

## **A decision support tool for current weather and weather forecast using the Android system**

*Jan Bohacik<sup>1</sup>, Department of Informatics at the University of Zilina, Slovakia*

**Abstract:** Nowadays, it is typical that people need detailed and easily available information about current weather and weather forecast. A solution to this requirement can be a mobile phone application which communicates with a data server and provides all useful information in an easily useable user interface. As a consequence, an Android based mobile application is provided. It uses XML data from OpenWeatherMap and analyses them with a parser. DOM parser and SAX parser are utilized in the application and they are compared experimentally. It is found that the SAX parser is several times quicker than the DOM parser.

**Key words:** XML parser, decision support, weather forecast, OpenWeatherMap, Android

### **1. Introduction**

Mobile devices such as smartphones or tablets are becoming common more and more and are widely used in banking, news reading, shopping, social networking and so on. The most popular operating system in mobile devices is Android (6). Therefore, it is sensible to create software applications for this platform. Decision support tools are an interesting subgroup of these applications and they are an active topic in research (4)(5)(7). Some important information for everyone is current weather and weather forecast. It is something that can always be available for all mobile devices owners with current technologies. A weather application in mobile devices is able to use data from the Internet, but it needs to process it quickly so that it is not too demanding for the resources of the mobile device.

The organization of the paper is as follows. The implemented application for current weather and weather forecast is described in Section 2. In Section 3, the algorithms used for parsing weather data from OpenWeatherMap are analyzed. The experimental results achieved with the parsing algorithms are in Section 4. Section 5 concludes the paper.

---

<sup>1</sup> Ján Boháčik with all Slovak diacritics.

## 2. Android application

The basic features of the application are downloading and representation of information about the current and future weather, selection of locality, setting of the measurement unit for temperature, and selection of the algorithm for data parsing. The data about weather is taken from online service OpenWeatherMap (1) which provides information through an Application Programming Interface (API). It is possible to use 60 different API calls per minute free of charge and get information about the current weather and the weather for the following five days in three-hour intervals. The information in the service is provided as JSON or XML data files. The provided application uses the XML format as it is also used in the Android system e.g. for definition of layouts, strings, animations and others. The XML files are available for the application only if an API key is automatically generated through the website of the OpenWeatherMap service. The key is then used in the URL of the XML document as <http://api.openweathermap.org/data/2.5/weather?q=city&appid=key&mode=xml>. An example of the XML document is as follows (2):

```
<current>
  <city id="3056508" name="Zilina">
    <coord lon="15.74" lat="49.22"/>
    <country>SK</country>
  </city>
  <temperature value="285.15" min="285.15" max="285.15" unit="kelvin"/>
  <humidity value="81" unit=""/>
  <pressure value="1006" unit="hPa"/>
  <wind>
    <speed value="1.5" name="Calm"/>
    <direction value="260" code="W" name="West"/>
  </wind>
```

```
<clouds value="75" name="broken clouds"/>
<visibility/.
<precipitation mode="no"/>
<weather number="803" value="oblacno" icon="04d"/>
<lastupdate value="2016-04-14T14:00:00"/>
</current>
```

A document as above is downloaded and parsed by the application through a parser.

### 3. Parsing algorithms

In many domains, XML has become the standard format for data representation and transfer but its wide adoption is mired by inefficient document-parsing methods (3). An XML parser is an effective tool reading an XML document and providing interface for access of its content and structure. Parsing is a core operation performed before an XML document can be navigated, queried or manipulated. However, it is a costly operation which may decrease XML processing performance. In the provided Android application, DOM and SAX parsers are used. DOM parser creates a tree structure in the computer memory and this structure corresponds to the XML document (8). It contains all elements, attributes and data. The DOM defines several interfaces. The most common ones are as follows:

- **Node** – The base datatype of the DOM.
- **Element** – The vast majority of the objects which are dealt with are Elements.
- **Attr** – It represents an attribute of an element.
- **Text** – The actual content of an Element or Attr.
- **Document** – It represents the entire XML document. A Document object is often referred to as a DOM tree.

SAX parser reads the whole document and it executes some particular event without placing the whole document into the memory when it encounters an element or attribute (9). Unlike a DOM parser, a SAX parser creates no parse tree and the XML document is analyzed as follows:

1. The XML document is read from top to bottom and tokens that make up a well-formed XML document are recognized.

2. Tokens are processed in the same order that they appear in the document.
3. The application program is reported the nature of tokens that the parser has encountered as they occur.
4. The application program provides an event handler that must be registered with the parser.
5. As the tokens are identified, callback methods in the handler are invoked with the relevant information.

The SAX parser should be used especially when the document does not contain elements which are inserted into each other deeply or when the document is too large for the computer memory. A disadvantage is that there is no random access to the XML document.

#### 4. Experimental results

Data parsing in the provided Android application was tested with measurement of duration in milliseconds as the criterion for both the DOM parser and the SAX parser. Duration was counted from the beginning of the parsing algorithm to its end. The data in the experiment were acquired through OpenWeatherMap API before testing so that the speed of the Internet connection did not influence the overall duration. There were twenty simulations in the experiment altogether and the results for particular parsers are shown in Table. 1. The SAX parser with 71.95 milliseconds on average for all simulations is much quicker than the DOM parser with 419.35 milliseconds.

*Table. 1: Speed of parsers in milliseconds for particular simulations (2).*

Simulation	Parser	
	DOM	SAX
No. 1	325	92
No. 2	414	107
No. 3	496	87
No. 4	403	50
No. 5	452	69
No. 6	443	66

No. 7	333	63
No. 8	378	52
No. 9	382	94
No. 10	493	63
No. 11	431	83
No. 12	386	60
No. 13	452	66
No. 14	387	72
No. 15	498	45
No. 16	473	65
No. 17	431	68
No. 18	363	82
No. 19	355	71
No. 20	492	84
<b>Maximum</b>	498	107
<b>Minimum</b>	325	45
<b>Average</b>	419.35	71.95

## 5. Conclusions

An Android based mobile application for current weather and weather forecast was created. The application used weather data from online service OpenWeatherMap and parsed them with a DOM/SAX parser. Both parsing algorithms were analyzed theoretically and compared experimentally through twenty simulations where the duration of parsing was counted. The SAX parser appeared to be much quicker with 71.95 milliseconds on average. As a consequence it seems to be a better choice for XML data processing than the DOM parser in the presented type of application. The Android application could be improved with addition of several functionalities such as automatic notifications for bad weather in the area where the

mobile phone is or a widget that would show the weather without execution of the application.

## **Bibliography**

- (1) Chen LH, Hsieh WF, Lin JY: Synthesizing a NPR navigating animation based on Route Recommendation, Proceedings of the Conference on Technologies and Applications of Artificial Intelligence, Pages: 71-76, Year: 2015.
- (2) Chudjak K.: Implementation of a Decision Support Tool Using the Android System (Bachelor's thesis) (Published by: University of Zilina; In: Zilina, Slovakia), Pages: 44, Year: 2016.
- (3) Deshmukh VM, Bamnote GR: An empirical study of XML parsers across applications, Proceedings of the International Conference on Computing Communication Control and Automation, Pages: 396-401, Year: 2015.
- (4) Koli RP, Jadhav VD: Agriculture decision support system as Android application, International Journal of Science and Research (Volume: 4, Number: 4), Pages: 903-906, Year: 2013.
- (5) Kuntagod N, Mukhrjee C: Mobile decision support system for outreach health worker, Proceedings of the 13th IEEE International Conference on e-Health Networking Applications and Services, Pages: 56-59, Year: 2011.
- (6) Thakkar NB: Google Android: An emerging innovative software platform for mobile devices, International Journal for Innovative Research in Science & Technology (Volume: 1, Number: 6), Pages: 272-278, Year: 2014.
- (7) Ventola CL: Mobile devices and apps for health care professionals: Uses and benefits, Pharmacy and Therapeutics (Volume: 39, Number: 5), Pages: 356-364, Year: 2014.
- (8) www.tutorialspoint.com: Java DOM Parser - Overview, Available at [http://www.tutorialspoint.com/java\\_xml/java\\_dom\\_parser.htm](http://www.tutorialspoint.com/java_xml/java_dom_parser.htm), Access: 15 April 2016 10:15
- (9) www.tutorialspoint.com: Java SAX Parser - Overview, Available at [http://www.tutorialspoint.com/java\\_xml/java\\_sax\\_parser.htm](http://www.tutorialspoint.com/java_xml/java_sax_parser.htm), Access: 16 April 2016 08:30

### **Address of the author:**

Ing. Jan Bohacik, PhD.  
Department of Informatics  
Faculty of Management Science and Informatics  
University of Zilina  
Univerzitna 8215/1  
010 26 Zilina  
SLOVAKIA  
Jan.Bohacik@fri.uniza.sk

### **Acknowledgements**

Software development and simulations were carried out by Kristian Chudjak during the author's supervision of his Bachelor's thesis.

The article was accepted for publication in September 2016 by the publisher of the Journal of Information Technologies (Vol. 9, No. 2, ISSN: 1337-7467), i.e. by the Faculty of Mass Media Communication, University of Ss. Cyril and Methodius in Trnava, Slovakia.