Enabling object-based documentation of existing bridge inspection data using Linked Data

Anne Göbels[1]

[1]Chair of Design Computation, RWTH Aachen University, Schinkelstraße 1,52062 Aachen, Deutschland

E-mail(s): goebels@dc.rwth-aachen.de

Abstract: Digital Twins (DTs) of existing bridges can provide a reliable data basis for predictive maintenance needed to ensure an operational infrastructure as the number of bridges in critical condition increases. An essential data source for a DT is the existing construction and inspection documentation stored in Germany in the relational database system *SIB-Bauwerke*. However, SIB-Bauwerke does not support object-specific documentation of inspection data, making it challenging to retrieve the condition of individual components and to link the data appropriately with other data sources of the DT. To address these issues, we present an automatic transmission of the existing SIB-Bauwerke data into an object-oriented structure, allowing detailed recording and analysis of inspection data and precise linking to external data objects. Based on a previously developed conversion of the inspection data into Linked Data graphs, we enhance the converted data by introducing direct links between components and damages and defining damaged areas. The newly implemented structure allows sophisticated and detailed statements about the condition of individual components or areas and can be easily linked at the object level with other resources of the DT.

Keywords: Infrastructure Maintenance, Digital Twin, Linked Data, SIB-Bauwerke, ASB-ING

1 Introduction

Bridge maintenance is currently performed reactively based on detected deficiencies during a manual inspection. However, as the number of bridges in critical condition increases, a predictive maintenance strategy is needed that ensures a sustainable working infrastructure through reliable predictions and preventive measures.

A prerequisite for predictive maintenance is a comprehensive and reliable data basis on which analyses can be performed. In this context, the TwinGen research project investigates how Digital Twins (DTs) of existing bridges can be generated using automated methods. The DT (in the project's scope) is a semantic-geometric model that represents a precise digital replica of a bridge. It is based on a geometry model linked on the object-level with complementary, external data sources, including point clouds data, damage pictures, construction plans, and inspection reports. Linked Data methods are used to achieve expressive and accurate links between these heterogeneous and distributed data sources.

Existing bridge documentation and inspection data are essential data sources to represent a bridge's entire lifespan as they provide information on the construction, previous maintenance, and the current condition. In Germany, this data is documented in *SIB-Bauwerke*, a documentation program that works with a relational database, whose structure is defined in the national data model *ASB-ING* [1]. This database is not object-oriented and only allocates damage information to the individual components via textual descriptions. Thus, it is challenging to assess individual components' conditions and to link the inspection information of SIB-Bauwerke appropriately to external data, such as IFC model elements.

To address these issues, we developed an automated transmission of the existing SIB-Bauwerke data into an object-oriented documentation structure, allowing precise linking and analyzing of the data. In prior work, we converted the inspection data into a Resource Description Format (RDF) graph [2] using the previously developed ASB-ING Ontology [3] . However, the graph still contains the same information and detail depth as the original data. In the work presented in this paper, we use Linked Data functionalities, such as SPARQL queries and the inclusion of external ontologies, to transform the implicit information about damage to components into explicit objects and links. We improve damage documentation by introducing precise linking between damages and individual components and defining damaged areas. At the same time, the original, mandatory data structure of the ASB-ING is maintained to ensure backward compatibility.

2 Related research activities

Representing and processing building and maintenance data using Linked Data technologies is subject to several research and development efforts. The ifcOWL Ontology [4] is the RDF-compatible version of the IFC schema, thus enabling the representation of IFC models in Linked Data. Bridge structures can be represented in Linked Data using the Bridge Ontology (BROT) [5]. It enables to describe bridge zones and components, including their topological relations. It is a generic, modular ontology that allows for domain-specific extension, e.g., building materials or component details.

The Damage Topology Ontology (DOT) [6] of Hamdan and Scherer provides classes and properties to document and classify damage elements or patterns and assign them to building elements. To highlight the specific damaged area of an element, they developed the Area of Interest (AOI) Ontology [7]. The AOI Ontology contains classes that subdivide an element into defined areas (e.g., bottom, top, horizontal/vertical center, peripheral areas, or external/internal area). The classes and properties of the AOI Ontology can be used in addition to other ontologies, such as the DOT Ontology, to provide a more detailed location of the damage.

An approach by Liu et al. presented in [8] also addresses the interlinking of BIM models and SIB-Bauwerke data using Linked Data. They store newly acquired element condition grades linked to model elements in an Information container for linked document delivery (ICDD) [9]. The BOT Ontology is used to represent the model elements, and the ASB-ING Condition Ontology [2] to store the condition grades compatible to SIB-Bauwerke in Linked Data. With a custom mapping ontology, the data of the ICDD gets transferred into the database tables of SIB-Bauwerke, by automatically generated SQL commands. However, their approach deals with newly acquired data and only covers a small part of the entire SIB-Bauwerke data. Nevertheless, the derivation of SQL commands from Linked Data graphs can support the backward compatibility of our approach.

Since the presented approaches and ontologies mainly focus on processing and storing new acquired (inspection) data, the applicability of existing data to the proposed methods is not addressed. However, we present in this paper how existing data can also benefit from these developments.

3 Background and previous work steps

The *Anweisung Straßeninformationsbank – Teilsystem Bauwerksdaten* (ASB-ING) (engl.:Instruction road database - subsystem structural data) [1] is the national data model for infrastructure documentation. It specifies how information on the structure and inspection must be recorded, including condition assessments, damages, and maintenance measures. To ensure uniform documentation, the ASB-ING provides predefined values stored in extensive key-value tables containing, e.g., possible component-, damage-, or material specifications.

The ASB-ING is implemented in the *Straßeninformationsbank- Bauwerke* (SIB-Bauwerke) (engl.:Road information database - structures), a relational database application that stores the inspection results and structural data of a bridge since its construction. According to the ASB-ING structure, the bridge documentation is separated into tables, covering different subjects. The structural data is divided into component groups (e.g., superstructure, substructure, bearings, coverings, etc). These groups are used for damage assignments, by which a condition grade is calculated for each component group following a defined algorithm [10]. These grades lead to the overall bridge condition assessment. Textual descriptions assign damages to individual components, but these are only intended to retrieve the damage on site.

The management of data that gets updated at each inspection (damages, measures, inspection report) is handled by using two tables for each topic. One table contains the current data of the latest inspection, and another table archives all previous information. For example, a damage existing for five years has been documented at each inspection and is finally stored once in the current data table and four times in the previous data table. These damage versions are indirectly linked by having the same Damage-ID.

The current published version of the ASB-ING is from 2013, but recently, there has been ongoing development of a new version to update the data structure and content, provided as a UML model. It

[2]https://roadotl.eu/static/eurotl-ontologies/testontologies/Germany/asb-ing-condition_doc/index-en.html

defines the content of the ASB-ING using 120 classes, 80 datatype definitions, and 500 properties. By converting this new ASB-ING UML model, we created the *ASB-ING Ontology* [3] [3].

The ASB-ING Ontology is the basis for converting the inspection data sets of SIB-Bauwerke into RDF, as it provides the required ontological framework to represent the data. Combined with mapping tables between the ASB-ING 2013 and the new version (provided by the official migration project), we enabled an automatic conversion process of the SIB-Bauwerke datasets into RDF graphs [2].

After converting the data, the graph of each bridge data set reflects the same information structure and quality as the original data. It records individual construction components and detailed damage documentation, but these are not linked, thus preventing an object-based organization of inspection data. In addition, the method of updating damage data leads to data repetition and makes it challenging to track damage conditions.

4 Methodological approach

A review of the current damage documentation of SIB-Bauwerke identifies its weaknesses. Each damage object contains information about the affected component group, element type, and the location of the element and/or damaged area. In addition, it provides information about the damage type and size, the damage ID, and the impact of the damage on the stability, traffic safety, and durability of the bridge.

This method of damage documentation merges many different types of information into one data object. These data types can be broadly categorized into static and dynamic/variable data. The static data, such as the description of the affected component and the damage ID, stays the same over time, while the variable data can change at each inspection. The combination of these data types in one object leads to repetition, ineffective data processing, and complicates the proper handling of the data concerning an object-based organization.

Thus, we split the static data into separate elements, each containing only one type of information. The separation allows independent and flexible processing of the different information.The new elements were defined using additionally developed classes of the ASB-ING Ontology (prefix: asb) or existing classes of the DOT and AOI Ontology. Thus, they also express a more specific semantic meaning.

Figure 1 shows a (simplified) example of an original damage object of our sample data set from SIB-Bauwerke after the conversion into RDF. The figure indicates how the damage information is separated and stored as new objects. We introduce a new class (*asb:SchadenObjekt*) for creating a parent damage object that serves as a container for the individual damage versions (the original damage objects). This parent damage object takes the damage ID of the original damage and is used as a reference for linking the damage to the component. Thus, only one link between the component and the damage object must be created instead of duplicated links for each damage version.

[3] https://w3id.org/asbingowl/core

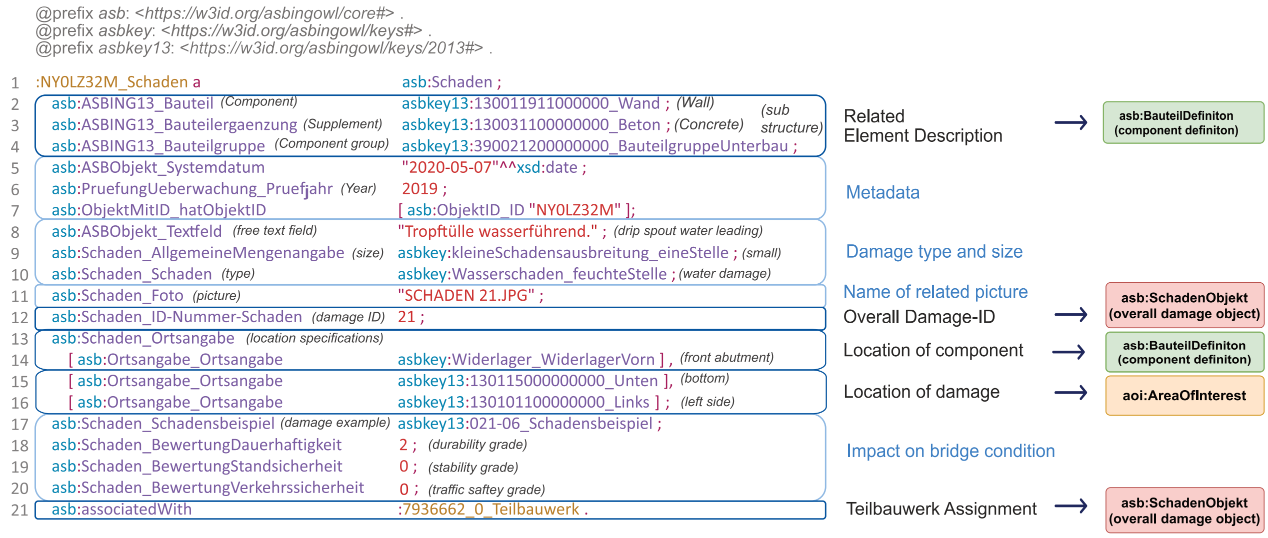

Figure 1: Example of an SIB-Bauwerke damage object after conversion into RDF and its separation

The properties that describe the component and its location are summarized with the new class *asb:BauteilDefinition* (Component definition). The location of the damage on the component is defined as an *aoi:AreaOfInterest*, according to the AOI Ontology of Hamdan and Scherer [7].The variable data stays in the original data objects (damage version) that remain unchanged to maintain compatibility with the original SIB-Bauwerke structure.

The interlinking of the new implemented elements using self-defined, and existing properties of the AOI and DOT Ontology, can be seen in figure 2. The damage versions are connected to the new parent object that is linked to the component definition and the area of interest. We retrieve the individual component in another process based on the component definition (see section 4.1). The component is then linked to the parent damage object or the area of interest. Thus, we achieved detailed and precise object-oriented documentation allowing direct and unambiguous assignment of damages to individual components.

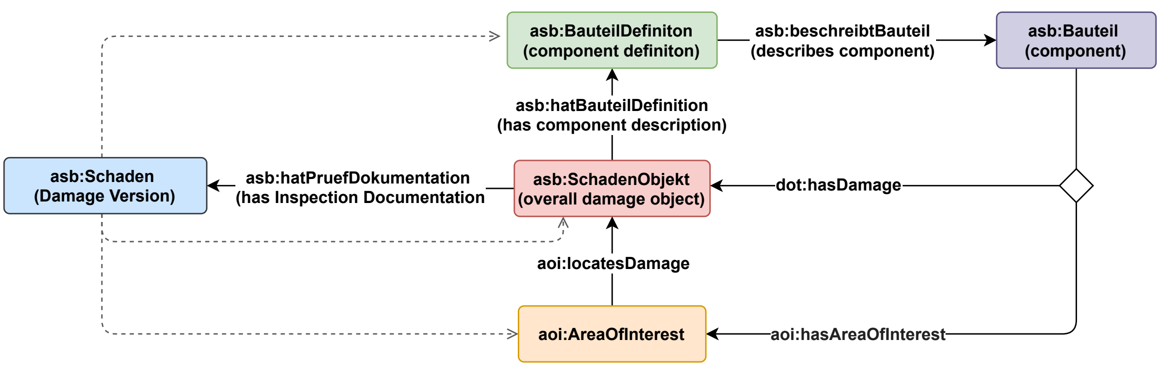

Figure 2: Proposed object-oriented structure for damage documentation

4.1 Implementation process

The implementation was tested on a dataset of SIB-Bauwerke containing the construction documentation and the inspection reports of a multi-span girder bridge from 2002 with 40 damages.The current code of the implementation process can be found at GitHub [4].

The starting point for implementing the proposed structure is the creation of the parent damage objects by querying the graph for all distinct damage IDs. Next, a component definition (*asb:BauteilDefinition*) is developed for each parent damage object, reusing the original damage version's component group and type specification (see rows 4 and 2, figure 1).

Adding the individual location to the component definition needs an intermediate step. The original structure does not distinguish between component locations and damage areas. Both are values of the property *asb:Schaden-Ortsangabe* (location specification) (see rows 13-16, figure 1). The meaning of location terms also depends on the component type, e.g., "left" can describe the left area of an abutment wall, but also the whole left cap. Thus, we developed a table that defines the meaning per location term and component type. Based on this table, the location information is either added to the component definition or used to create a damage area (area of interest).

Next, each parent damage object is related to a component definition via *asb:hatBauteilDefinition* and optionally to an area of interest via *aoi:locatesDamage* (see Figure 2).

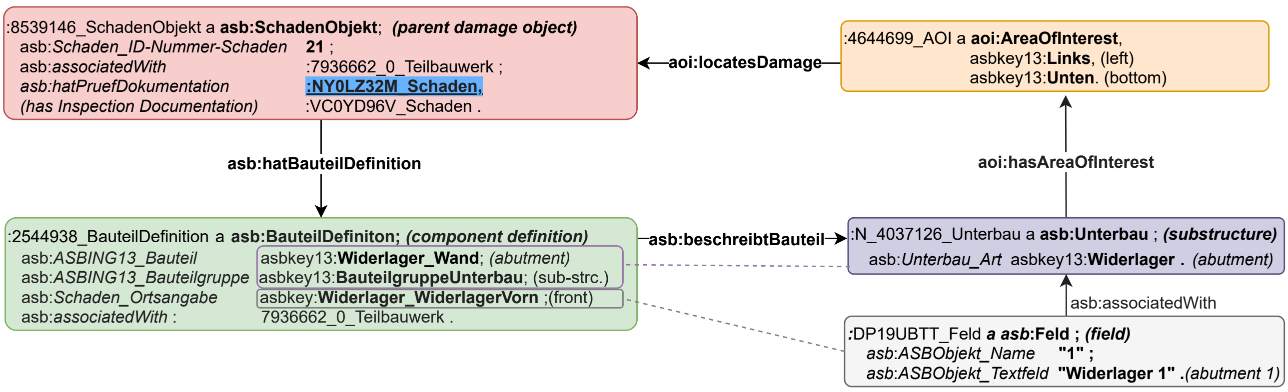

Figure 3: Implementation of the object-oriented structure for an example damage

The retrieval of the individual components based on the component definitions is done by comparing specific properties of both objects using SPARQL queries. Figure 3 shows the implemented structure for the damage object introduced in figure 1. In this example, the component definition is related to the front abutment because its component group and type (group: substructure, type: abutment wall) corresponds to the individual component's class and type (class: substructure, type: abutment). The location of "Front Abutment" of the component definition matches the values “1. Field” resp. “Abutment 1” of the field object associated with the abutment component. The relation between both objects is stored using *asb:beschreibtBauteil* (describes component).

[4]https://github.com/AnneGoebels/SIB-Bauwerke-To-Linked-Data

Thus, broadly defined, the value of the component group property matches the components' class, but all further details depend on the component type, which leads to individual handling in the script for each type. In the sample data set, this method was able to find most of the described components, except for smaller equipment, such as pipes and drains. Finally, the retrieved individual components are linked to the corresponding parent damage objects using *dot:hasDamage*, or to the area of interest, using *aoi:hasAreaOfInterest*. After this process is completed, the component definitions do not appear to be needed, but they can be used as a substitute for components that could not be found or are not present as individual components in the SIB-BW data set.

5 Conclusion

We achieved an automatic transformation of the existing inspection data from SIB-Bauwerke into an object-oriented structure. This structure enables precise linking with other object-based data models such as IFC models, allowing efficient use of existing inspection data in digital twins. Moreover, it is now possible to query for damages to individual components and most damaged areas of components or a bridge. If many SIB-Bauwerke data sets were converted using this process, an extensive knowledge database could be created to provide information about which component (area) is most affected by which type of damage and how they progress. The object-oriented structure also enables a detailed condition assessment of the bridge. Currently, an algorithm [10] is used in which the worst grade of a damage determines the grade of the component group. The worst component group determines the grade of the entire bridge. Applying this algorithm to individual components (i.e., the damage determines only the grade of the affected component) can lead to more precise, tailored, and efficient maintenance measures.

However, the process still needs to be tested on more data sets. Especially the assignment of component definitions to the components is currently highly adapted to one sample data set that describes a simple bridge construction with comparatively few damages. The extent to which this procedure can be generalized to more complex data sets needs to be investigated. In ongoing work, we are using the presented developments to enable an automatic linking of the components of SIB-Bauwerke to elements of the IFC model. Based on that, we create a prototypical viewer that can visualize the geometry model and the linked inspection data, including spatial representations of the damages. It will be able to perform space-, time-, and object-based queries to get new insights into the data.

Acknowledgements

We thank the German Federal Ministry for Digital and Transport (BMDV) for their support in funding the TwinGen research project and the Autobahn GmbH des Bundes - Niederlassung Südbayern for providing us the SIB-Bauwerke data set.

References

[1] Bundesministerium für Verkehr, Bau und Stadtentwicklung, Abteilung Straßenbau., *ASB-ING 2013 Anweisung Straßeninformationsbank Segment Bauwerksdaten*. 2013.

[2] A. Göbels, “Conversion of infrastructure inspection data into Linked Data models”, in *32. Forum Bauinformatik 2021*, Darmstadt, 2021.

[3] A. Göbels and J. Beetz, “Conversion of legacy domain models into ontologies for infrastructure maintenance”, in *Proceedings of the 9th Linked Data in Architecture and Construction Workshop - LDAC 2021*, vol. 3081, Luxembourg: CEUR-WS, 2021, p. 12.

[4] J. Beetz, J. van Leeuwen, and B. de Vries, “IfcOWL: A case of transforming EXPRESS schemas into ontologies”, en, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 23, no. 1, pp. 89–101, Feb. 2009.

[5] A.-H. Hamdan and R. J. Scherer, “Integration of BIM-related bridge information in an ontological knowledgebase”, in *Proceedings of the 8th Linked Data in Architecture and Construction Workshop - (LDAC)*, CEUR-WS, 2020, pp. 77–90.

[6] A.-H. Hamdan, M. Bonduel, and R. J. Scherer, “An ontological model for the representation of damage to constructions”, in *Proceedings of the 7th Linked Data in Architecture and Construction Workshop - (LDAC)*, Lisbon: CEUR-WS, 2019, pp. 64–77.

[7] H. Al-Hakam and R. J. Scherer, “Areas of Interest - Semantic description of component locations for damage assessment”, en, in *EG-ICE 2020 Proceedings: Workshop on Intelligent Computing in Engineering*, Berlin, Jul. 2020.

[8] L. Liu, P. Hagedorn, and M. König, “BIM-Based Organization of Inspection Data Using Semantic Web Technology for Infrastructure Asset Management”, in *Proceedings of the 1st Conference of the European Association on Quality Control of Bridges and Structures*, C. Pellegrino, F. Faleschini, M. A. Zanini, J. C. Matos, J. R. Casas, and A. Strauss, editors, vol. 200, Cham: Springer International Publishing, 2022, pp. 1117–1126.

[9] “ISO 21597-1:2020 Information container for linked document delivery - Exchange specification - Part 1: Container”, International Organization for Standardization, Geneva, Standard, 2020.

[10] P. Haardt, *Algorithmen der Zustandsbewertung von Ingenieurbauwerken*. Bergisch Gladbach, 1999.