EPA Public Access

Author manuscript

ACS Sustain Chem Eng. Author manuscript; available in PMC 2020 July 06.

About author manuscripts

Submit a manuscript

Published in final edited form as:

ACS Sustain Chem Eng. 2017 December 6; 6(2): 1961–1976. doi:10.1021/acssuschemeng.7b03379.

Toward Automated Inventory Modeling in Life Cycle Assessment: The Utility of Semantic Data Modeling to Predict Real-World Chemical Production

Vinit K. Mittal[†], Sidney C. Bailin[‡], Michael A. Gonzalez[§], David E. Meyer[§], William M. Barrett[§], Raymond L. Smith[§]

[†]Oak Ridge Institute of Science and Education (ORISE), Hosted by U.S. Environmental Protection Agency, Office of Research and Development, 26 West Martin Luther King Drive, Cincinnati, Ohio 45268, United States

[‡]Knowledge Evolution, Inc., 1748 Seaton Street NW, Washington, D.C. 20009, United States

[§]U.S. Environmental Protection Agency, National Risk Management Research Laboratory, 26 West Martin Luther King Drive, Cincinnati, Ohio 45268, United States

Abstract

A set of coupled semantic data models, i.e., ontologies, are presented to advance a methodology toward automated inventory modeling of chemical manufacturing in life cycle assessment. The cradle-to-gate life cycle inventory for chemical manufacturing is a detailed collection of the material and energy flows associated with a chemical's supply chain. Thus, there is a need to manage data describing both the lineage (or synthesis pathway) and processing conditions for a chemical. To this end, a Lineage ontology is proposed to reveal all the synthesis steps required to produce a chemical from raw materials, such as crude oil or biomaterials, while a Process ontology is developed to manage data describing the various unit processes associated with each synthesis step. The two ontologies are coupled such that process data, which is the basis for inventory modeling, is linked to lineage data through key concepts like the chemical reaction and reaction participants. To facilitate automated inventory modeling, a series of SPARQL queries, based on the concepts of ancestor and parent, are presented to generate a lineage for a chemical of interest from a set of reaction data. The proposed ontologies and SPARQL queries are evaluated and tested using a case study of nylon-6 production. Once a lineage is established, the process ontology can be used to guide inventory modeling based on both data mining (top-down) and simulation (bottom-up) approaches. The ability to generate a cradle-to-gate life cycle for a

Terms & Conditions

Electronic Supporting Information files are available without a subscription to ACS Web Editions. The American Chemical Society holds a copyright ownership interest in any copyrightable Supporting Information. Files available from the ACS website may be downloaded for personal use only. Users are not otherwise permitted to reproduce, republish, redistribute, or sell any Supporting Information from the ACS website, either in whole or in part, in either machine-readable form or any other form without permission from the American Chemical Society. For permission to reproduce, republish and redistribute this material, requesters must process their own requests via the RightsLink permission system. Information about how to use the RightsLink permission system can be found at <http://pubs.acs.org/page/copyright/permissions.html>.

The authors declare no competing financial interest.

chemical represents a key achievement toward the ultimate goal of automated life cycle inventory modeling.

Graphical Abstract

Keywords

Semantic data model; Lineage; Process; Ontology; Life cycle assessment; Life cycle inventory

Introduction

As chemicals are developed and enter commerce, there is an increasing need to evaluate the impacts on human health and the environment associated with their manufacture, use, and disposal. When a chemist designs and develops a new molecule or synthesis route (i.e., development of chemical alternatives) or stakeholders want to know about the effects of an existing chemical in production, they can turn to life cycle assessment (LCA) as an approach for holistic chemical management.(1, 2) LCA is an established method for evaluating the cradle-to-grave effects of chemicals and chemical products.(3, 4) Often, performing an LCA is a time-consuming and data-intensive task, requiring either the extensive use of proxy data or a lengthy amount of time to identify and analyze the cradle-to-grave processes.

Life cycle inventory (LCI) modeling, which describes the material and energy flows throughout the life cycle of a product or service, is the most time-consuming and resource-intensive activity.(5) To meet the need for accurate, fast, and transparent LCI methods, researchers have recently developed top-down and bottom-up approaches for estimating LCIs.(6, 7) The bottom-up approach is process specific and estimates material and energy inputs as well as releases.(7) The top-down approach uses data mining as a tool for identifying, extracting, and processing open-access EPA facility-level release data, with a proposed method to standardize and eventually automate LCI data collection for chemical manufacturing.(6) Both approaches focus on modeling a single unit process, whereas the overall life cycle is a collection of these individual processes. If one could first identify the chemicals, reactions, and processes within the life cycle, it would be possible to use both the top-down and bottom-up inventory techniques to potentially automate LCI modeling and reduce the intensive nature of LCA. Therefore, this contribution focuses on the advancement of a semantic-based methodology for establishing cradle-to-gate life cycle data

requirements. Such a methodology should be flexible and able to utilize and reconcile the top-down and bottom-up approaches to obtain the best and most accurate possible inventory.

Lineage of a Chemical and Its Underlying Processes

Of the four stages of the life cycle as defined in ISO 14040:2006, material acquisition and product manufacturing are perhaps the hardest to model due to the numerous processing steps they entail.⁽³⁾ Products are most often a combination of chemicals, and so efficiently modeling these chemical precursors is key to improving inventory modeling. The so-called cradle-to-gate life cycle inventory of each chemical describes the synthesis pathway of the chemical from raw starting materials to its desired usable form. Embodied in the pathway are the many relations that are involved in chemical reactions, including the different roles that chemicals can play in a reaction and the resulting relationships among the chemicals themselves. This structure is not unlike a family tree and so is defined here as the chemical lineage (see Figure 1). One can think of the product of interest (and any potential coproduct) as being the child of parents, which also have parents (called grandparents in Figure 1). The grandparents also have parents, and so on, until a complete lineage unfolds, from knowing only each successive set of parents, eventually going back to original chemicals which are analogous to first ancestors (i.e., the “Adam and Eve”). In addition to chemical relationships, the pathway contains knowledge of the operating conditions and parameters that guide a chemical process. In describing a particular example, FAME (fatty acid methyl esters; biodiesel) is the offspring of parents triglyceride and methanol. Proceeding along one possible lineage, triglyceride’s parent may be soybean biomass, which at the same time would be the grandparent of FAME.⁽⁸⁾ Methanol’s parents are the components of synthesis gas, which can be derived from many sources, such as CH₄ (methane), coal, or biomass, such as municipal solid waste, with this lineage pattern continuing all the way back to the extraction of elementary materials like crude oil and coal.^(9–11) If this knowledge is combined with details about the production processes at each step, it would then be possible to model a cradle-to-gate LCI for FAME. This coupling of chemistry and production knowledge highlights the need to decompose inventory modeling into two key concepts, “lineage”, which is qualitative, and “process”, which is quantitative, each comprising their own specific data needs. Therefore, the challenge of this work is consistently defining and relating these core concepts and data needs in a manner that will be compatible with the broader themes of LCA and LCI.

Semantic Data and Predictive Chemistry

The need to draw inferences from chemical processing data and predict a network, or pattern, of relationships suggests a methodology built on a semantic approach to data management, and linked open data (LOD) may be the most appropriate.⁽¹²⁾ In LOD, the data schema—the network of possible roles and relations—is specified in an ontology. Ontologies are concise, machine-processable models of a domain of discourse (in this case, chemistry). They provide a common language for automated applications to share information, and in this way support the objectives of LOD. Bridging between different ontologies enables multiple concepts to be joined to obtain richer and more powerful data

interpretations. One example of such modeling in science is the area of predictive chemistry. (13)

Predicting the outcomes of chemical reactions has always been an intriguing task. Several attempts,(14) mostly pertaining to organic chemicals, have been made to “teach” a computer to understand how and why a chemical reaction proceeds. Based on these established and predefined rules or algorithms, a computer can predict the products and byproducts of unknown chemical reactions, as long as the required knowledge is present in the hardcoded rules or algorithms. Some of these efforts are based on knowledge management, where existing information has been classified according to either functional group, named reactions, or databases of known reactions.(15–17) This has been facilitated by the development of a chemical ontology, for example, the Name Reaction Ontology RXNO(16) for chemical reactions. The results are classifications of the named organic reactions and their definition which can now be obtained by using an ontology query language, SPARQL (a recursive acronym for SPARQL Protocol and RDF Query Language), and through tools to represent the results graphically, e.g., Protégé.(18, 19) These ontologies are capable of predicting products of a reaction, but only for those reactions that exist in the database, which for the RXNO ontology are focused on predetermined sets of reactions.

Other efforts that have been undertaken to predict the outcome of a chemical reaction have met with some level of success using machine learning, neural networks, or rigorous algorithms based on the hard and fast rules of organic chemistry.(20–22) These methods only predict the products of a chemical reaction and provide no information on the preceding synthesis pathways, reaction conditions, or the ability to extrapolate or cross reference to other similar chemical synthesis routes. However, it is possible to create synthesis paths using either manual searches or by using semiautomated tools such as Reaxys.(23) Given the multiple routes by which organic chemicals can be synthesized, for a 5-step synthesis, there will be up to 10^{19} synthesis pathways, and it will be impossible to track all synthesis paths and more importantly identify the optimal or industrially relevant one.(24)

An approach that considers both the prediction of chemical reactions and their synthesis path(s) is Chematica, where an algorithm traverses a network of about 10 million organic chemistry reactions to fetch the synthesis path of a chemical. The synthesis path selected can be optimized based on a cost criterion, which is a combination of labor and starting material cost, or popularity of intermediate chemicals or yield.(14, 24) Despite several attractive features like speed, the extent of the database, and the ability to find numerous possible synthesis pathways, Chematica is unfortunately not sufficient for the purposes of LCA. For example, a result may generate a number of synthesis paths containing intermediate chemicals that may not be commercially viable. The fact that the synthesis paths are sorted based on internally assigned costs and yields is at odds with LCA, which often seeks to capture the real-world baseline for chemical and product manufacturing. More importantly, these previous efforts to predict the synthesis of a chemical also failed to capture other key aspects of an LCI, such as solvents, catalysts, promoters, wastes, and emissions.

A few research groups have proposed the development and use of ontologies and LOD for LCA. Earthster’s ECO ontology was one of the first core domain ontologies for LCA.(25)

CASCADE, another ontology project, was designed to interpret LCA information by using existing product data management.(26) The work by Edrisi et al. focused on the development of an ontology for information and knowledge sharing for an enterprise decision-making support tool.(27) A case study for a production process demonstrated the use of ontology in simultaneous consideration of environmental effects and an economic indicator. Bertin et al. applied an ontology to model and illustrate life cycle inventory data for U.S. electricity production,(28) and the work of Takhom et al. involved designing and constructing an ontology schema to store LCI data.(29) These examples primarily involved collecting and storing available LCI data and associated data quality indicators (DQI) for each individual chemical synthesis process. Zhang et al. developed an ontology for LCI modeling for the product life cycle and a semantic representation for LCA.(30) All of the LCI ontologies described rely on a gate-to-gate approach.

The present work expands upon previous efforts to develop ontologies for predictive chemistry and LCA by focusing on the creation of a chemical production network that integrates the concepts of lineage and process to enable cradle-to-gate inventory modeling. Therefore, the objectives of this paper are (1) develop a Lineage ontology capable of linking a chemical with its parents, grandparents, and so on, as well as other chemical needs such as catalysts and solvents, to establish a lineage; (2) develop a Process ontology by understanding the synthesis conditions for each step of a lineage and linking this information with typical LCI data types; (3) evaluate the functionality of the Lineage and Process ontologies by applying them to a case study chemical; (4) define data queries that can generate a lineage from a chemical reaction data set and test the queries on the case study chemical; and (5) qualitatively demonstrate how a Process ontology can be linked with inventory modeling tools, including both bottom-up (unit operation look-up tables and simulation) and top-down (data mining) approaches, to provide detailed LCI and improve the availability and utility of LCA.

Methodology

Ontology Modeling

An ontology is a model or specification of the kinds of things that exist in a domain of discourse, the attributes of these things, and the relationships between them. The Lineage and Process ontologies were developed in the Web Ontology Language (OWL).(31) In OWL, kinds of things are modeled as *classes*, and attributes and relationships are modeled as *properties*; the top-level class is *Thing*, and domain-specific classes are defined to be subclasses of *Thing*. To the extent possible, an ontology should build upon existing ontologies in order to avoid replicating work: replication muddies the range of available ontologies for future work, necessitating choices between similar ontologies, the differences and trade-offs between which may not be readily apparent. An ontology builds upon another one by importing it. The classes and properties of the imported ontology are then available to be referenced in the importing ontology. Thus, to determine the appropriate classes and properties of the Lineage ontology, we first examined existing chemical ontologies(32, 33, 16) and, from this information, determined that the existing ontologies did not provide an explicit, generic structure to formally describe the various components of a chemical

reaction. Such formal descriptions are needed in order to aggregate reactions into lineages by matching the products of one reaction with the reactants of another. We therefore decided to define, explicitly, the primary classes Chemical and ChemicalReaction, and we then focused the ontology design effort on determining the structural information—supporting classes and properties—that are required for the lineage computations. To aid in this determination, we populated the Chemical class with instances, i.e., specific chemicals, and similarly the ChemicalReaction class was populated with a small set of reactions, for which the reactants and products were specified.

The ontology design process then became a matter of deciding how best to represent reactants, products, and other chemicals that participate in a reaction, such as catalysts, solvents, and byproducts. As with any ontology modeling, a series of ontology design alternatives were considered. The alternatives we considered are standard choices that must be made in developing any substantial ontology, although the particulars vary with each ontology. The goals in evaluating these alternatives are *competence*—the ability to provide answers to the questions that will be asked of the ontology—and *maintainability*, the ability of the ontology to evolve as more information is added or new requirements appear. For the Lineage ontology, the first key choice was whether to depict subsets of chemicals via subclasses or via a controlled vocabulary. A controlled vocabulary is, by convention, a class within the ontology whose instances are the terms contained in the vocabulary. To see how this choice appears in elaborating the Lineage ontology, we could, for example, define Acid as a subclass of Chemical, since every acid is a chemical. Alternatively, we could define a vocabulary for characterizing chemicals, in which case Acid would be a term within the vocabulary, and a particular chemical would be described as an acid by linking the chemical (an instance of class Chemical) to this vocabulary term (an instance of the vocabulary class). These alternatives are illustrated in Figures S1 and S2.

Subclasses are the most obvious choice because describing more specific kinds of things (in this case, chemicals) is precisely what subclasses in OWL are meant to do. However, a controlled vocabulary is more easily sustained in the sense that when a new term is created, it is simply added to the vocabulary without changing the class-subclass structure of the ontology. Also, if subclasses are actually needed for some competency, OWL allows us to define them via the vocabulary terms. For example, a subclass Acid can be defined in OWL as *the set of all Chemicals that are assigned the vocabulary term Acids*. In this way, the appearance of the subclass Acid is through a definition rather than a declaration. This reduces the potential impact of any changes to the classification.

We decided to implement a design compromise. We represent the highest level grouping using subclasses, and under each such subclass we use a controlled vocabulary. This approach keeps the class structure of the ontology simple, while facilitating the classification of new chemicals that may be added to the ontology over time. Use of the controlled vocabulary implies, however, that we cannot take the classification down to a more detailed level, as vocabularies are flat sets of terms.

The second key design choice was whether to use classes, a controlled vocabulary, or properties to represent the different roles that a chemical can play in a reaction (e.g.,

reactant, solvent, catalyst, product, byproduct). Here we concluded that the benefits of a controlled vocabulary are minimal because the number of roles is limited and fixed. The benefit of a controlled vocabulary appears when the vocabulary is expected to evolve. Instead, we decided to use a combination of classes and properties. We created a property to represent each role, for example, hasReactant, hasSolvent, etc. A ReactionParticipant class was then added as a means to refer to all chemicals that are participating in a given reaction. Using this structure, a ReactionParticipant participates in a Reaction through some role.

The Process ontology was developed, similarly to the Lineage ontology, by first identifying the information that distinguishes a process from a chemical reaction, or that provides context for a chemical reaction within a process. This information was then allocated to a set of subsystems (Reactor, Separation, Storage, Treatment, etc.). The Lineage and Process ontologies were then bridged by comparing the data types within each and identifying data that were identical, similar, or related.

Chemical Case Study

A case study provides both a means to evaluate the effectiveness of the ontologies and a training set to guide development of the lineage generation queries. Nylon-6, a widely used polymer, was identified as an example for this study. Nylon-6 finds application in a variety of objects of day-to-day use, for example in gears, fittings, bearings, bristles for toothbrushes, threads, ropes, etc. Glass fiber grade nylon-6 can be used as a halogen-free fire retardant. The complete lineage of nylon-6 is appropriate for this study as it has multiple steps (demonstrates complexity) going back to the elementary extraction of either coal or crude oil, and it possesses multiple intermediate commodity chemicals such as toluene, cumene, phenol, and cyclohexanone.

Given nylon-6 is produced directly from caprolactam, a review of literature describing the production of caprolactam and its precursors was performed to manually generate the lineage for nylon-6 (i.e., a training set for future comparison). In addition, the caprolactam synthesis step was selected for this case study for developing and evaluating the Process ontology and required the collection of additional data describing the production process. Information regarding the caprolactam manufacturing process was obtained from Ullman's Encyclopedia of Chemical Technology.⁽³⁴⁾ Ullman's describes a number of processes for manufacturing caprolactam, stating that 90% is produced via a Beckmann rearrangement of cyclohexanone oxime, which is obtained by reacting cyclohexanone with hydroxyl amine. Production rates for caprolactam were obtained from the National Institute of Health's (NIH) Toxnet Hazardous Substance Database (HSDB).⁽³⁵⁾ Additional information on the synthesis of the intermediate cyclohexanone oxime was obtained from Internet web searches. A search of US patents identified two relevant patents: *Process for the Preparation of Cyclohexanone Oxime* (US 3,991,115) and *Process for the Preparation of Lactams* (US 3,914,217). These patents provide detailed descriptions of the reaction processes including reactor type, operating temperatures and pressures, reactor feed flows and concentrations, reaction yields/conversions, and reactor output flows and concentrations, all of which can be used during Process ontology data population and inventory modeling.

Once the full synthesis lineage of nylon-6 was established, Protégé was used to populate the classes of the Lineage ontology with instances from each synthesis step, thereby creating a Lineage database. Similarly, process related details were provided as instances to the Process ontology. Evaluation of the ontologies was based on the ability of each ontology to properly store and describe the data instances from the case study. SPARQL lineage generation queries were then applied to the synthesis data stored as instances of the Lineage ontology in the Lineage database in order to generate a predicted lineage for nylon-6. The success of the queries could be judged by how closely the predicted lineage matched the training set data obtained from the literature review.

Results and Discussion

Lineage Ontology

The basic structure (schema) of the proposed Lineage ontology, hereafter referred to as Lineage, is shown in Figure 2. Four broad classes are defined as subclasses of the class Thing (T in Figure 2). Two of the classes are the previously mentioned Chemical and ChemicalReaction classes. A third class, ReactionParticipant, is defined to classify chemicals according to how they participate in a reaction. A fourth class, ChemicalFamily, is used to classify chemicals according to their family origin (organic or inorganic) and eventually their structure, where a controlled vocabulary is used to describe things like functional groups. These terms are all instances of ChemicalFamily (in fact, they are instances of the Organic and Inorganic subclasses of ChemicalFamily). The Chemical class itself describes the use or behavior of the chemical and was defined to capture basic chemical attributes and chemical reactions from a chemist's point of view. Such information will also be beneficial for inventory modeling in cases where data based on a chemical read-across can serve as a proxy until more precise data is compiled. The precise definitions of the four broad classes are:

Chemical

A class whose instances are specific chemicals (e.g., caprolactam). A Chemical is linked to a ChemicalReaction as a ReactionParticipant, which allows the role of the chemical in the reaction to be stated. Use of reaction participant as an explicit concept in the ontology allows the chemical to perform different roles in different reactions. The property belongsToFamily connects a chemical to a family. Thus, for example, in Figure 2, the connection between Chemical and Acid is via the property belongsToFamily.

ChemicalReaction

A class whose instances are specific chemical reactions (e.g., acetic acid production reaction), with further description in terms of its ReactionParticipants. Chemical reactions in the ChemicalReaction class can be associated with unique reaction ID's that can be used to manage chemical reaction records in a data set and distinguish between similar reactions with different reaction conditions.

ReactionParticipant

A reaction participant is a chemical in the context of an identified reaction. This abstraction, which is not present in other chemical ontologies referenced herein, allows Lineage to define the role of a chemical in a reaction through a set of properties, including `isReactantOf`, `isCatalystOf`, `isSolventOf`, `isProductOf`, etc. Through these properties, a chemical can be assigned a single role or multiple roles within an identified chemical reaction. The unique stoichiometry for that chemical, playing a particular role in a particular reaction, can then be specified in the ontology.

ChemicalFamily

Chemicals can be categorized into families (e.g., organic, inorganic) based on their family origin. Controlled vocabularies are the most effective way to achieve this goal. Each such vocabulary is represented as a subclass of the `ChemicalFamily` class. Currently, the Lineage ontology utilizes two such subclasses: `Organic` and `Inorganic`. Instances of the `Inorganic` class represent families of inorganic chemicals, namely `Metallic` and `Organometallic`. Instances of the `Organic` class represent families of chemicals according to the functional group present. In the case of a chemical with multiple functional groups, the IUPAC convention of functional group priority will be followed, which is similar to the IUPAC nomenclature convention followed by chemists.⁽³⁶⁾ Functional group classification is an important feature of the Lineage ontology as it can serve as a potential bridging point to existing reaction ontologies (e.g., RXNO) that will allow reaction data managed with those schemas to be incorporated into a Lineage database.

Although classes provide a means to group and sort data, the ability to query and make inferences lies in understanding how the classes are related. In OWL, these relationships are known as “object properties” and can be used to link one object (e.g., a `Chemical`, a `Reaction`, etc.) to another, rather than to a data value (such as a text string or a number). These relationships are key for the current goal of inferring a lineage because without them the relationship of each upstream chemical to the product would have to be explicitly defined. Descriptions of the object properties defined in Lineage are provided in Table 1.

In addition to relationships to other objects, objects can be related to data. In OWL, these relationships are called data properties and involve values as strings or numbers as opposed to other objects. The data properties can define the properties of a chemical, its synonyms, special identifiers like Chemical Abstract Service (CAS) number or Simplified Molecular Input Line Entry System (SMILES) and details about a relevant literature source. A list of the data properties (with examples) included in Lineage is provided in Table 2. The ability to store references for the data will be useful when evaluating the quality of LCIs constructed using the data.⁽³⁷⁾ Often times, the knowledge described by data properties is necessary for inventory modeling and impact assessment. A benefit of LOD is that the data may be available from other data sources. Instead of storing the data in a database for lineage modeling, the data can be imported as needed through web services, provided the properties defined in Lineage are present. In this way, a database supporting lineage modeling can be maintained at a more reasonable size. As with other parts of Lineage, the list of data properties can be readily extended as new data needs are identified.

As previously stated, Lineage differs from other reported chemical reaction ontologies. For example, the RXNO ontology classification is based on organic named reactions or their effects on the “skeleton” of the molecule (e.g., Beckmann rearrangement). The database contains definitions of each named reaction and examples; however, several other details important for LCI and LCA such as other reaction participants and operating conditions are missing. Moreover, the RXNO does not include the concept of parent-child relationships between chemicals tracing back to the core elementary chemicals, which is central to the Lineage ontology enabling cradle-to-gate LCA and LCI analysis.

Process Ontology

The Process ontology, hereafter referred to as Process, describes process-related information for the industrial production of each chemical contained in a chemical map. The schema for Process is shown in Figure 3 and a complete listing of Process object properties and data properties are presented in the SI in Figures S3 and S4, respectively. Similar to Lineage, classes were defined in Process to capture the relevant information for modeling the production of a chemical. The basic chemistry underlying the production of a chemical remains the same, which means Process and Lineage should be linked wherever possible to share common data. Although classes can be defined in many possible ways, this desire to bridge Lineage and Process led to a class structure similar to Lineage, with two Process classes describing chemical reactions and inputs and outputs. However, the fact that industrial-scale chemical manufacturing often has more complex requirements than bench-scale synthesis necessitated the creation of new classes, such as Subsystem (for equipment), to capture the additional information. The main classes are:

InputsAndOutputs

This class is similar to ReactionParticipant in Lineage. A new class was defined to represent additional flows in and out of a process (inputs and outputs) besides those pertaining specifically to the chemical reaction. Subclasses of InputsAndOutputs are *Reactant*, *Catalyst*, *Promoter*, *Solvent*, *Buffer*, *Product*, *Coproduct*, *Waste*, *Air_emission*, *Land_release*, *Water_discharge*, and *Inert*.

ChemicalReaction

This builds on the ChemicalReaction class in Lineage. In a process plant there are additional intermediate and side reactions which are important for designing unit process equipment. Therefore, a reactor in Process can have multiple ChemicalReactions to account for this.

Chemical

This class directly bridges to the Chemical class in Lineage.

Subsystem

Subsystem represents different unit processes (e.g., reaction, separation, storage, etc.) and their required equipment within a production process. For example, the reactor subsystem can be a tank, tube, or reactive distillation column. Instances of the subclass will contain the equipment used for a particular process. For example, a tube reactor used in a process will

have an instance of *Tube_type_1*, and the properties of this instance will define the characteristics of the reactor used for that particular process. The numeric designation at the end of the reactor name is necessary as it is possible to have more than one instance of each subclass. For example, a reaction may actually be performed in a series of reactors, with each reactor constituting a separate instance of *Reactor_system*.

A key feature of *Process* is the recognition that industrial production of a chemical involves more than just the reaction step. The product must often be separated and recovered while dealing with regulatory compliance for air emissions and water discharges. *Process* accounts for this through the *Subsystem* class, which contains subclasses of basic unit processes, such as storage, pretreatment, reaction, separation, pollution abatement (or treatment), ancillary production of utilities (electricity, steam, water, and refrigeration), and additional miscellaneous processes. Similarly, instead of directly adopting the *ReactionParticipant* class from *Lineage*, the broader class *InputsAndOutputs* was created to allow more flexibility for specifying the role of a chemical in the various unit processes. For example, a coproduct of a bench-scale reaction may be considered as a waste at the industrial scale and require a treatment system to handle it. Essentially, a chemical reaction and its participants in *Lineage* can be connected to multiple unit operations in *Process* by defining appropriate relationships.

Similar to *Lineage*, object properties were defined in *Process* to establish relationships between instances of *Chemical*, *ChemicalReaction*, *InputsAndOutputs*, and *Subsystem* in the *Process* ontology. A representative sample of these properties are shown in Table 3 because the full set is too large to be included here (see Figure S3 in the SI). The data for the example used in Tables 3 and 4 correspond to bottom-up analysis of acetic acid production given in the work of Smith et al.⁽⁷⁾

Data properties were defined to specify equipment and other process-specific parameters. Only a representative sample of the full properties are shown in Table 4 because the complete list is too large to be included here (see Figure S4 in the SI). In contrast to *Lineage*, there are a larger number of data properties for *Process* as a result of all of the necessary information for modeling processes. For example, each unit process associated with the production of a chemical will operate under a particular set of conditions, and some unit processes, such as reactors, are more complex than others. These operating conditions must be defined through data properties. Reactors are complex because their design requires specific information about reactor configuration and material phase(s), desired product(s) and production rates for multioutput reactions, the limiting reactant, and stoichiometric coefficients. Other reaction-dependent production system details are the conversion based on the limiting reactant or reaction kinetics, purge fraction (for venting or bleeding impurities), feed impurity content, and selectivity when competing side reactions are possible. Each property actually involves a quantitative portion and a text portion, included as an annotation of the property, to describe the unit of the value. For example, carbon monoxide, as an instance of both *Chemical* class and the *Air Emission* subclass of *InputsAndOutputs*, has a *fugitive emissions* property of 79.93, where the *fugitive emissions* property has a *unit* of kilograms per year. The inclusion of unit information is necessary to make the actual (numeric) data useable in modeling calculations. The units have been included in the table

for demonstration purposes, but have not yet been defined in the preliminary version of Process. In the future, a units ontology such as Quantity, Unit, Dimension, and Type (QUDT; qudt.org) will be referenced to explicitly capture the intended units. With that change, the annotation of the data property will become an object property of the data property, specifying the data property's unit.

The chemical, reaction, and reaction participant data from Lineage are imported to Process as points of bridging to maintain continuity between the two ontologies. Through the import, Process makes use of the concepts already specified for Lineage. Some classes defined in Lineage are then given counterparts in Process, such that additional process-related context can be specified. Examples of these relationships are identified in Figure 4. Although not shown in the figure, all properties and relationships of classes in Lineage (such as the roles in a ChemicalReaction: reactant, solvent, catalyst, product, byproduct) are available and can be referred to in Process. For example, process:Chemical embeds a lineage:Chemical by means of the property *hasUnderlyingChemical* but includes additional process-related properties (not shown here).

Process differs in emphasis and level of abstraction from other previously reported process ontologies for LCA and manufacturing. For example, the ontology described by Zhang includes more detail than Process, but with a focus more about tools, machines, and their operation than the underlying chemistry.⁽³⁰⁾ Zhang also employs the Process Specification Language (PSL) to formulate rules for inferring new information, or for validating existing information. PSL is powerful, but also complex. The application of Lineage and Process to inventory modeling will use the more manageable SPARQL as the inference engine because it is adequate for this purpose, as demonstrated by the case study.

Applying Lineage and Process to the Case Study Chemical

Based on the literature review, approximately four million tons/y of nylon-6 is produced via the ring opening polymerization of caprolactam.⁽³⁸⁾ Caprolactam provides an example of an intermediate chemical with a number of precursor chemicals and potential manufacturing pathways to demonstrate the proposed ontologies.⁽³⁴⁾ The nylon-6 chemical lineage that was generated by hand from information retrieved during the literature review is shown in Figure 5. A sample of process data for caprolactam is included to demonstrate how each step of the synthesis in the figure can be connected to data defining the accompanying process. This approach to visualizing the chemical lineage was developed such that the primary chemical lineage can be established and its branches expanded to create a cradle-to-gate network, or map. An advantage of the map structure is the ability to represent common reactants as terminal nodes for upstream branches, which can simplify lineage modeling by allowing these modular branches to be reused for future modeling needs. After establishing a baseline chemical lineage for nylon-6, the next step in the evaluation is to populate a database for Lineage with instances (data) of the chemicals and reactions contained within the lineage.

While a large chemical reaction database will require a powerful triple store (LOD) platform, the small size of the case study enabled the Lineage database to be represented in a text-based form using the Turtle syntax (<https://www.w3.org/TR/turtle/>). For each synthesis

step in the chemical lineage, the data were first assigned as instances of the classes and subclasses, with each step constituting an instance of ChemicalReaction. For example, nylon-6 during the final step of its production appears in an instance of the class ReactionParticipant, in the role *product*, for the ChemicalReaction instance of Nylon6Production. The instances are connected through property assertions as shown by the example of Nylon6 (an instance of class Chemical) and Nylon6Production (an instance of class ChemicalReaction) in Figure 6. The sequence of these assertions relating instances of ReactionParticipant and ChemicalReaction provide the basis for inferring a lineage without having to explicitly predefine the connected lineage network in the system. The relationship between manual data entry and automated computation may be summarized by stating that each reaction is entered manually with a specification of its participants and their respective roles. From the individual reaction descriptions, lineages are computed automatically. In the future, if the information concerning reactions and their reactants, solvents, catalysts, and products is available online in a formal representation, it could be directly processed for lineage modeling using Lineage. As noted in the Ontology Modeling methodology section, the existing chemical reaction ontologies that we examined do not provide such information in explicit formal structures, but the source of such information need not be an ontology. It could just as well be a conventional database from which the information could be automatically extracted and processed using Lineage.

Similarly, the process data collected for caprolactam was used to create a database of instances for Process. The process plant design for caprolactam is a two-step process, with each step supporting its own set of instances. The first step involves the conversion of cyclohexanone (parent) and hydroxyl ammonium sulfate (parent) to cyclohexanone oxime (offspring), followed by the conversion of cyclohexanone oxime (now a parent) to caprolactam (offspring). After assigning the instances of class data, relationships are asserted by means of object properties to link the process reaction with its reactants and products, equipment needs, and subsequent separation. The specific processing conditions, such as the reaction conditions, are entered as values of data properties. For example, details of the size, volume, and material of construction may be included in a reactor description. If the reaction product is separated by decanting, an instance Decanter_type_1 must be specified for the *Liquid_separation_system* subclass of Separation_system and the details of the decanter operation captured as values of data properties. The object and data properties defined for the cyclohexanone oxime and caprolactam production processes are shown in Figure 7.

Generating a Chemical's Lineage Using SPARQL Data Queries

The purpose of creating Lineage was to support automated inventory modeling for LCA. As such, the true test of Lineage is the ability to extract a chemical's lineage from reaction data without having to predefine it in a database. The lack of predefinition is crucial to automation because the chemical of interest will change from assessment to assessment and the system must be flexible enough to accommodate this fact. Linked open data is well suited for this purpose because the underlying schema enables property composition, much like composition of functions in mathematics. These compositions are computed through data queries using a language such as SPARQL.

The SPARQL query language operates by specifying patterns of information that are then matched to the database contents. The authors first used an OWL *property chain* to define the relationship *isParentOf* as a composition of reactant and product relationships:

$$\text{Chemical_1 isReactantOf Reaction_1} \rightarrow \text{Caprolactam is ReactantOf Nylon6Production}$$
$$\text{Reaction_1 hasProduct Chemical_2} \rightarrow \text{Nylon6Production has Product Nylon6}$$

implies the following:

$$\text{Chemical_1 isParentOf Chemical_2} \rightarrow \text{Caprolactam is ParentOf Nylon6}$$

Then, given a specific chemical, the *isParentOf* property can be used to retrieve its lineage. This concept was implemented in the construction of the lineage generation SPARQL query as shown in Table 5. For sake of explanation, the query is shown for the example of nylon-6. While the *isReactantOf* and *hasProduct* facts had been entered manually in the ontology, the query selects only those relationships that are part of the lineage for nylon-6.

The query was run in TopBraid Composer Maestro Edition version 5.1.1 (Protégé or any other SPARQL engine could also be used) to generate the chemical lineage shown in Figure 8.(39) The lineage of nylon-6 obtained from the query (Figure 8) is able to reproduce the lineage developed manually in Figure 5. While this may seem fairly straightforward based on the limited training set of data, the queries were able to not only identify the steps for producing nylon-6, they also correctly identified instances where chemicals such as toluene or naptha can be produced from multiple sources. Even more impressive is that by understanding the concept of parent, the queries were able to string together a number of individual reactions and generate a lineage in a matter of moments while the manual lineage took hours, if not days, to produce. This result represents a significant step forward in the pursuit of automated inventory modeling because it will enable a modeling system to be able to generate supply chains for chemicals in the inventory, provided enough reaction instances are available to enable the chemical map to be completed. In addition, the ontology and queries are structured with enough flexibility to handle a single chemical appearing in multiple reactions with different roles and stoichiometries. In cases where options exist for a given step of the lineage, it will be necessary to introduce constraints such as reactant costs or industrial prevalence to resolve these choices. Such constraints have not been developed as of yet and are the subject of continued research in this area. If industrial production is split among a few paths (e.g., a 60/30/10% split for three paths) the resulting lineage can accommodate this split and communicate this information in the chemical map. All of this can be specified in SPARQL as long as the attributes to be considered are specified in the ontology (with the caveat that aggregating such percentages over the map is best done outside of SPARQL). It should be noted that for this initial stage of development, the query results only show main reactants and products. Other reaction participants, including coproducts (or “siblings”), are not included but can be added by modifying the SPARQL query to identify other participants from the same reaction. The chemical for which the

inventory is being developed will be defined as the product and other chemical products will be treated as coproducts until other designations such as waste are identified through process data.

From Lineage and Process to Life Cycle Inventory

The ontological modeling described here provides a means to identify and eventually predict the synthesis route of a chemical, while connecting this knowledge with relevant process information. The question one might ask is how exactly this relates to inventory modeling in LCA. Since the long-term goal of this work is to support an automated inventory modeling system as first described by Cashman et al.,⁽⁶⁾ the utility of the work is best realized by understanding where it fits within the knowledge flow of LCA, as shown in Figure 9.

Inventory modeling in LCA is a focused activity driven by the goods or services being studied and the functional unit defining how they will be studied. For example, one may be modeling the use of a toothbrush as a means to keep teeth clean with the goal of understanding how the potential impacts of incorporating nylon-6 bristles compare with other material choices. Thus, caprolactam becomes a chemical of interest with respect to inventory modeling because its inventory will be necessary to fully model the synthesis of nylon-6 to make the most insightful comparisons. The first step in modeling the inventory is understanding how caprolactam is made. This provides a basis for querying the reaction data stored as LOD within a triple-store graph and conforming to the Lineage schema. The query results define a qualitative life cycle inventory containing the list of upstream ancestors within the primary synthesis chain. In addition to the primary chain ancestors, the additional chemical participants and reaction conditions can be identified for each process link in the chain so as to complete (fill out) the inventory and capture the full cradle-to-gate requirements. During typical inventory modeling, a practitioner will often have access to a library of preexisting unit process data sets, some of which may correspond to target chemicals in the inventory. Performing a gap analysis by comparing the qualitative LCI with such a library will reduce the time and resources necessary for inventory modeling by honing in on true data needs and avoiding duplicative modeling activities.

At this point, quantitative inventory modeling tools are needed to generate actual material and energy flow data for each production step, which is where a unit process triple-store graph conforming to the Process schema plugs into the knowledge flow. For the production step of a given ancestor, the data contained in the Lineage data graph includes key information such as chemical roles and reaction conditions. These begin to define the process through the bridging as described above and as shown in the SI Figure S5. The types of data returned by queries of Process will vary, depending on the type of inventory modeling tool to be used.

The two main classifications for inventory modeling that were described earlier are top-down and bottom-up. The top-down method as defined in Cashman et al. relies on data mining and requires knowledge of facilities that currently produce or would be most likely to produce a chemical of interest.⁽⁶⁾ Such data might include facility references such as EPA's Facility Registry ID that identify a facility and link it with its appropriate identifiers in federal emissions and discharge monitoring systems. Thus, Process may include data that

link to EPA's Facility Registry Service (FRS) web service and enables data calls from EPA databases to acquire environmental release data for facilities manufacturing the chemical of interest.(40) For bottom-up modeling approaches, the process data graph can return the necessary process design specifications to drive process simulation tools like Aspen Plus and CHEMCAD.(41, 42)

An additional novel concept introduced in this contribution is the connection of Process query results with estimated process-specific data obtained from look-up tables that can lead to automated generation of a simplified bottom-up LCI. A look-up table for a unit operation contains relevant LCI data that has been correlated to the results of design equations solved over a range of values for key parameters. Conceptually, this approach is similar to a thermodynamic steam table or a log table, where the user can obtain the required value for a given set of entered information. In order to use a look-up table, a user provides parameters that are dependent on the unit operation (i.e., equipment type) to extract results from the table. Look-up tables are conducive to rapid LCI modeling because they provide approximate results pertaining to equipment design without having to perform complex simulations. For example, the look-up table for a reactor may require parameters such as temperature, pressure, extent of reaction, heat duty, rate of reaction, reaction rate constant, and others, which will be specified as data properties of the reaction. A look-up table or simulation tool can also provide information about the type and size of a reactor that can be used. Data properties of chemicals which might be of interest are density, viscosity, volatility, specific heat, and others. To implement this approach within the context of Figure 9, Process will need to include data properties that govern the look-up tables for the various pieces of equipment, or enough information to calculate these parameters with simple queries. For example, the heat duty required for a heat exchanger can be calculated if the heat capacity and mass of the chemical being heated is known along with the desired temperature change.

With this understanding of the roles Lineage and Process will play in inventory modeling, one can see why they are vital for successfully automating inventory modeling to improve the practicality of LCA as a decision support tool. An additional reason the Lineage and Process ontologies are well suited for this purpose is that they were intentionally designed from the onset with the goal of bridging them to existing ontologies describing LCA and LCI. For example, LCA ontologies define flows into and out of a unit process and classify them as either elementary (to the environment) or technosphere (to another unit process) flows. These flows can also be classified as inputs or outputs, where the chemical of interest is most often defined as the reference flow. The reference flow can easily be bridged to the participant role of product in either the Lineage or Process ontologies. Similarly, an elementary output flow to air can be bridged to a process participant role of hasAirEmission, a discharge to water related by hasWaterDischarge, and release to land by hasLandRelease. This ability to seamlessly bridge the necessary ontology pieces from Lineage and Process with the LCA ontology for practical purposes demonstrates the importance of developing ontologies based on actual data and applications as opposed to pure conceptual modeling.

Expanding the Intended Application: Alternatives Assessment

Another practical purpose for the ontologies connects to efforts from the National Academy of Sciences (NAS) that identified several elements, such as importance of accounting for the entire life cycle of a chemical and its alternatives, exposure to a chemical (source and quantity), social impact, the use of novel toxicological data streams, and applicability of *in silico* computational models and methods to estimate physicochemical information, which are often missing from existing frameworks. Consequently, the NAS produced the “Framework to Guide Assessment of Chemical Alternatives”, a 13-step alternative assessment framework, which aids in a decision-making process when assessing alternatives to a chemical of concern.⁽¹⁾

This decision framework contextualizes the aspects for arriving at potentially safer substitute chemicals with respect to human health and ecological risks. By incorporating aspects from previous guidance and research needs, this framework seeks to provide a greater level of standardization with the goal of providing a more harmonized approach to alternative assessment.^(43, 44) As mentioned previously, there is a great need to include life-cycle thinking, which addresses the potential human health and environmental impacts of a chemical at all of its life cycle stages (resource acquisition, production, use, disposal and/or recycle). Also included are the steps for quantifying a product for performance and economic evaluations.

This extended ontological framework as described in Figure 9 can also provide utility as a method for use in alternative assessment. In order to address this application, chemicals within the Lineage ontology will be assigned to a class relating to a functional group ontology and named reaction ontology. This will enable a computer to infer an alternative's synthesis path and corresponding Lineage based on the presence of a functional group and overall molecular structure, to identify potential chemical parents and thereby provide information for use in an alternative assessment.

Once the lineage and process related details for a particular chemical and its synthesis have been established, other chemicals having similar structures and functional groups that follow the same named organic reaction can be expected to have similar process synthesis. In addition to predicting the lineage of a chemical, the ontology could be queried to obtain information such as the uses of a particular chemical, all of the chemicals which could be used for a particular application, alternative routes to synthesize a chemical, and the fate of a chemical. And finally, bridging these ontologies with the Life Cycle Inventory (LCA-EPA) ontology, will enable us to integrate the data into OpenLCA. The connection to OpenLCA will allow linking information about chemical synthesis and the chemical manufacturing process to their effects on the environment and human health. Integrating all these approaches and tools yields a rapid and facilitated cohesive approach to generating data for the evaluation of potential chemical alternatives.

Future Outlook and Challenges to Overcome

The described approach to inventory modeling based on Lineage and Process can provide a means to automate the development of cradle-to-gate LCI for chemicals of interest, and,

being semantic in nature, will be flexible in bridging with other databases and tools to maximize what can be done with LCA for minimal resources. Some of the challenges that still exist for this research are conceptual, and some technical. Conceptual challenges include defining reaction properties that can adequately represent a chemical reaction for Lineage, obtaining process-related information and, in the case of multiple routes available for synthesis of a chemical, identifying and defining the most relevant industrial pathway. Chemical processes can be complicated, so identifying the level of detail required is a challenge. The major technical challenge is building the necessary Lineage and Process data graphs with sufficient content to drive the predictive queries. It is possible that growing trends in crowd-sourced data collection may provide the most cost-effective means to address this challenge. However, crowd-sourcing will require the development of a web-based data collection system that seamlessly integrates a data entry web form (example shown as Figure S6 in the SI) with the triple-store Lineage graph. In the present paper, Lineage and Process ontologies have been created to show the methodology and basic schema with a few examples. The full potential and usefulness of this approach can be realized only when sufficient data and information have been entered and tested to meet the desired objective. As data for more chemicals and processes are entered, new relationships and properties may have to be defined. Therefore, this approach is open-ended and will evolve with time and additional data.

Finally, it is worth addressing the possible misconception that some may have regarding the feasibility of this approach. One might incorrectly assume that the various data types referenced throughout the discussion will be stored in a single database, making an automated inventory modeling system impractical based on the sheer size of the underlying database. However, this is not an accurate vision of the system because it fails to harness the principles of linked open data. Instead of a centralized data repository, an automated inventory modeling system will incorporate data from multiple sources within the public domain and rely on web services to reduce the amount of data stored within any single system to something quite manageable. In addition, this approach eliminates the need for a single person or entity to oversee data curation and updates, which can be a costly part of data management.

The views expressed in this article are those of the authors and do not necessarily represent the views or policies of the U.S. Environmental Protection Agency. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgment

This research was supported in part by an appointment of V.M. to the Postmasters Research Program at the National Risk Management Research Laboratory, Office of Research and Development, U.S. Environmental Protection Agency (EPA), administered by the Oak Ridge Institute for Science and Education through Interagency Agreement No. DW-89-92433001 between the U.S. Department of Energy and the U.S. EPA.

References

1. National Research Council A Framework to Guide Selection of Chemical Alternatives 2014, DOI: 10.17226/18872
2. National Research Council Sustainability and the U.S. EPA 2011, DOI: 10.17226/13152
3. ISO. Environmental management–Life cycle assessment–Principles and framework; ISO no. 14040; ISO: Geneva, Switzerland, 2006.
4. ISO. Environmental management–Life cycle assessment–Requirements and guidelines; ISO no. 14044; ISO: Geneva, Switzerland, 1 07, 2006.
5. European Commission–Joint Research Centre–Institute for Environment and Sustainability. International Reference Life Cycle Data System (ILCD) Handbook–Specific guide for Life Cycle Inventory data sets, First ed; Publications Office of the European Union: Luxembourg, 3 2010; EUR 24709 EN.
6. Cashman S; Meyer D; Edelen A; Ingwersen W; Abraham J; Barrett W; Gonzalez M; Randall P; Ruiz-Mercado G; Smith R Mining available data from the United States Environmental Protection Agency to support rapid life cycle inventory modeling of chemical manufacturing Environ. Sci. Technol 2016, 50 (17) 9013–9025DOI: 10.1021/acs.est.6b02160 [PubMed: 27517866]
7. Smith R; Ruiz-Mercado G; Meyer D; Gonzalez M; Abraham J; Barrett W; Randall P Coupling Computer-Aided Process Simulation and Estimations of Emissions and Land Use for Rapid Life Cycle Inventory Modeling ACS Sustainable Chem. Eng 2017, 5 (5) 3786–3794DOI: 10.1021/acssuschemeng.6b02724 [PubMed: 30245942]
8. Ruiz-Mercado G; Gonzalez M; Smith M Sustainability Indicators for Chemical Processes: III. Biodiesel Case Study Ind. Eng. Chem. Res 2013, 52, 6747–6760DOI: 10.1021/ie302804x
9. Wittcoff HA; Reuben BG; Plotkin JS Ind. Org. Chem 2004, DOI: 10.1002/0471651540
10. Wise HE; Fahrenthold PD Predicting priority pollutants from petrochemical processes Environ. Sci. Technol 1981, 15 (11) 1292–1304DOI: 10.1021/es00093a002
11. U.S. EPA. Development Document for Effluent Limitations Guidelines and Standards for the OCPSF Point Source Category, EPA 440/1-87-009a and b, NTIS PB88–171335; U.S. EPA, Office of Water: Washington, D.C., 1987a; Vols. 1 and 2.
12. Bizer C; Heath T; Berners-Lee T Linked data – the story so far Int. J. Semantic Web Inform. Syst 2009, 5 (3) 1–22DOI: 10.4018/jswis.2009081901
13. Sankar P; Aghila GJ Design and development of chemical ontologies for reaction representation J. Chem. Inf. Model 2006, 46 (6) 2355–68DOI: 10.1021/ci050533x [PubMed: 17125179]
14. Szymkuc S; Gajewska EP; Klucznik T; Molga K; Dittwald P; Startek M; Bajczyk M; Grzybowski BA Computer-Assisted Synthetic Planning: The End of the Beginning Angew. Chem., Int. Ed 2016, 55, 5904–5937DOI: 10.1002/anie.201506101
15. Feldman HJ; Dumontier M; Ling S; Haider N; Hogue CW C.O. A chemical ontology for identification of functional groups and semantic comparison of small molecules FEBS Lett. 2005, 579, 4685–4691DOI: 10.1016/j.febslet.2005.07.039 [PubMed: 16098521]
16. Name Reaction Ontology RXNO. <https://www.ebi.ac.uk/ols/ontologies/rxno> (accessed August 24, 2017).
17. (a) <http://www.elsevier.com/online-tools/reaxys>; (b) <https://scifinder.cas.org>; (c) <http://www.chemspider.com>; (d) <http://www.infochem.de/products/databases/index.shtml> (accessed August 24, 2017).
18. SPARQL Query Language for RDF. <https://www.w3.org/TR/rdf-sparql-query/> (accessed August 24, 2017).
19. Protégé. <https://protege.stanford.edu/> (accessed August 24, 2017).
20. Coley CW; Barzilay R; Jaakkola TS; Green WH; Jensen KF Prediction of Organic Reaction Outcomes Using Machine Learning ACS Cent. Sci 2017, 3, 434–443DOI: 10.1021/acscentsci.7b00064 [PubMed: 28573205]
21. Wei JN; Duvenaud D; Aspuru-Guzik A Neural Networks for the Prediction of Organic Chemistry Reactions ACS Cent. Sci 2016, 2, 725–732DOI: 10.1021/acscentsci.6b00219 [PubMed: 27800555]

22. Chen JH; Baldi P No electron left behind: a rule-based expert system to predict chemical reactions and reaction mechanisms J. Chem. Inf. Model 2009, 49, 2034–2043 DOI: 10.1021/ci900157k [PubMed: 19719121]
23. Reaxys. www.reaxys.com (accessed August 24, 2017).
24. Kowalik M; Gothard CM; Drews AM; Gothard NA; Weckiewicz A; Fuller PE; Grzybowski BA; Bishop KJM Parallel optimization of synthetic pathways within the network of organic chemistry Angew. Chem., Int. Ed 2012, 51, 7928–7932 DOI: 10.1002/anie.201202209
25. McBride B; Norris GEarthster core ontology: description and rationale, version 0.1 SNAPSHOT; New Earth: Boston, 2010.
26. Cappellaro F; Masoni P; Moreno A; Scalbi S CASCADE In The 16th Internationale Conference: Informatics for Environment Protection, Vienna, Austria, 9 25–27, 2002; pp 490–493.
27. Muñoz E; Capón-García E; Laínez JM; Espuña A; Puigjaner L Considering environmental assessment in an ontological framework for enterprise sustainability J. Cleaner Prod 2013, 47, 149–164 DOI: 10.1016/j.jclepro.2012.11.032
28. Bertin B; Scuturici VM; Risler E; Pinon JM A semantic approach to life cycle assessment applied on energy environmental impact data management 2012 Joint EDBT/ICDT Workshops 2012, DOI: 10.1145/2320765.2320796
29. Takhom A; Ikeday M; Suntisrivaraporn B; Supnithi T Toward Collaborative LCA Ontology Development: a Scenario-Based Recommender System for Environmental Data Qualification 29th International Conference on Informatics for Environmental Protection, Copenhagen, Denmark, 9 7–9, 2015.
30. Zhang Y; Luo X; Buis JJ; Sutherland JW LCA-oriented semantic representation for the product life cycle J. Cleaner Prod 2015, 86, 146–162 DOI: 10.1016/j.jclepro.2014.08.053
31. OWL Web Ontology Language. <https://www.w3.org/TR/owl-ref/> (accessed August 24, 2017).
32. 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. 2007, 36, D344–D350 DOI: 10.1093/nar/gkm791 [PubMed: 17932057]
33. Hastings J; Chepelev L; Willighagen E; Adams N; Steinbeck C; Dumontier M The chemical information ontology: provenance and disambiguation for chemical data on the biological semantic web PLoS One 2011, 6 (10) e25513 DOI: 10.1371/journal.pone.0025513 [PubMed: 21991315]
34. Ritz J; Fuchs H; Kieczka H; Moran WC Caprolactam. In Ullmann's Encyclopedia of Industrial Chemistry; Wiley-VCH Verlag GmbH & Co. KGaA, 2000.
35. TOXNET. <http://toxnet.nlm.nih.gov/cgi-bin/sis/search2/r?dbs+hsdb:@term+@DOCNO+187> (accessed August 24, 2017).
36. Favre HA; Powell WH Nomenclature of Organic Chemistry: IUPAC Recommendations and Preferred Names; Royal Society of Chemistry: 2013.
37. Edelen A; Ingwersen W Guidance on Data Quality Assessment for Life Cycle Inventory Data; U.S. Environmental Protection Agency: Washington, DC, 2016; EPA/600/R-16/096.
38. Rubin E Synthetic Socialism: Plastics and Dictatorship in the German Democratic Republic; The University of North Carolina Press, 2014.
39. TopBraid Composer. <https://www.topquadrant.com/tools/modeling-topbraid-composer-standard-edition/> (accessed August 24, 2017).
40. Facility Registry Service (FRS). <https://www.epa.gov/enviro/facility-registry-service-frs>. (accessed August 24, 2017).
41. Aspen plus. <http://home.aspentech.com/products/engineering/aspen-plus> (accessed August 24, 2017).
42. Chemstations. <http://www.chemstations.com/> (accessed August 24, 2017).
43. Jacobs M; Malloy TF; Tickner JA; Edwards S Alternatives Assessment Frameworks: Research Needs for the Informed Substitution of Hazardous Chemicals Environ. Health Perspect. 2016, 124, 265–280 DOI: 10.1289/ehp.1409581

44. Tickner JA; Dorman DC; Shelton-Davenport MS Answering the Call for Improved Chemical Alternatives Assessments (CAA) Environ. Sci. Technol 2015, 49, 1995–1996DOI: 10.1021/es505446x

Synopsis

A reaction ontology is presented enabling development of an automated life cycle inventory for the assessment of chemical manufacturing processes.

Figure 1.
Visualizing chemical lineage as a family tree.

Figure 2.
Basic Lineage schema depicting the classes and subclasses.

Figure 3.
Overview of the Process ontology demonstrating the large amount of data required to describe chemical production processes.

Figure 4.
Some classes in Process are extensions of corresponding classes in Lineage.

Figure 5. Lineage of nylon-6 showing a sample of processing requirements.(9–11) Blue arrows represent a parent input (or reactant), and green arrows represent a child output (or product).

Figure 6.
Property assertions for Nylon6 and Nylon6Production.

Property assertions: CyclohexanoneOximeProduction	Property assertions: CaprolactamProduction
<div>Object property assertions +</div> <ul style="list-style-type: none">has_reactant Ammoniahas_product CyclohexanoneOxime'has separation system' Decanter_type_1has_reactant Cyclohexanonehas_reactant HydroxylAmmoniumSulfate'has reactor system' CSTR_type_1 <div>Data property assertions +</div> <ul style="list-style-type: none">has_conversion "99.6 percent"has_recirculation_rate "40 times feed"has_retention_time "7 minutes"'has temperature' "85-90 C"	<div>Object property assertions +</div> <ul style="list-style-type: none">'has separation system' distillation_column_type_2'has reactor system' CSTR_type_2'has separation system' distillation_column_type_1has_reactant CyclohexanoneOximehas_catalyst Oleum'has separation system' Decanter_type_2'has separation system' Crystallizer_type_1has_product Caprolactam'has separation system' liquid_extractor_column_type_1 <div>Data property assertions +</div> <ul style="list-style-type: none">has_feed_ratio "sulphuric acid to cyclohexanone oxime 1:1.05"'has temperature' "80 C"has_yield "99.1 percent of Cyclohexanone Oxime fed"has_recirculation_rate "40 times fresh feed"has_catalyst_contact_time "3 to 10 seconds"

Figure 7.
Object and data property assertions for cyclohexanone oxime and caprolactam production.

Figure 8. Lineage of nylon-6 generated from a SPARQL query of reaction data stored as linked open data using the Lineage schema.

Figure 9.
Framework describing the role of "Lineage" and "Process" in assessment tools.

Table 1.

Description and Examples of Lineage Object Properties

object property	description	example
belongs to family	specifies membership of a chemical in a family	AceticAcid <i>belongs to family</i> Acid.
has chemical	identifies the underlying chemical of a ReactionParticipant	MethanolInAceticAcidProduction <i>has chemical</i> Methanol.
has condition	defines reaction conditions; the subproperties are <i>has pressure</i> , <i>has reaction time</i> , <i>has temperature</i>	AceticAcidProduction <i>has temperature</i> 150 to 200 °C.
has reaction participant	defines a chemical's role in a reaction; the subproperties are <i>has byproduct</i> , <i>has catalyst</i> , <i>has product</i> , <i>has reactant</i> , <i>has solvent</i> , <i>etc.</i>	AceticAcidProduction <i>has product</i> AceticAcid.
isParentOf	describes the inferred relationship between two chemicals when one is a reactant of a reaction and the other is a product or byproduct of the same reaction	Methanol <i>isParentOf</i> AceticAcid.
isAncestorOf	an automatically inferred property describing the relationship between two chemicals via lineage	Hydrogen <i>isAncestorOf</i> AceticAcid.

Table 2.

Types and Examples of Lineage Data Properties

data property	data type	example
IUPAC name	text string	azepan-2-one
CAS number	text string	105-60-2
synonyms	text string	ϵ -caprolactam, 1-aza-2-cycloheptanone
SMILES structure	text string	<chem>O=C1NCCCCC1</chem>
chemical formula	text string	(CH ₂) ₅ C(O)NH
molecular weight	floating decimal	113.16
molecular weight units	text string	g mol ⁻¹
reaction type	text string	Beckmann rearrangement
number of reactants	integer	2
number of products	integer	1
number of byproducts	integer	0
reference	text string	Guo et al., Green Chem., 2006 , 8, 296–300
date added	text string	8-20-2017

Table 3.

Description and Examples of Process Object Properties

object property	description	example
has reactor system	relationship between the chemical reaction and the equipment used for the reaction	AceticAcidProduction has reactor system Tank_type_1.
has separation system	relationship between the chemical reaction and the separation system	AceticAcidProduction has separation system Distillation_column_type_1.
has storage system	relationship between the chemical reaction and the storage system	AceticAcidProduction has storage system Fixed_roof_tank_type_1.
has treatment system	relationship between the chemical reaction and the treatment system	AceticAcidProduction has treatment system Air_control_unit_type_1.
has heat integration system	relationship between the chemical reaction and heating requirements in a process	AceticAcidProduction has heat integration system Heat_exchanger_type_1.
has utility	relationship between the chemical reaction and utilities such as electricity and water	AceticAcidProduction has utility Cooling_tower_type_1.
has inputs and outputs	relationship between the ChemicalReaction and inputs and outputs in a process plant	AceticAcidProduction has air emission CarbonMonoxide.
has miscellaneous system	relationship between the ChemicalReaction and other equipment used in a process such as a mixer or pump	AceticAcidProduction has miscellaneous system Pump_type_3.

Table 4.

Types and Examples of Process Data Properties

data property	data type	example
has production volume	floating decimal	300000
has production volume unit	text string	ton/y
has concentration	floating decimal	11.4
has concentration unit	text string	wt %
has temperature	floating decimal	189
has temperature unit	text string	deg C
has reaction rate	floating decimal	20
has reaction rate unit	text string	g mol/L h
has conversion	floating decimal	98.5
has conversion unit	text string	%
has fugitive emissions	floating decimal	79.93
has fugitive emissions unit	text string	kg/y

Table 5.

SPARQL Query for Establishing a Chemical Lineage

SPARQL Code Line	Meaning
CONSTRUCT {	Tells the SPARQL processor to select these facts to constitute the desired lineage:
?anc1 isReactantOf ?reaction.	Selected fact: ancestor_1 is a reactant of the Reaction
?reaction hasProduct ?anc2.	Selected fact: Ancestor_2 is a product of the Reaction
}	Closes the CONSTRUCT command
WHERE {	Tells the SPARQL processor to look for this pattern:
?anc1 :isParentOf* :Nylon6.	Ancestor_1 is an ancestor of Nylon6
?anc2 :isParentOf* :Nylon6.	Ancestor_2 is an ancestor of Nylon6
?anc1 :isParentOf ?anc2.	Ancestor_1 is a parent of Ancestor_2
?anc1 :isReactantOf ?reaction.	Ancestor_1 is a reactant of some Reaction
{	Either:
?reaction ;hasProduct ?anc2	Ancestor_2 is a product of the Reaction
{ UNION {	Or:
?reaction :hasByProduct ?anc2 .	Ancestor_2 is a byproduct of the Reaction
}	Closes the Either-Or
}	Closes the pattern definition