A New Wireless Web Access Mode Based on Cloud Computing

Yunpeng Xiao\textsuperscript{1}, Yang Tao\textsuperscript{2}, Qian Li\textsuperscript{3}
\textsuperscript{1,2}College of Software, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
\textsuperscript{3}Department of Academic Affairs, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
shineagle2005@hotmail.com, taoyang64@cta.cq.cn, liqian31992@hotmail.com

Abstract

As most websites are designed for desktop PCs, it is extremely uncomfortable to browse these large pages on a wireless handheld device with small screen and limited user interface. So it is necessary to adapt these web pages to small screen devices. Besides, as the limited computing ability and capacity of storage of wireless handheld devices, it is also extremely challenging to deploy existing web page adaptation engine. By referring to huge computing ability and storage resource of cloud computing infrastructure, a new wireless web access mode is proposed. Firstly, the system framework is present. Subsequently, the two key components of system are described in detail: the one is distributed web page adaptation engine, which is designed for the purpose that the engine can be carried by computing cloud distributed and parallel; the other is distributed web page blocks management based on cloud computing, which is proposed so that the web page adaptation engine can be deployed reasonably. Moreover, a prototype system and a set of evaluation experiments have been implemented.

1. Introduction

With the rapid development of wireless telecommunication technology and increment of bandwidth, more people are using their wireless handheld devices for web surfing when they are on the move. However, wireless handheld devices with constrained functionality such as small screen, limited computing power and so on, restrict the user oriented QoS provided for wireless web surfing. It is difficult for wireless mobile users to browse large web page designed for PC users comfortably.

Researchers have proposed some effective algorithms which split web page into multiple blocks suitable for displaying on small screen device reasonably \cite{1,2}. Meanwhile, it is necessary to put these existing web adaptation engines into practice. Most of existing engines are carried by proxy or server when they are implemented.

We have implemented a web page adaptation engine for wireless devices based on proxy mode \cite{3}. Whereas, by experiments, it is found that the proxy will be more inefficient as more wireless terminals appear. To solve the problem, we proposed a P2P collaborative mode for wireless web surfing that wireless devices connect each other, forming a P2P collaborative system \cite{4}. Each device of the system can contribute their computing ability and capacity of storage to the one that needs them. On the contrary, the system will be more efficient as more wireless terminals join the system. However, the characteristics of wireless computing environment such as mobility, high frequency disconnection, network condition diversity, and network communications dissymmetry and so on, cause the system owns few peers at most time, exhibiting a low efficiency.

There is a strong desire to analyze the data in the emerging cloud computing infrastructure because of its strong value propositions \cite{5,6}. A cloud computing infrastructure, such as Amazon Elastic Computing Cloud (Amazon EC2) \cite{7}, provides huge potential computing ability, typically in the form of virtual machines, to end users on-demand from remote locations in the Internet. Cloud presents many value propositions which include: the cloud provided an abstraction of some part of the environment such that the user didn't have to perceive nor care about it; besides, cloud is currently a storage place for data and applications and that it will move to being “like electricity” in its accessibility to developers and users alike. In addition, because of its scale and expertise, Cloud could provide a much more reliable and secure infrastructure than most enterprises could afford. For example, Amazon has multiple data centers and each data center also has multiple backup power generators. Amazon also has strong security measures to guard against not only physical security breach but also hacker attacks.

By means of the existing web page adaptation engine we proposed \cite{3}, and by referring to cloud computing infrastructure, a new wireless web access mode is proposed. A distributed web page adaptation engine is designed firstly for the convenience of the following...
deployment so that the engine can be carried by computing cloud distributed and parallel. Then a distributed web page blocks management based on cloud computing is provided and a specific algorithm is designed to assign engine processing task, coordinate working between each computing cloud. Moreover, a prototype system and a set of evaluation experiments have been implemented.

The rest of the paper is organized as follows. Section 2 presents the software framework of our system. We describe the key component modules of the system in section 3. A prototype system is implemented and verified by a set of experiments in section 4. Finally, we indicate our future work and conclude the approach in section 5.

2. Framework

The original intention of designing the system is utilizing huge computing ability and storage capacity of cloud computing infrastructure to carry web page adaptation engine. Figure 1 shows the framework of the whole system. Cloud portal is an interface between wireless end users and original websites. Here, it represents the new service for wireless web accessing. From the system framework, we can conclude that a distributed web page adaptation engine and distributed web page blocks management based on cloud computing must be designed so that the engine can be carried by computing cloud distributed and parallel.

3. Component modules

3.1. Distributed web page adaptation engine

The distributed web page adaptation is composed of two modules. They are Structure Processor and Iterative Page Blocker. Structure processor parses normal web page and creates DOM tree as output. Iterative page blocker splits web page into small blocks and makes the refinement of adaptation engine can be carried by computing cloud distributed and parallel.

(1) Structure processor

Structure processor parses normal web page based on HTML criterion fills in missing html tag like <li>, <hi> and so on. Then the module constructs DOM tree corresponding to the page based on DOM criterion. Web page’s content and structure are reflected by the DOM tree, including those elements that build up web page like element name, element content and element attributes, including relationship between those elements as well. Figure 2 shows an example that a DOM tree corresponds to its HTML document. A DOM tree of structure HTML is more easily to be accessed and operated than the raw web page stream data.

![Figure 2](image)

Figure 2  DOM tree corresponding to HTML document

(2) Iterative page blocker

The main works of iterative page blocker are designing a page blocker algorithm and making the algorithm can be deployed into cloud computing model distributed and parallel. So how to split a web page into small blocks by a reasonable and iterative way is the key issue to this module.

Up to now, because of the low available intelligence of semantic technology in practices, most of current studies on web page adaptation are limited to special pages, special websites, or special page format. These methods include structure HTML analysis method [8], process method based on nature language [9], machine learning [10] and Ontology [11]. However, these researches are too specific to be applied broadly. As random of user surfing behavior and high require of response time, when the wireless devices access web pages of new field or new structure, these systems usually can not work successfully.

![Figure 3](image)

Figure 3  Layered iterative relationship at vertical viewpoint

To design an iterative page blocker, page structure is analyzed firstly here. Actually, no matter in vertical direction or in horizontal direction, there is some hierarchy iterative relationship between pages and between blocks inside a web page.

From vertical viewpoint, we can organize all the web pages of a website layered iterative relationship by using...
theory of layered network for reference. As shown in figure 3, the page in the first layer is index.html usually, and the page will be parsed into many blocks by page blocker we proposed [3]. Each block contains many hyperlinks that link to the pages in the next lower layer. Repetition goes on until the current block only contains content information without any links or the hyperlink which the current block contains links back the page in upper layer.

From horizontal viewpoint, there is the similar situation to different blocks of a web page. Figure 4 shows a normal web page is split into three blocks at first step by page blocker mentioned above. If we refine the accuracy of the page blocker, some blocks will be split further.

![Layered iterative relationship at horizontal viewpoint](http://www.cq upt.edu.cn/english/index.php)

**Algorithm**

Since there is some hierarchy iterative relationship between pages and between blocks inside a web page no matter in vertical direction or in horizontal direction, and referring to the table tag method and DOM criterion, an innovative and simple distributed iterative web page blocker algorithm is presented here.

After a computing cloud receives HTML code of a web page or a page block, its corresponding DOM tree is created at first. Then a top-down traverse is used to select all different table type subtrees which are the nearest to the current root node. For each table subtree, to the table root node, if all its direct child nodes only have attributes without value, ignore this kind of subtree. Thirdly, to the remaining table subtrees, the mainly works are in vertical or horizontal direction logically. If there is only one subtree needs processing and the web page which the subtree belongs to is the page the user is viewing, then vertical processing will begins. The web pages which hyperlinks of the current page blocks link will be parser in other computing cloud distributed and parallel. If there are more than one subtree, horizontal processing will begins, the blocks will be sent to other computing cloud and split further. The pseudocode of the algorithm is given as fellows:

1. Initialize and create a DOM tree corresponding to original web page
2. From current root node, use a top-down traverse: select all different 'table' subtrees that are the nearest to root node
3. For i=1: subtrees.Number/ignore subtrees which keep page style only
   - Boolean flag=false
   - For j=1:subtrees[i].childNum
     - if child[j].value=null
       - [flag=true, break]
     - If flag=false
       - Ignore subtrees[i]
4. If subtrees.Number=1/vertical processing
   - If verticalLayer=1/processing 2nd layer for cache only
     - For k=1:hyperlinks.Number
       - Send the hyperlink request to other cloud
     - 5.else if subtrees.Number>1/horizontal processing
       - For i=1:subtrees.Number
         - Send the page block Html code which the subtrees[i] corresponds to the other computing cloud

3.2. Distributed web page blocks management

After the original web page is split into small blocks and allocated into cloud computing infrastructure by distributed web page adaptation engine, it is important to manage these distributed web page blocks subsequently. To meet the requirement, two issues must be done: data structure definition and page blocks mapping.

(1) Data structure

To manage these distributed web page blocks, a simple Distributed Hash Table (DHT) named Blocks Table is defined to record the blocks. The table mainly contains three attributes:

*Blocks Table*{BlockID, BlockURL, HTMLcode} 

**BlockID** is the unique identification of each block, which is created by distributed web page adaptation engine. It is a digital sequence which is segmented by dot like a.b.c. Here a, b and c are 1Bit random number. Dot represents the layer number of iterative page blocker. No matter in vertical direction or in horizontal direction, once the block is going to be parser further, a dot will be appended at the rear of current BlockID.

**BlockURL** is the URL which cloud portal uses it to obtain each block. Interrogation expression is used here to construct the URL. For example, as shown in figure 4, if the original page’s URL is http://www.cq upt.edu.cn, and if the blockID of the block, which is at the top of the original page, is ****, then the block URL will be http://www.yahoo.com?blockid=****.
HTMLcode is HTML source code of each block which is created by iterative page blocker.

(2) Page blocks mapping

This issue focuses on how to get HTML source code of target page block in distributed cloud computing infrastructure by using Blocks Table defined above. By using the Distributed Hash Table and layered iterative relationship while constructing BlockID, a distributed mapping algorithm is designed. The pseudocode for the function is as follows:

```
distributedMappingFunc(String BlockID): return HTMLcode
if (BlocksTable.contain(BlockID))
    return BlocksTable.getHTMLcode(BlockID)
String newBlockID = BlocksTable.maxSubString(BlockID)
String blockURL = BlocksTable.getBlockURL(newBlockID)
request.forward(blockURL)
```

4. Implementation and evaluation

4.1. Implementation of prototype

The system was implemented based on the above design. We implemented the prototype system entirely in Java. Our scheme was carried out in Amazon EC2 Cloud. Amazon promises each server is equivalent to a system with a 1.7GHz x86 processor, 1.75GB of RAM, 160GB of local disk, and 250Mb/s of network bandwidth. We use a cloud tool named EC2Deploy [12], which is a set of tools for deploying, managing and testing Java EE applications on Amazon EC2, to deploy the scheme. A Java open source project named HTMLParser [13], which can parser semi-structured HTML page, fulfills missing tag and creates DOM tree. Besides, it supports powerful JAVA library which can make multiple analysis and operation based on DOM document. SUN Java ME Wireless Toolkit2.5.2 simulator is used to simulate handheld terminal device. Bandwidth is set at 30Kbps to simulate the current China Mobile GPRS’s access bandwidth. Figure 5 shows process result corresponding to part of the figure 4.

![Figure 5 Snapshot of prototype system](image)

4.2. System evaluation

Evaluation is divided into efficiency evaluation and effectiveness evaluation. Evaluation scheme is similar to the scheme implemented in [3,4], where we proposed deployment based on proxy and P2P Collaborative Mode respectively. We also compare experiment data under the three different schemes. For comparison, we call the system in [3] as traditional system (TS), system in [4] as P2P system (PS), whereas the system described in this paper is referred as cloud computing system (C2S).

For efficiency evaluation, 3000 web pages of 10 typical portal websites such as CQUP, sina and yahoo etc. are chosen to build up efficiency evaluation test set. Total files’ size is 51600kB and 17.2kB per file. When in data collection time, the collection system uses thread pool model and set the pool with 50 threads. System efficiency is mainly determined by system processing speed. Table 1 shows test result, it reaches the conclusion that efficiency of the cloud computing system can meet the demand of processing large number online web pages swiftly. And the C2S’s efficiency is higher than the TS’s and PS’s because the C2S makes full use of huge computing ability and storage resource of cloud computing infrastructure.

<table>
<thead>
<tr>
<th>Deployment scheme</th>
<th>Execute time(s)</th>
<th>Speed(files/s)</th>
<th>Speed(kB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS</td>
<td>18.93</td>
<td>158.48</td>
<td>2725.83</td>
</tr>
<tr>
<td>PS</td>
<td>23.83</td>
<td>125.89</td>
<td>2165.34</td>
</tr>
<tr>
<td>C2S</td>
<td>13.47</td>
<td>222.72</td>
<td>3830.73</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test items</th>
<th>TS</th>
<th>PS</th>
<th>C2S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response time</td>
<td>86.47%</td>
<td>92.12%</td>
<td>94.76%</td>
</tr>
<tr>
<td>Distinguish page block</td>
<td>99.03%</td>
<td>88.32%</td>
<td>87.29%</td>
</tr>
<tr>
<td>Page block identify</td>
<td>94.54%</td>
<td>90.81%</td>
<td>90.32%</td>
</tr>
<tr>
<td>Page block merge</td>
<td>87.30%</td>
<td>88.34%</td>
<td>85.61%</td>
</tr>
<tr>
<td>Content block title identify</td>
<td>81.21%</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

For effectiveness of system, especially for accuracy of distributed web page adaptation engine, it is hard to give formal evaluation conclusion. It is also subjective during the evaluation of system effectiveness. So we use general accuracy as evaluation standard, artificial evaluation method in practical process. Thirty students are invited to finish a questionnaire with five test items in table 2. All the participants are active users of web in their daily lives. Firstly, we give two days to the participants to familiar with operating the three systems; after that, each participant is asked to finish the questionnaire. Here we extract 100 web pages random and make them artificial evaluated for evaluation of system effect, according to “successful” or “unsuccessful” to give evaluation result for each performance index shown in table 2. We can see from the table that effect of each system is acceptable and each performance index is above 80%.
In particular, customer satisfaction is tested and compared with the three different modes. We simulate different number of wireless terminal devices respectively from 20 to 200. Twenty students are invited to finish the questionnaire, expressing their satisfaction including speed, GUI and whatever that their feeling about the three ways. From figure 6, we can conclude that the proxy mode will be more inefficient as more wireless terminals appear. On the contrary, the P2P collaborative mode will be more inefficient when few terminals exist. As the huge computing ability and storage resource of cloud computing infrastructure, no matter how many wireless terminals exist, the cloud computing mode always has a high customer satisfaction.

5. Conclusions and future work

With the evolution of wireless telecommunication technology and information technology, it seems more important to provide wireless users with a swift, convenient, economical, personalized and shining appearance way for web surfing. A new wireless web access mode based on cloud computing infrastructure is proposed in this paper. Compared with existing modes, the scheme makes full use of huge computing ability and storage resource of cloud computing infrastructure, provides a new wireless web access service with more swift and better user experience by experiments.

In the next step, we will focus on optimizing the web page adaptation algorithm for dealing with complex structured web page and strengthening security of cloud computing model.

References