UDC 528.9:004 DOI: 10.30929/1995-0519.2018.6.21-26

METHOD OF MAKING A SYMBOL LIBRARY FOR THE WE-MAPS

D. Brazhnyk, Yan Haowen
Lanzhou Jiaotong University
vul. Anning Xi, 88, Gansu province, People republic of China. E-mail: dariabrazhnyk@gmail.com

Purpose. This article is devoted to the review of the modern methods of the designations creation for the We-maps type geographic information systems (GIS). Those kinds of designations are wide-ranged and are gathered in the whole libraries of symbols. **Methodology.** The methodologies involve determining the sequential steps that must be followed when creating information designations that are appropriate for the We-maps use. The process of creating any map includes a generalization of reality. Ordinary users and commercial companies both are interested in geographic information data, which accelerate this area developing speed today. **Originality.** The authors introduces their own authentic creation method for the We-maps type GIS designations. The main key of this idea is the usage of QR-codes; the combination of the symbol with QR-code which will minimalize the losses during the recognition. Basic information about the object is stored in the QR-code. The implementation of the idea was done in JavaScript with the use of the API service Amap.com which is the leader of geographic information systems in China. The API service was selected, because the other GIS systems are either banned (Google maps) or uninformative (Yandex maps). This implementation is represented by a marked map with authentic symbols. The geocenter of map is the Lanzhou Jiaotong University. **Results.** This map itself is a web page that is materialized by using hyper markup language (Html). When loading the web page sends a request using JavaScript language to the web service to build the map. Several tools are situated on the map: the zoom bar and the user's geolocation. Information about the user's location is received from the GPS coordinates of the user or by the IP address of the nearest point on the network. The map has a 3D layer of the objects representation for the ground orientation convenience. This information is located on the free git service GitHub in the public domain. A code was generated for users to open this map. The code can be scanned by any device that has a camera and a scanning program.

Key words: Symbol Library; We-maps; GIS; JavaScript; QR-code; Amap; API; Git.

СПОСІБ СТВОРЕННЯ БІБЛІОТЕКИ СИМВОЛІВ ДЛЯ WE-КАРТ

Д. В. Бражник, Ян Хаовен
Ланджоуський транспортній університет
Аннин Си Лу 88, Провінція Ганьсу, Китайська народна республіка. E-mail: dariabrazhnyk@gmail.com

Надано огляд сучасних методів створення позначень для геоінформаційних систем (ГІС) по типу "We-карт". Такі позначення складають цілі бібліотеки умовних позначень. Методології включають визначення послідовних кроків, яких необхідно дотримуватися при створенні інформаційних позначень, придатних для використання в We-картах. Процес створення будь-якої карти, включає в себе узагальнення дійсності. Як звичайні користувачі, так і комерційні компанії зацікавлені в геоінформаційних даних, тому цей напрям активно розвивається. Розроблено власну методологію створення інформаційних позначень для геоінформаційних систем з використанням QR-коду. Головна ідея полягає в об'єднанні умовного позначення і QR-коду з мінімальними втратами для розпізнання. Основна інформація про об'єкт зберігається в QR-коді. Реалізація ідеї виконана на JavaScript з використанням API сервісу Amap.com який є лідером географічних інформаційних систем в Китаї. Сервіс API був обраний тому, що інші ГІС-системи або заборонені (Google maps), або неінформативні (Yandex maps). Дана реалізація представлена розміченою картою з власними умовними позначеннями. Геоцентром карти є Ланджоуський транспортний університет. Сама карта являє собою веб-сторінку, яка реалізована за допомогою мови гіпер-розмітки (Html). Під час завантаження веб-сторінки, надсилається запит за допомогою мови JavaScript у веб-сервіс для побудови карти. На карті реалізовані кілька інструментів: панель масштабування і геолокація користувача. Інформація про місцезнаходження користувача визначається за допомогою GPS-координат користувача або за IP-адресою найближчої точки мережі. Карта має 3D шар представлення об'єктів на карті, для зручності орієнтації на місцевості. Дана реалізація знаходиться на безкоштовному Git сервісі GitHub у відкритому доступі. Для користувачів був згенерований код для переходу до даної карти. Код може бути відсканований будь-яким пристроєм, що має камеру і програму для сканування QR-кодів.

Ключові слова: Бібліотека символів; We-карти; ГІС; JavaScript, QR-код, Amap, API, Git.

PROBLEM STATEMENT. We- mapping is a technique of utilizing maps that are obtained from an information system for spatial and geographical data. We-maps are defined by the point of view of the consumer. They usually involve a web browser on the clients end or any other program that makes the interaction possible. The geographic information system plays a vital role in the service of developing we-maps. Experts often use GIS and we-maps interchangeably to ensure the investigative competences are achieved. GIS is essential for we-maps since it makes it more interactive and helps in data acquisition, storage and software architecture. Geographic information consumers develop an understanding of geographic phenomenon through the use of visual representations of the immediate environment on the maps. Cartographic designers use a wide range of graphical and non-graphical generalization operators to simplify reality and communication [1]. For cartographic designers it is presumed that there is never a single perfect design. Decisions made usually using GIS depend on: desired output format, the audience, the message and the function the map should support [2].

Communication of geographic knowledge is made by cartographic designers by using graphic symbols to present features on the map. Users of We-maps have certain expectations since most we-maps use current information hence while using this maps data has to be updated from time to time.

MATERIAL AND RESULTS. *Literature view*. For decades, most digital geographic information was confined for use on desktop-based personal computers (PCs) or in-house mainframes and could not be easily shared with other organizations. GIS analysts would access data from their own workplace computers that were often connected to a central file server somewhere in the office. Specialized software was required to view or manipulate the data, effectively narrowing the audience that could benefit from the data [3, 1]. With mass uptake of the Internet in the mid-1990s, people began thinking about how maps and other geographic information could be shared across computers, both within the organization and with the general public. The first step was to post static images of maps on HTML pages; however, people soon realized the potential for interactive maps. The first of these, served out by newborn versions of software such as Map Server and Esri ArcIMS, were horrendously pixilated, slow, and clunky by today's standards [4, 5]. Limited by these tools, cartographers had not yet arrived in large amounts on the web mapping scene, and most of the maps looked hideous [6]. However, these early interactive web maps were revolutionary at the time. The idea that you could use your humble web browser to request a map anywhere you wanted and see the resulting image was liberating and exciting. To get a feel for the web mapping landscape at this time period.

These early, dynamically drawn web maps ran into challenges with speed and scalability (the ability to handle many simultaneous users). The server could only accommodate a limited number of map requests at a time before slowing down (at best) and crashing (at worst). Web maps matured significantly in these two metrics when websites began to serve out tiled map images from regenerated caches [7, 8]. Why ask the server to draw every single map dynamically when you could just put forward an initial investment to pre-draw all possible map extents at a reasonable set of scales? Once you had the map images drawn and cached, you could serve out the images as a tiled mosaic. Each tiled map request was satisfied exponentially faster than it would take to serve the map dynamically, allowing for a server to accommodate hundreds of simultaneous users. Following the lead of Google Maps, many sites began serving out "pre-cooked" tiled map images using a creative technique known as Asynchronous JavaScript and XML (AJAX) that eliminated the ubiquitous and annoying blink that occurred after any navigation action in earlier web maps. Now you could pan the map forever without your server gasping for breath as it tried to catch up [4, 1].

Cartographers, who had largely been resigned to trading aesthetics for speed in web maps, also realized the potential of the tiling techniques. No longer would the number of layers in a map slow down the server: once you had regenerated the tiles, you could serve a beautiful map just as fast as an ugly one. Web maps became an opportunity to exercise cartographic techniques and make the most attractive map possible. Thus were born the beautiful, fast, and detailed "web 2.0" base maps that are common today on Google, Microsoft Bing, Open Street Map, and other popular websites. As web browsers increased in their ability to draw graphics using technologies such as SVG and later WebGL, the possibilities for interactivity arose [8]. On-the-fly feature highlighting and HTML-enriched popup windows became common elements. For several years, developers experimented with plug-ins such as Adobe Flash and Microsoft Silverlight for smooth animation of map navigation and associated widgets. More recently, developers are abandoning these platforms in favor of new HTML5 standards recognized by the latest web browsers without the need for plug-ins [9]. Although maps had arrived on the browser by the mid-2000s, they were still largely accessed through desktop PCs. The widespread adoption of smart phones and tablets in subsequent years only increased the demand for web maps. Mobile devices could not natively hold large collections of GIS data, nor could they install advanced GIS software; they relied on web or cellular connections to get maps on demand. These connections were either initiated by browsers on the device, such as Safari, or native applications installed on the device and built for simple, focused purposes. In both cases, GIS data and maps needed to be pulled from the organization's traditional data silos and made available on the web [8].

Methodologies. Making We- maps comprises of different activities and these activities range from designing the information on the maps, map designing, user experience designing and promoting finished web maps. During the designing process consideration of completeness, timeliness and authority is vital not only the data modeling. Web interface is a necessity when designing maps since they aid in the communication of the map's message and makes it appealing to the audience. When compiling a web map some few things have to be determined. The size and geographical extent has to be shown on the map, resolution of this maps is determined by this parameters. For making we-maps the following parameters are considered

(a) Size

When making we-maps most designers usually estimate the 17 – or 19- inch monitors since most people have that on their desktops. But with increasing technology the We- maps can be used on other devices such as Tablets PCs, smart phones, or iPads. These maps may work well on devices other than the primary mode but at times they might not.

(b) Geographical extent

The geographical extent of We-maps can be pan or zoomed at times this can cause greater extent than what is shown on the screens. Restricting map extent is usually advisable in order to make sense in providing a global view depending on the maps purpose.

(c) Map scale

Different maps may be given variable scales for readers to zoom in and out. Separate maps are compiled for each map scale to ensure that the zooming experience appears seamless.

(d) Map projection

When considering projections of the We-maps overlapping with other web maps should be avoided. When considering using these maps on Arc GIS Online, Bing, or Google, Mercator projection is required. If different projections are used anyone can use the map in mash up using the same projection. Consideration of alternate projections such as modified Winkel Tripel Projection can be used when you don't want anyone to use these maps.

(e) Color

Technology used nowadays can display millions of colors, using web-safe colors is a moot issue when making we-maps. Most of the we-maps are in color but color on the web is usually different to color on print. In print colors are made of ink pigments, on computers colors are created by combining red green and blue light in different proportions and intensities. For instance, white is a product of red, green, and blue light at full intensity. The disadvantages of using an additive color system is that light colors viewed on a computer monitor are overly luminous and too harsh on the eye for extended viewing [2]. The intensity of light radiating from a screen displaying pure white can affect the clarity of fine detail in type, point symbols, and line symbols as well as intricate patterns, such as raster's used to show hill shades or elevation tints.

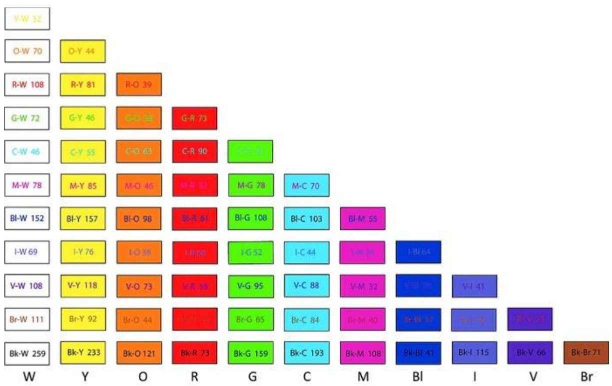


Figure 1 – Table showing variation of colors in a We- map

(f) Symbols

When making We-maps the symbols must be legible with large enough texts which can be distinguished from the background. A rule of thumb is that text and symbols should be at least 10pixels high. This means that depending on the PC one is using the points should be accommodative enough. Contrast is required to enable one to have the ability to distinguish a symbol from its background.

(g) Fonts

Fonts designed for the web are encouraged to use when making we-maps. Recommendation have shown that Arial (or Helvetica on Macintosh), Verdana, Georgia, Trebuchet, and Century Gothic (all installed on Windows systems), and Lucinda Grande and Palatino (installed on most systems) as the most popular fonts for web design. When considering good fonts then one has enough space between characters and within characters. A tall x-height also opens up the space within a character. These properties make fonts legible on screen. With the exception of Georgia and Palatino, these are sans-serif fonts. Serifs are the small lines or decorations added to the ends of the main strokes of the character that theoretically help the letters flow and lead the eye across text during reading. Serif fonts are very popular in print. However, many designers and cartographers believe that sans-serif fonts are more suitable for web map design because serifs compromise the space between characters [10]. This holds true for small blocks of text (e.g., labels on maps, titles, legend text), but for large blocks of text, serif fonts are still easier to read.

Figure 2 – web fonts as used in we-maps

(h) Resolution

Computer display resolution is low when compared to print maps. For desktop computers, it is common to design for a resolution of 96 dpi (dots per inch) because all LCD monitors support this resolution. Newer LCDs typically have a native pixel density of 120 dpi and 144 dpi. Choose resolution based on the type of computer your target audience will mostly likely use. This low resolution, coupled with the color projection issue, will impact the cartographic design of a web map. Because screen displays are pixels, no orthogonal lines and sharp edges appear jagged. These jagged edges can be softened by adding pixels of intermediate color between the object and the background (antialiasing), which fools the eye into seeing a jagged edge as a smooth one

(i) Map marginalia

Maps have two basic components: the map itself and information about the map, commonly called marginalia (additional information outside the edge of the map displayed in the margins). Map marginalia includes titles, legends, scale bars, scale text, and north arrows, as well as information about the data used, map projection, author, and publication date. With web maps, it makes sense to include some of these items, but not all. Characteristics of good web fonts (from sitepoint.com/anatomy-web-fonts) All maps should have a title. For symbology that may be unclear or confusing, include a legend, especially if the map is for an international audience. Cartographic conventions vary. Whether to include a map scale depends on how much area is shown on the map. If your map covers a large area or is 3D (i.e., is in a perspective rather than plan metric view), scale will vary across the map, and a scale bar or scale text would be inaccurate for all mapped locations. For 3D maps and maps that use a projection other than web Mercator, you may not want to include a north arrow because orientation may vary across the map. Instead, including a graticule (latitude and longitude lines) or other grid is a good alternative that helps address the scale issue as well.

For web maps, it is very useful to include the author and publication date and information on the data. Users of web maps expect data to be current and accurate and sometimes expect to be able to access the data. Knowing who made the map, when it was published, and what data was used to make it helps users assess the validity of the information on or linked to the map.

Findings & discussion. Users are likely to have expectations for We-maps content. Users of We-maps expect current data and sometimes continuously updated data this includes maps that show monitoring sites. We-maps users also expect interactive maps that support zooming at a minimum but also potentially support query, analysis and customization. For larger We-maps users expect detailed and realism. They may even also expect the data used to make the map to be downloadable and free. For these maps users expect data to be complete, consistent, and authoritative.

Originality. Subjects of research associated We-maps at this point is relevant, as these maps are used in daily life, in a variety of scientific research and commercial campaigns. The main criterion for the use of such maps is: the speed and correctness of the display of information on the device, as well as an intuitive understood interface. Therefore, We-maps are becoming an integral part of modern life. But they, also require improvement.

For this purpose, the task is to create a library of symbols with special symbols. The main idea consists in the combine the symbol and QR-code, which stores detailed information about the object or place.

In today's world, QR-codes are increasingly gaining popularity. They are introduced into use in all areas of activity. A QR code is a two-dimensional barcode that encodes a variety of information consisting of symbols (including Cyrillic, numbers, and special symbols). The principle of working with QR code is very simple. After scanning it from the device, the user will automatically pass to the embedded link in it. Except links, the code can contain a phone number, address, email, coordinates, names, and so on. You can add information about different objects to QR codes.

On one matrix, depending on the type of symbols, their can be placed a different quantity:

- 7089 digits;
- 4296 Latin letters and numbers;
- 2953 bytes of the binary code;
- 1817 asian characters.

The minimum size of the QR matrix is only 21x21 pixels (plus fields) and it is considered the first version. The maximum size is the 177x177 pixels – the 40th version.

The type of coding depends on the used symbols and has accordant name:

• digital - 10 bit/3 digits;

• alphanumeric - 10 digits, special symbols and all Latin letters (11 bit/2 symbols);

• byte;

• kanji characters – 13bit/1 character.

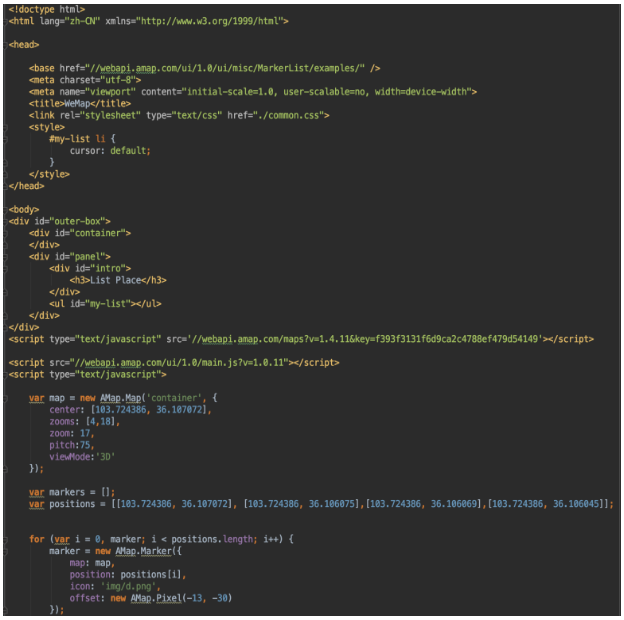
For correction of errors in QR coding, Reed-Solomon's codes are used. The code word in it consists of 8 bits. Exactly thanks to these codes there is a possibility of drawing a picture inside the QR matrix, without loss of readability by it [11, 12].

Thus, without overloading the map itself with unnecessary information, the user has the opportunity to obtain detailed information about a particular location or object of interest. The essence of this map is that it is accessible, mobile, easy to use, but at the same time contains all the necessary information.

Figure 3 – Example map with special symbols

The Google Maps, as it is known, are blocked in China and rather unreliable, as well as due to the fact, that Google maps and Yandex maps don't have a markup, was made the decision, for more convenience to use the Amaps maps, as in China, they are the leader.

The map was created using JavaScript language, based in API.



```
<!doctype html>
<html lang="zh-CN" xmlns="http://www.w3.org/1999/html">

<head>

    <base href="//webapi.amap.com/ui/1.0/ui/misc/MarkerList/examples/" />
    <meta charset="utf-8">
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no, width=device-width">
    <title>WeMap</title>
    <link rel="stylesheet" type="text/css" href="./common.css">
    <style>
        #my-list li {
            cursor: default;
        }
    </style>
</head>

<body>
<div id="outer-box">
    <div id="container">
    </div>
    <div id="panel">
        <div id="intro">
            <h3>List Place</h3>
        </div>
        <ul id="my-list"></ul>
    </div>
</div>
<script type="text/javascript" src='//webapi.amap.com/maps?v=1.4.11&key=f393f3131f6d9ca2c4788ef479d54149'></script>

<script src="//webapi.amap.com/ui/1.0/main.js?v=1.0.11"></script>
<script type="text/javascript">

    var map = new AMap.Map('container', {
        center: [103.724386, 36.107072],
        zooms: [4,18],
        zoom: 17,
        pitch:75,
        viewMode:'3D'
    });

    var markers = [];
    var positions = [[103.724386, 36.107072], [103.724386, 36.106075],[103.724386, 36.106069],[103.724386, 36.106045]];


    for (var i = 0, marker; i < positions.length; i++) {
        marker = new AMap.Marker({
            map: map,
            position: positions[i],
            icon: 'img/d.png',
            offset: new AMap.Pixel(-13, -30)
        });
```

Figure 4 – Listing of map's code in JavaScript language

Map's code to the JavaScript language:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
<meta name="viewport" content="initial-scale=1.0,
user-scalable=no, width=device-width">
<link rel="stylesheet"
href="https://a.amap.com/jsapi_demos/static/demo-
center/css/demo-ce.." />
<style>
html,
body,
#container {
width: 100%;
height: 100%;
}
.amap-icon img {
width: 25px;
height: 25px;
}
</style>
<title>We-Maps</title>
</head>
<body>
<div id="container"></div>
<script
src="https://webapi.amap.com/maps?v=1.4.10&key=f3
93f3131f6d9ca.."></script>
<script
src="https://a.amap.com/jsapi_demos/static/demo-
center/js/demoutil.."></script>
<script>
var map = new AMap.Map('container', {
center: [103.724386, 36.107072],
layers: [
new AMap.TileLayer.Satellite(),
new AMap.TileLayer.RoadNet()
],
zooms: [4,18],
zoom: 17
});
var markers = [];
var positions = [[103.724386, 36.107072], [103.724386,
36.106075],[103.724386, 36.106069],[103.724386,
36.106045]];
map.on("complete", function(){
log.success("Complete loading map！ ");
});
for (var i = 0, marker; i < positions.length; i++) {
marker = new AMap.Marker({
map: map,
position: positions[i],
icon: 'qr-code.png',
offset: new AMap.Pixel(-13, -30)
});
markers.push(marker);
}
</script>
</body>
</html>
```

The map is located on Github. You can follow the next link:

https://dariabrazhnyk.github.io/

Or by scanning the QR-code.

Figure 5 – QR-code for to pass to the created map

As an example, was taken a map of the city of Lanzhou (China) with its center in Lanzhou transport University. On the created map, QR codes act as special symbols with additional information. The object icons were connected with the QR code.

A relatively large part of the QR-code can be painted over or hidden behind any pictures, but it is still readable. With the aim, to provide bigger understanding for users, of what the QR-code refers to, the icon of a specific object was placed inside QR-code.

Figure 6 – Example of special symbols with the QR-code, inside which is placed the icon of a specific object

In addition, by using QR-code control system can monitor the productivity of each code to within one scan. The manufacturer can access various data relating to the widespread QR-codes. In addition to the frequency of scaning of codes, it is possible to receive number of unique scannings, to thereby define how many people scaned the code. Moreover, it is possible to obtain information about the location, date, time, on the used device and its operating system at each scanning. Determine the place where the code has been scan, it is possible by using the IP address of the equipment. And although the location is not defined accurately, this data are quite enough for analysis and statistics. All these data are received in real time (online), that is, each new scanning is displayed within a few seconds.

CONCLUSIONS**.** When making We-maps selection of symbols to the map information must follow a systematic approach. This is process is to ensure a systematic approach in categorizing data and ensuring the users and cartographers' perception corresponds. When selecting the appropriate symbol one has to; determine the nature of the information, the perception of the property to be conveyed, the recommended visual variables and the reproduction facilities available, appropriate visual variable are considered under the map phenomenon(Cobb & Olivero). When selecting appropriate symbols, it means drawing methodology awareness by introducing and using Cartographers' toolbox. Selection of appropriate symbols for We-maps is identified with multiple professional back grounds and making use of symbol libraries to display GIS results.

Also on the basis of this article can draw the following conclusions:

If you use a library with symbols of this type, then the map itself remains easy and fast to use, and does not have a complex content. At the same time, users can easily get complete information about a particular place or object that interests them, without spending time and traffic on loading extra unnecessary information. The biggest advantages are the speed and mobility of access, as well as the simplicity and convenience in usage for both PC and mobile phone including other devices.

The purpose of this work is to identify the most convenient and understandable for ordinary users, as well as suitable for We-maps symbols and determine the most optimal method of creating these symbols.

REFERENCES

1. Mikheeva, A.V., Marchuk, An.G., Dyadkov, P.G. (2014), Geoinformation Systems for Studying Seismicity and Impact Cratering using Remote Sensing Data, Geographic Information Systems (GIS): Techniques, Applications and Technologies. – Nantes University, France: Nova Science Publishers, p. 151–216.

2. Levis, P., Madden, S., Polastre, J., Szewczyk, R., Whitehouse, K., Woo, A., Gay, D., Hill, J., Welsh, M., Brewer, E., Culler, D. (2005), TinyOS: An operating system for sensor networks In: Weber, W., Rabaey, J.M. and Aarts, E., Eds., Ambient Intelligence, Springer Berlin Heidelberg, Berlin, 115-148.

3. Krygier, J., Wood, D. (2016), *Making maps: a visual guide to map design for GIS*: Guilford Publications.

4. Knowles, A. K. J. P. h. H. m., spatial data,, & scholarship, G. a. c. h. (2008). GIS and History. 1-13.

5. Van Oosterom, P., Zlatanova, S., Penninga, F., Fendel, E. (2008), *Advances in 3D geoinformation systems*: Springer.

6. Istomin, E., Sokolov, A., Abramov, V., Gogoberidze, G., Fokicheva, A. (2015), Methods for external factors assessing within geoinformation management of territories, *15th SGEM GeoConference on Informatics, Geoinformatics and Remote Sensing*.

7. Dresen, B., Jandewerth, M. (2012), Integration of spatial analyses into LCA—calculating GHG emissions with geoinformation systems, *The International Journal of Life Cycle Assessment*, *17*(9), 1094-1103.

8. Kowal, K. (2002), Tapping the web for GIS and mapping technologies: For all levels of libraries and users, *Information Tehnology Libraries 21*(3), 109-114.

9. Acemoglu, D., Robinson, J. A. (2005), *Economic origins of dictatorship and democracy*: Cambridge University Press.

10. Taylor, W., Plewe, B. (2006), "The effectiveness of interactive maps in secondary historical geography education", Cartographic Perspectives, 55, 16-33.

11. QR-code capability, 4.10.2015. URL: https://sfztn.com/qr-code-generator/vozmozhnosti-qr-koda.

12. 张莉琴，万春晖.自媒体的传播特征与公民媒介素养的提升[J]. 传媒，2014，30(13)：49-50.

СПОСОБ СОЗДАНИЯ БИБЛИОТЕКИ СИМВОЛОВ ДЛЯ WE-КАРТ

Д. В. Бражник, Ян Хаовен

Ланджоуский транспортный университет

Аннин Си Лу, 88, Провинция Ганьсу, Китайская народная республика. E-mail: dariabrazhnyk@gmail.com

Приведен обзор современных методов создания обозначений для геоинформационных систем (ГИС) по типу "We-карт". Такие обозначения составляют целые библиотеки условных обозначений. Методологии включают определение последовательных шагов, которые необходимо соблюдать при создании информационных обозначений, подходящих для использования в We-картах. Процесс создания любой карты, включает в себя обобщение действительности. Как обычные пользователи, так и коммерческие компании заинтересованы в геоинформационных данных, поэтому сегодня данное направление активно развивается. Разработана собственная методология создания информационных обозначений для геоинформационных систем с использованием QR-кода. Главная идея состоит в объединении условного обозначения и QR-кода с минимальными потерями в качестве для распознания. Основная информация об объекте хранится в QR-коде. Реализация идеи выполнена на JavaScript с использованием API сервиса Amap.com, который является лидером географических информационных систем в Китае. Сервис API был выбран потому, что другие ГИС-системы либо запрещены (Google maps), либо неинформативны (Yandex maps). Данная реализация представлена размеченной картой с собственными условными обозначениями. Геоцентром карты является Ланджоуский транспортный университет. Сама карта представляет собой веб-страницу, которая реализована с помощью языка гипер-разметки (Html). При загрузке веб-страницы отправляет запрос с использованием языка JavaScript в веб-сервис для построения карты. На карте реализованы несколько инструментов: панель масштабирования и геолокация пользователя. Информация о местоположении пользователя определяется с помощью GPS-координат пользователя или по IP-адресу ближайшей точки сети. Карта имеет 3D слой представления объектов на карте, для удобства ориентации на местности. Данная реализация находится на бесплатном Git сервисе GitHub в открытом доступе. Для пользователей был сгенерирован код для перехода к данной карте. Код может быть отсканирован любым устройством, имеющим камеру и программу для сканирования QR-кодов.

Ключевые слова: Библиотека символов; We-карты; ГИС, JavaScript, QR-код, Amap, API, Git.

Стаття надійшла 08.10.2018.