Automation in Construction 77 (2017) 24–51

Contents lists available at ScienceDirect

Automation in Construction

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

JavaScript Object Notation (JSON) data serialization for IFC schema in web-based BIM data exchange

Kereshmeh Afsari [a,*], Charles M. Eastman [a], Daniel Castro-Lacouture [b]

[a] School of Architecture, Georgia Institute of Technology, 245 4th St NW, Atlanta, GA 30332, USA
[b] School of Building Construction, Georgia Institute of Technology, 276 Fifth Street, Atlanta, GA 30332, USA

ARTICLE INFO

Article history:
Received 6 May 2016
Received in revised form 11 November 2016
Accepted 10 January 2017
Available online xxxx

Keywords:
JSON
IFC
ifcXML
Cloud interoperability
Precast concrete

ABSTRACT

In the building industry, building data such as objects and processes are described in Industry Foundation Classes (IFC) data model schema to support a neutral data exchange format for BIM tools interoperability. While IFC specification has been encoded in ifcXML format by buildingSMART to support XML-based data transmission, there is a lack of studies on the implementation of IFC specification using JavaScript Object Notation (JSON) serialization. JSON is a key-value style lightweight data exchange format that has higher parsing efficiency than XML and due to the inadequacies of XML, JSON has been widely used in Web applications, specifically in Asynchronous JavaScript and XML (AJAX) Web services. This paper highlights the need for JSON implementation of IFC specification and introduces ifcJSON Schema and its data content. The main objective of this study is to outline how IFC specification can be represented in JSON format. Therefore, the study explains the implementation of the IFC standard as a JSON schema to guide the creation of JSON documents. The ifcJSON documents can be used for web-based data transfer as an alternative to XML documents. Since current IFC specification release is IFC4 Add1, the implementation of ifcJSON4 schema is specified and guidelines for generating and validating ifcJSON documents are described. Additionally, this paper implements ifcJSON4 schema in a use case within the precast concrete domain by indicating the data content for a precast building element with its corresponding geometry representation, product placement, and owner history data. The analysis of results indicates that ifcJSON4 schema developed in this paper is a valid JSON schema that can guide the creation of valid ifcJSON documents to be used for web-based data transfer and to improve interoperability of Cloud-based BIM applications.

1. Introduction

Existing methodologies for Cloud-BIM data integration and data exchange has been studied in a prior research [1] that highlighted challenges in current model-based data exchange and pointed out that current methodologies have not fully exploit the potential of the Cloud. Most importantly, while vendor specific data formats are quite diverse, they are based on multiple and different data schemas, and data standards for Cloud-based cross-platform data exchange purposes are limited. In the building industry, building data such as objects and processes is described in Industry Foundation Classes (IFC) data model schema to support a neutral data format for BIM tools interoperability. IFC schema defines a set of generic building objects with associated attributes and properties as well as multiple shape definition methods for the objects [2].

IFC data can be encoded in multiple file formats for various purposes [3]. In fact, data serialization is defined as encoding objects or translating data into a format that can be stored in a file, memory or a database or that can be sent to other application. In fact, a proper data serialization format can affect data transmission rates and performance significantly [4]. So far, IFC specification is provided as EXPRESS and XSD definitions [5] and IFC data files that are being exchanged between applications use three main data formats including ".ifc" using the STEP physical file structure, ".ifcXML" using the eXtensible Markup Language (XML) document structure, and ".ifcZIP" using the PKzip 2.04 g compression algorithm [3].

On the other hand, XML and JavaScript Object Notation (JSON) are two different data serialization formats used in web applications [6,7]. These two approaches are applied in data transmission between web applications which typically are the application in Asynchronous JavaScript and XML (AJAX). Since XML and JSON have different features, they have been used for different situations but often only one approach is used in development to ensure unity and readability [7]. buildingSMART International has implemented the IFC standard using XML technologies as ifcXML specification [8,9]. XML is a platform independent language for representing data and has been used in the development of web service applications. However, the performance of web services have shown a significant decrease when using XML data because of the low efficiency of reading and parsing XML data during the

* Corresponding author.
E-mail address: keresh@gatech.edu (K. Afsari).

http://dx.doi.org/10.1016/j.autcon.2017.01.011
0926-5805/

Table 1
Definition of JSON document and JSON schema in BIMserver.

BIMserver JSON document

```
"5374396": {
    "Owner History": {
        "Owning User": {
            "The Person": {},
            "The Organization": {}
        },
        "Owning Application": {},
        "State": "NULL",
        "Change Action": "NOCHANGE",
        "Creation Date": 1215091190
    },
    "Tag": "BF42B38D-335C-4620-80-93-9FFDAC5BDD4C"
},
```

Parts of BIMserver JSON Schema

```
"IfcOwnerHistory": {
  "domain": "ifcutilityresource",
  "superclasses": [],
  "fields": {
    "OwningUser": {
      "type": "IfcPersonAndOrganization",
      "reference": true,
      "many": false
    },
    "OwningApplication": {
      "type": "IfcApplication",
      "reference": true,
      "many": false
    },
    "State": {
      "type": "enum",
      "reference": false,
      "many": false
    },
```

execution of services [10]. Based on the measurement of metrics such as the number of objects sent, total time to send the number of objects, average time per object transmission, user CPU utilization, system CPU utilization, and memory utilization, it has been proved that JSON is significantly faster and has higher parsing efficiency than XML [4,7]. Besides, AJAX has become one of the popular technologies for developing web services [11]. AJAX is a web technique to transfer data between a browser and a server asynchronously and has several advantages over the classic web applications since it reduces response time, server load, and bandwidth of web applications [12]. Initially, XML was used in a wide range of AJAX developments, but gradually XML showed inadequacies because its performance reduces significantly when it is applied to interactive pages [7]. In order to address the issues of XML-based services, AJAX decreases the server workload by applying JavaScript at the client side. JSON is a native data for JavaScript and this feature makes JSON a proper format to be used for data exchanging in AJAX applications [10]. JSON is a key-value style lightweight data exchange format which is independent of any programming language and unlike XML it is easy for machines to parse and generate while it is also human readable [7].

While JSON has become an obvious choice over XML for web services, so far there is a lack of studies on implementing IFC specification using JSON serialization. This paper highlights that there is a need to provide guidelines on how to translate IFC data to JSON to indicate how IFC schema can be represented as JSON specification. Therefore, in this paper the main objective is to outline that IFC specification can be represented in JSON format. The ifcJSON Schema introduced in this paper, is an alternative to IFC EXPRESS specification and the ifcJSON document is an alternative to the Step Physical File (SPF) representation. In fact, this study addresses how to implement IFC standard as a JSON schema to generate JSON documents for web-based data transfer. In our previous study [13], the opportunities for Cloud-BIM interoperability and the need for a Web compatible data format that follows common agreements and terminologies have been pointed out. The ifcJSON schema and document developed in this paper can facilitate standardization of JSON-based BIM data and can have a major impact on interoperability of Web-based BIM applications by unifying BIM data representation based on both industry-wide standard i.e. IFC and Web compatible data format i.e. JSON. This approach to interoperability of Web-based BIM applications can expedite the production and revision of construction documents through using Web compatible BIM documents.

In this paper, first a review of the IFC schema in EXPRESS and XML specification is presented in Section 2 and accordingly, the study will point out the significance of developing ifcJSON representation in Section 3 in more details. Then, the methodology for JSON implementation of IFC data schema is specified in Section 4. In Section 5, the data model mapping for ifcJSON schema is defined, and the implementation of the ifcJSON document as the exchange data for the ifcJSON schema is specified. Also, the data validation methods that are used in this paper to ensure the data is correct and useful are described. In addition, the ifcJSON is implemented in a use case explained in Section 6 for the precast concrete domain to specify the geometry representation, product placement, owner history and a building element data both in ifcJSON schema and ifcJSON document. Then, the results of the study are presented and discussed in Section 7 to outline how the objective of the paper has been addressed and the limitations of this approach are explained in Section 8. Recommendations and the direction for future studies are also described in Section 9.

2. IFC schema and data serialization

IFC data model provides the basis for a common understanding of the building processes and the required information results from their execution [14]. IFC represents an open specification by introducing object classes and provides a useful structure for data sharing among applications [15] in a building construction or facility management project.

IFC specification has a data schema that is represented as an EXPRESS schema specification. ISO 16739:2013 consists of the IFC data schema in an EXPRESS schema specification, and reference data for the description and definitions of property and quantity names [16]. EXPRESS itself is an information model specification language which is defined as ISO10303-11 by the ISO TC184/SC4 committee [5]. It is developed as part of the STEP standard for product model data exchange [17]. The IFC exchange file structure, with the extension ".ifc" or ".stp", is known as "STEP Physical File" (SPF) format. It is an ASCII file format using a clear text encoding of product data for exchanging IFC data between different applications [5,8,18]. Current IFC release is IFC4 Add1 [19].

Alternatively, there is an XML Schema specification for IFC data model (i.e. ifcXML) represented as XML definitions [3]. The ifcXML file structure, with ".ifcXML" or ".ifx" or ".xml" extension, is the XML document structure [5]. The XML schema is automatically created from the EXPRESS representation of IFC schema by a language binding described by "XML representation of EXPRESS schemas and data", defined as

```
{
    "type": "Feature",
    "id": "f1",
    "geometry": {...},
    "properties": {...},
    "title": "Example Feature"
}
```

Fig. 1. Example of a GeoJSON "Feature" [31].