Advances in Engineering Software 000 (2017) 1–21

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

Research paper

An information modeling framework for bridge monitoring

Seongwoon Jeong[a,*], Rui Hou[b], Jerome P. Lynch[c], Hoon Sohn[c], Kincho H. Law[a]

[a] *Department of Civil and Environmental Engineering, Stanford University, 473 Via Ortega, Stanford, CA 94305-4020, USA*
[b] *Department of Civil and Environmental Engineering, University of Michigan, 2350 Hayward St., Ann Arbor, MI 48109-2125, USA*
[c] *Department of Civil and Environmental Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea*

ARTICLE INFO

Article history:
Received 25 January 2017
Revised 5 April 2017
Accepted 13 May 2017
Available online xxx

Keywords:
Bridge information modeling
Bridge management
Bridge monitoring
NoSQL database

ABSTRACT

Bridge management involves a variety of information from different data sources, including geometric model, analysis model, bridge management system (BMS) and structural health monitoring (SHM) system. Current practice of bridge management typically handles these diverse types of data using isolated systems and operates with limited use of the data. Sharing and integration of such information would facilitate meaningful use of the information and improve bridge management, as well as enhance bridge operation and maintenance and public safety. In many industries, information models and interoperability standards have been developed and employed to facilitate information sharing and collaboration. Given the success of building information modeling (BIM) in the Architecture, Engineering and Construction (AEC) industry, efforts have been initiated to develop frameworks and standards for bridge information modeling (BrIM). Current developments of BrIM focus primarily on the physical descriptions of bridge structures, such as geometry and material properties. This paper presents an information modeling framework for supporting bridge monitoring applications. The framework augments and extends the prior work on the OpenBrIM standards to further capture the information relevant to engineering analysis and sensor network. Implementation of the framework employs an open-source NoSQL database system for scalability, flexibility and performance. The framework is demonstrated using bridge information and sensor data collected from the Telegraph Road Bridge located in Monroe, Michigan. The results show that the bridge information modeling framework can potentially facilitate the integration of information involved in bridge monitoring applications, and effectively support and provide services to retrieve and utilize the information.

1. Introduction

Bridge management involves copious and diverse information, including geometry, engineering model, inspection reports and monitoring data. Current practice of bridge management employs isolated systems to manage and process different types of information wherein information managed in one system is neither shared among other systems nor integrated with information managed by other systems. However, as bridge monitoring and management technologies advance and the number of bridge monitoring and management applications increases, the demand for efficient information sharing and data exchange will grow. Sharing and integration of such information would enable meaningful uses of information and improve bridge management services, as well as enhance bridge operation, maintenance and public safety.

Much research has been conducted in developing data exchange and interoperability standards in many industry domains to avoid error-prone and time-consuming manual data conversion as well as to facilitate automated exchange of information and machine-to-machine interaction [18,27]. In the architecture, engineering and construction (AEC) industry, building information modeling (BIM) has been widely adopted as a means to support integrated project delivery process and data exchange throughout the project lifecycle [12]. One of the *de facto* BIM standard data models is the Industry Foundation Classes (IFC) [3]. The IFC standard specifies platform-neutral file format using EXPRESS modeling language to enable digital data exchange among building design and analysis systems. The IFC-EXPRESS schema has been translated into XML (eXtensible Markup Language) format, a commonly used representation of industry standards. [3].

Given the success of BIM, research efforts have been initiated to develop frameworks and standards for bridge information modeling (BrIM). The main objectives of BrIM research are twofold: enabling an integrated bridge data repository and developing elec-

---

[*] Corresponding author.
E-mail address: swjeong3@stanford.edu (S. Jeong).

http://dx.doi.org/10.1016/j.advengsoft.2017.05.009
0965-9978/© 2017 Published by Elsevier Ltd.

Please cite this article as: S. Jeong et al., An information modeling framework for bridge monitoring, Advances in Engineering Software (2017), http://dx.doi.org/10.1016/j.advengsoft.2017.05.009

tronic data exchange standards for bridge applications [5]. Focusing on the spatial and physical entities of bridge structures, IFC-Bridge for bridge modeling has been proposed [39]. There have also been research efforts towards developing information modeling framework for bridge management. Marzouk and Hisham [26] proposed a BrIM framework to support bridge management by connecting a 3-dimensional bridge model, inspection sheets and structural condition assessment modules. Samec et al. [36] developed a BrIM system, along with a 3-dimensional visualization tool and a mobile application, to manage bridge life cycle information that includes inspection and maintenance data. To facilitate information interoperability in the bridge domain, one of the most notable efforts for developing software-neutral BrIM data schema is the OpenBrIM standards [6]. Supported by the US Federal Highway Administration (FHWA), OpenBrIM is “a bridge industry consensus standard for engineering data description, modeling, and interoperability for integrated structural design, construction, and life-cycle management of bridges” (see https://collaboration.fhwa.dot.gov/dot/fhwa/ascbt/brim/). OpenBrIM uses XML as the basic syntax to define object-based information model [30]. OpenBrIM (version 1.0 and version 2.0) was originally developed by the research team at the State University of New York at Buffalo [4]. The current version (version 3.0) of OpenBrIM is led by CH2M Hill and sponsored by FHWA [1]. The efforts so far have been focused mostly on the 3-dimensional representation of bridge structures. As such, current OpenBrIM standards lack the data entities for representing the information pertinent to bridge monitoring applications. In our framework, we use the OpenBrIM as the base schema and extend it to include data entities for capturing bridge monitoring information. Specifically, we extend the standard OpenBrIM schema to include engineering entities for analysis modeling and sensor information.

Relational database (RDB) systems are often employed as the primary data storage for bridge modeling, bridge management and bridge monitoring applications [23,24,26,35,38,41]. However, the tabular structure of RDB is not convenient for handling semi-structured data (e.g., XML document and tagged-text) and unstructured data (e.g., text and image), which are commonly found in engineering applications. For bridge monitoring, the collected sensor data need to be managed and processed efficiently. Because of the variable lengths in the data records, the data sets are not conveniently structured in a RDB system. Research has shown that RDB systems have fundamental drawbacks in satisfying the performance and scalability requirements for the new era of “big data” with a variety of formats and a large volume of data [37]. To overcome the shortcomings of RDB systems for handling big data, NoSQL (Not-Only-SQL) database systems have been widely adopted in applications such as real-time analysis, knowledge representation and large-scale data management [15]. Recent studies have shown that, in comparing to RDB systems, NoSQL systems enable higher scalability, better flexibility and faster performance by reducing some of the consistency requirements and supporting more flexible data schemas [14,25]. Because of their flexibility, scalability and performance, NoSQL database systems can be an effective alternative for handling bridge monitoring and management data and supporting bridge monitoring applications [20].

In this paper, we present a bridge information modeling (BrIM) framework for bridge monitoring applications. The framework facilitates data exchange and integration of information involved in bridge management applications. The BrIM framework adopts and extends the data schema of the OpenBrIM standards to facilitate data interoperability among bridge monitoring and management applications. We define data entities to capture information that is needed for bridge engineering analysis, sensor description and bridge monitoring. The BrIM framework also provides data link to the time-series sensor data and image data so as to allow users to locate the data via the information model. Apache Cassandra database [21], an open-source column family NoSQL database, is employed as the backend database system to guarantee flexibility and scalability of the framework. Scripts are written to illustrate data exchange between different data formats, such as Cassandra data schema, BrIM model and data model required by bridge monitoring applications. To demonstrate the NoSQL-based BrIM framework, the bridge model and the sensor data collected from the Telegraph Road Bridge (TRB) located in Monroe, Michigan are employed in this study.

2. Bridge information model

This section describes a bridge information model designed for bridge monitoring applications. Engineering information modeling, such as BIM, typically adopts an object-based approach in which an information model is composed of objects, each of which contains attributes about the object [11]. Information modeling standards and tools specify a predefined set of object families that are used to capture the data entities involved in the targeted domain. For instance, the OpenBrIM standards include object families for describing 3-dimensional geometry of bridge structure [2]. The bridge information modeling (BrIM) schema described herein extends the OpenBrIM schema with newly defined objects for representing engineering analysis model and sensor information. New objects are identified based on relevant standards and software tools. Specifically, we examine CSI Bridge (a structural modeling and analysis software tool) and SensorML (an open standard for sensor description) to ensure that the BrIM is capable of supporting typical applications in bridge engineering.

2.1. OpenBrIM

The OpenBrIM standards describe a bridge structure as a collection of hierarchical objects and their parameters [2]. Each object represents either a physical entity (e.g., beam, column and deck) or a conceptual entity (e.g., project, group and unit system) of a bridge structure. On the other hand, each parameter either represents a property (e.g., length, width and thickness) of an object or refers to another object. Fig. 1(a) and (b) show the schema definitions of a basic “`Object`” entity and a “`Parameter`” entity, respectively, in OpenBrIM [29]. The data schema of the basic “`Object`” entity includes attributes, such as `N` (name), `X`, `Y` and `Z` (coordinates), `RX`, `RY` and `RZ` (angles of rotation), and `AX`, `AY` and `AZ` (angles of rotation about the origin of the 3-dimensional workspace). Similarly, the data schema of basic “`Parameter`” entity includes attributes, such as `V` (value), `T` (type), `D` (description), `UC` (name of unit system), `UT` (type of unit), `Category` (category of the parameter) and `Role` (specifying whether a user can edit the parameter). The data schema of any other data entities in OpenBrIM is defined by extending the basic “`Object`” and “`Parameter`” entities.

To encode bridge information, OpenBrIM standards use ParamML, which is a variation of the extensible markup language (XML) for engineering applications [2]. For example, Fig. 2(a) shows the data schema of the `Shape` object written in the XML schema definition (XSD) format [29]. In the schema definition, `xs` refers to the XML schema namespace (http://www.w3.org/2001/XMLSchema). The definitions of the data components from the XML schema namespace are as follows (http://www.w3schools.com/xml).

- `xs:complexType` includes other elements and/or attributes.
- `xs:complexContent` specifies extensions or restrictions on a `xs:complexType` element.
- `xs:extension` extends `xs:complexType` element.

Please cite this article as: S. Jeong et al., An information modeling framework for bridge monitoring, Advances in Engineering Software (2017), http://dx.doi.org/10.1016/j.advengsoft.2017.05.009