

# Application Of A New Printing System For Smart Devices

Zhang Shujun

Department of computer and Information Engineering  
Chengde Petroleum College  
Cheng De, China  
zhsjun@sina.com

Zhang Liming

Department of Mechanical Engineering  
Chengde Petroleum College  
Cheng De, China  
36752920@qq.com

**Abstract**—Different from the existing printing methods, such as adding ActiveX control and converting the document to other forms (PDF and etc.), through the research on the smart device application system, this paper puts forward the printing method for general report forms based on mobile intelligent device according to the characteristics of print statements. In order to solve the problems of the complex print statements and multi page continuous printing, this printing method uses HTML5, CSS to define the format of the report form, uses ASP.NET technology to realize the network data access of mobile intelligent device and uses JavaScript technology to build a report template processing engine. And also, this printing method solves the problem of complex header print. This research supports all access to WEB devices and can be applied in all reporting system. The current printing system can process 500 pages one time with no errors occurred, which can improve work efficiency effectively.

**Keywords**—web report; template; printing system; smart device; web device

## I. INTRODUCTION

A refined reporting and printing function is one of the crucial characters that defines a good system, no matter such is used for enterprise application system or public information platform. Its final product takes the shape of tables. Accordingly report printing system becomes the issue that Information system development personnel concerned[1]. This issue becomes more relevant when systems are not isolated anymore, but instead connected by the Web technology, which requires more efficiency of collecting data and printing it. The existing system only provides solutions as converting document to different forms.

Cloud introduces Smart Devices into the field of intelligent office, which consequently makes the need of an efficient printing organ more desirable. The present solution is to add Active X Control to the Web platform that generally used as bridge to connect the sever and users. This requires the Web to convert tables into other forms of documents: Excel, Word, PDF, etc. However this cannot be achieved in smart devices[2]. Furthermore this gives user a number of process extra.

After immense research and experiments on the smart device application system, this paper provides a new solution that enables the Web 2.0 (HTML 5, CSS,

JavaScript) to provide a cross-platforms system that prints templates.

## II. TEMPLATES

A template will fill in the collected data to a specific form, which makes the process of turning complicate data into tables possible. We used HTML 5 and CSS to define the template, and further strengthen the process by JavaScript. Users of Smart Devices or PC can print out data with no Plug-in[3]. Here, we mainly resolved the following questions:

How to define and process the template by using static technology as HTML 5, CSS and JavaScript?

How to print complicate tables?

How to print multiple pages documents?

How to print bills collectively and precisely?

How to print out tables with different headers in different pages?

How to apply in Smart Devices?

How to make the printings be consistent with the appearance in Browser?

This paper uses the Ajax and the ASP.NET to process the report parameter and data loading. The printing system structure is shown in Figure 1. The printing system is built on the Web application system[4], uses the related technologies in Web2.0 without additional plug-ins. The Ajax technology makes the data without refresh, so as to print paginated reports. Because the intelligent equipment can not be directly connected to the printer, therefore, the print content needs to be sent to the print queue and to be uniformly handle by the print server[5].

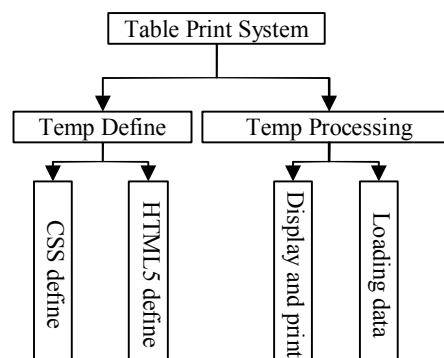


Figure 1. The Construction of the Printing System

The definition of the template can be freely determined in accordance with the need of tables, which generally include: header, body and footer. Each part can be further divided if necessary.

HTML 5 define template by using DIV and tables. CSS is used to render if necessary. This is especially true when processing multiple pages is needed, where CSS will be chosen with character of “page-break-before” or “page-break-after”. What’s interesting is different browser analyzes CSS character differently. This will usually happen when a single cell takes more than one page or the cell is cut off into two in different pages. Changes will be needed accordingly with the browser[6].

Relevant issues needed to be taken into account will change according to the content, either by using DIV or tables, or both. The process of defining template usually takes the following steps:

#### A. Determine the parameter in the tables

The names of parameters need to be written in lower case in order to be processed by JavaScript:

title: Topic of the tables.

teachername: the name of teachers and staffs which indicate the individuals that are shown.

department: department of the teachers.

printer: indicating the person who print the table.

printdate: the date of forming the tables.

#### B. Definition of HTML 5 documents.

The tags being used in HTML 5 mainly include DIV, SPAN, P, table, tr, th, td and etc. Specifies the unique id number for the data labels, which make JavaScript identify each group of data. Header definition[7]:

```
<div id="template">
<p id="title">{title}</p>
<div><span id="teachername">
{teachername}</span>
<span id="department">
{department}</span>
<div class="clear"></div></div>
```

Data will usually be transformed into tables that have headers. The header of the data will also be shown in the tables as header. The headers can be handled differently in multiple pages documents.

```
<table><thead><tr><th>&nbsp;</th>
<th>&nbsp;</th>
<th>order</th><th>items</th>
<th>January</th><th>February</th>.....
<th>December</th>
<th>Total</th></tr></thead>
```

Tag "thead" indicate the header. Tag "tbody" indicate the starting point of data.

```
<tbody id="data">{data}</tbody>
<tfoot id="sum">
<tr>
<td colspan=3>
Total
</td>
</tr>
</tfoot>
</table>
```

#### C. Defining the format of data.

For different table, each has its own form. However they can be collectively divided into repeated or un-repeated data. The repeated ones will be shown in groups while the un-repeated will be shown directly. If data can be displayed in form of JSON, it will be more efficient since these data can be directly process by JavaScript. The following is data in JSON format:

```
{ "year":2013,"teachername":"Teacher: name",
"department":"Department: Mechanical Engineering",
"printer":"Tom","printdate":"Date: 2014-07-22",
"multitem":true,"itemshow":"single","salary":{
"foreach":true,"colsum":true,"rowsum":true,
"sumcolumn":"1-12","data":{
"iname":"position value","i1":"730",.....,"i12":"0"},{
"iname":"Grade salary","i1":"365",.....,"i12":"0"},
.....}}
```

Most of the data in Web system comes from the data base, which should be transformed in JSON forms from time to time.

```
for (int i = 0; i < irow; i++){
Dictionary<string, string> d = new
Dictionary<string, string>();
for (int j = 0; j < icol; j++) {
string cn = dt.Columns[j].ColumnName;
d.Add(cn, dt.Rows[i][cn].ToString());
}
list.Add(d);}
```

Here, This Paper have a list of data that can be form in order, which will be further converted into JSON:

```
for (int i = 0; i < ilist; i++){
json += CreateJson(list[i]) + ",";
```

ASP.NET will send these data to users. It can be printed when users send these data to server when needed[8].

#### D. Defining the appearance of the table.

Display and style can be defined by CSS. The most used CSS format are shown as follows:

```
# template{width:100%}
#title{text-align:center; font-size:20px;}
.clear{clear:both;}
#teachername{float:left;}
#department{float:right;}
table{border-collapse:collapse;}
th,td{ border: 1px solid #ccc;}

```

Here, in order to differentiate styles, “@media” can be used (setting printing machine):

```
@media print{
table{border-collapse:collapse;}
th,td{ border: 1px solid black;}
}
```

The process being taken by browser has three steps: loading template, loading data, print. The former two can be achieved by Ajax technology using ASP.NET within the server un-synchronizedly. With such, data can be loaded separately according to the table.

```
p.appendChild(createPageBreak());
for (var i = 0; i < igzlen; i++) {
if (dTabData.rows[i].className == 'nextpage') {
tb=dTabData.cloneNode(false); tb.className += "
clear";
```

```
if(showhead.value == 'yes') {
    tb.appendChild(copyTableRow(head)); }
```

The sever program is relatively simple comparing with users' application. It receives printing instructions from users devices. For both users' applications and server, back-up connectors should be provided for R & D personals or system application personals[9].

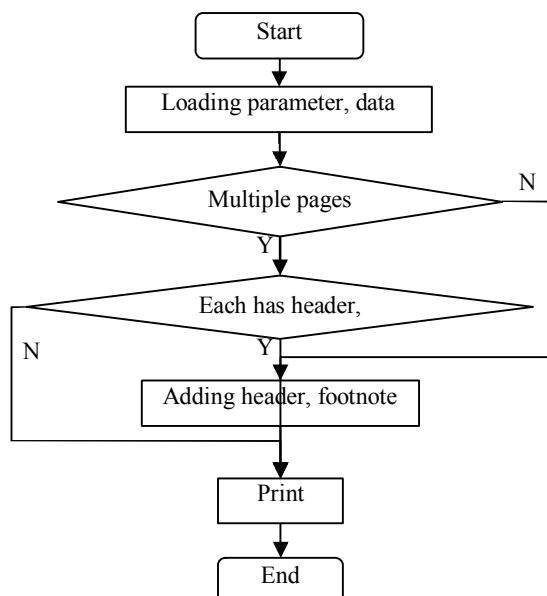


Figure 2. Server Process with Users Instructions

### III. EFFECT

All constructions described above have been applied in practices, No matter how the report format, according to which forms of the table can easily be changed with different definition of the template. Smart Devices can print out data table whenever and wherever by sending the print command. In order to support the ability of intelligent equipment will directly connect the printer to print, system provides the interface. With no further steps of converting documents, HTML 5 example can be shown below in Chart 3. So developers and system operators are very convenient. Through the template to create report results as shown in Figure 3. For multiple pages, the current system can process 500 pages one time with no errors occurred.

The personal income system in enterprises and institutions as an example shows the application of intelligent mobile printing. When the user browses the report table by intelligent devices such as mobile phone, iPad etc, he or she can be input to the print queue server by selecting the print command only. If the server is connected to the printer, the report data will be printed. It should be noted that, the print server and the application server is not a print system. Of course, it can also that the application server and the print server are combined using according to the actual

situation. In the personal income system, The print server independent is used, As shown in Figure 3.

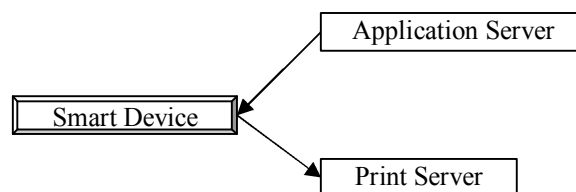


Figure 3. Smart Device Printing

On the basis of personal income distribution system in enterprises concerning the individual annual summary statistics report required, define the report template, Summary format considers the system shown in table 1.

TABLE1 Report Form

2014 summary of personal income distribution

Teacher: Teacher's name														Department: Computer and Information											
		Engineering																							
No	Items	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total											
1	Subtotal	5828	5828	5828	5828	5828	5828	5828	5828	5828	5828	5828	5828	69936											
2	1 Post Salary	750	750	750	750	750	750	750	750	750	750	750	750	9000											
3	2 Base Salary	365	365	365	365	365	365	365	365	365	365	365	365	4380											
4	3 Grade Salary	1550	1550	1550	1550	1550	1550	1550	1550	1550	1550	1550	1550	18600											
5	4 Allowance	1386	1386	1386	1386	1386	1386	1386	1386	1386	1386	1386	1386	16632											
6	5 Class Fee	1850	1850	1850	1850	1850	1850	1850	1850	1850	1850	1850	1850	22200											
7	6 Health Costs	5	5	5	5	5	5	5	5	5	5	5	5	60											
8	7 Workload	50	50	50	50	50	50	50	50	50	50	50	50	600											
9	Subtotal	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	7793.6	93523.2											
10	Only Child	120												120											
11	Stroke						100							100											
12	Examiners			235										235											
13	Teacher's Day									500				500											
14	Civilization	5000												5000											
15	Course Source			100										100											
16	Training	1800		500	500	500			1600					5400											
17	Journal													100											
18	Subsidies	400		200	200	200	200	400		200				1800											
19	Accumulation	573.6	573.6	573.6	573.6	573.6	573.6	573.6	573.6	573.6	573.6	573.6	573.6	6883.2											
20	Total	13721.8	6501.1	7595	7360	7450	6450	2000	500					31590.8											

The content HTML5 templates to define references table 1.

First to define the report title, which uses label p, sets the alignment for center and adds the variables {year} to define aggregate annual.

```
<p align="center">{year} summary of personal income distribution</p>
```

The data show in the tables. The top of the data tables show the title. The title includes the current staff name and unit employees. The names and units are displayed in the table at the top of the left and right area[10].

```

<table
border="1" id="table1"
bordercolorlight="#000000"
bordercolordark="#FFFFFF"
cellspacing="0" cellpadding="0">
<caption>
<span id="empname">Teacher: {tname}</span>
<span id="department">Department: {department}</span>
</caption>

```

The display mode of the header data uses the id selector in CSS to control in the table display. CSS is defined as follows:

```
#empname{left:10px;position:absolute;}
#department{right:10px;position:absolute;}
```

Here is the absolute position of the elements of the display control and need to set the relative position for the

title tag caption of form. The data header required to be print begins to show below the title. Defining the header and getting header data can be achieved by the various factors.

```
{tablehead}
```

As the table shows the longitudinal split into two large projects with the same transverse project, so, it is need to show the two items by the inherent format definition. The first definition:

```
<td rowspan="8">s<br>a<br>l<br>a<br>r<br>y</td>
```

First define the summary data template of the first item, here realize the summary data acquisition through using the variable {subtotalforsolory}.

```
<td colspan="2">Subtotal</td>
```

```
{subtotalforsolory}
```

The data shown uses the variable {subtotalforsolory}.

```
{dataforsolory}
```

After the first data is completed, the second items of data begin to be defined define. The contents are as follows:

```
<td rowspan="12">o<br>t<br>h<br>e<br>r</td>
```

The collection template definition of the second items data:

```
<td colspan="2">Subtotal</td>
```

```
{subtotalforother}
```

The processing mode for the detailed item data of the second items is similar with the first item. It is shown directly in variable {dataforother}.

```
{dataforother}
```

After all the details of data processing is completed, the summary data needs to be displayed in the interface, which is realized in variable {totaldata}

In the process of last definition template, The bottom data in the report form should be added into.

```
<div id="footer">
```

```
<span id="printer">{printer}</span>
```

```
<span id="printdate">{now}</span>
```

```
</div>
```

So far, the template definition is complete. The print are realized by calling printing system. The result is same with table 1.

#### IV. CONCLUSION

Printing through Smart Device with no plug-in or converting forms has been proved possible with the solution this paper provided. The only drawback this paper can say for now is this cannot have an effective control over the printing machine and papers. However this has been made up by the setting options provided by browser itself. For printing with Smart Devices system provide options backstage. Among all the things, printing multiple pages is the leading issues, which can be solved by JavaScript that can precisely locate, calculate what has been displayed.

#### REFERENCES

- [1] Li Yongfeng and Gu Chuan, "Designing and implementation of a template web printing method," Computer Applications and Software, vol. 27, Apr. 2010, pp. 117-119, doi: 10.3969/j.issn.1000-386X.2010.04.038.
- [2] GU Hongbin, "Research and implementation of Web Bills Precise Printing Technology," Computer Programming Skills & Maintenance, Dec. 2013, pp. 4-5,14, doi: 10.3969/j.issn.1006-4052.2013.10.001.
- [3] Zhu Min, "Design and Implement implementation of Web - based visualised," Computer Applications and Software, vol. 29, Jul. 2010, pp. 4-5,14, doi: 10.3969/j.issn.1000-386X.2012.07.057.
- [4] Flanagan, "JavaScript Definitive Guide," Beijing: Machinery Industry Press, 2012.
- [5] Nicholas C. Zakas, "JavaScript advanced programming," Beijing: People's Posts and Telecommunications Press, 2012.
- [6] Liu Xijie and Liu Lin, "HTML、CSS、JavaScript Webpage production from entry to the master," Beijing: People's Posts and Telecommunications Press, 2012.
- [7] Keith,J and Sambells,J, "JavaScript DOMArt of programming," Beijing: Electronics industry Press, 2011.
- [8] Douglas Crockford, "JavaScript the essence of language," Beijing: Electronics industry Press, 2012.
- [9] Addy Osmani, "JavaScript design patterns," Beijing: People's Posts and Telecommunications Press, 2013.
- [10] Ni Baotong and Wu Dongwei, "HTML 5+CSS 3 Web Site Design and Layout Webpage," Beijing:: Tsinghua University press 2013