC_Vitamin_D6_structural_derivative LC_Vitamin_D6 LC_Vitamin_D6_derivative
Vitamin D7 and derivatives [ST0306]	LC_Vitamin_D7_structural_derivative LC_Vitamin_D7 LC_Vitamin_D7_derivative
Bile acids and derivatives [ST04]	LC_Bile_acid_structural_derivative

C24 bile acids, alcohols, and derivatives [ST0401]	LC_C24_bile_acid_structural_derivative LC_C24_bile_acid LC_C24_bile_acid_alcohol_derivative
C26 bile acids, alcohols, and derivatives [ST0402]	LC_C26_bile_acid_structural_derivative LC_C26_bile_acid LC_C26_bile_acid_alcohol_derivative
C27 bile acids, alcohols, and derivatives [ST0403]	LC_C27_bile_acid_structural_derivative LC_C27_bile_acid LC_C27_bile_acid_alcohol_derivative
C28 bile acids, alcohols, and derivatives [ST0404]	LC_C28_bile_acid_structural_derivative LC_C28_bile_acid LC_C28_bile_acid_alcohol_derivative
C22 bile acids, alcohols, and derivatives [ST0405]	LC_C22_bile_acid_structural_derivative LC_C22_bile_acid LC_C22_bile_acid_alcohol_derivative
C23 bile acids, alcohols, and derivatives [ST0406]	LC_C23_bile_acid_structural_derivative LC_C23_bile_acid LC_C23_bile_acid_alcohol_derivative
C25 bile acids, alcohols, and derivatives [ST0407]	LC_C25_bile_acid_structural_derivative LC_C25_bile_acid LC_C25_bile_acid_alcohol_derivative
C29 bile acids, alcohols, and derivatives [ST0408]	LC_C29_bile_acid_structural_derivative LC_C29_bile_acid LC_C29_bile_acid_alcohol_derivative
Steroid conjugates [ST05]	LC_Steroid_conjugate
Glucuronides [ST0501]	LC_Glucuronide
Sulfates [ST0502]	LC_Sulfate
Glycine conjugates [ST0503]	LC_Glycine_conjugate
Taurine conjugates [ST0504]	LC_Taurine_conjugate
Other Sterol lipids [ST00]	-
Prenol Lipids[PR]	LC_Prenol_Lipid
Isoprenoids [PR01]	LC_Isoprenoid
C5 isoprenoids (hemiterpenes) [PR0101]	LC_C5_isoprenoid
C10 isoprenoids (monoterpenes) [PR0102]	LC_C10_isoprenoid_par_monoterpene_par_
Other monoterpenoids [PR010214]	-

C15 isoprenoids (sesquiterpenes) [PR0103]	LC_C15_isoprenoid_par_sesquiterpene_par_
Other sesquiterpenoids [PR010389]	-
C20 isoprenoids (diterpenes) [PR0104]	LC_C20_isoprenoid_par_diterpene_par_
Other diterpenoids [PR010455]	-
C25 isoprenoids (sesterterpenes) [PR0105]	LC_C25_isoprenoid_par_sesterterpene_par_
Other sesterterpenoids [PR010507]	-
C30 isoprenoids (triterpenes) [PR0106]	LC_C30_isoprenoid_par_triterpene_par_
Other triterpenoids [PR010625]	-
C40 isoprenoids (tetraterpenes) [PR0107]	LC_C40_isoprenoid_par_tetraterpene_par_
Polyterpenes [PR0108]	LC_Polyterpene
Retinoids [PR0109]	LC_Retinoid
Quinones and hydroquinones [PR02]	LC_Quinone_par_inclusive_of_hydroquinone_par_
Ubiquinones [PR0201]	LC_Ubiquinone
Vitamin E [PR0202]	LC_Vitamin_E
Vitamin K [PR0203]	LC_Vitamin_K
Polyprenols [PR03]	LC_Polyprenol
Bactoprenols [PR0301]	LC_Bactoprenol
Bactoprenol monophosphates [PR0302]	LC_Bactoprenol_monophosphate
Bactoprenol diphosphates [PR0303]	LC_Bactoprenol_diphosphate
Phytoprenols [PR0304]	LC_Phytoprenol
Phytoprenol monophosphates [PR0305]	LC_Phytoprenol_monosphophate
Phytoprenol diphosphates [PR0306]	LC_Phytoprenol_diphosphate
Dolichols [PR0307]	LC_Dolichol
Dolichol monophosphates [PR0308]	LC_Dolichol_monophosphate
Dolichol diphosphates [PR0309]	LC_Dolichols_diphosphate
Hopanoids [PR04]	LC_Hopanoid
Other Prenol lipids [PR00]	-
Saccharolipids [SL]	LC_Saccharolipid
Acylaminosugars [SL01]	LC_Acylaminosugar
Monoacylaminosugars [SL0101]	LC_Monoacylaminosugar
Diacylaminosugars [SL0102]	LC_Diacylaminosugar

Triacylaminosugars [SL0103]	LC_Triacylaminosugar
Tetraacylaminosugars [SL0104]	LC_Tetraacylaminosugar
Pentaacylaminosugars [SL0105]	LC_Pentaacylaminosugar
Hexaacylaminosugars [SL0106]	LC_Hexaacylaminosugar
Heptaacylaminosugars [SL0107]	LC_Heptaacylaminosugar
Acylaminosugar glycans [SL02]	LC_Acylaminosugar_glycan
Acyltrehaloses [SL03]	LC_Acyltrehalose
Acyltrehalose glycans [SL04]	LC_Acyltrehalose_glycan
Other acyl sugars [SL05]	-
Other Saccharolipids [SL00]	-
Polyketides[PK]	LC_Polyketide
Linear polyketides [PK01]	LC_Linear_polyketide
Halogenated acetogenins [PK02]	LC_Halogenated_acetogenin
Annonaceae acetogenins [PK03]	LC_Annonaceae_acetogenin
Macrolides and lactone polyketides [PK04]	LC_Macrolides_and_lactone_polyketide
Ansamycins and related polyketides [PK05]	LC_Ansamycin_related_polyketide
Polyenes [PK06]	LC_Polyene
Linear tetracyclines [PK07]	LC_Linear_tetracycline
Angucyclines [PK08]	LC_Angucycline
Polyether antibiotics [PK09]	LC_Polyether_antibiotic
Aflatoxins and related substances [PK10]	LC_Aflatoxin_related_substance
Cytochalasins [PK11]	LC_Cytochalasin
Flavonoids [PK12]	LC_Flavonoid
Anthocyanidins [PK1201]	LC_Anthocyanidins
Flavans, Flavanols and Leucoanthocyanidins [PK1202]	LC_Flavans_Flavanols_and_Leucoanthocyanidins
Proanthocyanidins [PK1203]	LC_Proanthocyanidins
Biflavonoids and polyflavonoids [PK1204]	LC_Biflavonoids_and_polyflavonoids
Isoflavonoids [PK1205]	LC_Isoflavonoids
Rotenoid flavonoids [PK1206]	LC_Rotenoid_flavonoids
Pterocarpans [PK1207]	LC_Pterocarpans
Isoflavans [PK1208]	LC_Isoflavans

Coumestan flavonoids [PK1209]	LC_Coumestan_flavonoids
Neoflavonoids [PK1210]	LC_Neoflavonoids
Flavones and Flavonols [PK1211]	LC_Flavones_and_Flavonols
Chalcones and dihydrochalcones [PK1212]	LC_Chalcones_and_dihydrochalcones
Aurone flavonoids [PK1213]	LC_Aurone_flavonoids
Flavanones [PK1214]	LC_Flavanones
Dihydroflavonols [PK1215]	LC_Dihydroflavonols
Other Flavonoids [PK1216]	-
Aromatic polyketides [PK13]	LC_Aromatic_polyketide
Monocyclic aromatic polyketides [PK1301]	LC_Monocyclic_aromatic_polyketides
Naphthalenes and naphthoquinones [PK1302]	LC_Naphthalenes_and_naphthoquinones
Benzoisochromanquinones [PK1303]	LC_Benzoisochromanquinones
Anthracenes and phenanthrenes [PK1304]	LC_Anthracenes_and_phenanthrenes
Anthracyclinones [PK1305]	LC_Anthracyclinones
Dibenzofurans, griseofulvins, dibenzopyrans and xanthones [PK1306]	LC_Dibenzofurans_griseofulvins_dibenzopyrans_and_xanthones
Diphenylmethanes, acylphloroglucinols and benzophenones [PK1307]	LC_Diphenylmethanes_acylphloroglucinols_and_benzophenones
Depsides and depsidones [PK1308]	LC_Depsides_and_depsidones
Diphenyl ethers, biphenyls, dibenzyls and stilbenes [PK1309]	LC_Diphenyl_ethers_s_biphenyls_dibenzyls_and_stilbenes
Benzofuranoids [PK1310]	LC_Benzofuranoids
Benzopyranoids [PK1311]	LC_Benzopyranoids
Other aromatic polyketides [PK1312]	-
Nonribosomal peptide/polyketide hybrids [PK14]	LC_Non-ribosomal_peptide_polyketide_hybrid
Other Polyketides [PK00]	-

Appendix C: DL difinitions of lipids

Legends:

Bold – Lipid concepts where cardinality axioms have been applied.

Manchester OWL Syntax:

Boolean class constructors symbols-

⊓ = And

⊔ = Or

Concepts	DL Definition
Lipid	NA
LC_Fatty_Acyl	Lipid (hasPart some Carboxylic_Acid_derivative_Group ⊓ hasPart some Acyl_Chain) ⊔ hasPart some Alkyl_Chain;
LC_Docosanoid	LC_Fatty_Acyl (hasPart some Carboxylic_Acid) ⊓ (hasPart some Alcohol) ⊓ (hasPart some Alkenyl_Group) ⊓ (hasPart some Cyclopentenone) ⊓ (hasAcyl_Chain exactly 1); hasPart only (Carboxylic_Acid or Alcohol or Alkenyl_Group or Cyclopentenone or Acyl_Chain);
LC_Eicosanoid	LC_Fatty_Acyl (hasPart some Carboxylic_Acid) ⊓ (hasPart some Alkenyl_Group) ⊓

	(hasAcyl_Chain exactly 1);
LC_Clavulone_structural_derivative	LC_Eicosanoid hasPart some (Cyclopentenone or Halogenated_Hydroxy_ Cyclopentenone) ⊓ hasPart some (Carboxylic_Acid_Ester or Lactone_Group); hasPart only (Carboxylic_Acid_Ester or Lactone_Group or Cyclopentenone or Halogenated_Hydroxy_Cyclopentenone or Alkenyl_Group or Carboxylic_Acid or Acyl_Chain)
LC_Clavulone	-
LC_Clavulone_derivative	-
LC_Eicosapentaenoic_acid_derivative	LC_Eicosanoid hasPart some (Hydroperoxide or Alcohol or Peroxide) ⊓ hasPart exactly 5 Alkenyl_Group; hasPart only (Hydroperoxide or Alkenyl_Group or Alcohol or Carboxylic_Acid or Peroxide or Acyl_Chain)
LC_Hydroperoxyeicosapentaenoic_acid	-
LC_Hydroxyeicosapentaenoic_acid	-
LC_Eicosatetraenoic_acid_derivative	LC_Eicosanoid hasPart some (Hydroperoxide or Alcohol or Peroxide) ⊓ hasPart exactly 4 Alkenyl_Group; hasPart only (Hydroperoxide or Alkenyl_Group or Alcohol or Carboxylic_Acid or Peroxide or Acyl_Chain);
LC_Hydroperoxyeicosatetraenoic_acid	-
LC_Hydroxyeicosatetraenoic_acid	-
LC_Eicosatrienoic_acid_derivative	LC_Eicosanoid

	hasPart some (Hydroperoxide or Alcohol or Peroxide) ⊓ hasPart exactly 3 Alkenyl_Group; hasPart only (Hydroperoxide or Alkenyl_Group or Alcohol or Carboxylic_Acid or Peroxide or Acyl_Chain);
LC_Hydroperoxyeicosatrienoic_acid	-
LC_Hydroxyeicosatrienoic_acid	-
LC_Epoxyeicosatrienoic_acid	LC_Eicosanoid (hasPart some Epoxy) ⊓ (hasPart exactly 3 Alkenyl_Group) ⊓ (hasPart some Alcohol); hasPart only (Alkenyl_Group or Epoxy or Carboxylic_Acid or Acyl_Chain);
LC_Hepoxilin	LC_Eicosanoid (hasPart exactly 1 Epoxy) ⊓ (hasPart exactly 1 Alcohol); hasPart only (Epoxy or Alcohol or Alkenyl_Group or Carboxylic_Acid or Acyl_Chain);
LC_Isoprostane	LC_Eicosanoid hasPart some Alcohol ⊓ hasPart some (Hydroxy_Cyclopentanone or Cyclopentenone or Di-hydroxy_Cyclopentane) ⊓ hasPart some Ketone; hasPart only (Acyl_Chain or Alcohol or Hydroxy_Cyclopentanone or Cyclopentenone or Di-hydroxy_Cyclopentane or Ketone or Alkenyl_Group or Carboxylic_Acid);
LC_Leukotriene	LC_Eicosanoid

	hasPart some Alcohol; hasPart only (Alcohol or Alkenyl_Group or Carboxylic_Acid or Acyl_Chain);
LC_Levuglandin	LC_Eicosanoid hasPart only (Alcohol or Ketone or Aldehyde or Carboxylic_Acid or Acyl_Chain); (hasPart some Aldehyde) ⊓ (hasPart some Ketone) ⊓ (hasPart some Alcohol);
LC_Lipoxin	LC_Eicosanoid hasPart exactly 3 Alcohol hasPart only (Alkenyl_Group or Alcohol or Carboxylic_Acid or Acyl_Chain);
LC_Prostaglandin	LC_Eicosanoid hasPart some (Bicyclic_5_membered_Heterocyclic_Group or Di-hydroxy_Cyclopentane or num5_membered_Cycloketone_Group) ⊓ hasPart some Alkenyl_Group ⊓ hasPart some Alcohol ⊓ hasPart some Ketone; hasPart only (Acyl_Chain or Alkenyl_Group or Ketone or Bicyclic_5_membered_Heterocyclic_Group or Di-hydroxy_ Cyclopentane or Alcohol or num5_membered_Cycloketone_Group);
LC_Thromboxane	LC_Eicosanoid hasPart some Alcohol ⊓ hasPart some (Tetrahydropyran_Group or Bicyclic_Tetrahydropyran-Oxetane_Group);

	hasPart only (Alcohol or Alkenyl_Group or Tetrahydropyran_Group or Carboxylic_Acid or Bicyclic_Tetrahydropyran-Oxetane_Group);
LC_Fatty_acid	LC_Fatty_Acyl hasPart some Carboxylic_Acid ⊓ hasAcyl_Chain exactly 1;
LC_Amino_fatty_acid	LC_Fatty_acid hasPart some Amine_Group; hasPart only (Amine_Group or Carboxylic_Acid or Acyl_Chain);
LC_Carbocyclic_fatty_acid	LC_Fatty_acid hasPart some (Cycloalkane_Group or Cycloalkene_Group); hasPart only (Cycloalkane_Group or Cycloalkene_Group or Carboxylic_Acid or Acyl_Chain);
LC_Cyano_fatty_acid	LC_Fatty_acid hasPart some Nitrile_Group; hasPart only (Nitrile_Group or Carboxylic_Acid or Acyl_Chain);
LC_Dicarboxylic_acid	LC_Fatty_acid hasPart exactly 2 Carboxylic_Acid; hasPart only (Carboxylic_Acid or Acyl_Chain);
LC_Epoxy_fatty_acid	LC_Fatty_acid hasPart some Epoxy;

	hasPart only (Epoxy or Carboxylic_Acid or Acyl_Chain);
LC_Halogenated_fatty_acid	LC_Fatty_acid hasPart some Alkyl_Halide_Group; hasPart only (Alkyl_Halide_Group or Carboxylic_Acid or Acyl_Chain);
LC_Heterocyclic_fatty_acid	LC_Fatty_acid hasPart some num5_membered_Heterocyclic_Group; hasPart only (num5_membered_Heterocyclic_Group or Carboxylic_Acid or Acyl_Chain);
LC_Hydroperoxy_fatty_acid	LC_Fatty_acid hasPart some Hydroperoxide; hasPart only (Hydroperoxide or Carboxylic_Acid or Acyl_Chain);
LC_Hydroxy_fatty_acid	LC_Fatty_acid hasPart some Alcohol; hasPart only (Alcohol or Carboxylic_Acid or Acyl_Chain);
LC_Methoxy_fatty_acid	LC_Fatty_acid hasPart some Methoxy_Group; hasPart only (Methoxy_Group or Carboxylic_Acid or Acyl_Chain);
LC_Methyl_branched_fatty_acid*	LC_Fatty_acid hasPart some Methyl_Derivative;

	hasPart only (Methyl_Derivative or Carboxylic_Acid or Acyl_Chain);
LC_Mycolic_acid	LC_Fatty_acid hasPart some Alpha-Hydroxy_Acid_Group ⊓ hasMeromycolic_ Chain exactly 1;
LC_General_alpha_mycolic_acid	LC_Mycolic_acid hasPart some (Cyclopropane or Alkenyl_Group);
LC_Alpha_mycolic_acid	LC_General_alpha_mycolic_acid hasPart some Cyclopropane; hasPart only (Alkenyl_Group or Alpha-Hydroxy_Acid_Group or Cyclopropane or Alcohol or Carboxylic_Acid or Meromycolic_Chain);
LC_Oxygenated_alpha_mycolic_acid	LC_General_alpha_mycolic_acid hasPart some (Ketone or Methoxy_Group);
LC_General_Methoxy_mycolic_acid	LC_Oxygenated_alpha_mycolic_acid hasPart some Methoxy_Group;
LC_Methoxy_mycolic_acid	LC_General_Methoxy_mycolic_acid hasPart some non-terminal_Methoxy; hasPart only (non-terminal_Methoxy or Alpha-Hydroxy_Acid_ Group or Carboxylic_Acid or Alcohol or Meromycolic_Chain or Cyclopropane or Alkenyl_Group);

LC_Omega-1_methoxy_mycolic_acid	LC_General_Methoxy_mycolic_acid hasPart some Terminal_Methoxy; hasPart only (Terminal_Methoxy or Alpha-Hydroxy_Acid_Group or Carboxylic_Acid or Alcohol or Meromycolic_Chain or Cyclopropane or Alkenyl_Group);
LC_Keto_mycolic_acid	LC_Oxygenated_alpha_mycolic_acid hasPart some Ketone; hasPart only (Ketone or Alpha-Hydroxy_Acid_Group or Alcohol or Carboxylic_Acid or Meromycolic_Chain or Cyclopropane or Alkenyl_Group);
LC_Wax_ester_mycolic_acid	LC_General_alpha_mycolic_acid hasPart some Carboxylic_Acid_Ester_Group; hasPart only (Alkenyl_Group or Alpha-Hydroxy_Acid_Group or Meromycolic_Chain or Cyclopropane or Carboxylic_Acid_ Ester_Group or Methyl or Alcohol or Carboxylic_Acid);
LC_General_methylated_mycolic_acid	LC_Mycolic_acid hasPart some Alkenyl_Group ⊓ hasPart some Methyl;
LC_Alpha_2_mycolic_acid	LC_General_methylated_mycolic_acid hasPart only (Methyl or Alpha-Hydroxy_Acid_Group or Meromycolic_Chain or Alkenyl_Group or Alcohol or Carboxylic_Acid);
LC_Epoxy_mycolic_acid	LC_General_methylated_mycolic_acid hasPart some Epoxy;

	hasPart only (Methyl or Alpha-Hydroxy_Acid_Group or Alkenyl_ Group or Meromycolic_Chain or Epoxy or Carboxylic_Acid or Alcohol);
LC_General_mycolic_acid	LC_Mycolic_acid hasPart some Alkenyl_Group;
LC_Alpha_1_mycolic_acid	LC_General_mycolic_acid hasPart only (Alkenyl_Group or Alpha-Hydroxy_Acid_Group or Meromycolic_Chain or Alcohol or Carboxylic_Acid);
LC_Alpha_prime_mycolic_acids	LC_Alpha_1_mycolic_acid hasPart some proximal_Alkenyl; hasPart only (Alkenyl_Group or Alpha-Hydroxy_Acid_Group or Meromycolic_Chain or Alcohol or Carboxylic_Acid);
LC_Nitro_fatty_acid	LC_Fatty_acid hasPart some Nitro_Group; hasPart only (Nitro_Group or Carboxylic_Acid or Acyl_Chain);
LC_Oxo_fatty_acid	LC_Fatty_acid hasPart some Ketone; hasPart only (Ketone or Carboxylic_Acid or Acyl_Chain);
LC_Straight_chain_fatty_acid	LC_Fatty_acid hasPart exactly 1 Carboxylic_Acid; hasPart only (Carboxylic_Acid or Acyl_Chain);

LC_Thia_fatty_acid	LC_Fatty_acid hasPart some Thiol_Group; hasPart only (Thiol_Group or Carboxylic_Acid or Acyl_Chain);
LC_Unsaturated_fatty_acid	LC_Fatty_acid hasPart some Alkenyl_Group; hasPart only (Alkenyl_Group or Carboxylic_Acid or Acyl_Chain);
LC_Fatty_acyl_derivative	LC_Fatty_acid hasPart some Alkyl_Chain;
LC_Fatty_alcohol	LC_Fatty_acyl_derivative hasPart some Alcohol ⊓ hasAlkyl_Chain exactly 1; hasPart only (Alcohol or Alkyl_Chain);
LC_Fatty_ether	LC_Fatty_acyl_derivative hasPart some Ether ⊓ hasAlkyl_Chain exactly 1; hasPart only (Ether or Alkyl_Chain);
LC_Fatty_nitrile	LC_Fatty_acyl_derivative hasPart some Nitrile_Group ⊓ hasAlkyl_Chain exactly 1; hasPart only (Nitrile_Group or Alkyl_Chain);
LC_Hydrocarbon	LC_Fatty_acyl_derivative

	hasAlkyl_Chain exactly 1; hasPart only Alkyl_Chain;
LC_Fatty_aldehyde	LC_Fatty_Acyl hasPart some Aldehyde ⊓ hasAcyl_Chain exactly 1; hasPart only (Aldehyde or Acyl_Chain);
LC_Fatty_amide	LC_Fatty_Acyl hasPart some Carboxylic_Acid_Amide_Group ⊓ hasAcyl_Chain exactly 1;
LC_Fatty_acyl_homoserine_lactone	LC_Fatty_amide hasPart some Carboxylic_Acid_Homoserine_Lactone; hasPart only (Carboxylic_Acid_Homoserine_Lactone or Acyl_ Chain);
LC_N-acyl_amine	LC_Fatty_amide hasPart some Carboxylic_Acid_Secondary_Amide; hasPart only (Carboxylic_Acid_Secondary_Amide or Acyl_Chain);
LC_N-acyl_ethanolamine_par_endocannabinoid_par_	LC_Fatty_amide hasPart some Carboxylic_Acid_Ethanolamine; hasPart only (Carboxylic_Acid_Ethanolamine or Acyl_Chain);
LC_Primary_amide	LC_Fatty_amide

	hasPart some Carboxylic_Acid_Primary_Amide; hasPart only (Carboxylic_Acid_Primary_Amide or Acyl_Chain);
LC_Fatty_ester	LC_Fatty_Acyl hasPart some Carboxylic_Acid_Ester_Group ⊓ hasAcyl_Chain exactly 1;
LC_Cyano_ester	LC_Fatty_ester hasPart some Carboxylic_Acid_Ester ⊓ hasPart some Nitrile_Group; hasPart only (Nitrile_Group or Carboxylic_Acid_Ester or Acyl_Chain);
LC_Fatty_acyl_ACP	LC_Fatty_ester hasPart some Carboxylic_Acid-ACP; hasPart only (Carboxylic_Acid-ACP or Acyl_Chain);
LC_Fatty_acyl_adenylate	LC_Fatty_ester hasPart some Carboxylic_Acid_Adenosine_ Monophosphate; hasPart only (Carboxylic_Acid-Adenosine_Monophosphate or Acyl_Chain);
LC_Fatty_acyl_carnitine	LC_Fatty_ester hasPart some Carboxylic_Acid-Carnitine; hasPart only (Carboxylic_Acid-Carnitine or Acyl_Chain);

LC_Fatty_acyl_CoA	LC_Fatty_ester hasPart some Carboxylic_Acid-CoA; hasPart only (Carboxylic_Acid-CoA or Acyl_Chain);
LC_Lactone	LC_Fatty_ester hasPart some Lactone_Group; hasPart only (Lactone_Group or Acyl_Chain);
LC_Wax_diester	LC_Fatty_ester hasPart exactly 2 Carboxylic_Acid_Ester; hasPart only (Carboxylic_Acid_Ester or Acyl_Chain);
LC_Wax_monoester	LC_Fatty_ester hasPart exactly 1 Carboxylic_Acid_Ester; hasPart only (Carboxylic_Acid_Ester or Acyl_Chain);
LC_Octadecanoid	LC_Fatty_Acyl hasPart some Carboxylic_Acid ⊓ hasAcyl_Chain exactly 1 ⊓ hasPart some Alkenyl_Group;
LC_Jasmonic_acid	LC_Octadecanoid hasPart some Cyclopentanone; hasPart only (Alkenyl_Group or Cyclopentanone or Carboxylic_Acid or Acyl_Chain);

LC_num12-oxophytodienoic_acid_metabolite	LC_Octadecanoid hasPart some Cyclopentenone; hasPart only (Alkenyl_Group or Cyclopentenone or Carboxylic_Acid or Acyl_Chain);
LC_Oxygenated_hydrocarbon	LC_Fatty_Acyl hasPart some (Terminal_Acyl_Group or Ketone) ⊓ hasAcyl_Chain exactly 1; hasPart only (Terminal_Acyl_Group or Ketone or Acyl_Chain);
LC_Glycerolipid	Lipid hasPart some (Carboxylic_Acid_Ester or Ether) ⊓ hasPart some Carbon_Chain_Group;
LC_Diradylglycerol	LC_Glycerolipid hasPart some Glycerol ⊓ hasCarbon_Chain exactly 2;
LC_Alkylacylglycerol	LC_Diradylglycerol hasAcyl_Chain exactly 1 ⊓ hasAlkyl_Ether_Chain exactly 1;
LC_1-acyl_2-alkylglycerol	-
LC_1-alkyl_2-acylglycerol	-
LC_Di-glycerol_tetraether	LC_Diradylglycerol hasAlkyl_Ether_Chain exactly 2 ⊓ hasEther_Group exactly 2 ⊓ hasGlycerol_Group exactly 2; hasPart only (Glycerol or Ether or Alkyl_Ether_Chain);

LC_Di-glycerol_tetraether_glycan	LC_Diradylglycerol hasAlkyl_Ether_Chain exactly 2 ⊓ hasPart some Glycerolglycan ⊓ hasEther_Group exactly 2 ⊓ hasGlycerol_Group exactly 2; hasPart only (Glycerol or Ether or Glycerolglycan or Alkyl_Ether_ Chain);
LC_Diacylglycerol	LC_Diradylglycerol hasAcyl_Chain exactly 2; hasPart only (Glycerol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Dialkylglycerol	LC_Diradylglycerol hasAlkyl_Ether_Chain exactly 2 ⊓ hasGlycerol_Group exactly 1 hasPart only (Glycerol or Alkyl_Ether_Chain or Ether);
LC_num1Z-alkenylacylglycerol	LC_Diradylglycerol hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Vinyl_Ether_Chain or Acyl_Chain or Carboxylic_Acid_Ester or Ether or Glycerol);
LC_Glycosyldiradylglycerol	LC_Glycerolipid hasPart some Glycerolglycan ⊓ hasCarbon_Chain exactly 2;
LC_Glycosylalkylacylglycerol	LC_Glycosyldiradylglycerol hasAcyl_Chain exactly 1 ⊓ hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerolglycan or Acyl_Chain or

	Carboxylic_Acid_Ester or Ether or Alkyl_Ether_Chain);
LC_Glycosyldiacylglycerol	LC_Glycosyldiradylglycerol hasAcyl_Chain exactly 2; hasPart only (Glycerolglycan or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycosyldialkylglycerol	LC_Glycosyldiradylglycerol hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerolglycan or Alkyl_Ether_Chain or Ether);
LC_Glycosylmonoradylglycerol	LC_Glycerolipid hasPart some Glycerolglycan ⊓ hasCarbon_Chain exactly 1;
LC_Glycosylmonoacylglycerol	LC_Glycosylmonoradylglycerol hasAcyl_Chain exactly 1; hasPart only (Glycerolglycan or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycosylmonoalkylglycerol	LC_Glycosylmonoradylglycerol hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerolglycan or Alkyl_Ether_Chain or Ether);
LC_Monoradylglycerol	LC_Glycerolipid hasCarbon_Chain exactly 1 ⊓ hasPart some Glycerol;

LC_Mono-par-1Z-alkenyl-par-glycerol	LC_Monoradylglycerol hasVinyl_Ether_Chain exactly 1; hasPart only (Glycerol or Vinyl_Ether_Chain or Ether);
LC_Monoacylglycerol	LC_Monoradylglycerol hasAcyl_Chain exactly 1; hasPart only (Glycerol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Monoalkylglycerol	LC_Monoradylglycerol hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerol or Alkyl_Ether_Chain or Ether);
LC_Triradylglycerol	LC_Glycerolipid hasCarbon_Chain exactly 3 ⊓ hasPart some Glycerol;
LC_Alkyldiacylglycerols	LC_Triradylglycerol hasAcyl_Chain exactly 2 ⊓ hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerol or Carboxylic_Acid_Ester or Ether or Acyl_Chain or Alkyl_Ether_Chain);
LC_Dialkylmonoacylglycerols	LC_Triradylglycerol hasAcyl_Chain exactly 1 ⊓ hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerol or Carboxylic_Acid_Ester or Ether or Acyl_Chain or Alkyl_Ether_Chain);

LC_Estolides	LC_Triradylglycerol hasAcyl_Chain exactly 2 ⊓ hasAcyl_Estolide_Chain exactly 1; hasPart only (Acyl_Chain or Acyl_Estolide_Chain or Carboxylic_Acid_Ester or Glycerol);
LC_num1Z-alkenyldiacylglycerols	LC_Triradylglycerol hasAcyl_Chain exactly 2 ⊓ hasVinyl_Ether_Chain exactly 1; hasPart only (Vinyl_Ether_Chain or Acyl_Chain or Carboxylic_ Acid_Ester or Ether or Glycerol);
LC_Triacylglycerols	LC_Triradylglycerol hasAcyl_Chain exactly 3; hasPart only (Glycerol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycerophospholipid	Lipid hasPart some (Carboxylic_Acid_Ester or Ether) ⊓ hasPart some Glycerophosphate_Group ⊓ hasPart some Carbon_Chain_Group;
LC_CDP-Glycerol	LC_Glycerophospholipid hasPart some Cytidine_Diphosphate_Glycerol ⊓ hasPart some Acyl_Chain;
LC_CDP-1-alkyl_glycerol	LC_CDP-Glycerol hasAlkyl_Ether_Chain exactly 1; hasPart only (Cytidine_Diphosphate_Glycerol or Alkyl_Ether_Chain or Ether);

LC_CDP-1Z-alkenylglycerol	LC_CDP-Glycerol hasVinyl_Ether_Chain exactly 1; hasPart only (Cytidine_Diphosphate_Glycerol or Vinyl_Ether_Chain or Ether);
LC_CDP-Alkenylacylglycerol	LC_CDP-Glycerol hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Vinyl_Ether_Chain or Acyl_Chain or Carboxylic_ Acid_Ester or Ether or Cytidine_Diphosphate_Glycerol);
LC_CDP-1Z-alkenyl_2-acylglycerol	-
LC_CDP-Alkylacylglycerol	LC_CDP-Glycerol hasAcyl_Chain exactly 1 ⊓ hasAlkyl_Ether_Chain exactly 1; hasPart only (Cytidine_Diphosphate_Glycerol or Carboxylic_Acid _Ester or Ether or Acyl_Chain or Alkyl_Ether_ Chain);
LC_CDP-1-alkyl_2-acylglycerol	-
LC_CDP-diacylglycerol	LC_CDP-Glycerol hasAcyl_Chain exactly 2; hasPart only (Cytidine_Diphosphate_Glycerol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_CDP-dialkylglycerol	LC_CDP-Glycerol hasAlkyl_Ether_Chain exactly 2; hasPart only (Cytidine_Diphosphate_Glycerol or Alkyl_Ether_Chain or Ether);

LC_CDP-monoacylglycerol	LC_CDP-Glycerol hasAcyl_Chain exactly 1; hasPart only (Cytidine_Diphosphate_Glycerol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Di-glycerol_tetraether_phospholipid_par_caldarchaeol_par_	LC_Glycerophospholipid hasPart some Glycerophosphatidylethanolamine ⊓ hasGlycerol_Group exactly 2 ⊓ hasEther_Group exactly 4 ⊓hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerophosphatidylethanolamine or Alkyl_Ether_Chain or Glycerol or Ether);
LC_Glycerol-nonitol_tetraether_phospholipid	LC_Glycerophospholipid hasPart some Glycerophosphatidylethanolamine ⊓ hasPart some Glycerol-nonitol ⊓ hasAlkyl_Ether_Chain exactly 2 ⊓ hasGlycerol_Group exactly 2 ⊓ hasEther_Group exactly 4; hasPart only (Glycerol-nonitol or Glycerophosphatidylethanolamine or Alkyl_Ether_Chain or Ether);
LC_Glycerophosphate	LC_Glycerophospholipid hasPart some Glycerophosphatidic_acid;
LC_Alkenylacylglycerophosphate	LC_Glycerophosphate hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Vinyl_Ether_Chain or Acyl_Chain or Carboxylic_Acid_Ester or Ether or Glycerophosphatidic_acid);

LC_num1Z-alkenyl_2-acylglycerophosphate	-
LC_Alkylacylglycerophosphate	LC_Glycerophosphate hasAcyl_Chain exactly 1 ⊓ hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidic_acid or Carboxylic_Acid_Ester or Ether or Acyl_Chain or Alkyl_Ether_Chain);
LC_num1-alkyl_2-acylglycerophosphate	-
LC_Diacylglycerophosphate	LC_Glycerophosphate hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidic_acid or Carboxylic_Acid_Ester or Acyl_Chain);
LC_Dialkylglycerophosphate	LC_Glycerophosphate hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerophosphatidic_acid or Alkyl_Ether_Chain or Ether);
LC_Monoacylglycerophosphate	LC_Glycerophosphate hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidic_acid or Carboxylic_Acid_Ester or Acyl_Chain);
LC_num1-alkyl_glycerophosphate	LC_Glycerophosphate hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidic_acid or Alkyl_Ether_Chain or Ether);

LC_num1Z-alkenylglycerophosphate	LC_Glycerophosphate hasVinyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidic_acid or Vinyl_Ether_Chain or Ether);
LC_Glycerophosphocholine	LC_Glycerophospholipid hasPart some Glycerophosphatidylcholine;
LC_Alkenylacylglycerophosphocholine	LC_Glycerophosphocholine hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylcholine or Acyl_Chain or Carboxylic_Acid_Ester or Vinyl_Ether_Chain or Ether);
LC_num1Z-alkenyl_2-acylglycerophosphocholine	-
LC_Alkylacylglycerophosphocholine	LC_Glycerophosphocholine hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylcholine or Acyl_Chain or Carboxylic_Acid_Ester or Alkyl_Ether_Chain or Ether);
LC_num1-acyl_2-alkylglycerophosphocholine	-
LC_num1-alkyl_2-acylglycerophosphocholine	-
LC_Diacylglycerophosphocholine	LC_Glycerophosphocholine hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidylcholine or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphocholine	LC_Glycerophosphocholine

	hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerophosphatidylcholine or Alkyl_Ether_Chain or Ether);
LC_Monoacylglycerophosphocholine	LC_Glycerophosphocholine hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylcholine or Acyl_Chain or Carboxylic_Acid_Ester);
LC_num1-alkyl_glycerophosphocholine	-
LC_num1Z-alkenylglycerophosphocholine	-
LC_Glycerophosphoethanolamine	LC_Glycerophospholipid hasPart some Glycerophosphatidylethanolamine;
LC_Alkenylacylglycerophosphoethanolamine	LC_Glycerophosphoethanolamine hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Ether or Carboxylic_Acid_Ester or Glycerophosphatidylethanolamine or Acyl_Chain or Vinyl_Ether_Chain);
LC_Alkylacylglycerophosphoethanolamine	LC_Glycerophosphoethanolamine hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylethanolamine or Ether or Carboxylic_Acid_Ester or Acyl_Chain or Alkyl_Ether_Chain);
LC_Diacylglycerophosphoethanolamine	LC_Glycerophosphoethanolamine

	hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidylethanolamine or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphoethanolamine	LC_Glycerophosphoethanolamine hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerophosphatidylethanolamine or Alkyl_Ether_Chain or Ether);
LC_Monoacylglycerophosphoethanolamine	LC_Glycerophosphoethanolamine hasAcyl_Chain exactly 1; hasPart some (Glycerophosphatidylethanolamine or Acyl_Chain or Carboxylic_Acid_Ester);
LC_num1-alkyl_glycerophoethanolamine	LC_Glycerophosphoethanolamine hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidylethanolamine or Alkyl_Ether_Chain or Ether);
LC_num1Z-alkenylglycerophosphoethanolamine	LC_Glycerophosphoethanolamine hasVinyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidylethanolamine or Vinyl_Ether_Chain or Ether);
LC_Glycerophosphoglycerol	LC_Glycerophospholipid hasPart some Glycerophosphotidylglycerol;
LC_Alkenylacylglycerophosphoglycerol	LC_Glycerophosphoglycerol

	hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphotidylglycerol or Acyl_Chain or Carboxylic_Acid_Ester or Vinyl_Ether_Chain or Ether);
LC_num1Z-alkenyl_2-acylglycerophosphoglycerol	-
LC_Alkylacylglycerophosphoglycerol	LC_Glycerophosphoglycerol hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1;
LC_num1-alkyl_2-acylglycerophosphoglycerol	-
LC_Diacylglycerophosphodiradylglycerol	LC_Glycerophosphoglycerol hasAcyl_Chain exactly 2 ⊓ hasCarbon_Chain exactly 4; hasPart only (Carbon_Chain_Group or Glycerophosphotidylglycerol or Carboxylic_Acid_Ester);
LC_Diacylglycerophosphoglycerol	LC_Glycerophosphoglycerol hasAcyl_Chain exactly 2; hasPart only (Acyl_Chain or Glycerophosphotidylglycerol or Carboxylic_Acid_Ester);
LC_Diacylglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerol hasAcyl_Chain exactly 2 ⊓ hasCarbon_Chain exactly 3; hasPart only (Carbon_Chain_Group or Glycerophosphotidylglycerol or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphoglycerol	LC_Glycerophosphoglycerol hasAlkyl_Ether_Chain exactly 2; hasPart only (Alkyl_Ether_Chain or Glycerophosphotidylglycerol or

	Ether);
LC_Monoacylglycerophosphoglycerol	LC_Glycerophosphoglycerol hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Acyl_Chain or Glycerophosphotidylglycerol or Carboxylic_Acid_Ester);
LC_Monoacylglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerol hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerol or Carboxylic_Acid_Ester or Carbon_Chain_Group);
LC_num1-alkyl_glycerophosphoglycerol	LC_Glycerophosphoglycerol hasAlkyl_Ether_Chain exactly 1; hasPart only (Alkyl_Ether_Chain or Glycerophosphotidylglycerol or Ether);
LC_num1Z-alkenylglycerophosphoglycerol	LC_Glycerophosphoglycerol hasVinyl_Ether_Chain exactly 1; hasPart only (Vinyl_Ether_Chain or Glycerophosphotidylglycerol or Ether);
LC_Glycerophosphoglycerophosphate	LC_Glycerophospholipid hasPart some Glycerophosphotidylglycerophosphate;
LC_Alkenylacylglycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1;

	hasPart only (Glycerophosphotidylglycerophosphate or Acyl_Chain or Carboxylic_Acid_Ester or Vinyl_Ether_Chain or Ether);
LC_num1Z-alkenyl_2-acylglycerophosphoglycerophosphate	-
LC_Alkylacylglycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphate or Acyl_Chain or Carboxylic_Acid_Ester or Alkyl_Ether_Chain or Ether);
LC_num1-alkyl_2-acylglycerophosphoglycerophosphate	-
LC_Diacylglycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate hasAcyl_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphate or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphate or Alkyl_Ether_Chain or Ether);
LC_Monoacylglycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphate or Acyl_Chain or Carboxylic_Acid_Ester);
LC_num1-alkyl_glycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate

	hasAlkyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphate or Alkyl_Ether_Chain or Ether);
LC_num1Z-alkenylglycerophosphoglycerophosphate	LC_Glycerophosphoglycerophosphate hasVinyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphate or Vinyl_Ether_Chain or Ether);
LC_Glycerophosphoglycerophosphoglycerol	LC_Glycerophospholipid hasPart some Glycerophosphotidylglycerophosphoglycerol ⊓ hasPart some Carboxylic_Acid_derivative_Group;
LC_Alkenylacylglycerophosphoglycerophosphodiradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carboxylic_Acid_Ester or Ether or Carbon_Chain_Group);
LC_num1Z-alkenyl_2-acylglycerophosphoglycerophosphodiradylglycerol	-
LC_Alkenylacylglycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carboxylic_Acid_Ester or Ether or Carbon_Chain_Group);
LC_num1Z-alkenyl_2-acylglycerophosphoglycerophosphomonoradylglycerol	-
LC_Alkylacylglycerophosphoglycerophosphodiradylglycerol	LC_Glycerophosphoglycerophosphoglycerol

	hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carboxylic_Acid_Ester or Ether or Carbon_Chain_Group);
LC_num1-alkyl_2-acylglycerophosphoglycerophosphodiradylglycerol	-
LC_Alkylacylglycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carboxylic_Acid_Ester or Ether or Carbon_Chain_Group);
LC_num1-alkyl_2-acylglycerophosphoglycerophosphomonoradylglycerol	-
LC_Diacylglycerophosphoglycerophosphodiradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAcyl_Chain exactly 2 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carbon_Chain_Group or Carboxylic_Acid_Ester);
LC_Diacylglycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAcyl_Chain exactly 2 ⊓ hasCarbon_Chain exactly 3; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carbon_Chain_Group or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphoglycerophosphodiradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAlkyl_Ether_Chain exactly 2 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Ether

	or Carbon_Chain_Group);
LC_Dialkylglycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAlkyl_Ether_Chain exactly 2 ⊓ hasCarbon_Chain exactly 3; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Ether or Carbon_Chain_Group);
LC_Monoacylglycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAcyl_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Carbon_Chain_Group or Carboxylic_Acid_Ester);
]LC_num1-alkyl_glycerophosphoglycerophosphodiradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAlkyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 3; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Ether or Carbon_Chain_Group);
LC_num1-alkyl_glycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasAlkyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Ether or Carbon_Chain_Group);
LC_num1Z-alkenylglycerophosphoglycerophosphodiradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasVinyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 2; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Ether or Carbon_Chain_Group);

LC_num1Z-alkenylglycerophosphoglycerophosphomonoradylglycerol	LC_Glycerophosphoglycerophosphoglycerol hasVinyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Glycerophosphotidylglycerophosphoglycerol or Ether or Carbon_Chain_Group);
LC_Glycerophosphoinositol	LC_Glycerophospholipid hasPart some Glycerophosphatidylinositol ⊓ hasPart some Acyl_Chain;
LC_Alkenylacylglycerophosphoinositol	LC_Glycerophosphoinositol hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylinositol or Acyl_Chain or Vinyl_Ether_Chain or Carboxylic_Acid_Ester or Ether);
LC_num1Z-alkenyl_2-acylglycerophosphoinositol	-
LC_Alkylacylglycerophosphoinositol	LC_Glycerophosphoinositol hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylinositol or Acyl_Chain or Alkyl_Ether_Chain or Carboxylic_Acid_Ester or Ether);
LC_num1-alkyl_2-acylglycerophosphoinositol	-
LC_Diacylglycerophosphoinositol	LC_Glycerophosphoinositol hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidylinositol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphoinositol	LC_Glycerophosphoinositol

	hasAlkyl_Ether_Chain exactly 2; hasPart only (Glycerophosphatidylinositol or Alkyl_Ether_Chain or Ether);
LC_Monoacylglycerophosphoinositol	LC_Glycerophosphoinositol hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylinositol or Acyl_Chain or Carboxylic_Acid_Ester);
LC_num1-alkyl_glycerophosphoinositol	LC_Glycerophosphoinositol hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidylinositol or Alkyl_Ether_Chain or Ether);
LC_num1Z-alkenylglycerophosphoinositol	LC_Glycerophosphoinositol hasVinyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidylinositol or Vinyl_Ether_Chain or Ether);
LC_Glycerophosphoinositol_bisphosphate	LC_Glycerophospholipid hasPart some Glycerophosphatidylinositol_Diphosphates ⊓ hasPart some Acyl_Chain;
LC_Diacylglycerophosphoinositol_bisphosphate	LC_Glycerophosphoinositol_bisphosphate hasAcyl_Chain exactly 2; hasPart some (Glycerophosphatidylinositol_Diphosphates or Acyl_Chain or Carboxylic_Acid_Ester);

LC_Glycerophosphoinositol_monophosphate	LC_Glycerophospholipid hasPart some Glycerophosphatidylinositol_Monophosphate ⊓ hasPart some Acyl_Chain;
LC_Diacylglycerophosphoinositol_monophosphate	LC_Glycerophosphoinositol_monophosphate hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidylinositol_Monophosphate or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycerophosphoinositol_trisphosphate	LC_Glycerophospholipid hasPart some Glycerophosphatidylinositol_Triphosphates ⊓ hasPart some Acyl_Chain;
LC_Diacylglycerophosphoinositol_trisphosphate	LC_Glycerophosphoinositol_trisphosphate hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidylinositol_Triphosphates or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycerophosphoinositolglycan	LC_Glycerophospholipid hasPart some Glycerophosphatidylinositol_Glycans;
LC_Alkenylacylglycerophosphoinositolglycan	LC_Glycerophosphoinositolglycan hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylinositol_Glycans or Carboxylic_ Acid_Ester or Ether or Acyl_Chain or Vinyl_Ether_Chain);
LC_num1Z-alkenyl_2-acylglycerophosphoinositolglycan	-

LC_Alkylacylglycerophosphoinositolglycan	LC_Glycerophosphoinositolglycan hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylinositol_Glycans or Carboxylic_ Acid_Ester or Ether or Acyl_Chain or Alkyl_Ether_Chain);
LC_num1-alkyl_2-acylglycerophosphoinositolglycan	-
LC_Diacylglycerophosphoinositolglycan	LC_Glycerophosphoinositolglycan hasAcyl_Chain exactly 2; hasPart only (Glycerophosphatidylinositol_Glycans or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Monoacylglycerophosphoinositolglycan	LC_Glycerophosphoinositolglycan hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylinositol_Glycans or Acyl_Chain or Carboxylic_Acid_Ester);
LC_num1-alkyl_glycerophosphoinositolglycan	LC_Glycerophosphoinositolglycan hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidylinositol_Glycans or Ether or Alkyl_Ether_Chain);
LC_num1Z-alkenylglycerophosphoinositolglycan	LC_Glycerophosphoinositolglycan hasVinyl_Ether_Chain exactly 1; hasPart only (Glycerophosphatidylinositol_Glycans or Ether or Vinyl_Ether_Chain);

LC_Glycerophosphonocholine	LC_Glycerophospholipid hasPart some Glycerophosphonotidylcholine ⊓ hasPart some Acyl_Chain;
LC_Diacylglycerophosphonocholine	LC_Glycerophosphonocholine hasAcyl_Chain exactly 2; hasPart only (Glycerophosphonotidylcholine or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycerophosphonoethanolamine	LC_Glycerophospholipid hasPart some Glycerophosphonotidylethanolamine ⊓ hasPart some Acyl_Chain;
LC_Diacylglycerophosphonoethanolamine	LC_Glycerophosphonoethanolamine hasAcyl_Chain exactly 2; hasPart only (Glycerophosphonotidylethanolamine or Acyl_Chain or Carboxylic_Acid_Ester);
LC_Glycerophosphoserine	LC_Glycerophospholipid hasPart some Glycerophosphatidylserine;
LC_Alkenylacylglycerophosphoserine	LC_Glycerophosphoserine hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylserine or Carboxylic_Acid_Ester or Ether or Acyl_Chain or Vinyl_Ether_Chain);
LC_num1Z-alkenyl_2-acylglycerophosphoserine	-
LC_Alkylacylglycerophosphoserine	LC_Glycerophosphoserine

	hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycerophosphatidylserine or Carboxylic_Acid_Ester or Ether or Acyl_Chain or Alkyl_Ether_Chain);
LC_num1-alkyl_2-acylglycerophosphoserine	-
LC_Diacylglycerophosphoserine	LC_Glycerophosphoserine hasAcyl_Chain exactly 2; hasPart only (Acyl_Chain or Glycerophosphatidylserine or Carboxylic_Acid_Ester);
LC_Dialkylglycerophosphoserine	LC_Glycerophosphoserine hasAlkyl_Ether_Chain exactly 2; hasPart some (Alkyl_Ether_Chain or Glycerophosphatidylserine or Ether);
LC_Monoacylglycerophosphoserine	LC_Glycerophosphoserine hasAcyl_Chain exactly 1; hasPart only (Acyl_Chain or Glycerophosphatidylserine or Carboxylic_Acid_Ester);
LC_num1-alkyl_glycerophosphoserine	LC_Glycerophosphoserine hasAlkyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Alkyl_Ether_Chain or Glycerophosphatidylserine or Ether);
LC_num1Z-alkenylglycerophosphoserine	LC_Glycerophosphoserine

	hasVinyl_Ether_Chain exactly 1 ⊓ hasCarbon_Chain exactly 1; hasPart only (Vinyl_Ether_Chain or Glycerophosphatidylserine or Ether);
LC_Glyceropyrophosphate	LC_Glycerophospholipid hasPart some Glyceropyrophosphatidic_acid ⊓ hasPart some Carboxylic_Acid_derivative_Group;
LC_Diacylglyceropyrophosphate	LC_Glyceropyrophosphate hasAcyl_Chain exactly 2; hasPart only (Acyl_Chain or Glyceropyrophosphatidic_acid or Carboxylic_Acid_Ester);
LC_Monoacylglyceropyrophosphate	LC_Glyceropyrophosphate hasAcyl_Chain exactly 1; hasPart only (Acyl_Chain or Glyceropyrophosphatidic_acid or Carboxylic_Acid_Ester);
LC_Glycosylglycerophospholipid	LC_Glycerophospholipid hasPart some Phosphate_Group;
LC_Alkenylacylglycosylglycerophospholipid	LC_Glycosylglycerophospholipid hasVinyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycosylglycerophosphate_Group or Carboxylic_Acid _Ester or Ether or Acyl_Chain or Vinyl_Ether_Chain);
LC_num1Z-alkenyl_2-acylglycosylglycerophospholipid	-

LC_Alkylacylglycosylglycerophospholipid	LC_Glycosylglycerophospholipid hasAlkyl_Ether_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Glycosylglycerophosphate_Group or Carboxylic_Acid _Ester or Ether or Acyl_Chain or Alkyl_Ether_Chain);
LC_num1-alkyl_2-acylglycosylglycerophospholipid	-
LC_Diacylglycosylglycerophospholipid	LC_Glycosylglycerophospholipid hasAcyl_Chain exactly 2; hasPart only (Glycosylglycerophosphate_Group or Carboxylic_Acid _Ester or Acyl_Chain);
LC_Monoacylglycosylglycerophospholipid	LC_Glycosylglycerophospholipid hasAcyl_Chain exactly 1; hasPart only (Glycosylglycerophosphate_Group or Carboxylic_ Acid_Ester or Acyl_Chain);
LC_num1-alkylglycosylglycerophospholipid	LC_Glycosylglycerophospholipid hasAlkyl_Ether_Chain exactly 1; hasPart only (Glycosylglycerophosphate_Group or Ether or Alkyl_Ether_Chain);
LC_num1Z-alkenylglycosylglycerophospholipid	LC_Glycosylglycerophospholipid hasVinyl_Ether_Chain exactly 1; hasPart only (Glycosylglycerophosphate_Group or Ether or Vinyl_Ether_Chain);
LC_Prenol_Lipid	Lipid

	hasPart some (Prenyl or Phytyl);
LC_Hopanoid	LC_Prenol_Lipid hasPart some Hopanoid_ring_system ⊓ hasPart some (Epoxy or Methyl or Carboxylic_Acid_Ester_Group or Ether) ⊓ hasPart some (Acyl_Chain or Alkyl_Chain or Bicyclic_Heterocycle_System or Monocyclic_Ring_Group); hasPart some (Carboxylic_Acid_Amide_Group or Alcohol or Alkenyl_Group or Amine_Group); hasPart some Prenyl; hasPart only (Hopanoid_ring_system or Bicyclic_Heterocycle_System or Monocyclic_Ring_Group or Isoprenoid_ring_derivative or Acyl_Chain or Alkyl_Chain or Prenyl or Ether or Carboxylic_Acid_Ester_Group or Methyl or Epoxy or Carboxylic_Acid_Amide_Group or Alcohol or Alkenyl_Group or Amine_Group);
LC_Isoprenoid	LC_Prenol_Lipid hasPart some Prenyl; hasPart some (Isoprene_Chain or Isoprenoid_ring_derivative);
LC_C10_isoprenoid_par_monoterpene_par_	LC_Isoprenoid hasPart some (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) ⊓ hasPrenyl_Group exactly 2; hasPart some (Alcohol or Alkenyl_Group or Ketone or Ethyl or Methyl or Carboxylic_Acid_Ester_Group or Ether or Aldehyde or Carboxylic_Acid or Phosphate_Group); hasPart only (Alcohol or Alkenyl_Group or Ketone or Ethyl or

	Methyl or Carboxylic_Acid_Ester_Group or Ether or Aldehyde or Carboxylic_Acid or Phosphate_Group) ⊔ (Isoprene_Chain ⊔ Monocyclic_Terpenoid ⊔ Polycyclic_Terpenoid) ⊔ Prenyl;
LC_C15_isoprenoid_par_sesquiterpene_par_	LC_Isoprenoid hasPart some (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) ⊓ hasPrenyl_Group exactly 3; hasPart some (Acyl_Chain or Methyl or Epoxy or Alcohol or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or Aldehyde); hasPart only ((Acyl_Chain or Methyl or Epoxy or Alcohol or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or Aldehyde) or (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) or Prenyl);
LC_C20_isoprenoid_par_diterpene_par_	LC_Isoprenoid hasPart some (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) ⊓ hasPrenyl_Group exactly 4; hasPart some (Alcohol or Methyl or Phosphate_Group or Carboxylic_Acid_Amide_Group or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or Aldehyde or Carboxylic_Acid); hasPart only (Alcohol or Methyl or Phosphate_Group or Carboxylic_Acid_Amide_Group or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or Aldehyde or Carboxylic_Acid or (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) or Prenyl);
LC_C25_isoprenoid_par_sesterterpene_par_	LC_Isoprenoid

	hasPart some (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) ⊓ hasPrenyl_Group exactly 5; hasPart some (Alcohol or Epoxy or Methyl or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or Aldehyde); hasPart only (Alcohol or Epoxy or Methyl or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or Aldehyde or (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_ Terpenoid) or Prenyl);
LC_C30_isoprenoid_par_triterpene_par_	LC_Isoprenoid hasPart some (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_Terpenoid) ⊓ hasPrenyl_Group exactly 6; hasPart some (Alcohol or Epoxy or Methyl or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether); hasPart only (Alcohol or Epoxy or Methyl or Alkenyl_Group or Ketone or Carboxylic_Acid_Ester_Group or Ether or (Isoprene_Chain or Monocyclic_Terpenoid or Polycyclic_ Terpenoid) or Prenyl);
LC_C40_isoprenoid_par_tetraterpene_par_	LC_Isoprenoid hasPart some (Isoprene_Chain or Monocyclic_Terpenoid or Bicyclic_terpenoid) ⊓ hasPrenyl_Group exactly 8; hasPart some (Monocyclic_Ring_Group or Acyl_Chain or Epoxy or Monomeric_Glycan_Group or Alcohol or Alkenyl_Group or Ketone or Ether or Carboxylic_Acid_Ester_Group or Sulfuric_Acid_ derivative_Group); hasPart only ((Isoprene_Chain or Monocyclic_Terpenoid or Bicyclic_terpenoid) or Monocyclic_Ring_Group or Acyl_Chain or

	Epoxy or Monomeric_Glycan_Group or Alcohol or Alkenyl_Group or Ketone or Ether or Carboxylic_Acid_Ester_Group or Sulfuric_Acid_derivative_Group or Prenyl);
LC_C5_isoprenoid	LC_Isoprenoid hasPrenyl_Group exactly 1 ⊓ hasPhosphate_Group exactly 2; hasPart only (Prenyl or Phosphate_Group or Isoprene_Chain);
LC_Polyterpene	LC_Isoprenoid hasPart some Isoprene_Chain ⊓ hasPart some Monocyclic_Terpenoid ⊓ hasPrenyl_Group min 9; hasPart only (Isoprene_Chain or Monocyclic_Terpenoid or Prenyl);
LC_Retinoid	LC_Isoprenoid hasPart some Isoprene_Chain ⊓ hasPart some Monocyclic_Ring_Group; hasPart some (Acyl_Chain or Ketone or Alcohol or Aldehyde or Carboxylic_Acid or Methyl or Epoxy or Methyl or Ether or Carboxylic_Acid_Ester_Group or Carboxylic_Acid); hasPart only ((Acyl_Chain or Ketone or Alcohol or Aldehyde or Carboxylic_Acid or Methyl or Epoxy or Methyl or Ether or Carboxylic_Acid_Ester_Group or Carboxylic_Acid) or Isoprene_Chain or Monocyclic_Ring_Group or Prenyl);
LC_Polyprenol	LC_Prenol_Lipid hasPart some Prenyl; hasPart some ((Alcohol or Carboxylic_Acid) or Phosphate_Group) ⊓

	hasPart some Isoprene_Chain;
LC_Bactoprenol	LC_Polyprenol hasPrenyl_Group exactly 11 ⊓ hasPart some Alcohol; hasPart only (Isoprene_Chain or Alcohol or Prenyl);
LC_Bactoprenol_diphosphate	LC_Polyprenol hasPhosphate_Group exactly 2 ⊓ hasPrenyl_Group exactly 11 ⊓ hasPart some Glycan_Group; hasPart only (Isoprene_Chain or Phosphate_Group or Glycan_Group or Prenyl);
LC_Bactoprenol_monophosphate	LC_Polyprenol hasPhosphate_Group exactly 1 ⊓ hasPrenyl_Group exactly 11 ⊓ hasPart some Glycan_Group; hasPart only (Isoprene_Chain or Phosphate_Group or Glycan_Group or Prenyl);
LC_Dolichol	LC_Polyprenol hasPrenyl_Group max 21 ⊓ hasPart some (Alcohol or Carboxylic_Acid) ⊓ hasPrenyl_Group min 12; hasPart only (Isoprene_Chain or Alcohol or Carboxylic_Acid or Prenyl);
LC_Dolichol_monophosphate	LC_Polyprenol hasPhosphate_Group exactly 1 ⊓ hasPrenyl_Group max 21 ⊓ hasPrenyl_Group min 12;

	hasPart only (Isoprene_Chain or Phosphate_Group or Prenyl);
LC_Dolichols_diphosphate	LC_Polyprenol hasPhosphate_Group exactly 2 ⊓ hasPrenyl_Group max 21 ⊓ hasPrenyl_Group min 12; hasPart only (Isoprene_Chain or Phosphate_Group or Prenyl);
LC_Phytoprenol	LC_Polyprenol hasPart some Alcohol ⊓ hasPrenyl_Group exactly 5; hasPart only (Isoprene_Chain or Alcohol or Prenyl);
LC_Phytoprenol_diphosphate	LC_Polyprenol hasPhosphate_Group exactly 2 ⊓ hasPrenyl_Group exactly 5; hasPart only (Isoprene_Chain or Phosphate_Group or Prenyl);
LC_Phytoprenol_monosphophate	LC_Polyprenol hasPhosphate_Group exactly 1 ⊓ hasPrenyl_Group exactly 5; hasPart only (Isoprene_Chain or Phosphate_Group or Prenyl);
LC_Quinone_par_inclusive_of_hydroquinone_par_	LC_Prenol_Lipid hasPart some Quinone_ring_system;
LC_Ubiquinone	LC_Quinone_par_inclusive_of_hydroquinone_par_ hasPart some Isoprene_Chain ⊓ hasPrenyl_Group min 3 ⊓ hasPrenyl_Group max 10; hasPart some Ubiquinone_ring ⊓ hasPart some Alkenyl_Group ⊓

	hasPart some Ketone ⊓ hasPart some Ether; hasPart only (Isoprene_Chain or Ubiquinone_ring or Prenyl or Alkenyl_Group or Ketone or Ether);
LC_Vitamin_E	LC_Quinone_par_inclusive_of_hydroquinone_par_ hasPart some Tocopherol_ring ⊔ hasPart some Tocopherolquinone_ring ⊔ hasPart some Tocoquinone_ring; hasPart only (Tocoquinone_ring or Tocopherol_ring or Tocopherolquinone_ring or Prenyl);
LC_Vitamin_K	LC_Quinone_par_inclusive_of_hydroquinone_par_ hasPart some Menaquinone_ring; hasPart only (Menaquinone_ring or Phytyl or Prenyl);
LC_Saccharolipid	Lipid hasPart some Acyl_Chain ⊓ hasPart some Glycan_Group;
LC_Acylaminosugar	LC_Saccharolipid hasPart some (Monomeric_Glycan_Group or (Dimeric_Glycan_Group and not Trehalose)) ⊓ hasPart some Phosphate_Group;
LC_Diacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 2 ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid; hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or

	Amino_Acid or Acyl_Chain);
LC_Heptaacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 7 ⊓ hasPart some Carboxylic_Acid_Ester_ Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid; hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_ Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain);
LC_Hexaacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 6 ⊓ hasPart some Carboxylic_Acid_Ester_ Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid; hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_ Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain);
LC_Monoacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 1 ⊓ hasPart some Carboxylic_Acid_Ester_ Group ⊓ hasPart some Amine_Group ⊓ hasPart some Amino_Acid; hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain);
LC_Pentaacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 5 ⊓ hasPart some Carboxylic_Acid_Ester_ Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid;

	hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain);
LC_Tetraacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 4 ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid; hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain);
LC_Triacylaminosugar	LC_Acylaminosugar hasAcyl_Chain exactly 3 ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid; hasPart only (Phosphate_Group or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain);
LC_Acylaminosugar_glycan	LC_Saccharolipid hasPart some Phosphate_Group ⊓ hasPart some Acyl_Estolide_Chain ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Amino_Acid ⊓ hasPart some Carboxylic_Acid; hasPart only (Phosphate_Group or Carboxylic_Acid or Glycan_Group or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Amino_Acid or Acyl_Chain or Acyl_Estolide_Chain);

LC_Acyltrehalose	LC_Saccharolipid hasPart some Trehalose ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Amide_Group; hasPart only (Trehalose or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Acyl_Chain);
LC_Acyltrehalose_glycan	LC_Saccharolipid hasPart some Trehalose ⊓ hasPart some Phosphate_Group ⊓ hasPart some Acyl_Estolide_Chain ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Amide_Group ⊓ hasPart some Carboxylic_Acid; hasPart only (Phosphate_Group or Carboxylic_Acid or Trehalose or Carboxylic_Acid_Ester_Group or Carboxylic_Acid_Amide_Group or Acyl_Chain or Acyl_Estolide_Chain);
LC_Sphingolipid	Lipid hasPart some Sphingoid_Base_Chain_Group ⊓ hasPart some (Primary_Amine or Carboxylic_Acid_Secondary_Amide) ⊓ hasPart some Alcohol;
LC_Acidic_glycosphingolipid	LC_Sphingolipid hasSphingoid_Base_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Secondary_Amide ⊓ hasPart some (Organic_Sulfur_Group or Glycan_series or Organic_Phosphate_Group);
LC_Ganglioside	LC_Acidic_glycosphingolipid

	hasPart some Ganglioside_series; hasPart only (Ganglioside_series or Carboxylic_Acid_Secondary _Amide or Sphingoid_Base_Chain_Group or Alcohol or Carboxylic _Acid_Ester_Group or Acyl_Chain);
LC_Glucuronosphingolipid	LC_Acidic_glycosphingolipid hasPart some Glucuronic_acid; hasPart only (Glucuronic_acid or Carboxylic_Acid_Secondary _Amide or Sphingoid_Base_Chain_Group or Alcohol or Carboxylic_Acid_Ester_Group or Acyl_Chain);
LC_Phosphoglycosphingolipid	LC_Acidic_glycosphingolipid hasPart some (Charged_Glycan_series or Neutral_Glycan_series) ⊓ hasPart some Phosphate_Group; hasPart only (Charged_Glycan_series or Neutral_Glycan_series or Phosphate_Group or Carboxylic_Acid_Secondary_Amide or Sphingoid_Base_Chain_Group or Alcohol or Carboxylic_Acid_ Ester_Group or Acyl_Chain);
LC_Sulfoglycosphingolipid_par_sulfatide_par_	LC_Acidic_glycosphingolipid hasPart some (Charged_Glycan_series or Neutral_Glycan_series) ⊓ hasPart some Sulfuric_Acid_derivative_Group; hasPart only (Charged_Glycan_series or Neutral_Glycan_series or Sulfuric_Acid_derivative_Group or Carboxylic_Acid_Secondary _Amide or Sphingoid_Base_Chain_Group or Alcohol or Carboxylic_Acid_Ester_Group or Acyl_Chain);
LC_Amphoteric_glycosphingolipid	LC_Sphingolipid hasSphingoid_Base_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓

	hasPart some Carboxylic_Acid_Ester_Group ⊓ hasPart some Carboxylic_Acid_Secondary_Amide ⊓ hasPart some Glycan_series ⊓ hasPart some Phosphoethanolamine; hasPart only (Glycan_series or Phosphoethanolamine or Carboxylic_Acid_Secondary_Amide or Sphingoid_Base_Chain_ Group or Alcohol or Acyl_Chain or Carboxylic_Acid_Ester_Group);
LC_Basic_glycosphingolipid	LC_Sphingolipid hasSphing-4-enine_Chain exactly 1 ⊓ hasPart some Primary_Amine ⊓ hasPart some Monomeric_Glycan_Group; hasPart only (Monomeric_Glycan_Group or Sphing-4-nine _par_Sphingosine_par_Chain or Primary_Amine or Alcohol);
LC_Ceramide	LC_Sphingolipid hasSphingoid_Base_Chain exactly 1 ⊓ hasPart some Acyl_Chain ⊓ hasPart some Carboxylic_Acid_Secondary_Amide ⊓ hasPart some Carboxylic_Acid_Ester_Group;
LC_Acylceramide	LC_Ceramide hasPart some Alkenyl_Group ⊓ hasSphing-4-enine_Chain exactly 1 ⊓ hasAcyl_Ester_Chain exactly 1; hasPart only (Alkenyl_Group or Sphing-4-nine_par_Sphingosine _par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_ Acid_Ester_Group or Acyl_Chain or Alcohol);
LC_Ceramide_1-phosphate	LC_Ceramide hasAcyl_Chain exactly 1 ⊓ hasPart some Phosphate_Group ⊓ hasPart some Alkenyl_Group ⊓ hasSphing-4-enine_Chain exactly 1;

	hasPart only (Alkenyl_Group or Sphing-4-nine_par_Sphingosine_ par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_ Acid_Ester_Group or Acyl_Chain or Alcohol or Phosphate_Group);
LC_N-acy-4-hydroxysphinganine_par_phytoceramide_par_	LC_Ceramide has4-hydroxysphinganine_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (num4-hydroxysphinganine_par_Phytosphingosine_ par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester_Group or Acyl_Chain or Alcohol);
LC_N-acylsphinganine_par_dihydroceramide_par_	LC_Ceramide hasSphinganine_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Alcohol or Sphinganine_par_Dihydrosphingosine_par _Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid _Ester_Group or Acyl_Chain);
LC_N-acylsphingosine_par_ceramide_par_	LC_Ceramide hasPart some Alkenyl_Group ⊓ hasSphing-4-enine_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1; hasPart only (Alkenyl_Group or Sphing-4-nine_par_Sphingosine _par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic _Acid_Ester_Group or Acyl_Chain or Alcohol);
LC_Neutral_glycosphingolipid	LC_Sphingolipid hasPart some Alkenyl_Group ⊓ hasPart some Carboxylic_Acid_ Secondary_Amide ⊓hasSphingoid_ Base_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasPart some (Neutral_Glycan_series or

	Monomeric_Glycan_Group);
LC_Gal-_par_Gala_series_par_	LC_Neutral_glycosphingolipid hasPart some Galactose; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain _Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_ Ester_Group or Acyl_Chain or Alcohol or Galactose);
LC_Galb1-3GlcNAcb1-3Galb1-4Glc-_par_Lacto_series_par_	LC_Neutral_glycosphingolipid hasPart some Lacto_series; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Lacto_series);
LC_Galb1-4GlcNAcb1-3Galb1-4Glc-_par_Neolacto_series_par_	LC_Neutral_glycosphingolipid hasPart some Neo-lacto_series; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Neo-lacto_series);
LC_GalNAcb1-3Gala1-3Galb1-4Glc-_par_Isoglobo_series_par_	LC_Neutral_glycosphingolipid hasPart some Isoglobo_series; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Isoglobo_series);
LC_GalNAcb1-3Gala1-4Galb1-4Glc-_par_Globo_series_par_	LC_Neutral_glycosphingolipid hasPart some Globo_series;

	hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Globo_series);
LC_GalNAcb1-4Galb1-4Glc-_par_Ganglio_series_par_	LC_Neutral_glycosphingolipid hasPart some Ganglio_series; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Ganglio_series);
LC_GalNAcb1-4GlcNAcb1-3Manb1-4Glc-_par_Arthro_series_par_	LC_Neutral_glycosphingolipid hasPart some Arthro_series; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Arthro_series);
LC_GlcNAcb1-2Mana1-3Manb1-4Glc-_par_Mollu_series_par_	LC_Neutral_glycosphingolipid hasPart some Mollu_series; hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Mollu_series);
LC_Simple_Glc_series_par_GlcCer__LacCer__etc__par_	LC_Neutral_glycosphingolipid hasPart some (Lactose or Glucose); hasPart only (Alkenyl_Group or Sphingoid_Base_Chain_Group or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester _Group or Acyl_Chain or Alcohol or Lactose or Glucose or Galactose);

LC_Phosphonosphingolipid	LC_Sphingolipid hasSphingoid_Base_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasPart some Phosphono_Group ⊓ hasPart some Carboxylic_Acid_Ester_Group; hasPart only (Sphingoid_Base_Chain_Group or Acyl_Chain or Carboxylic_Acid_Secondary_Amide or Alcohol or Carboxylic_Acid_Ester_Group or Phosphono_Group);
LC_Phosphosphingolipid	LC_Sphingolipid hasSphingoid_Base_Chain exactly 1 ⊓ hasAcyl_Chain exactly 1 ⊓ hasPart some Phospho_Group;
LC_Ceramide_phosphocholine_par_sphingomyelin_par_	LC_Phosphosphingolipid hasPart some Phosphocholine ⊓ hasPart some Alkenyl_Group ⊓ hasSphing-4-enine_Chain exactly 1 ⊓ hasPart some Carboxylic_Acid_Secondary_Amide; hasPart only (Phosphocholine or Alkenyl_Group or Sphing-4-nine_par_Sphingosine_par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester_Group or Acyl_Chain or Alcohol);
LC_Ceramide_phosphoethanolamine	LC_Phosphosphingolipid hasPart some Phosphoethanolamine ⊓ hasPart some Alkenyl_Group ⊓ hasSphing-4-enine_Chain exactly 1 ⊓ hasPart some Carboxylic_Acid_Secondary_Amide; hasPart only (Phosphoethanolamine or Alkenyl_Group or Sphing-4-nine_par_Sphingosine_par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester_Group or Acyl_Chain or Alcohol);

LC_Ceramide_phosphoinositol	LC_Phosphosphingolipid hasPart some Phosphoinositol ⊓ hasPart some Alkenyl_Group ⊓ hasSphing-4-enine_Chain exactly 1 ⊓ hasPart some Carboxylic_Acid_Secondary_Amide; hasPart only (Phosphoinositol or Alkenyl_Group or Sphing-4-nine_par_Sphingosine_par_Chain or Carboxylic_Acid_Secondary_Amide or Carboxylic_Acid_Ester_Group or Acyl_Chain or Alcohol);
LC_Sphingoid_base	LC_Sphingolipid hasSphingoid_Base_Chain exactly 1;
LC_Lysoglycosphingolipid	-
LC_Lysosphingomyelin	LC_Sphingoid_base hasPart some Phosphocholine ⊓ hasPart some Alkenyl_Group; hasPart only (Phosphocholine or Primary_Amine or Alcohol or Sphingoid_Base_Chain_Group or Alkenyl_Group);
LC_N-methylated_sphingoid_base	-
LC_num4-Hydroxysphinganine_par_Phytosphingosine_par_	LC_Sphingoid_base has4-hydroxysphinganine_Chain exactly 1; hasPart only (num4-hydroxysphinganine_par_Phytosphingosine_par_Chain or Primary_Amine or Alcohol);
LC_Sphing-4-enine_par_Sphingosine_par_	LC_Sphingoid_base hasSphing-4-enine_Chain exactly 1 ⊓ hasPart some Alkenyl_Group; hasPart only (Alkenyl_Group or Primary_Amine or Alcohol or

	Sphing-4-nine_par_Sphingosine_par_Chain);
LC_Sphinganine	LC_Sphingoid_base hasSphinganine_Chain exactly 1; hasPart only (Primary_Amine or Alcohol or Sphinganine _par_Dihydrosphingosine_par_Chain);
LC_Sphingoid_base_1-phosphate	LC_Sphingoid_base hasPart some Phosphate_Group; hasPart only (Phosphate_Group or Primary_Amine or Alcohol or Sphingoid_Base_Chain_Group);
LC_Sphingoid_base_homolog_structural_derivative	-
LC_Sphingoid_base_homolog	-
LC_Sphingoid_base_homolog_variant	-
LC_Sterol_Lipid	Lipid hasPart some Cyclopenta-a-Phenanthrene_Ring_ System;
LC_Bile_acid_structural_derivative	LC_Sterol_Lipid hasPart some Cyclopenta-a-Phenanthrene_Ring ⊓ hasPart some (Carboxylic_Acid or Ketone or Alcohol);
LC_C22_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some nor-Cholane ⊓ hasPart some Propyl_Derivative; hasPart only (nor-Cholane or Propyl_Derivative or Alcohol or Carboxylic_Acid or Ketone);

LC_C22_bile_acid	-
LC_C22_bile_acid_alcohol_derivative	-
LC_C22_bile_acid_derivative	-
LC_C23_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some nor-Cholane ⊓ hasPart some Butyl_Derivative; hasPart only (Ketone or Alcohol or nor-Cholane or Butyl_Derivative);
LC_C23_bile_acid	-
LC_C23_bile_acid_alcohol_derivative	-
LC_C23_bile_acid_derivative	-
LC_C24_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some Cholane ⊓ hasPart some Penthyl_Derivative ⊓ hasPart some (Aldehyde or Alkenyl_Group); hasPart only (Cholane or Penthyl_Derivative or Aldehyde or Alkenyl_Group or Alcohol or Carboxylic_Acid);
LC_C24_bile_acid	-
LC_C24_bile_acid_alcohol_derivative	-
LC_C24_bile_acid_derivative	-
LC_C25_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some nor-Cholestane ⊓ hasPart some Hexanoyl_Derivative; hasPart only (nor-Cholestane or Hexanoyl_Derivative or Alcohol or Ketone);
LC_C25_bile_acid	-
LC_C25_bile_acid_alcohol_derivative	-

LC_C25_bile_acid_derivative	-
LC_C26_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some nor-Cholestane ⊓ hasPart some Hepthyl_Derivative; hasPart only (nor-Cholestane or Hepthyl_Derivative or Ketone or Alcohol);
LC_C26_bile_acid	-
LC_C26_bile_acid_alcohol_derivative	-
LC_C26_bile_acid_derivative	-
LC_C27_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some Cholestane ⊓ hasPart some Iso-Octyl_Derivative ⊓ hasPart some Alkenyl_Group; hasPart only (Cholestane or Alkenyl_Group or Iso-Octyl_Derivative or Alcohol or Ketone or Carboxylic_Acid);
LC_C27_bile_acid	-
LC_C27_bile_acid_alcohol_derivative	-
LC_C27_bile_acid_derivative	-
LC_C28_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some (Ergostane or Campestane) ⊓ hasPart some Methyl-Iso-Octyl ⊓ hasPart some (Methyl or Alkenyl_Group); hasPart only (Ergostane or Campestane or Methyl or Alkenyl_Group or Alcohol or Carboxylic_Acid or Methyl-Iso-Octyl);
LC_C28_bile_acid	-
LC_C28_bile_acid_alcohol_derivative	-
LC_C28_bile_acid_derivative	-

LC_C29_bile_acid_structural_derivative	LC_Bile_acid_structural_derivative hasPart some homo-Cholestane ⊓ hasPart some Decanoyl_ Derivative; hasPart only (homo-Cholestane or Decanoyl_Derivative or Alcohol or Carboxylic_Acid);
LC_C29_bile_acid	-
LC_C29_bile_acid_alcohol_derivative	-
LC_C29_bile_acid_derivative	-
LC_Secosteroid	LC_Sterol_Lipid hasPart some Cyclopenta-a-Phenanthrene_fissile_variant;
LC_Vitamin_D2_structural_derivative	LC_Secosteroid hasPart some Ergocalciferol ⊓ hasPart some Methyl-Iso-Octyl ⊓ hasPart some (Alcohol or Methyl or Sulfuric_Acid_derivative _Group or Alkenyl_Group); hasPart only (Ergocalciferol or Alcohol or Sulfuric_Acid_derivative _Group or Alkenyl_Group or Methyl or Methyl-Iso-Octyl);
LC_Vitamin_D2	-
LC_Vitamin_D2_derivative	-
LC_Vitamin_D3_structural_derivative	LC_Secosteroid hasPart some (Seco-Choladiene or Seco-Cholatriene or Seco-Pregnatriene or Seco-Cholestatriene or Seco-Cholestapentaene or Seco-Cholestatetraene or Penthyl_Derivative or Iso-Octyl_ Derivative or Alcohol or Epoxy or Carboxylic_Acid_derivative_ Group or Alkynyl_Group or Alkenyl_Group or Ether or Ketone); hasPart only (Seco-Choladiene or Seco-Cholatriene or Seco-

	Pregnatriene or Seco-Cholestatriene or Seco-Cholestapentaene or Seco-Cholestatetraene or Penthyl_Derivative or Iso-Octyl_ Derivative or Alcohol or Epoxy or Carboxylic_Acid_derivative_ Group or Alkynyl_Group or Alkenyl_Group or Ether or Ketone);
LC_Vitamin_D3	-
LC_Vitamin_D3_derivative	-
LC_Vitamin_D4_structural_derivative	LC_Secosteroid hasPart some Seco-Ergostatriene ⊓ hasPart some Methyl-Iso-Octyl ⊓ hasPart some (Alcohol or Methyl or Alkenyl_Group); hasPart only (Seco-Ergostatriene or Methyl-Iso-Octyl or Alcohol or Methyl or Alkenyl_Group);
LC_Vitamin_D4	-
LC_Vitamin_D4_derivative	-
LC_Vitamin_D5_structural_derivative	LC_Secosteroid hasPart some Seco-Poriferastatriene ⊓ hasPart some Ethyl-Iso-Octyl ⊓ hasPart some (Alcohol or Ethyl or Alkenyl_Group); hasPart only (Seco-Poriferastatriene or Ethyl-Iso-Octyl or Alcohol or Ethyl or Alkenyl_Group);
LC_Vitamin_D5	-
LC_Vitamin_D5_derivative	-
LC_Vitamin_D6_structural_derivative	LC_Secosteroid hasPart some Seco-Poriferastatetraene ⊓ hasPart some Ethyl-Iso-Octyl ⊓ hasPart some (Alcohol or Alkenyl_Group or Ethyl); hasPart only (Seco-Poriferastatetraene or Ethyl-Iso-Octyl or Alcohol or Alkenyl_Group or Ethyl);

LC_Vitamin_D6	-
LC_Vitamin_D6_derivative	-
LC_Vitamin_D7_structural_derivative	LC_Secosteroid hasPart some Seco-Campestatriene ⊓ hasPart some Methyl-Iso-Octyl ⊓ hasPart some (Alkenyl_Group or Alcohol or Ethyl); hasPart only (Seco-Campestatriene or Methyl-Iso-Octyl or Alkenyl_Group or Alcohol or Ethyl);
LC_Vitamin_D7	-
LC_Vitamin_D7_derivative	-
LC_Steroid	LC_Sterol_Lipid hasPart some Cyclopenta-a-Phenanthrene_Ring;
LC_C18_steroid_par_estrogen_par__structural_derivative	LC_Steroid hasPart some Estrane ⊓ hasPart some (Alkenyl_Group or Carboxylic_Acid_Ester_Group or Ketone or Alcohol or Ether) ⊓ hasPart some Methyl; hasPart only (Estrane or Alkenyl_Group or Carboxylic_Acid_Ester _Group or Ketone or Alcohol or Ether or Methyl);
LC_C19_steroid_par_androgen_par__structural_derivative	LC_Steroid hasPart some Androstane ⊓ hasPart some (Alkenyl_Group or Carboxylic_Acid_Ester_Group or Ketone or Aldehyde or Alcohol or Ether) ⊓ hasPart some Methyl; hasPart only (Androstane or Alkenyl_Group or Carboxylic_Acid_ Ester_Group or Ketone or Aldehyde or Alcohol or Ether);
LC_C21_steroid_structural_derivative	LC_Steroid

	hasPart some Pregnane ⊓ hasPart some (Acyl_Chain or Acyl_Ester_ Chain or Alkenyl_Group or Alkynyl_Group or Carboxylic_Acid_ Ester_Group or Carboxylic_ Acid or Ketone or Aldehyde or Alcohol or Cycloalkane_Group) ⊓ hasPart some Ethyl; hasPart only (Pregnane or Acyl_Chain or Acyl_Ester_Chain or Alkenyl_Group or Alkynyl_Group or arboxylic_Acid_Ester_Group or Carboxylic_Acid or Ketone or Aldehyde or Alcohol or Cycloalkane_Group);
LC_Steroid_conjugate	LC_Sterol_Lipid hasPart some Cyclopenta-a-Phenanthrene_Ring ⊓ hasPart some (Amino_Acid or Organic_Sulfur_Group or Glucuronic_acid);
LC_Glucuronide	LC_Steroid_conjugate hasPart some Glucuronic_acid ⊓ hasPart some (Estrane or Androstane or Pregnane) ⊓ hasPart some (Methyl or Ethyl); hasPart only (Glucuronic_acid or Estrane or Androstane or Pregnane or Methyl or Ethyl);
LC_Glycine_conjugate	LC_Steroid_conjugate hasPart some Glycine ⊓ hasPart some Cholane ⊓ hasPart some Penthyl_Derivative; hasPart only (Glycine or Cholane or Penthyl_Derivative;
LC_Sulfate	LC_Steroid_conjugate hasPart some Sulfuric_Acid_derivative_Group ⊓ hasPart some (Estrane or Androstane or Pregnane or Cholestane) ⊓ hasPart some (Methyl or Ethyl or Iso-Octyl_Derivative);

	hasPart only (Sulfuric_Acid_derivative_Group or Estrane or Androstane or Pregnane or Cholestane or Methyl or Ethyl or Iso-Octyl_Derivative);
LC_Taurine_conjugate	LC_Steroid_conjugate hasPart some Taurine ⊓ hasPart some Cholane ⊓ hasPart some Penthyl_Derivative; hasPart only (Taurine or Cholane or Penthyl_Derivative);
LC_Sterol	LC_Sterol_Lipid hasPart some Cyclopenta-a-Phenanthrene_Ring;
LC_Brassinolide_structural_derivative	LC_Sterol hasPart some Campestano-lactone ⊓ hasPart some (Alcohol or Methyl or Carboxylic_Acid_Ester_Group) ⊓ hasPart some Methyl-Iso-Octyl; hasPart only (Campestano-lactone or Alcohol or Methyl or Carboxylic_Acid_Ester_Group or Methyl-Iso-Octyl);
LC_Brassinolide	-
LC_Brassinolide_derivative	-
LC_Bufanolide_structural_derivative	LC_Sterol hasPart some Bufanolide ⊓ hasPart some (Alcohol or Carboxylic_Acid_Ester_Group or Alkenyl_Group); hasPart only (Bufanolide or Alcohol or Carboxylic_Acid_Ester_Group or Alkenyl_Group);
LC_Bufanolide	-
LC_Bufanolide_derivative	-

LC_C24-propyl_sterol_structural_derivative	LC_Sterol hasPart some Cholestane ⊓ hasPart some (Alcohol or Propyl or Alkenyl_Group) ⊓ hasPart some Propyl-Iso-Octyl; hasPart only (Cholestane or Alcohol or Propyl or Alkenyl_Group or Propyl-Iso-Octyl);
LC_C24-propyl_sterol	-
LC_C24-propyl_sterol_derivative	-
LC_Calysterol_structural_derivative	LC_Sterol hasPart some Cyclo-stigmastane ⊓ hasPart some Cyclopropene ⊓ hasPart some (Alcohol or Alkenyl_Group) ⊓ hasPart some Iso-Octyl_Derivative; hasPart only (Cyclo-stigmastane or Cyclopropene or Alcohol or Alkenyl_Group or Iso-Octyl_Derivative);
LC_Calysterol	-
LC_Calysterol_derivative	-
LC_Cardanolide_structural_derivative	LC_Sterol hasPart some Cardanolide ⊓ hasPart some (Alcohol or Carboxylic_ Acid_Ester_Group or Alkenyl_Group); hasPart only (Cardanolide or Alcohol or Carboxylic_Acid_ Ester_Group or Alkenyl_Group);
LC_Cardanolide	-
LC_Cardanolide_derivative	-
LC_Cholesterol_structural_derivative	LC_Sterol hasPart some Cholestane ⊓ hasPart some Iso-Octyl_Derivative ⊓ hasPart some (Alcohol or Ketone or Alkenyl_Group or Epoxy);

	hasPart only (Cholestane or Iso-Octyl_Derivative or Alcohol or Ketone or Alkenyl_Group or Epoxy);
LC_Cholesterol	-
LC_Cholesterol_derivative	-
LC_Cholesteryl_ester	LC_Sterol hasPart some Cholestane ⊓ hasPart some (Nitro_Group or Hydroperoxide or Carboxylic_Acid_Ester_Group or Alkenyl_ Group) ⊓ hasPart some Acyl_Chain ⊓ hasPart some Iso-Octyl_Derivative; hasPart only (Cholestane or Nitro_Group or Hydroperoxide or Carboxylic_Acid_Ester_Group or Alkenyl_Group or Acyl_Chain or Iso-Octyl_Derivative);
LC_Cycloartanol_structural_derivative	LC_Sterol hasPart some Cyclo-lanostane ⊓ hasPart some (Alcohol or Cyclopropane or Alkenyl_Group) ⊓ hasPart some Iso-Octyl_Derivative; hasPart only (Cyclo-lanostane or Alcohol or Cyclopropane or Alkenyl_Group or Iso-Octyl_Derivative);
LC_Cycloartanol	-
LC_Cycloartanol_derivative	-
LC_Ergosterol_structural_derivative	LC_Sterol hasPart some (Ergostane or Campestane) ⊓ hasPart some (Alcohol or Methyl or Epoxy or Ketone or Alkenyl_Group) ⊓hasPart some Methyl-Iso-Octyl; hasPart only (Ergostane or Campestane or Alcohol or Methyl or

	Epoxy or Ketone or Alkenyl_Group or Methyl-Iso-Octyl);
LC_C24-methyl_ergosterol_derivative	-
LC_Ergosterol	-
LC_Furospirostanol_structural_derivative	LC_Sterol hasPart some Furospirostane ⊓ hasPart some (Alcohol or Ether or Alkenyl_Group); hasPart only (Furospirostane or Alcohol or Ether or Alkenyl_Group);
LC_Furospirostanol	-
LC_Furospirostanol_derivative	-
LC_Furostanol_structural_derivative	LC_Sterol hasPart some Furostane ⊓ hasPart some (Alcohol or Carboxylic_ Acid_Ester_Group or Alkenyl_Group or Glucose) ⊓ hasPart some Iso-Penthyl_Derivative; hasPart only (Furostane or Alcohol or Carboxylic_Acid_Ester _Group or Alkenyl_Group or Glucose or Iso-Penthyl_Derivative);
LC_Furostanol	-
LC_Furostanol_derivative	-
LC_Gorgosterol_structural_derivative	LC_Sterol hasPart some Gorgostane ⊓ hasPart some (Cyclopropane or Alcohol or Alkenyl_Group) ⊓ hasPart some Cyclopropanoyl-Iso-Octyl; hasPart only (Gorgostane or Cyclopropane or Alcohol or Alkenyl_Group or Cyclopropanoyl-Iso-Octyl);
LC_Gorgosterol	-

LC_Gorgosterol_derivative	-
LC_Solanidine_structural_derivative	LC_Sterol hasPart some (Solanidane or Spirosolane) ⊓ hasPart some (Alcohol or Secondary_Amine or Alkenyl_Group); hasPart only (Solanidane or Spirosolane or Alcohol or Secondary_Amine or Alkenyl_Group);
LC_Alkaloid_derivative	-
LC_Solanidine	-
LC_Solanidine_derivative	-
LC_Spirostanol_structural_derivative	LC_Sterol hasPart some Spirostane ⊓ hasPart some (Ether or Alcohol); hasPart only (Spirostane or Ether or Alcohol);
LC_Spirostanol	-
LC_Spirostanol_derivative	-
LC_Stigmasterol_structural_derivative	LC_Sterol hasPart some (Stigmastane or Poriferastane) ⊓ hasPart some (Ethyl or Alcohol or Alkenyl_Group) ⊓ hasPart some Ethyl-Iso-Octyl; hasPart only (Stigmastane or Poriferastane or Ethyl or Alcohol or Alkenyl_Group or Ethyl-Iso-Octyl);
LC_C24-ethyl_stigmasterol_derivative	-
LC_Stigmasterol	-
LC_Stigmasterol_derivative	-

CLASSES OF LIPID THAT WERE ARBITRARILY CREATED TO ASSERT OBO COMPLIANCE IN LiCO DO NO HAVE EQUIVALENT CLASS IN THE ORIGINAL LIPIDMAPS HIERARCHY. THEREFORE, WE DO NOT PROVIDE DL DEFINITION FOR THESE CLASSES OF LIPID. IN ADDTION TO THAT, WE ALSO DO NOT PROVIDE DL DEFINITION FOR THE CLASS POLYKETIDE AS IT IS BEYOND THE SCOPE OF THE THESIS.