MEDINFO 2019: Health and Wellbeing e-Networks for All
L. Ohno-Machado and B. Séroussi (Eds.)

doi:10.3233/SHTI190193

Aggregation and Visualization of Laboratory Data by Using Ontological Tools Based on LOINC and SNOMED CT

Cora Drenkhahn[a], Petra Duhm-Harbeck[b], Josef Ingenerf[a,b]

[a] *Institute of Medical Informatics (IMI), University of Lübeck, Lübeck, Germany,*
[b] *IT Center for Clinical Research, Lübeck (ITCR-L) University of Lübeck, Lübeck, Germany*

Abstract

With the proliferation of digital communication in healthcare, the reuse of laboratory test data entails valuable insights into clinical and scientific issues, basically enabled by semantic standardization using the LOINC coding system. In order to extend the currently limited potential for analysis, which is mainly caused by structural peculiarities of LOINC, an algorithmic transformation of relevant content into an OWL ontology was performed, which includes LOINC Terms, Parts and Hierarchies. For extending analysis capabilities, the comprehensive SNOMED CT ontology is added by transferring its contents and the recently published LOINC-related mapping data into OWL ontologies.

These formalizations offer rich, computer-processable content and allow to infer additional structures and relationships, especially when used together. Consequently, various reutilizations are facilitated; an application demonstrating the dynamic visualization of fractional hierarchy structures for user-supplied laboratory data was already implemented. By providing element-wise aggregation via superclasses, an adaptable, graph representation is obtained for studying categorizations.

Keywords:

Biomedical Ontologies, LOINC, SNOMED CT

Introduction

As a prerequisite for any reuse of clinical data, it is essential to communicate in a semantic interoperable way so that the content's meaning is preserved independently from locally used terms and external influences like the applied software system. This is achieved by coding information explicit with a standardized terminology. The most widespread coding scheme in the laboratory domain is the *Logical Observation Identifier Names and Codes* (LOINC) terminology [1]. Each laboratory test is assigned a unique code representing its characteristics through the values of at least five axes. By using mapping methods, prepared and integrated laboratory data can provide valuable insights into patient care [2; 3].

However, activities to extend identifying LOINC codes by hierarchical structures or formal specifications have only recently begun, so that so far their use is limited to the unambiguous identification of laboratory tests with language-independent codes. Repeatedly it was demanded that LOINC codes should be enriched with a hierarchical structure., e.g. for aggregating reportable diseases in public health [4] or for improving semantic interoperability of LOINC-coded data [5].

So, the project aimed to transform the given LOINC terminology contents into a different, computer-processable representation subsequently enabling a larger scale of aggregation, analysis, and visualization of coded laboratory data.

Therefore, relevant LOINC components are converted into an ontology based on the well-known Web Ontology Language (OWL), i.e. a knowledge representation often defined as an "explicit specification of a conceptualization" [6], which uses formal logic to infer reasoner-based conclusions.

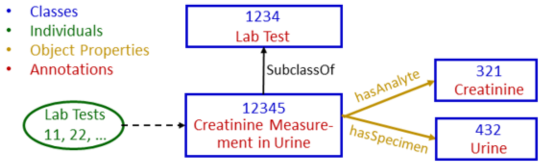

Figure 1 - Formalization of lab test classes and individuals

In order to enrich the given LOINC contents with additional information and hierarchical structures, the extensive *Systematized Nomenclature of Medicine – Clinical Terms* (SNOMED CT) shall be incorporated into the resulting ontology, as well. By using formal definitions throughout its scope, SNOMED CT is already specified as an ontology and can thus be integrated easily.

But to establish the connections between equivalent or otherwise related concepts in LOINC and SNOMED CT another source of data is needed. This is provided by the recently released official mapping between both terminologies, resulting from the publishers' cooperation agreement in 2013 [7]. Generating an OWL ontology for these mapping statements completes the first step, shown in figure 2, as a starting point for their integration and use.

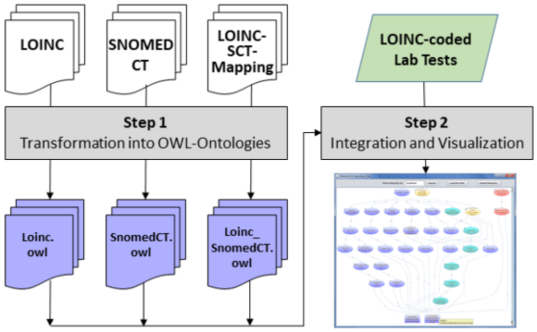

Figure 2 - LOINC aggregation and visualization tool: outline

The authors are aware that there are ontological challenges in the representation of laboratory tests, especially in connection with result values [8]. In this paper, purpose-specific LOINC

Hierarchies based on hierarchized LOINC *Part* terms (e.g. *Multi-Axial Hierarchy*), as well as logically formalized hierarchies, shall be provided via LOINC/SNOMED CT mapping in an integrated way to classify lab tests as LOINC-coded individuals per reasoner. We are convinced that LOINC- and SNOMED CT-based hierarchies serve different purposes, which complement each other but cannot replace each other [9]. This is supported by Vreeman speaking from "clinically-relevant aggregations" within LOINC by using multi-axial hierarchies and the newly introduced LOINC *Groups* [10].

Coming back to figure 2, this work concentrates on technical aspects for making use of OWL reasoners in order to integrate the three mentioned formalizations in step 2 and visualize LOINC-coded lab tests as individuals with respect to the integrated hierarchies. While there have already been attempts to build an ontology for LOINC [11; 12], this approach includes a larger scale of contents. Here, the main focus is laid onto arranging LOINC tests in hierarchical structures based on both explicit subclass relations as well as on their implicit features inferred by the LOINC/SNOMED CT-mapping.

Methods

LOINC resources

As the starting point for the project, the latest LOINC version 2.64 was utilized, published by the Regenstrief Institute in June 2018. This release contains a total of 87,863 LOINC *Terms* representing distinct laboratory tests or other clinical observations. Each *Term* is defined by a number of atomic components within. These LOINC *Parts* can be identified by their unique LP-Code and correspond to LOINC's main axes or their subcomponents describing the test's properties in detail. An example of usage is shown in figure 3. All in all, there are currently 52,000 *Parts* available which are used in about 900,000 relationships to define LOINC *Terms*.

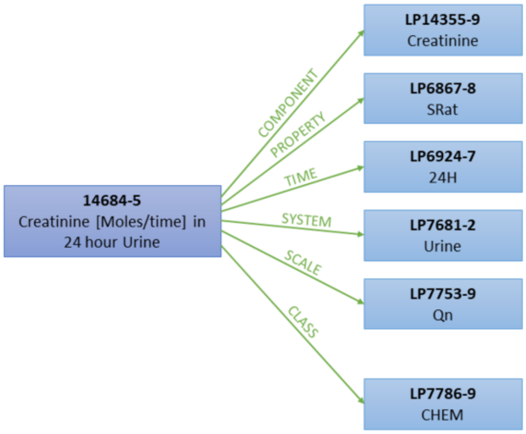

Figure 3 - Interrelation between a LOINC Term (left) and its definition by a unique combination of LOINC Parts (right).

To include even more contents into the transformation process we accessed the database containing background information within RELMA 6.23, a software offered by the Regenstrief Institute to enable searching in and mapping to LOINC. In these database tables, the information mentioned before can be found as well as more hierarchical structures. After further investigation a number of 10 so-called *Part Hierarchies* could be identified, each referencing a different aspect of laboratory tests. Even though the *Hierarchies* are built using LOINC *Parts* they are made to arrange complete LOINC *Terms* in a hierarchical structure based on their properties. As a special case the *Multi-Axial Hierarchy* is formed and made public, which uses composite LOINC *Parts* to structure *Terms* by their properties of two or more axes [13].

Converting LOINC to OWL ontology

In order to create an OWL ontology for the LOINC coding system in an automated and reusable way, an algorithmic approach was implemented using *Java*. For parsing all source files, present or previously transformed in CSV format, the Java CSV library is incorporated. Subsequent construction of OWL aspects is conducted by means of the OWL API [14].

```
Class: <http://imi.uni-luebeck.de/loinc#14684-5>

  Annotations:
    rdfs:comment "Creatinine:SRat:24H:Urine:Qn",
    rdfs:label "Creatinine [Moles/time] in 24 hour Urine"

  EquivalentTo:
    <http://imi.uni-luebeck.de/loinc#L_0>
     and (<…/loinc#hasClass> some <…/loinc#LP7786-9>)
     and (<…/loinc #hasComponent> some <…/loinc#LP14355-9>)
     and (<…/loinc #hasProperty> some <…/loinc#LP6867-8>)
     and (<…/loinc #hasScale> some <…/loinc#LP7753-9>)
     and (<…/loinc #hasSystem> some <…/loinc#LP7681-2>)
     and (<…/loinc #hasTime> some <…/loinc#LP6924-7>)
```

Figure 4 - OWL definition in Manchester Syntax for the LOINC Term shown in figure 3.

LOINC *Terms* are represented as distinct OWL classes with their code as an identifier, complemented by human-readable descriptions as label and comment. To model fully-defined LOINC *Terms* on the detailed level, they receive an equivalence statement combining all related *Parts* into one axiom. Here, an additionally defined superclass L_0, referring to LOINC Tests in general, is used as a starting point. Then, each LOINC *Part* is integrated as a specifically defined OWL class and added via an *ObjectProperty* derived from the *Part's* category. An exemplary result of this OWL translation is given in figure 4. For some LOINC *Terms,* not all required *Parts* are stated so that the combined definition is represented as subclass axiom, limiting logical inferences later on.

```
Class: <http://imi.uni-luebeck.de/loinc#LH56020>

  Annotations:
    rdfs:label "Urine Test"

  EquivalentTo:
    <http://imi.uni-luebeck.de/loinc#L_0>
     and (<…/loinc#hasSystem> some <…/loinc#LP7681-2>)

  SubClassOf:
    <http://imi.uni-luebeck.de/loinc#LH647>
```

Figure 5 - OWL concept of a hierarchy element describing all tests measured in the specimen "urine". The partially identical definition to figure 4 shall be noted.

For the ontological representation of LOINC *Hierarchies* each of their elements found as tree nodes in the source files is translated into a separate OWL class. The respectively specified parent node is referenced in a subclass axiom, building the hierarchy's backbone structure. In order to add its conceptual meaning, each element is furthermore defined using the LOINC *Part* associated with the hierarchy node in question. So, an equivalence axiom is constructed from the conjunction of the lab test basic class, the appropriate *Object-Property* and the *Part's* OWL class, as shown in figure 5. By using this kind of definition for hierarchy elements, the similarity to those used for LOINC *Terms* leads to logically inferred subsumptions of test terms into hierarchical classes.

But as there are some divergent structures in the LOINC *Part Hierarchies*, these implicit definitions can't be used throughout. Regarding both the *Component* and the *Multi-Axial Hierarchy* duplicate uses of the same LOINC *Part* can be found, causing equivalent definitions of hierarchy elements by the previously described approach. To avoid false inferences, the affected *Hierarchies* are therefore limited to primitive definitions. LOINC *Terms* are thereupon subsumed explicitly by the means of subclass axioms.

SNOMED CT resources

To include SNOMED CT contents, both the source files of the terminology in itself as well as those specifying the mapping to LOINC are needed. For the latter, we could obtain the latest version of the LOINC/SNOMED CT Cooperative package, published as Production Release in July 2017. According to the version used in these mapping files, we decided to utilize the International release 20170731 of SNOMED CT.

As the largest terminologies in medicine, SNOMED CT covers a much wider range of application than LOINC. Though laboratory tests and observations are mainly described by SNOMED CT *Procedure* and *Observable Entity* concepts, many other categories can be applied to define individual aspects of lab data as well, especially by combining them into *Post-coordinated Expressions*. Because of this, the complete SNOMED CT ontology shall be taken into consideration.

The cooperative package includes two main types of relations between LOINC and SNOMED CT. Firstly, entire LOINC *Terms* are mapped to combined SNOMED CT post-coordinated expressions, using the *Observable Entity* concept as a basic class. Any other components required to describe the LOINC *Term* are added as pairs of attributes and concepts, adhering to a compositional model. Secondly, the elementary LOINC *Parts* are associated with corresponding SNOMED CT components, once again built from attribute-concept-pairs.

Convert SNOMED CT to OWL ontology

For the transformation of SNOMED CT into an OWL ontology a Perl script is provided by SNOMED International. It was applied according to its instruction, converting given RF2 files into the OWL XML/RDF format.

Regarding the ontology generation for the mapping contents, the same procedure as for the LOINC terminology could be utilized. So, another Java algorithm using the OWL API was implemented.

```
Class: <http://imi.uni-luebeck.de/loinc#14684-5>

  EquivalentTo:
    <http://snomed.info/id/363787002>
     and (<http://snomed.info/id/370132008> some <http://snomed.info/id/30766002>)
     and (<http://snomed.info/id/704318007> some <http://snomed.info/id/118562009>)
     ...
```

Figure 6 - Excerpt of the equivalence definition for the previous LOINC Term mapped to SNOMED CT components.

Relationships of LOINC *Terms* to SNOMED CT post-coordinated expressions are hereby represented in the same way already used for their definition by combined LOINC *Parts*, as shown in figure 6. Where applicable, another equivalence or subclass statement is added to the OWL class of the LOINC *Term* accordingly, composed of the conjunction of all required SNOMED CT attributes and concepts.

The ontological representation of LOINC *Part* mappings turned out to be more difficult, resulting partly from the given unbalanced definitions, which refer to single LOINC *Parts* on one side but combined SNOMED CT expressions on the other. As a portion of each *Part's* meaning is implicitly included in its category, the corresponding *ObjectProperty* is used to form a combined expression for the LOINC *Part*. Additionally, both sides are complemented by the inclusion of the respective basic class (*LOINC Test* and *Observable Entity*) to ensure conformity with any other ontological components.

As a result of these adjustments, both terms needed for the part mapping transformation consist of complex class definitions, so that *General Concept Inclusions* (GCIs) have to be used for an appropriate OWL representation (see figure 7).

```
Class: "Observable Entity"
     and "Has Specimen" some "Urine specimen"

  EquivalentTo:
    "LOINC Test"
    and "hasSystem" some "Urine"
```

Figure 7 - Simplified OWL example of a SNOMED CT expression (above) mapped to a LOINC Part (below) using labels instead of identifiers.

Another problem occurred during the conversion of non-equivalent part mappings, with LOINC *Parts* being broader or narrower in content than the corresponding SNOMED CT expression. Due to the exactly specified compositional model of SNOMED CT, some LOINC *Parts* are mapped to a rather complicated composition of elements. In the given source files these relations were found to be not differentiated sufficiently, leading to imprecise definitions and thus false inferences. Because of this, only LOINC *Parts* with equivalent SNOMED CT expressions were considered for the envisioned prototype.

Visualization of created hierarchies

In order to demonstrate the resulting OWL ontologies' possibility of usage and to gain a graphical representation of the hierarchical structures inferred by them, another Java application was implemented afterward. It is based on the OWL API as well and uses an implementation (*ElkOwlApi*) of the ELK reasoner for computing inferences in addition [15]. Furthermore, the *JGraphX library* was chosen to provide graph drawing functionality. Accordingly, the complete graphical user interface was designed using Java *Swing*.

The application's input mostly consists of the three previously created OWL ontologies, each providing LOINC, SNOMED CT or mapping knowledge, which are then inferred conjointly by the ELK reasoner. Based on LOINC-coded laboratory individuals provided by the user, all relevant classes of these terminologies are determined. A recursive algorithm is hereby used to traverse the inferred hierarchies while collecting all visited superclasses. These are added as nodes to a tree-like graph structure, including their relationships as edges.

When the root element is reached, the completed graph is visualized on the GUI. A number of stylesheets are utilized to differentiate between elements by defining distinct representations according to source terminology or class type. Furthermore, some interactive functions are implemented in order to improve usability beyond basic operations. By folding and expanding subtrees, the graph's complexity can be reduced as needed and by highlighting a path's course becomes more concise. Finally, the implemented tool allows adding user-defined hierarchies that are transferred to a light-weight ontology which is as well included in the inference and visualization, shown as red elements on the right in figure 8.

Results

For the LOINC terminology, two different OWL ontologies were generated successfully: A full version and one comprised

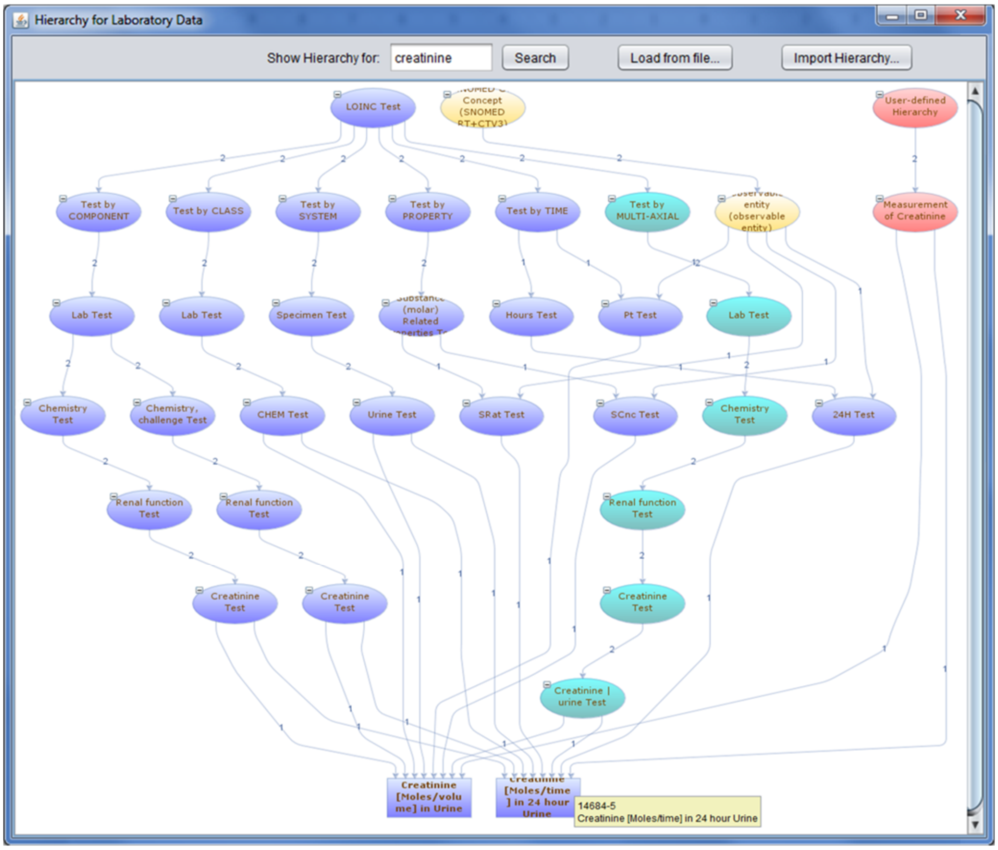

Figure 8 - Visualization of inferred hierarchies for the LOINC Terms 14684-5 and 14683-7 as leaf nodes. Each tree node represents a distinct OWL class, each edge denotes a subclass relation. The three root elements refer to the LOINC terminology (left), the SNOMED CT ontology (middle) and a user-defined hierarchy (right). The OWL ontology generated for the LOINC/SNOMED CT-Mapping does not include any named classes, so its contents is solely represented by edges.

of laboratory tests only. Both of these are divided into two main areas separating single 'LOINC Parts' from composite 'LOINC Tests'. The latter contains hereby all transformed hierarchy elements in their defined tree-structure as well as all LOINC *Terms*, which are classified into these hierarchies. A basic *Term* without subcomponents is typically included into six different hierarchies, one for each of its definitional *Parts* (except 'Scale') and the *Multi-Axial Hierarchy*.

For SNOMED CT an OWL representation could be generated easily by using the given script. The output contains the entire terminology contents keeping its extensive predefined relations and hierarchies. Another OWL ontology could be built for the cooperative package linking LOINC to SNOMED CT. Here, all of the LOINC *Term* mappings were transformed into OWL axioms using classes already defined in the other ontologies. In terms of the LOINC *Parts*, only equivalent mappings could be included as explained before, leaving one quarter of the enclosed relationships out of the result.

All of the created OWL ontologies can be applied as input in suitable software applications, either separately or combined. By importing them into the ontology editor *Protégé,* their contents and characteristics can be examined, see Table 1.

Additionally, the ELK reasoner was used to infer all ontologies, requiring about ten seconds in the case of their combined usage. Afterwards, both stated and inferred axioms were evaluated manually.

Table 1 - Extent of generated OWL ontologies

	LOINC (lab only)	SNOMED CT	Mapping
Axioms	1 004 634	1 520 034	62 155
Classes	277 159	335 225	34 272
ObjectProperties	16	97	25
AnnotationProp.	2	6	0
SubClassOf	366 292	253 406	2 355
EquivalentClasses	62 306	81 818	25 503
GCIs	0	0	5 969

No conflicts or contradictions could be found up to this point, concluding that the ontologies are consistent and valid. Regarding the inferred hierarchical order, the LOINC *Hierarchies* were found to yield a large amount of structuring information. This becomes particularly apparent through the graphs generated by the visualization application, as shown for two exemplary LOINC *Terms* in figure 8. Most of the pictured nodes refer to OWL classes generated based on LOINC contents. The same applies to the edges depicting subclass axioms, both stated and inferred.

By the inclusion of SNOMED CT knowledge and the cooperative mapping further hierarchical deductions are inferred into the arrangement of LOINC *Terms*. These are computed by the reasoner in a multi-level process, typically based on one of the *Term's Parts*, its representation in SNOMED CT and the subclass definitions therein. Though there is more background information used in the inferred hierarchies, the graph visualization includes only two SNOMED CT elements: The root concept and the basic *Observable Entity*. All other involved SNOMED CT components are defined as post-coordinated expressions and thus not as named classes, which could be represented as tree nodes. The visualization application allows the rendering of multiple instances of LOINC *Terms*. These are displayed collectively in one hierarchy graph, taking into account their interrelations and frequencies of occurrence. As a result, an aggregated analysis in different levels of granularity is facilitated for a set of LOINC-coded laboratory tests.

Discussion

Despite the heterogeneous structure and non-formal definitions of LOINC contents, an extensive OWL ontology could be created comprised of LOINC *Terms, Parts* and *Hierarchies*. By using this formalization new insights into the terminology's hidden information are granted, whereupon the already existing hierarchies appeared to yield a particularly large potential by structuring the otherwise unsorted LOINC *Terms* according to their characteristics.

Additionally, the integration of SNOMED CT contents adds even more knowledge and relationships, enabling a wider range of application. The tree-like graphs created based on a user-supplied input of LOINC codes of interest provide a clearly arranged hierarchical structure and thus new possibilities to evaluate laboratory data. In this representation LOINC *Terms* and hierarchy elements are already displayed comprehensively, whereas contents derived from SNOMED CT or the cooperative mapping is included in a more subtle way that requires further improvement in order to present meaningful information.

For all of the generated OWL ontologies and their inferences, a profound evaluation is needed to ensure validity and to specify statistical properties. Amongst others, it is planned to use this tool within the "LOINC-300"-activities within the BMBF-funded Medical Informatics Initiative in Germany [16].

Conclusions

In this project we could develop an approach to improve the reusability of LOINC-coded laboratory data by converting relevant terminology contents into OWL ontologies, hence facilitating advanced analysis based on formally-defined representations and logical inferences.

By including the otherwise only internally used LOINC *Hierarchies* as well as SNOMED CT knowledge, extensive structures could be formed that build novel hierarchical classifications for LOINC *Terms*. For a concrete visualization, an application could be implemented that creates tree-like hierarchy graphs for LOINC-coded lab data, thereby enabling individualized and aggregated evaluations.

Acknowledgements

We thank the Regenstrief Institute for providing the structured LOINC source data and for their support.

The research was supported in part by the BMBF Medical Informatics Initiative. For using SNOMED CT source data an affiliate development license was obtained from IHTSDO.

References

[1] C.J. McDonald, et al. Case, and P. Maloney, LOINC, a universal standard for identifying laboratory observations: a 5-year update, *Clin Chem* **49** (2003), 624-633.
[2] C. Zunner, et al., Mapping local laboratory interface terms to LOINC at a German university hospital using RELMA V.5: a semi-automated approach, *J Am Med Inform Assoc* **20** (2013), 293-297.
[3] M.F. Emons, Integrated patient data for optimal patient management: the value of laboratory data in quality improvement, *Clin Chem* **47** (2001), 1516-1520.
[4] S. Steindel, J.W. Loonsk, A. Sim, T.J. Doyle, R.S. Chapman, and S.L. Groseclose, Introduction of a hierarchy to LOINC to facilitate public health reporting, *Proc AMIA Symp* (2002), 737-741.
[5] M.C. Lin, D.J. Vreeman, et al., Auditing consistency and usefulness of LOINC use among three large institutions - using version spaces for grouping LOINC codes, *J Biomed Inform* **45** (2012), 658-666.
[6] N. Guarino, D. Oberle, and S. Staab, What is an ontology?, In: *Handbook on ontologies*, 2009, pp. 1-17.
[7] D. Vreeman, New Regenstrief and IHTSDO agreement to make EMRs more effective at improving health care, *LOINC News* (2013), see https://loinc.org/news/new-regenstrief-and-ihtsdo-agreement-to-make-emrs-more-effective-at-improving-health-care/ (accessed November 5, 2018)
[8] M. Mary, et al., Ontological Representation of Laboratory Test Observables: Challenges and Perspectives in the SNOMED CT Observable Entity Model Adoption, In: *AIME 2017, LNAI 10259*, 2017, pp. 14-23.
[9] J. Ingenerf and R. Linder, Assessing applicability of ontological principles to different types of biomedical vocabularies, *Methods Inf Med.* **48** (2009), 459-467.
[10] D.J. Vreeman, Grouping data coded with LOINC, see https:// danielvreeman.com/grouping-data-loinc/ (accessed November 5, 2018).
[11] M. Mary, L.F. Soualmia, and X. Gansel, Formalisation de la terminologie LOINC et évaluation de ses avantages pur la classification des tests de laboratoire, In: *28es Journées francophones d'Ingénierie des Connaissances IC 2017*, 2017, pp. 2-13.
[12] T. Adamusiak and O. Bodenreider, Quality assurance in LOINC using Description Logic, *AMIA Annu Symp Proc* **2012** (2012), 1099-1108.
[13] Regenstrief Institute, LOINC Multiaxial Hierarchy, see https:// loinc.org/downloads/accessory-files/ (accessed November 5, 2018).
[14] M. Horridge and S. Bechhofer, The OWL API: A Java API for OWL ontologies *Semantic Web* **2** (2011), 11-21.
[15] Y. Kazakov, M. Krötzsch, and F. Simančík, The Incredible ELK: From Polynomial Procedures to Efficient Reasoning with EL Ontologies, *Journal of Automated Reasoning* **53** (2014), 1-61.
[16] S.C. Semler, F. Wissing, and R. Heyder, German Medical Informatics Initiative, *Methods Inf Med* **57** (2018), e50-e56.

Address for correspondence

Cora Drenkhahn, Institute of Medical Informatics, University of Lübeck, Ratzeburger Allee 160, 23538 Lübeck, Germany

Email: c.drenkhahn@uni-luebeck.de