A RESTFUL API FOR THE EXTENDED WHAT3WORDS ENCODING

W. Jiang, E. Stefanakis*

Department of Geodesy and Geomatics Engineering, University of New Brunswick, P.O.Box 4400, Fredericton, E3B 5A3 Canada
wjiang3@unb.ca, estef@unb.ca

Commission IV, WGIV/7

ABSTRACT:

Geocoding is a fundamental task of Geographic Information System (GIS) processing and analysis. It allows the transformation between a location reference (i.e., an address or an alphanumeric code) and coordinates, which is often an essential step when performing spatial analysis, mapping, and other geolocation related processes. Providing software functionality through RESTful APIs is a common practice in geospatial applications. Client programs are able to access or process geospatial data easily through a lightweight and scalable RESTful web service. Existing geocoding RESTful API providers include Google Maps Geocoding API, ArcGIS Geocoding REST API, MapQuest Open Geocoding API and what3words (w3w) Geocoding API. Extensions of what3words geocoding system have recently been proposed to overcome its limitation of fixed resolution and lack of consideration of the third dimension. This paper considers one of the extensions, the Quadtree Extension Model (QTEM) and introduces a RESTful API that provides operations for forward geocoding, reverse geocoding, single line and polygon encoding, and centre points encoding for a given area. The resources published by the web service could be implemented by software programs performing indoor and outdoor location referencing, location marking and path finding. This API could facilitate various geographic applications such as facility management, emergency evacuation, navigation and student survey data management in a university campus environment.

1. INTRODUCTION

Methods to determine a location on a digital map or find the optimal path to get there are becoming increasingly relevant to everyday life. Location encoding (also known as geocoding) is an approach that transforms an address, postal code, place name or another geographic reference to geographic coordinates (Goldberg, 2011; Lee, 2009; Karimi *et al.,* 2011). This allows spatial analysis, mapping, and other geolocation related processes to be performed in GIS software packages. Reverse geocoding functions perform the opposite task and convert coordinates into the address, name, or code of the location. Address-based geocoding methods have been applied for decades. Despite numerous efforts to improve address-based geocoding, not all limitations have been resolved (Jiang and Stefanakis, 2018). Alphanumeric code geocoding systems have emerged more recently and provide alternative methods to overcome some of the problems. Example geocoding systems include: MapCode, c-Squares, WMO squares, Open Post Code, Google's Open Location Code. what3words (w3w) is one of them and of particular interest to this study.

1.1 what3words Geocoding and Extensions

w3w divides the earth's surface into a grid of 3-meter by 3-meter squares; each assigned a unique code consisting of three dictionary words separated by periods, e.g., the entrance to the CN Tower in Toronto, Canada is located at "select.threaten.shelters" (Barr, 2015). This method allows for easier memorization of locations and is supported in multiple languages. The static and unchangeable code meets the fundamental need of addressing people and places no matter the existence of a reliable street address. However, w3w also presents limitations: it has a fixed resolution of 3 m x 3 m and provides no consideration of elevation.

To overcome these limitations, Jiang and Stefanakis (2018) proposed a series of extended models of w3w towards the two aspects mentioned above. With a variable resolution, w3w grid cells could have a smaller or larger size to satisfy the needs of a

finer or coarser resolution respectively. The w3w code was extended by a fourth word to present the new cell size in a finer or coarser resolution. The QTEM model recursively subdivides the original cell in quadrants to obtain resolutions of 1.5-meter, 0.75-meter, 0.375-meter, 0.1875-meter and 0.09-meter (Figure 1). At each resolution level, all the fourth words start with the same letter so that the resolution level can implicitly be determined from the code (Table 1). A fifth element can also be attached to the extended code to represent the variety of elevation types. The elevation can be described in three ways: absolute height, relative height (ground reference; above or below the ground), and floor (floor label and relative height within the floor) for indoor applications (Table 1). The space can be regarded as a 3D grid of cubic blocks, where each block has assigned a five-word code referencing its geographic location. Figure 2 shows an example of blocks representing the occupied space in dark colour and the free space in light colour, in the resolution of 1.5-meter and 0.75-meter.
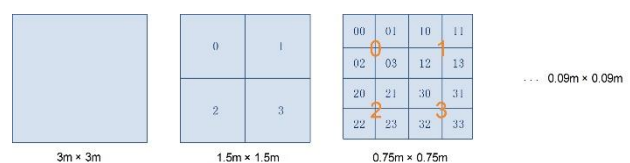

Figure 1. QTEM-a finer resolution extension model of w3w
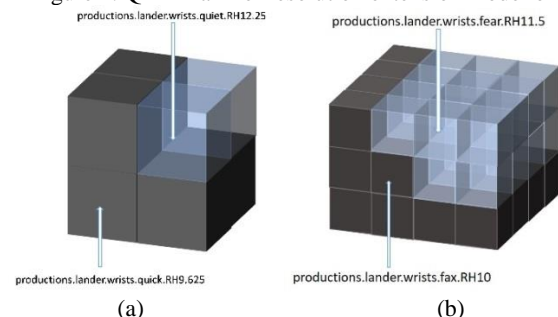

(a)   (b)

Figure 2. Blocks representing the 3D space. (a) blocks of 1.5m resolution; (b) blocks of 0.75m. (dark blocks represent occupied space, while light colour blocks represent free space).

---

* Corresponding author

Resolution (m)	Extended Code
3×3	word1.word2.word3..H(RH/F) 10
1.5×1.5	word1.word2.word3.**q**···H(RH/F)
0.75×0.75	word1.word2.word3.**f**··· H(RH/F)
0.375×0.375	word1.word2.word3.**c**··· H(RH/F)
0.1875×0.1875	word1.word2.word3.**p**··· H(RH/F)
0.09×0.09	word1.word2.word3.**s**··· H(RH/F)

Table 1. Extended five-word code of w3w

According to the resolution, the maximum residual offset of the geocoded location from the centroid of the cell is ranging from 1.06 meter to 0.06 meter. The w3w extensions provide higher precisions by simply adding one more word to the original three-word code. This coding schema retains the advantage of w3w as being easier to remember (Barr, 2015). While the MapCode provides precisions ranging from 5 meters to 0.16 meter by increasing the code from 5 digits to 8 digits with mixing numbers, letters, and symbols (Geelen, 2015). Similarly, Google's Open Location Code uses a 10-digit code to represent a square of 14-meter by 14-meter in size, which defines the precision to be roughly 10 meters. To increase the precision, the code needs to be lengthened (Open Locaiton Code, 2015).

1.2 The RESTful API

To implement the two-way transformation between the extended code and the geographic coordinates, an application programming interface (API) for the extended w3w geocoding system was designed.

"Representational State Transfer" (REST) is an architectural style of network that commonly applied to API designs for web services (Masse, 2011). A Web API conforming to the constrains of the architectural style is considered RESTful. A client program can use the API to communicate with the web services. It can acquire data from the web service by sending a request to a specific Uniform Resource Locator (URL) through HTTP protocol. The RESTful API developed for this study implements the QTEM extension and provides the functions of (i) forward and reverse geocoding; (ii) encoding single line and polygon feature; (iii) encoding a section of centre points for a given area.

1.3 Scope and Structure of the Paper

The scope of this study is the development and testing of a HTTP-based RESTful API implementing the QTEM geocoding extensions of w3w.

The paper content is organized into five Sections. Section 2 provides a summary of existing geocoding systems including a discussion on the advantages and disadvantages of these systems. It also reviews the definition and implementation of geographic RESTful APIs. Section 3 introduces the operations supported by w3w geocoding RESTful API. Section 4 demonstrates the development of the extended RESTful API and example results. Section 5 summarizes the geocoding operations provided by the extended RESTful API and lists some potential applications.

2. LITERATURE REVIEW

Geocoding is one of the basic geospatial operations. It converts addresses, postal codes, place names or other geographic references to geographic coordinates (Goldberg, 2011; Lee, 2009; Karimi *et al.,* 2011). It plays a vital role in spatial analysis, as geocoding technology has been utilized in many application areas such as epidemiology, environmental science, emergency management, marketing, planning and location-based services. These applications involve a broad range of disciplines including,

but not limited to: geography; geographic information science; computer science; digital libraries; history; and economics (Goldberg, 2011; Karimi *et al.,* 2011). Geocoded data provides a basis for subsequent spatial analysis and mapping. Errors associated with the geocoded data are likely to propagate through subsequent processing, analysis, modelling, and decision making (Goldberg, 2011; Karimi *et al.*, 2011). Therefore, it is important to obtain accurate locations from geocoding processes.

There are two categories of geocoding schemes adopted by geocoding systems: address-based and alphanumeric code based. Address-based geocoding systems have many constraints including coverage, standardization, maintenance, and precision. The coverage issues occur because these systems are unable to geocode locations with no official address (Barr, 2015). Standardization is an issue because address-based systems require properly formatted input, whereas address formats vary from location to location (Lee, 2009). Also, these systems require high maintenance as address databases must be updated regularly to reflect real-world changes for the entire area (Lee, 2009). Last but not least, precision can be a major concern. Geocoding in rural areas is often offset-prone (Kellison, 2012). Even in urban areas, where geocoders are typically more precise, they are referenced to the centroid of structures or parcels. Entire university buildings, business parks, and farms are abstracted to single points that do not carry sufficient precision for many applications (Chen *et al.*, 2016; Zandbergen, 2007; Karimi *et al.*, 2011).

The alphanumeric code geocoding scheme provides an alternative way for describing a geographic location. Multiple alphanumeric code based geocoding systems are already available to convert between alphanumeric codes and geographic coordinates: Geohash, MapCode, c-Squares, WMO squares, Open Post Code, Google's Open Location Code, and what3words (Barr, 2015; Stefanakis, 2016). These systems assign systematic alphanumeric labels to locations (polygons) over the earth, which are converted to geographic coordinates using mapping formulas, instead of graphs and an address database (Barr, 2015). Alphanumeric code geocoding systems share several advantages: (i) every cell is assigned a unique and static code, (ii) codes are simple to encode and decode; (iii) codes are efficient for communication (Chen *et al.*, 2016).

w3w, compared to other alphanumeric code geocoding systems, has three significant advantages. First, is the cleanness of the coding scheme, as the use of dictionary words is less error-prone than a code mixing letters and numbers. Secondly, the codes are easier to remember (Barr, 2015). Finally, it supports multiple languages. Other alphanumeric code geocoding systems provide a code combining random Latin characters with numbers, which can be as long as 10-characters for high resolution. The codes provided by these systems are hard to remember, while they are not language or culture-independent (Rhind, 2015). However, w3w still has limitations, such as a fixed resolution and no consideration of elevation, which may impede efficient modelling in applications. Jiang and Stefanakis (2018) introduced a series of extensions of the w3w geocoding system to overcome these limitations. The Ternary-Tree Extension Model (TTEM) and Quadtree Extension Model (QTEM) proposed would extend the w3w grid to provide variable resolutions. These models in conjunction with elevation extension models support indoor applications where a finer resolution and height value are desired. These extensions require attaching a fourth-word and a fifth-word to original w3w codes to represent the resolution and elevation respectively.

Geocoding as a basic geoprocessing tool is commonly implemented by geographic applications (Zhao et al, 2012). Providing its functionalities through RESTful web service is an increasing trend in industry. The REST architecture, as a new paradigm on Web service is now a favorite method of designing web services that process and deliver geographic data (Foerster et al, 2011). The lightweight and scalable properties of the RESTful APIs meet the needs of simplicity and flexibility for many applications. It is beneficial for non-computer science researchers as it can be easily implemented in existing applications (Mazzetti et al, 2009). Many RESTful APIs have been developed for geospatial applications to publish geospatial resources over a network for client programs' access (Sun et al, 2017). The standardized interfaces allow for the interoperability between client programs and web services. Examples of geocoding RESTful APIs are Google Maps Geocoding API, ArcGIS Geocoding REST API, MapQuest Open Geocoding API and what3words Geocoding API.

3. W3W RESTFUL API

W3w RESTful API, like many other geocoding APIs, provides programmatic access to transform a three-word address into coordinates (forward geocoding) in the WGS84 spatial reference system and to convert coordinates to a three-word address (reverse geocoding). It also provides access to a geographically bounded section of 3m x 3m what3words grid which is comprised of multiline features. The delivery of the resources is in XML, JSON and GeoJSON format. Other non-spatial operations such as obtaining suggestions based on a full or partial three-word address (AutoSuggest and StandardBlend) and retrieving a list of supported languages are provided as well (what3words API, 2016).

Each of the API resources can be accessed from a specific URI that consists of an endpoint https://api.what3words/com/v2, a path (i.e. /forward?, /reverse?, /grid?) and a set of query options. Examples in Table 2 (a) demonstrate the URIs of w3w RESTful API.

Operations	URI
forward geocoding	https://api.what3words.com/v2/forward?addr=index.home.raft&key=MY-API-KEY&lang=en&format=json&display=full
reverse geocoding	https://api.what3words.com/v2/reverse?coords=51.521251%2C-0.203586&key=MY-API-KEY&lang=en&format=json&display=full
grid	https://api.what3words.com/v2/grid?bbox=52.208867%2C0.117540%2C52.207988%2C0.116126&key=MY-API-KEY&format=json

Table 2 (a) Example URIs of w3w RESTful API

Each URI consists of both required and optional queries (Table 2 (b)). The parameters of the queries are the input that will be passed to the web server when a request has been made. The optional queries can be left blank and they will be assigned default values designed by w3w RESTful API.

Example representations of the resources of w3w forward and reverse geocoding were shown below (Figure 3). The requested format is JSON, and the "display" parameter in the URIs were determined as "minimal".

Operations	Queries	Parameters
forward geocoding	addr	three-word address
	key	API key of w3w
	lang (optional)	language it supports
	format (optional)	return data format type
	display (optional)	return data display type
reverse geocoding	coord	coordinates
	key	API key of w3w
	lang (optional)	language it supports
	format (optional)	return data format type
	display (optional)	return data display type
grid	bbox	the northeast and southwest corner coordinates of the specified grid
	key	API key of w3w
	format (optional)	return data format type

Table 2 (b) Example URIs' parameters of w3w RESTful API

URI
https://api.what3words.com/v2/forward?addr=basks.lenses.rugged&key=myAPIkey&lang=en&format=json&display=minimal

Result

```
{
  "words":"basks.lenses.rugged",
  "geometry":{
    "lng":66.642151,
    "lat":45.949237
  },
  "language":"en",
  "status":{
    "reason":"OK",
    "status":200
  }
}
```

(a)

URI
https://api.what3words.com/v2/reverse?coords=45.92222%2C-66.67777&key=myAPIkey&lang=en&format=json&display=minimal

Result

```
{
  "words":"muddy.finishes.sued",
  "geometry":{
    "lng":-66.677771,
    "lat":45.922232
  },
  "language":"en",
  "status":{
    "reason":"OK",
    "status":200
  }
}
```

(b)

Figure 3. (a) Result of w3w forward geocoding API request in JSON format, (b) Result of w3w reverse geocoding API request in JSON format

The w3w API is utilized by the extended w3w API, to provide forward and reverse geocoding of the first three words of each extended code.

4. EXTENDED W3W RESTFUL API

The development of the extended RESTful API for the QTEM model is presented. Like the w3w RESTful API, the extended RESTful API also provides forward and reverse geocoding operations. Operations of single line and polygon feature encoding and multi-points encoding are implemented as well.

4.1 Development

4.1.1 URI Structure and HTTP Method: A RESTful API exposes a set of resources over the network that are identified by URIs (Balani et al, 2009). Each unique URI directs to a specific resource. The accessibility of URI is essential for web applications. Therefore, the architecture structure of URI needs to be clearly formed. The generic syntax of a URI is defined as URI= scheme "://" authority "/" path ["?" query] ["#" fragment] (RFC 3986). The scheme "https" is used for the interaction with the resource through Hypertext Transfer Protocol (HTTP) over Secure Sockets Layer (SSL). HTTP provides a uniform interface and a set of methods such as GET, POST, PUT, DELETE to manipulate the resource. The extended w3w RESTful API was developed exclusively using the GET method, which retrieves the representation of the resource. "gaia.gge.unb.ca" host was used for the authority portion to identify the owner of the RESTful API. It can only be accessed within UNB network at this time. Based on the requested resource the path and query parameters vary. Table: 3-7 demonstrate URI templates of the extended w3w RESTful API.

The operations can be performed by four-word and five-word extensions, denoted as "w4w" and "w5w" respectively in the URI path. The following word (i.e., forward, reverse, line, polygon, area) indicates the operation of the API. The format parameter can be assigned "xml", "json" or "geojson" to obtain the specific format of the resource.

Operation	Forward Geocoding
URI Root	https://gaia.gge.unb.ca
URI Path	/WebService1/geocoding/w4w/forward
	(/WebService1/geocoding/w5w/forward)
Parameters	addr — extended four-word (five-word) code
	lang — supported language
	format — supported format
HTTP method	GET
Format	XML, JSON, GeoJSON

Table 3. Extended w3w forward geocoding API reference

Operation	Reverse Geocoding
URI Root	https://gaia.gge.unb.ca
URI Path	/WebService1/geocoding/w4w/reverse
	(/WebService1/geocoding/w5w/reverse)
Parameters	lat — latitude of the coordinates
	lng — longitude of the coordinates
	(h — elevation of the coordinates)
	resolution — resolution of the extended square
	lang — supported language
	format — supported format
HTTP method	GET
Format	XML, JSON, GeoJSON

Table 4. Extended w3w reverse geocoding API reference

Operation	Line Encoding
URI Root	https://gaia.gge.unb.ca
URI Path	/WebService1/geocoding/w4w/line
	(/WebService1/geocoding/w5w/line)
Parameters	start — four-word (five-word) code of endpoint
	end — four-word (five-word) code of the other endpoint
	lang — supported language
	format — supported format
HTTP method	GET
Format	XML, JSON, GeoJSON

Table 5. Extended w3w line encoding API reference

Operation	Polygon Encoding
URI Root	https://gaia.gge.unb.ca
URI Path	/WebService1/geocoding/w4w/polygon
	(/WebService1/geocoding/w5w/polygon)
Parameters	vertices — an array of four-word (five-word) codes separated by "%"
	lang — supported language
	format — supported format
HTTP method	GET
Format	XML, JSON, GeoJSON

Table 6. Extended w3w polygon encoding API reference

Operation	Centre Points Encoding
URI Root	https://gaia.gge.unb.ca
URI Path	/WebService1/geocoding/w4w/area
	(/WebService1/geocoding/w5w/area)
Parameters	bbox — the coordinates of southwest and northeast corner for a given area
	resolution — resolution of the extended square
	format — supported format
HTTP method	GET
Format	XML, JSON, GeoJSON

Table 7. Extended w3w centre points encoding API reference

4.1.2 Resources Design: The resource is the data returned to the client. The information the resource conveys is determined by the geocoding operations. For the QTEM model, the operation algorithms were implemented on the JAVA programming platform. The extended w3w forward and reverse geocoding algorithms are the basic procedures, utilized by the algorithms of line, polygon and centre points encoding.

Algorithm 1 demonstrates the forward geocoding operation that transforms a four-word code to a pair of coordinates (φ, λ) (Table 8 (b)). An array of the fourth words, representing the positions, in the second level QTEM model is displayed in Table 8 (a).

The algorithm can be mapped to a work flow (Figure 4) in which the extended four-word code as input were separated into two parts—the original w3w address code and the fourth word. The first three words are transformed to the corresponding coordinates of the w3w square by using w3w RESTful API. By finding the fourth word with the 2D array, the position of the sub-square is located. The latitude and longitude offsets between these two points are calculated and applied to obtain the coordinates of the sub-square.

Similarly, for the five-word implementation, the first four words were processed using Algorithm 1 to obtain latitude and longitude. The fifth word was processed separately to retrieve the

height value. Figure 5 shows the work flow of the transformation from five-words code to coordinates with elevation.

Algorithm 2 demonstrates the reverse geocoding operation that converts a pair of coordinates (φ, λ) to a four-word code given a specific resolution (r) (Table 9). The work flow in Figure 6 illustrates the process of Algorithm 2, in which the coordinates (φ, λ) and resolution (r) in darker blocks were passed as input parameters. The output four-word code was obtained by combining the original three words retrieved from w3w API and the fourth word retrieved from the "find Sub-square" method.

To convert coordinates with elevation value to a five-word code, the above work flow was utilized to obtain the first four words. The elevation was processed separately and converted to the corresponding symbol as the fifth word (Figure 7).

fan	far	fee	few
fat	fax	fear	feat
file	film	flag	flam
fire	five	flap	flat

(a)

Algorithm 1: Forward geocoding, from four-words code to position

Input:
    An array of string data type of four words
Output:
    Corresponding coordinates of latitude φ' and longitude λ'
Steps:
  1. Create a new array to store the first three words of the four-word code;
  2. Retrieve coordinates φ, λ of the center point of the original what3words square through what3words API by providing the new array;
  3. Determine resolution based on the fourth word:
    i. If it starts with letter 'q', resolution r = 1.5 m
    ii. If it starts with letter 'f', resolution r = 0.75 m
    iii. If it starts with letter 'c', resolution r = 0.375 m
    iv. If it starts with letter 'p', resolution r = 0.1875 m
    v. If it starts with letter 's', resolution r = 0.09375 m
  4. find its position in the two-dimensional array [row, column]
  5. find the offset distance value
    i. calculate the x, y distance between the center point of the sub-square and the center point of the original square
      x = f (row, column, r)
      y = f (row, column, r)
    ii. convert the x, y value into Δλ, Δφ respectively
      $\Delta\varphi = y \div M$[1]
      $\Delta\lambda = x \div (N$[2]$\cdot\cos\varphi)$
  6. Add offset Δφ, Δλ to φ and λ respectively
      φ' = φ + Δφ
      λ' = λ + Δλ
  End

(b)

Table 8. (a) An example 2D array of the fourth words of QTEM model in resolution of 0.375m; (b) Algorithm of transforming four-word code to coordinates

4.2 RESTful API Request Results

Table 10 lists the operations the extended w3w RESTful API provides, associated with example URIs.
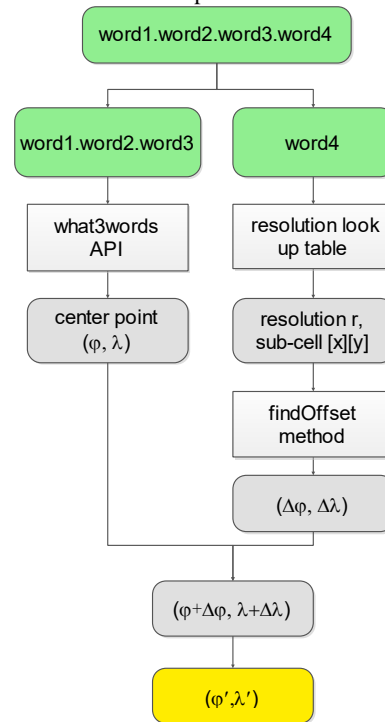

Figure 4. Work flow of extended forwarding geocoding with four-word code

The following examples display some of the results of the extended RESTful API requests. The four-word code "handlebars.overlooks.pitch.fat" representing a sub-square in resolution 0.75m (Table 8 (a)) was passed on to the web server. The coordinates of the sub-square associated with the bounding coordinates and the latitude and longitude offsets from the centre point of the 3m square (in meters) were returned in XML format as determined in the URI (Figure 8). Similarly, the five-word address containing elevation information can be transformed to the corresponding coordinates (Figure 9).

The coordinates retrieved from the forwarding geocoding API in GeoJSON format was uploaded to ESRI ArcGIS Online platform and presented in Figure 14 (a).

The reverse geocoding that converts coordinates into extended codes were also tested (Figure 10). The resolution parameter from the client input may differ. However, it will be classified to fit the specific resolution supported by the QTEM extensions.

The single line encoding operation using four-word code was tested by sending the following request (Figure 11). The GeoJSON object was mapped as a line feature following along the hallway of Head Hall Building in UNB (Figure 14 (b)). The polygon encoding API provides the function of retrieving a simple polygon by inputting the codes of its vertex (Figure 12). The centre points encoding for a defined area return a collection of points that represent the sub-squares (Figure 13).

The features of the polygon and the multi-points obtained from the above API requests were visualized in Figure 14 (c). To

---

[1] $M = \dfrac{a(1-e^2)}{(1-e^2\sin^2\varphi)^{\frac{3}{2}}}$,

[2] $N = \dfrac{a}{(1-e^2\sin^2\varphi)^{\frac{1}{2}}}$, a is the semi-major axis, e is the numerical eccentricity of the ellipsoid (Torge et al, 2012).

demonstrate the corresponding four-word code for sub-squares, four example points were selected from each original square. The fourth word in each code indicates the position of the sub-square inside the original cell (Table 8 (a)).
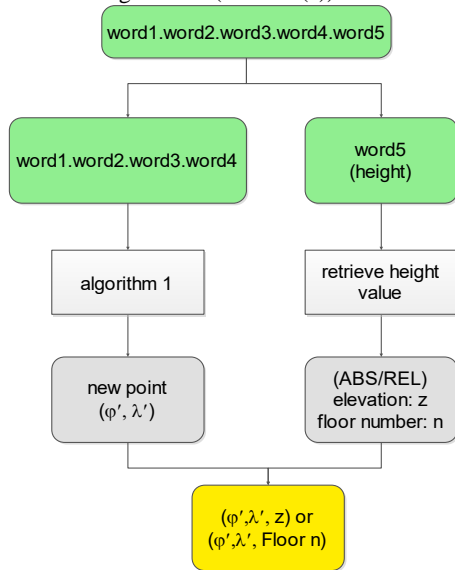

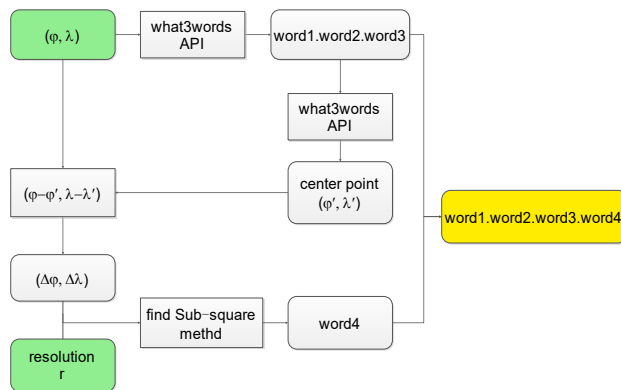Figure 5. Work flow of extended forward geocoding with five-word code

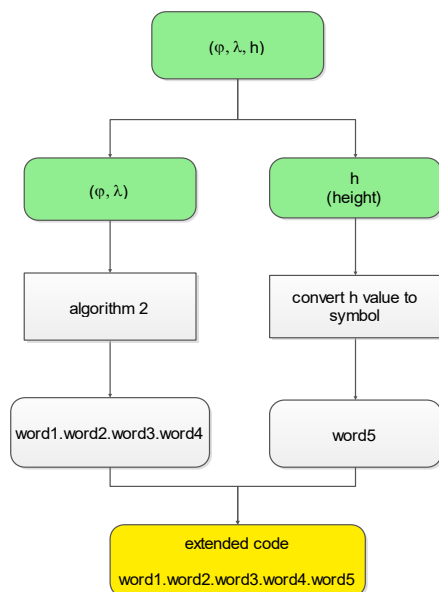Figure 6. Work flow of extended reverse geocoding with four-word code

Figure 7. Work flow of extended reverse geocoding with five-word code

5. CONCLUSION

The extended w3w RESTful API implementing QTEM extensions has been designed and tested in order to validate the extensions and provide a web service for client geographic applications. The extended geocoding API provides operations for (i) forward geocoding, that transforms the extended four-word or five-word code to corresponding coordinates; (ii) reverse geocoding, that converts a set of coordinates into the extended code; (iii) line and polygon encoding, that retrieve single line or polygon feature; and (iv) centre points encoding, that retrieve a section.

The URIs of the developed API were designed in the pattern with a clear architectural structure that is human-readable and easy to understand. The requested resources can be obtained by identifying the URI path and the corresponding parameters. XML, JSON and GeoJSON format of the resources are supported by the API. The algorithms that perform forward and reverse geocoding were tested by implementing the API requests of centre points for a given area.

The extended w3w RESTful API is a lightweight and scalable application. The resources published by the web service could be implemented by software programs performing indoor and outdoor location referencing, location marking and path finding. Such applications include facility management, emergency evacuation, route navigation planning and other location-based services inside university campus.

URI
https://gaia.gge.unb.ca/WebService1/geocoding/w4w/forward?
addr=handlebars.overlooks.pitch.fat&lang=en&format=xml

Result

```
<response>
   <bound>
      <northeast>
         <latitude>45.95010674780873</latitude>
         <longitude>-66.64177367334382</longitude>
      </northeast>
      <southwest>
         <latitude>45.95010000019127</latitude>
         <longitude>-66.64178334665617</longitude>
      </southwest>
   </bound>
   <geometry>
      <latitude>45.950103374</latitude>
      <longitude>-66.64177851</longitude>
   </geometry>
   <offset>
      <delta_x>-1.12500244212281</delta_x>
      <delta_y>0.3750212596528374</delta_y>
   </offset>
</response>
```

Figure 8. Forward geocoding a four-word code

URI
https://gaia.gge.unb.ca/WebService1/geocoding/w5w/forward?
addr=handlebars.overlooks.pitch.fat.**RH13**&lang=en&format=xml

Result

```
</response>
      ...
   <geometry>
      <latitude>45.950103374</latitude>
      <longitude>-66.64177851</longitude>
      <elevation>relative 13</elevation>
   </geometry>
      ...
</response>
```

Figure 9. Forward geocoding a five-word code

```
URI
https://gaia.gge.unb.ca/WebService1/geocoding/w4w/reverse?
lat=45.949803&lng=-66.641686&resolution = 0.75&
lang=en&format=json

Result

{
        "first":"tram",
        "second":"sullen",
        "third":"registration",
        "fourth":"flag"
}
```

Figure 10. Reverse geocoding coordinates to four-word code

```
URI
https://gaia.gge.unb.ca/WebService1/geocoding/w4w/line?
start=breathless.trains.machine.far&end=rituals.munched.cheater
.flap&lang=en&format=geojson

Result

{
  "type": "LineString",
  "coordinates": [
    [
      -66.641651837,
      45.950083121
    ],
    [
      -66.641565163,
      45.950197879
    ]
  ]
}
```

Figure 11. Encoding with four-word code

```
URI
https://gaia.gge.unb.ca/Webservices1/geocoding//w4w/po
lygon?vertices= vendetta.invocation.dairy.few%
clay.risk.lockup.flag%ungreased.thumbnail.glitters.flat%
faithless.term.breakdown.fat&lang=en&format=geojson

Result
{
  "type": "Polygon",
  "coordinates": [
   [
    [-66.64174949, 45.950137121],
    [ -66.641720163, 45.950177626],
    [-66.64163249, 45.950143879],
    [-66.64166151, 45.950103374],
    [-66.64174949, 45.950137121]
   ]
  ]
}
```

Figure 12. Polygon encoding by providing the codes of vertex

ACKNOWLEDGEMENTS

REFERENCES

Balani, N., Hathi, R., 2009. *Apache CXF Web Service Development: Develop and Deploy SOAP and RESTful Web Services*. Birmingham: Packt Publishing.

Barr, R., 2015. *About what3words*. Retrieved from what3words.com: https://what3words.com/about/ (26/3/ 2018)

Berners-Lee, T., Fielding, R., Masinter, L., 2005. *RFC 3986 Internet Standard*. Retrieved March 26, 2018, from Uniform Resource Identifier (URI): Generic Syntax: https://tools.ietf.org/html/rfc3986 (26/3/2018)

Chen, D., Cheng, C., Tong, X., Yuan, J., (2016. Research on the Multi-scale Spatial Location Coding Model for Address. *Geo-information Science*, 18(6), 727-733.

Foerster, T., Bruhl, A., Schaffer, B., 2011. RESTful Web Processing Service. *AGILE 2011*, 18-22.

Goldberg, D. W., 2011. Advances in Geocoding Research and Practice. *Transactions in GIS*, 15(6), 727-733.

Algorithm 2: Reverse geocoding, from position to four-words code

Input:
  1. Coordinates of latitude φ and longitude λ
  2. Resolution r
Output:
  Corresponding four-words code
Steps:
  1. Retrieve original what3words code [word1.word2.word3] through what3words API by providing coordinates (φ, λ);
  2. Retrieve the coordinates of the center point of the what3words square φ_c, λ_c through what3words API by providing the code [word1.word2.word3] from step 1;
  3. Find the Sub-square method
    i. Calculate the latitude and longitude offset between input point and the center point of what3words
    $$\Delta\varphi = \varphi - \varphi_c$$
    $$\Delta\lambda = \lambda - \lambda_c$$
    ii. Convert Δφ, Δλ into horizontal and vertical distance
    $$y = M \times \Delta\varphi$$
    $$x = (N \cdot \cos\varphi) \times \Delta\lambda$$
  4. Resolution categorized:
    If r <=2m and r >=1m, resolution r = 1.5 m search the words start with letter "q"
    If r <1m and r >=0.5m, resolution r = 0.75 m search the words start with letter "f"
    If r <0.5m and r >=0.25m, resolution r = 0.375 m search the words starts with letter "c"
    If r <0.25m and r >=0.125m, resolution r = 0.1875 m, search the words starts with letter "p"
    If r <0.125m and r >0m, resolution r = 0.09375 m search the words starts with letter "s"
  5. Find the fourth word in the array
    row = f (x, y, r)
    column = f (x, y, r)
  6. Add the fourth word to the words tuple [word1.word2.word3] retrieved from step 1.
    End

Table 9. Algorithm of converting coordinates to four-word code

```
URI
https://gaia.gge.unb.ca/WebService1/geocoding/w4w/area?
bbox=45.94925049480924,-
66.64216067341889,45.94928,-66.64219&resolution=
0.75&format=xml

Result
{
  "type": "FeatureCollection",
  "features":[
    { "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates":
          [-66.64216551,45.949280879]
        },
      "properties": {
        "words": "newsstands.ulcer.scored.fire"
        }
    },
    { "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates":
          [-66.6421558368284,45.949280879]
        },
      "properties": {
        "words": "newsstands.ulcer.scored.five"
        }
    },
    ......
  ]
}
```

Figure 13. Centre points encoding for a given area

Geelen, P., 2015. *Mapcode Downloads and Developer Resources.* http://www.mapcode.com/downloads.html?iso3= 112&mapcode=49.4V (26/3/2018)

Jiang, W., Stefanakis, E., 2018. What3words Geocoding Extensions. *Journal of Geovisualization and Spatial Analysis*, 2:7, 1-17. Available online: http://rdcu.be/G7fB

Karimi, H. A., Sharker, M. H., Roongpiboonsopit, D., 2011. Geocoding Recommender: An Algorithm to Recommend Optimal Online Geocoding Services for Applications. *Transactions in GIS*, 15(6), 869-886.

Kellison, M. T., 2012. *Address Points and a Master Address File: Improving Efficiency in the City of Chino*. (Master's thesis). University of Southern California.

Lee, J., 2009. GIS-based Geocoding Methods for Area-based Addresses and 3D Addresses in Urban Areas. *Environment and Planning B: Planning and Design*, 36, 86-106.

Mazzetti, P., Nativi , S., Caron, J., 2009. RESTful Implementation of Geospatial Services for Earth and Space Science Applications. *International Journal of Digital Earth*, 2, 40-61. doi:10.1080/17538940902866153

Masse, M., 2011. *REST API Design Rulebook*. Sebastopol, CA, US: O'Reilly.

Open Location Code. 2018. *Google/open locaiton code*: https://github.com/google/open-location-code (30/6/2018).

Rhind, G., 2015. *How Google Could Improve Open Locatoin Code*. http://resources.pcapredict.com/index.php/google-open-location-codes/#comment-3294363901 (26/3/2018)

Stefanakis, E., 2016. Location Encoding Systems – Could geographic coordinates be replaced and at what cost? *GoGeomatics Magazine*, 1-4.

Sun, Z., Di, L., Heo, G., Zhang, C., Fang, H., Yue, P., Lin, L., 2017. GeoFairy: Towards a one-stop and location based Service for Geospatial Information Retrieval. *Computers, Environment and Urban Systems*, 62, 156-167.

Torge, W., Muller, J., 2012. *Geodesy* (4th ed.). Berlin: Deutsche Nationalbibliothek. pp. 95

what3words. 2016. *what3words API V2. RESTful API*: https://docs.what3words.com/api/v2/#description (26/3/ 2018)

Zhao, P., Foerster , T., Yue, P., 2012. The Geoprocessing Web. *Computers & Geosciences,* 47, 3-12.

Zandbergen, P. A., 2007. A Comparison of Address Point, Parcel and Street Geocoding Techniques. *Computers, Environment and Urban Systems,* 32(2008), 214-232. doi:10.1016/j.compenvurbsys.2007.11.006

Operation	URI
forward geocoding	https://gaia.gge.unb.ca/Webservices1/geocoding/w4w/forward?addr=handlebars.overlooks.pitch.fat&lang=en&format=xml
	https://gaia.gge.unb.ca/Webservices1/geocoding/w5w/forward? addr= handlebars.overlooks.pitch.fat.RH13&lang=en&format=xml
reverse geocoding	https://gaia.gge.unb.ca/Webservices1/geocoding/w4w/reverse?lat=45.949803& lng=-66.641686&resolution=0.75&lang=en&format=json
	https://gaia.gge.unb.ca/Webservices1/geocoding/w5w/reverse?lat=45.949803& lng=-66.641686&h=RH13&resolution=0.75&lang=en&format=json
line encoding	https://gaia.gge.unb.ca/Webservices1/geocoding/w4w/line?start=breathless.trains.machine.far& end=rituals.munched.cheater.flap&lang=en&format=geojson
	https://gaia.gge.unb.ca/Webservices1/geocoding/w5w/line?start= breathless.trains.machine.far.RH13& end= rituals.munched.cheater.flap.RH13&lang=en&format=geojson
polygon encoding	https://gaia.gge.unb.ca/Webservices1/geocoding//w4w/polygon?vertices= vendetta.invocation.dairy.few%clay.risk.lockup.flag%ungreased.thumbnail.glitters.flat%faithless.term.breakdown.fat&lang=en&format=geojson
	https://gaia.gge.unb.ca/Webservices1/geocoding//w5w/polygon? vertices=vendetta.invocation.dairy.few.F5%clay.risk.lockup.flag.F5%ungreased.thumbnail.glitters.flat.F5%faithless.term.breakdown.fat.F5&lang=en&format=geojson
area encoding	https://gaia.gge.unb.ca/WebService1/geocoding/w4w/area?bbox=45.94925049480924, -66.64216067341889,45.94928,-66.64219&resolution=0.75&format=xml
	https://gaia.gge.unb.ca/Webservices1/geocoding//w5w/area?bbox=45.948479,-66.641608, 45.948430,-66.641590&h=35.5&resolution=0.75&format=xml

Table 10 Extended w3w RESTful API operations and example URIs
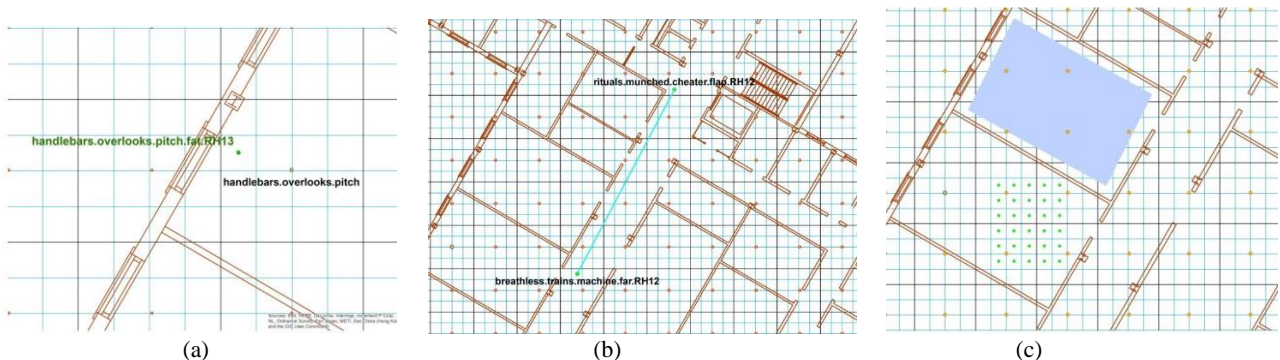

(a) (b) (c)

Figure 14. Visualization of (a) forward geocoding of a five-word code, (b) line encoding of Head Hall hallway, (c) polygon encoding of room E4 and centre points encoding of defined area in room E5