

Article

DCC Terminology Service—An Automated CI/CD Pipeline for Converting Clinical and Biomedical Terminologies in Graph Format for the Swiss Personalized Health Network

Philip Krauss ¹, Vasundra Touré ², Kristin Gnodtke ², Katrin Cramer ² and Sabine Österle ^{2,*}¹ Trivadis—Part of Accenture, 4051 Basel, Switzerland; Philip.Krauss@trivadis.com² Personalized Health Informatics Group, SIB Swiss Institute of Bioinformatics, 4051 Basel, Switzerland; Vasundra.Toure@sib.swiss (V.T.); Kristin.Gnodtke@sib.swiss (K.G.); Katrin.Cramer@sib.swiss (K.C.)

* Correspondence: Sabine.Oesterle@sib.swiss

Abstract: One goal of the Swiss Personalized Health Network (SPHN) is to provide an infrastructure for FAIR (Findable, Accessible, Interoperable and Reusable) health-related data for research purposes. Semantic web technology and biomedical terminologies are key to achieving semantic interoperability. To enable the integrative use of different terminologies, a terminology service is an important component of the SPHN Infrastructure for FAIR data. It provides both the current and historical versions of the terminologies in an SPHN-compliant graph format. To minimize the usually high maintenance effort of a terminology service, we developed an automated CI/CD pipeline for converting clinical and biomedical terminologies in an SPHN-compatible way. Hospitals, research infrastructure providers, as well as any other data providers, can download a terminology bundle (currently composed of SNOMED CT, LOINC, UCUM, ATC, ICD-10-GM, and CHOP) and deploy it in their local terminology service. The distributed service architecture allows each party to fulfill their local IT and security requirements, while still having an up-to-date interoperable stack of SPHN-compliant terminologies. In the future, more terminologies and mappings will be added to the terminology service according to the needs of the SPHN community.

Keywords: semantics; terminology; terminology service; clinical research infrastructure; ontologies; semantic web



Citation: Krauss, P.; Touré, V.; Gnodtke, K.; Cramer, K.; Österle, S. DCC Terminology Service—An Automated CI/CD Pipeline for Converting Clinical and Biomedical Terminologies in Graph Format for the Swiss Personalized Health Network. *Appl. Sci.* **2021**, *11*, 11311. <https://doi.org/10.3390/app112311311>

Academic Editor: Elena Cardillo

Received: 28 October 2021

Accepted: 25 November 2021

Published: 29 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Many national and international terminologies are available to describe biomedical concepts and to encode health-related data in Switzerland. Within the framework of the Swiss Personalized Health Network [1] (SPHN) national research infrastructure initiative, the SPHN Data Coordination Center (DCC) is developing a data ecosystem [2] using semantic web technologies to manage and link data according to the SPHN Semantic Framework [3] and in compliance with the FAIR principles [4]. The main terminologies used in the SPHN Semantic Framework are the Systematized Nomenclature of Medicine—Clinical Terms [5] (SNOMED CT); the Logical Observation Identifiers Names and Codes [6] (LOINC); the International Statistical Classification of Diseases and Related Health Problems, 10th revision, German modification [7] (ICD-10-GM); the Swiss classification for procedures [8] (“Schweizerische Operationsklassifikation” CHOP [9]); the Anatomical Therapeutic Chemical Classification System [10] (ATC) and the Unified Code for Units of Measure [11] (UCUM). Using these semantic standards across different hospitals enables better interoperability between all kinds of data, ranging from routine care data to clinical research data, but also leverages the additional information that these semantic standards provide. Classifications, such as ATC, ICD-10-GM, or CHOP, provide hierarchical information, whereas multi-axial terminologies such as LOINC allow for the use of contextual machine-readable information. For LOINC, additional attributes have been generated for

the six axes (component, property, time, system, scale, and method) to provide individual LOINC parts that describe the laboratory's measurements. Lastly, SNOMED CT provides a full ontology with polyhierarchical classifications, which can be explored in different ways to infer and retrieve knowledge (e.g., simple identification of a finding site: "lung" or of a causative agent ("virus") of a disorder: "viral pneumonia", or hierarchical information that an "adenoviral pneumonia (disorder)" is the child of a "viral pneumonia (disorder)").

It is the nature of biomedical terminologies to change, for example, as new research findings or adapted coding guidelines need to be reflected. Terms and codes can become deprecated, and new terms are added with new releases. Therefore, a reliable continuous process is required to stay up to date and to keep track of the release cycles of these terminologies. Providing historical versions of terminology is required since researchers receiving real-world data over a large time span need to be able to connect the applicable versions of the terminology to the data they receive from the data provider. While the information itself is present, one of the challenges is to seamlessly integrate and use all of these terminologies, which follow different concept models, and have their own formats and release cycles. Various communities and countries, therefore, host so-called terminology services that provide terminologies in several formats and representations that are compatible with different software applications. The following two architectural designs can be found in the literature (see Figure 1): either a centralized system with one terminology service or a federated design with a local terminology service. The most well-known example of a centralized terminology service is the BioPortal [12] repository, from the National Center for Biomedical Ontology (NCBO), which provides over 900 biomedical ontologies. Another example is Ontoserver [13], which is based on the Fast Health Interoperability Resources (FHIR) standard and includes a syndication mechanism to facilitate the versioning of terminologies. Ontoserver is available via a commercial software license and has been selected by NHS Digital UK and Nictiz Netherlands [14]. Several organizations make use of the HL7[®] FHIR[®] Application Programming Interface (API) to enable access to current terminology content, e.g., LOINC is offering the LOINC FHIR Terminology Server [15] based on the HL7[®] FHIR[®] API. However, this solution provides access to one single standard and only the current version. The Unified Medical Language System (UMLS) offers a meta-level view on concepts from many biomedical vocabularies and provides a crosswalk between them [16]. It can also be observed that hospitals start to develop their own terminology server in order to increase the quality of the clinical data [17]. All of these solutions have their benefits and advantages specifically targeted to certain application scenarios and they support semantic interoperability.

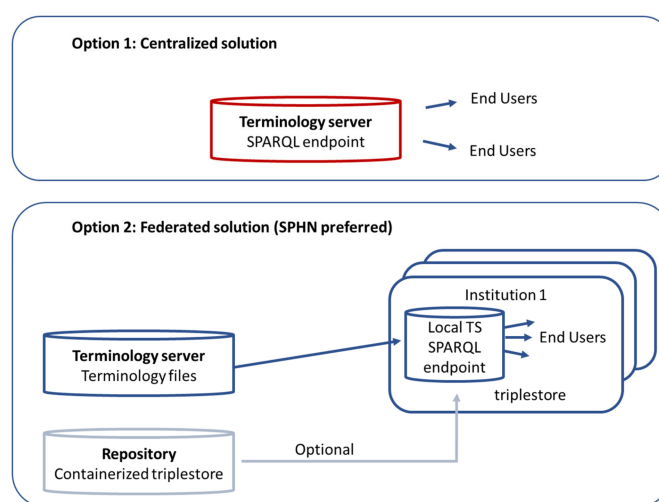


Figure 1. Two possible options for the architecture of a terminology service. Option 1 shows a centralized terminology service while Option 2 shows a federated terminology service (the SPHN-preferred option).

For the SPHN DCC Terminology Service, the following five main requirements have been identified: the terminology service must fit into the distributed project setting of the SPHN; comply with the stringent security requirements necessary to analyze clinical data; support the local Swiss setting and at the same time, ensure it is compatible with the SPHN Resource Description Framework (RDF) schema; deal with the continuous change in the terminologies and cope with the different license models of the terminologies.

Distributed project setting in the SPHN: The distributed project setting demands that a data producer, for example, a hospital, can obtain the terminologies in RDF (Turtle and OWL format); therefore, they can also use the data for validating the data assets being submitted. A data consumer, for example, a research project, needs the terminologies in an RDF format to link the terminologies to the hospital data. To avoid increasing the volume of the hospital data and to reduce the possibility of errors, the terminology data, such as hierarchy or property information, should not be sent by hospitals to the research project space together with the patient data, but should be provided to the researcher separately through the terminology service.

Security requirements: SPHN projects are hosted on the BioMedIT network [18], a service provider infrastructure for research with sensitive data built as an integral part of the SPHN. Each project is provided with a secure isolated project space for data storage and analysis on a BioMedIT node. A project space in BioMedIT is set up to fulfill the stringent security requirements to ensure the necessary data protection. The connection from the project space to the internet is limited and data, as well as other resources such as terminologies, need to come from a trusted source. A similar setup is observed in the hospitals. Additionally, when sending a SPARQL Protocol and RDF Query Language (SPARQL) query to a SPARQL endpoint, a SPARQL federation blends the data; therefore, it is important to ensure that no confidential data are sent to the SPARQL interface outside an institution.

Local Swiss requirements: Swiss-specific systems, such as Swiss Diagnosis Related Groups (SwissDRG), need to be taken into account, i.e., the terminology service needs to cover the terminologies required for accounting, e.g., CHOP, which is Swiss-specific and only valid for coding procedures for inpatients. CHOP itself does not provide official Internationalized Resource Identifiers (IRI) for each CHOP code; therefore, IRIs need to be created in an SPHN/BioMedIT namespace. In addition, CHOP can be properly used in the SPHN RDF schema only if IRIs are available, including IRIs for different versions. Hosting our own terminology service allows us to keep control of the compatibility of the terminology versions and the SPHN data schema versions represented in RDF [19].

Timing the release and use of versions of terminology: When a new version of a terminology is released, it does not necessarily mean that it is immediately used in hospitals. For example, the German Modification of ICD-10 is valid in Switzerland for coding the diagnoses of inpatients. However, due to the time needed for translation into French and Italian, a new version of ICD-10-GM is only adopted every two years, whereas the German Federal Institute of Drugs and Medical Devices (BfArM) in Germany releases a German Modification of ICD-10 every year.

Licensing: Each piece of terminology comes with its own license conditions, which must be respected when setting up a terminology service. There are a number of terminologies that are freely available, such as LOINC. Additionally, there are licensing models based on the membership of an organization, e.g., Switzerland is a member of SNOMED International and, thus, users can exploit SNOMED CT free of charge if they use the terminology in Switzerland, by registering with the national release center (NRC). However, other organizations that develop terminologies are more restrictive when it comes to the usage with multiple partners. Therefore, sharing terminology is only allowed under certain conditions, and hurdles have to be overcome to make the semantic standard available to a community in an integrated environment. Special attention is required for the use of these terminologies in multi-institutional settings, such as the SPHN projects.

An assessment of these requirements has led to the federated design of a terminology service for the SPHN (see option two in Figure 1). Hospitals and service providers have the ability to host their own local SPHN terminology services supporting exactly the terminologies, versions, modifications or editions that are used in the SPHN, rather than connecting to a general existing service such as BioPortal.

Keeping a terminology service up to date is a resource-intensive process, as the individual steps, such as converting new versions into a specific format and validating and uploading them, are very time-consuming. The use of a continuous integration/continuous deployment (CI/CD) pipeline is commonly used in software development to automate the delivery process of a tool, with code builds and test runs [20]. We, therefore, developed such a pipeline to automate the conversion of the provided terminologies into an SPHN-compatible RDF format, their validation and upload to the terminology server.

2. Materials and Methods

2.1. Design

A federated system was designed where terminology bundles, which are offered through a simple and secure programmatic interface, can be pulled by the partners and provided as an institution-internal SPARQL endpoint. These bundles were created through a continuous integration/continuous deployment (CI/CD) pipeline (see Figure 2). This pipeline simplifies the provisioning of these bundles. The central component of the SPHN DCC Terminology Service consists of the following two main parts, as depicted in Figure 2: the terminology server, where the source files and ready-to-use terminologies are stored in the object storage, and second, the CI/CD pipeline. The terminology server is an object storage and was implemented using the MinIO object storage. For the CI/CD pipeline, GitLab runners were chosen. The terminology server consists of the following two storage areas, called buckets: “Sources” and “Ontologies”. The CI/CD pipeline picks up the sources and applies the converters to the source files, tests the conversions and repacks the bundles according to the specifications. The bundle is signed and then made available in the ontologies area. From this area, other processes pick up the bundles to make the content available in the BioMedIT research environments, hospitals, or the BioMedIT Portal, which provides a download area for individual users. Both storage areas have an archive; therefore, old data can be accessed when needed.

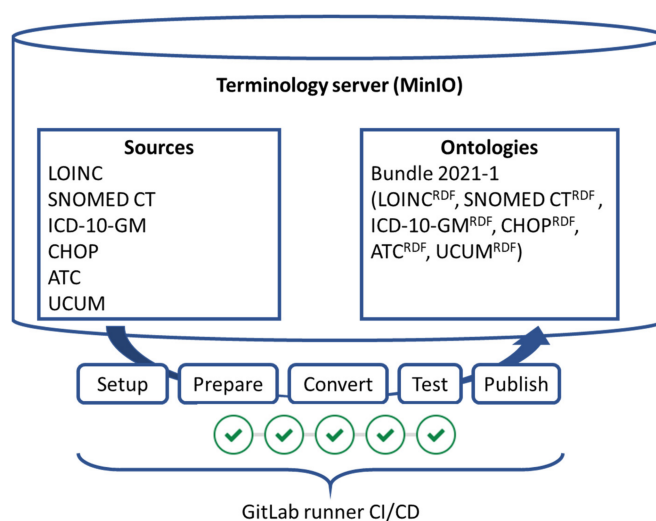


Figure 2. Design of the SPHN DCC Terminology Service. The terminology service consists of the terminology server and the GitLab runner pipeline. Terminologies are converted from their source format through the CI/CD pipeline into SPHN-compliant versions in the RDF (Resource Description Framework) standard.

2.2. Implementation

2.2.1. Data Collection

The different terminologies, usually in tabular formats, were collected from their source providers. Some of the standard development organizations (SDOs) required personalized accounts to access the terminology files, while others required a detailed description of the environment in which the terminology would be used, as their license model was not (yet) tailored for electronic use in distributed environments. In addition, the release cycle and release dates of each terminology are different, ranging from yearly releases to several ones per year. To automate the login, the download and maintenance of changes in the login procedures required some continuous efforts. Therefore, this step was executed manually by the DCC team.

2.2.2. Conversion Scripts

The SNOMED CT in RDF was generated with the SNOMED OWL Toolkit [21]. The resulting OWL files were additionally converted to the Turtle file format with a home-made tool. For the other terminologies, Python scripts were developed to convert the original terminologies to a machine-readable SPHN-compatible RDF format. For ATC, CHOP, ICD-10 GM and LOINC, codes are represented as classes (rdfs:Class) and names (labels) by the property rdfs:label, such as, in the axes of LOINC, which are represented as individual properties (e.g., “hasComponent”, “hasMethodType”). Hierarchies of the codes are represented by the property rdfs:subClassOf. Where necessary, root classes were created under the SPHN namespace: <https://biomedit.ch/rdf/sphn-resource>. For LOINC, specific properties for representing the 6-axis given as contextual information for each LOINC term (e.g., “hasComponent”, “hasMethodType”) were created. For UCUM, the library of pre-built valid UCUM codes from the National Library of Medicine, National Institutes of Health, U.S. Department of Health and Human Services (v1.5, Released in June 2020 [22]) was used as an input file for the Python pipeline. UCUM codes are provided as owl:NamedIndividual, represented with the code rdfs:label and the label rdfs:comment. All developed scripts are available under the GNU General Public License v3.0 license on https://git.dcc.sib.swiss/dcc/biomedit-rdf/-/tree/master/ontology_pipelines (accessed on 30 October 2021).

2.2.3. Test

Output files were loaded in a local Apache Jena Fuseki [23] instance and verified. In this step, technical correctness was tested, e.g., the files are valid RDF files that can be parsed. Furthermore, semantic correctness was checked, e.g., that the terminologies contain at least one hierarchy in each terminology. Every terminology conversion pipeline ingests their own test cases with them to assure that the specific tests can be run when needed. There are clear parallels with test cases for software applications; however, the scripts were not tested directly with artificial input data, but with each conversion of the actual terminology content. Table 1 summarizes the queries executed to test the terminology files. For Q1 in CHOP, for example, it checked that there were no roots other than the known nodes (chapters), C0 to C18, in CHOP present in the RDF files.

Table 1. Verification queries on the converted terminologies.

ID	Description
Q0	Every element must be at least of type rdfs:Class
Q1	Only the listed roots are present in the terminology
Q2	The number of concepts is between x and y
Q3	Every concept must have at least one rdfs:label

Currently, the tests are deployed as a set of SPARQL ASK [24] scripts. All of the answers of these scripts have to be “true”, otherwise the pipeline fails. Using SPARQL, in

contrast to Shapes Constraint Language (SHACL), also allows quantitative queries to be made on the data, e.g., querying the count of triples present in a certain terminology.

2.2.4. Bundling

The terminologies are provided in bundles. A basic logic determines which elements are copied over to a bundle and which ones are not. Due to different update cycles of the source files, not all the conversions are run in every execution. The bundling determines which terminologies are updated and which are kept. Every bundle is signed using a dedicated GNU Privacy Guard (GPG) key to provide trust in the origin of the terminologies.

2.2.5. GitLab Runner Pipeline

All terminologies are stored in an MinIO server hosted on the BioMedIT node sciCORE at the University of Basel [18]. In the MinIO server, there are the following four folders: sources/current, sources/archive, ontologies/current and ontologies/archive. Input files (original files in .csv, .xlsx, .zip or release format 2 (RF2) for SNOMED CT) are downloaded by the DCC staff and uploaded into the sources/current folder. The GitLab runner CI/CD pipeline (Available online: <https://git.dcc.sib.swiss/dcc/biomedit-rdf>, accessed on 30 October 2021) includes the following steps (see Figure 2):

1. Setup: The pipeline run is initialized and some local temporary exchanges are created.
2. Prepare: The files are loaded from the sources/current folder on the virtual machine (VM) where the GitLab runner is executed.
3. Convert: The provided terminology is converted with the corresponding conversion script. Each terminology has its own job that is only executed in case there is valid input data for the job.
4. Test: Technical and semantic tests are run. All tests are run in a single job.
5. Publish: A single zip file is created containing all the newest RDF files of SNOMED CT, LOINC, ATC, ICD-10-GM, UCUM and CHOP. This bundle is signed and copied to the MinIO server in the ontologies/current folder. The previous bundle is moved to the ontologies/archive folder. If all steps are successful, the source files are moved from sources/current to sources/archive.

2.2.6. Containerized Execution

All stages depend on the previous step; therefore, if one fails, the pipeline fails. The GitLab Runner pipelines allow us to use a different docker image to run the steps in each job. This functionality is extensively used, as most conversion scripts are based on Python, while the SNOMED CT and the overall test job are based on Java.

In the convert and test steps, Makefiles [25] are used to abstract from the container and the GitLab Runner pipeline. With the Makefile, a developer can use a single step without having a docker installed. In addition, Makefiles offer a concise way of expressing the different targets, “install”, “build” and “test”, that fit into the pipeline well. The template for the Makefile is reused across the conversion pipelines.

The following four different docker containers are used:

1. GitLab Runner: This is the main container that registers to the GitLab CI/CD pipeline. Besides the MinIO service, it is the only prerequisite to be installed on a machine in order for it to be able to execute the pipelines.
2. Python: Besides the pipeline for SNOMED CT, all pipelines use Python in their conversion. The Python container gives a simple abstraction of a plain Linux with a basic Python installation. Additional packages that are needed are installed using a requirements.txt file through the Python package manager, pip, in the “install” target of the Makefile of the individual pipeline.
3. Ubuntu Linux: For the SNOMED CT conversion, Java is used. As additional tools are needed in the conversion, the generic Ubuntu Linux container is used, and the necessary tools are installed using the “install” target of the Makefile. The generic Ubuntu Linux container is also used for interaction with MinIO using the mc client tool.

4. Apache Jena Fuseki: As part of the pipeline, the triplestore Apache Jena Fuseki is used to test the terminologies in RDF. To be able to make use of the scripts locally in a hospital or BioMedIT node, the triplestore is also offered as a docker container. The same docker container is used in the testing step to ensure correctness. For service providers that do not have a local triplestore of choice, this container can be used to deploy a triplestore. Apache Jena Fuseki is open-source software under the Apache 2.0 license.

3. Results

The developed terminology server relies on open-source standards and technologies and not on a specific vendor. The SPHN is in full control of the content and the versioning, and the service is as automated as possible to ease its use and maintenance. Data providers, such as hospital IT personnel, can download terminology bundles from the central terminology server on BioMedIT and load them into their local triplestore. In case they do not have a triplestore, the Apache Jena Fuseki container can be used. For researchers using the BioMedIT environment, the locally installed terminology service comes with the basic setup of a project space. The solution is minimizing the resources needed for maintenance as each repeating step is automated as much as possible. New versions of the existing terminologies only need to be downloaded by the DCC staff from the original source and be provided as an input for the CI/CD pipeline.

Currently, the SPHN DCC Terminology Service includes SPHN-compatible, machine-readable versions of the following terminologies:

- CHOP (including historical versions since 2016)
- ICD-10-GM (including historical versions since 2014)
- SNOMED CT (including historical versions since 31 January 2021)
- LOINC (including historical versions since v2.69)
- ATC (2021 version)
- UCUM

For individual researchers, DCC provides an additional webservice in the BioMedIT Portal (<https://portal.dcc.sib.swiss/>, accessed on 30 October 2021), where these terminology files are available for Swiss researchers (via SWITCH edu-ID) to download.

4. Discussion

A terminology server can be implemented as a single service unit where the terminology information can be retrieved centrally (see option one in Figure 1). In the distributed nature of the SPHN initiative, where different legal entities are license holders of individual terminologies and have different security requirements, a single service is not feasible. To overcome these hurdles, a distributed setting was developed (see option two in Figure 1). The terminologies are converted once and offered to the institutions for sharing with their end users. The architecture allows any connected institution to run their own terminology service based on the content provided by the main terminology server (see Figure 2). Institutions have the flexibility to decide which software components to work with to implement their own service. They may already have a preferred triplestore database, which they can use, or they can set up new graph-based data infrastructures, i.e., implement a containerized triplestore based on the open-source software Apache Jena Fuseki. Apache Jena Fuseki is used in the pipeline for testing the terminologies. The same version and configuration of Apache Jena Fuseki is offered as a container to the institutions that could facilitate the installation of a terminology service.

Searching for individual codes and their meaning is usually one of the functions a terminology server provides via an API. This function is also desired by the SPHN community; however, due to security requirements and licensing conditions, it needs to be implemented at local terminology services instead of at the main terminology server. By making terminologies available in a locally accessible triplestore, institutions can provide SPARQL endpoints as an API to their users, while ensuring the necessary security require-

ments of the individual institution are met. It is the responsibility of the institution to comply with licensing terms and requirements. The federated setting, where there is one terminology server per institution, is the key element, since a SPARQL federation can blend data and it is challenging to ensure that no confidential data are sent to the SPARQL interface. With our federated approach, we ensure that confidential data stay within the boundaries of a local institution, and that projects with highly sensitive data can have their own terminology services. However, the solution requires each service provider to supply the local terminology service to their end users and download the terminology bundle on a regular basis. To support this, an HTTPS API is provided on the central site to programmatically download the terminology packages.

In addition, while automating our pipeline using CI/CD has substantial benefits with respect to automation, testing and reproducibility, it also introduces additional complexity where different programming languages, Infrastructure as Code, external systems, and pipelining are used. On one hand, this limits the changes to the system that need to be executed, but on the other hand, it ensures that the changes are conducted carefully and are well tested. Another limitation introduced by the CI/CD approach, primarily based on the use of the programming language Python for the majority of the conversions, is the runtime performance. Python and the library used for handling RDF provide an elegant way to handle RDF data, but the speed is not as optimal as when using a triplestore. However, as a release is not time-sensitive and the runtime of 40 min for a complete run is currently acceptable, the limitation does not affect the service.

At the deployment's end, there is the choice of using the open-source MinIO Object Storage, which offers good performance but only limited types of operations on the filesystem compared to POSIX-compliant filesystems. In addition, the permission model of the chosen storage is rather simple. Neither of the limitations restrict the service, as most of the roles only have read access; the separate buckets for the sources and ontologies enable the management of the roles also on the bucket level, and the available file system operations enable the maintainer to move, obtain, put, or delete content, which is sufficient for our usage.

On the terminology end, our choice of providing terminologies in an RDF graph-format enables the full exploration of complex ontologies, for instance, SNOMED CT, which has poly-hierarchical and attribute relations. Furthermore, wherever possible, we use the original identifiers and only create our own URIs (e.g., for CHOP) where no URIs are provided by the SDO. This approach benefits from having resolvable (dereferenceable) URIs (e.g., for SNOMED CT) where a user can enter the URI in a browser and be redirected to the concept on the SDO homepage. However, we rely on the proper identifier management of the SDO, including backward compatibilities. Finally, it is the responsibility of the institution to comply with the licensing terms and requirements for each terminology of the SDO.

The SPHN DCC Terminology Service, in its current version, does not support mappings between terminologies, i.e., the process does not currently generate interlinks between the individual codes. For example, relations between ICD-10-GM diagnosis codes and SNOMED CT codes from the hierarchy of disorders are not considered. All the terminologies are handled individually. Mappings exist between the terminologies provided by several SDOs, but these are under separate licenses, which creates an additional administrative burden that hinders the process of making them available through a terminology service for a distributed environment. Nevertheless, since maps are of high value in the SPHN environment, due to the use of terminologies with partly overlapping content, we plan to host mappings and relations between semantic standards on the SPHN DCC Terminology Service in the future. In addition, depending on the community's needs, further terminologies will be included in this pipeline and made available in the future. For example, the Swiss list of pharmaceutical specialities (Spezialitätenliste) [26], which is a list of all the medication approved and reimbursed in Switzerland, is likely to be included. The list provides the Global Trade Item Number (GTIN) code of medicinal products, to-

gether with the ATC code(s) of their active ingredient(s). Therefore, a researcher can easily derive the active ingredients of a medicinal product, which is normally of interest, from its product code.

5. Conclusions

Despite the availability of many terminologies to clarify the clinical and biomedical terms used in different domains and to be shared amongst users, their adoption and use in the clinical setting occurs at a very slow pace. By providing the relevant terminologies via a terminology service, in the context of the SPHN, the goal is to facilitate the use and the adoption of these terminologies in the SPHN community. The SPHN DCC Terminology Service offers an automated platform incorporating various clinical and biomedical terminologies of interest that comply with the institutional security restrictions. With the federated distribution of these terminologies, SPHN data users have access to an integrated set of resources of standard terminologies while the safety of sensitive resources is ensured when exploring them.

Author Contributions: Conceptualization, P.K., K.G., K.C. and S.Ö.; methodology, P.K.; software, V.T. and P.K.; validation, P.K.; writing—original draft preparation, S.Ö. and P.K.; writing—review and editing, all authors. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded in the framework of the Swiss Personalized Health Network (SPHN).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable for studies not involving humans.

Code Availability Statement: All code is available under the GNU General Public License v3.0 license on <https://git.dcc.sib.swiss/dcc/biomedit-rdf> accessed on 30 October 2021.

Acknowledgments: The authors would like to thank Shubham Kapoor and the sciCORE team of the University of Basel for their support in the setup of the MinIO server.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Lawrence, A.K.; Selter, L.; Frey, U. SPHN—The Swiss Personalized Health Network Initiative. *Stud. Health Technol. Inform.* **2020**, *270*, 1156–1160. [CrossRef] [PubMed]
- Österle, S.; Touré, V.; Crameri, K. The SPHN Ecosystem Towards FAIR Data. *Preprints* 2021. [CrossRef]
- Gaudet-Blavignac, C.; Raisaro, J.L.; Touré, V.; Österle, S.; Crameri, K.; Lovis, C. A national, semantic-driven, three-pillar strategy to enable health data secondary usage interoperability for research within the swiss personalized health network: Methodological study. *JMIR Med. Inform.* **2021**, *9*, 1–10. [CrossRef] [PubMed]
- Wilkinson, M.D.; Dumontier, M.; Aalbersberg, I.J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.-W.; da Silva Santos, L.B.; Bourne, P.E.; et al. Comment: The FAIR Guiding Principles for Scientific Data Management and Stewardship. *Sci. Data* **2016**, *3*, 1–9. [CrossRef] [PubMed]
- SNOMED CT. Available online: <https://www.snomed.org/> (accessed on 1 October 2021).
- McDonald, C.J.; Huff, S.M.; Suico, J.G.; Hill, G.; Leavelle, D.; Aller, R.; Forrey, A.; Mercer, K.; DeMoor, G.; Hook, J.; et al. LOINC, a universal standard for identifying laboratory observations: A 5-year update. *Clin. Chem.* **2003**, *49*, 624–633. [CrossRef] [PubMed]
- ICD-10 GM. Available online: <https://www.dimdi.de/dynamic/en/classifications/icd/icd-10-gm> (accessed on 1 October 2021).
- Bundesamt für Statistik. Medizinisches Kodierungshandbuch. *Der Offizielle Leitfaden der Kodierrichtlinien in der Schweiz*. 2022. Available online: <https://www.bfs.admin.ch/bfs/de/home/statistiken/gesundheits.assetdetail.17304223.html> (accessed on 26 November 2021).
- Schweizerische Operationsklassifikation (CHOP). Bundesamt für Statistik, 2021 (BFS). Available online: <https://www.bfs.admin.ch/bfs/de/home/statistiken/kataloge-datenbanken/publikationen.assetdetail.14880301.html> (accessed on 1 October 2021).
- ATC. Available online: <https://www.whocc.no> (accessed on 1 October 2021).
- UCUM. Available online: <http://unitsofmeasure.org> (accessed on 1 October 2021).
- Whetzel, P.L.; Noy, N.F.; Shah, N.H.; Alexander, P.R.; Nyulas, C.; Tudorache, T.; Musen, M. BioPortal: Enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic Acids Res.* **2011**, *39*, 541–545. [CrossRef] [PubMed]
- Metke-Jimenez, A.; Steel, J.; Hansen, D.P.; Lawley, M. Ontoserver: A Syndicated Terminology Server. *J. Biomed. Semant.* **2018**, *9*, 1–10. [CrossRef] [PubMed]

14. Ontoserver. Available online: <http://ontoserver.csiro.au/> (accessed on 16 October 2021).
15. LOINC FHIR Terminology Server. Available online: <https://loinc.org/fhir/> (accessed on 17 October 2021).
16. Bodenreider, O. The Unified Medical Language System (UMLS): Integrating Biomedical Terminology. *Nucleic Acids Res.* **2004**, *32*, D267–D270. [[CrossRef](#)] [[PubMed](#)]
17. De Quirós, F.G.B.; Otero, C.; Luna, D. Terminology Services: Standard Terminologies to Control Health Vocabulary. *Yearb. Med Inform.* **2018**, *27*, 227–233. [[CrossRef](#)]
18. Coman Schmid, D.; Crameri, K.; Oesterle, S.; Rinn, B.; Sengstag, T.; Stockinger, H. SPHN—The BioMedIT Network: A Secure IT Platform for Research with Sensitive Human Data. *Stud. Health Technol. Inform.* **2020**, *270*, 1170–1174. [[CrossRef](#)] [[PubMed](#)]
19. WC3 RDF. Available online: <https://www.w3.org/RDF/> (accessed on 1 September 2021).
20. Shahin, M.; Babar, M.A.; Zhu, L. Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices. *IEEE Access* **2017**, *5*, 3909–3943. [[CrossRef](#)]
21. SNOMED OWL Toolkit. Available online: <https://github.com/IHTSDO/snomed-owl-toolkit> (accessed on 1 September 2021).
22. UCUM list. Available online: <https://ucum.org/trac/wiki/adoption/common> (accessed on 5 August 2021).
23. Apache Jena Fuseki. Available online: <https://jena.apache.org/documentation/fuseki2/> (accessed on 3 July 2021).
24. SPARQL ASK. Available online: <https://www.w3.org/TR/2013/REC-sparql11-query-20130321/#ask> (accessed on 26 November 2021).
25. Makefiles. Available online: <https://www.gnu.org/software/make/manual/make.html> (accessed on 29 November 2021).
26. Schweizer Spezialitätenliste. Available online: <http://www.spezialitätenliste.ch/> (accessed on 29 November 2021).