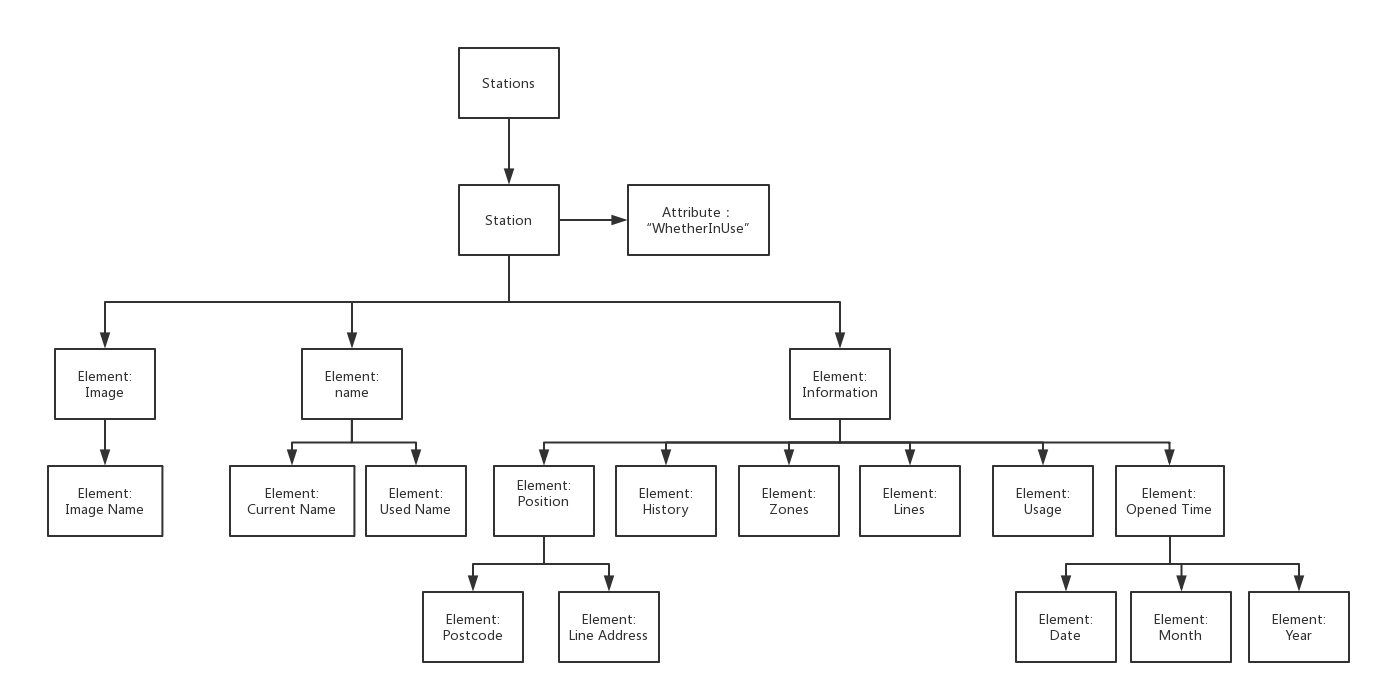
1. **Rational of Choice**

Due to that the webpage should contain the information which is not updated from time to time, we need to extract a kind of data which will exist for long time, and also is useful for user to read and learn.

In this condition, the information about London Underground Stations come to my mind so that I can collect their details such as name, location, lines, usage and history and so on, and then use the technology of Document Type Definition(DTD) to produce several XML files which will be used to make Extensible Stylesheet Language(XSL) files to produce HTML webpage so as to demonstrate the whole information for readers to learn.

Here the topic was chosen is My favourite 10 London Underground Stations which include the image of the 10 stations, their used name in the past, their current name, position, opened time, zones, lines, usage and some basic and simple history.



**Figure1: Tree Structure of This Project**

1. **Document Analysis**
2. DTD

Firstly, we need to write a DTD file as a schema to make 10 XML files for each station, which means we can use a single DTD as the XML structure to produce similar 10 XML files to describe the elements of ten stations which was stated above. The structure is shown in figure 1 which give a clearly description of the whole structure we will use in DTD.

Firstly, we have the root element called Stations, which will contain kinds of station information. Then for element of station, we can have three main elements to conclude which are image, name and information respectively. For image, we have image name to contain the path of image. Property of name has current name and used name which is used for holding the name used before and the current name. For information, we have position which contains two elements, postcode and address in detail, opened time which holds the date, month and year when the station started in using, history, zones, lines, and usage.

1. XML

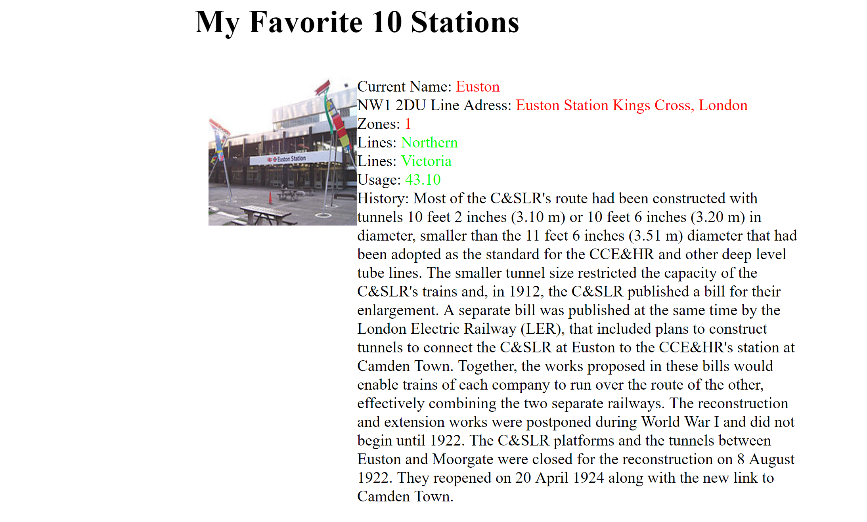
After using DTD file to produce 10 XML with necessary tags, then we fill in these tags with specific information. After then we can combine them into one file called stations.xlm which contains the whole ten stations’ details, which is convenient for XSLT to extract information with these details.

1. XSLT

For this project we will write two kinds of XSLT files which will be used to produce two kinds of forms to represent these information we extracted. The first form is table which will mainly use for-each select and when-otherwise logic sentence to control the whole frame. The second one is list which will be based on the apply-template structure.

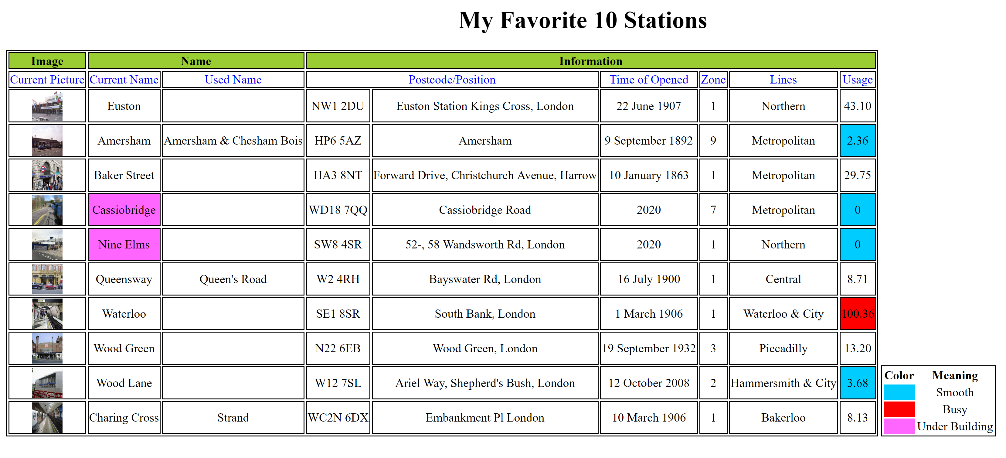
1. HTML

Then we will have two kinds of HTML forms to represent extracted information we have made above, which are shown below:



**Figure 2: List Form**

s



**Figure 3: Table Form**

1. **Implementation**

**Figure 5: Process of this Project**

1. DTD
2. Firstly, we use <!ELEMENT> to define all the nodes. For stations, the root node, we need to set it include ‘station+’ to make 1 or more station in XML, then we need to define station.
3. Element of station has an attribute called type, which is ‘#REQUIRED’ to define whether the station is currently under use. We use ‘opened’ to represent under using, and ‘building’ to mean under constructing.
4. Then we have three main details which are image, name and information in details. Image have a single element called ‘imgName’ to contain the src of the image in the file.
5. Element of name will have two basic elements which is ‘currentName’ meaning in use nowadays and ‘usedNmae\*’ in the past respectively. Here ‘\*’ means this station has or did not have any used name before.
6. Information has 6 different details information which are position, opened time, zones, lines+, usage and history accordingly.
7. Position contains two elements which are postcode and address.
8. Time of open has three elements which are date, month and year. Here we need to set date and month as ‘\*’ due to that if the station is under building, it can only have year to open in the future.
9. XML

From the DTD defined above, we will have ten XML files containing the 10 stations with the necessary element tags and then we fill in all the tags with the detail information we collected from website, which are Euston, Amersham, Baker Street, Cassiobridge, Nine Elms, Queensway, Waterloo, Wood Green, Wood Lane and Charing Cross. Especially, Cassiobridge and Nine Elms are under building now. Then these 10 files are combined into one file called stations.xml. Here, stations.xml still satisfy the schema defined in DTD.

1. XSLT
2. Table Form:

Firstly we use table form to illustrate the information we have filled in. In this situation, we use “<xsl:for-each>” to go through stations.xml to get all the station element. Then we want to mark up these stations which are under building, so we use when-otherwise statement to choose the attribute of station equals to “building” and then make the background color of this cell as plum. Then we add another statement to mark the usage above 50 as busy using with red and under 10 as smooth using with blue. Due to the appearance of the table, here we omit the information of brave history.

1. List Form: Another form is a list, which is used with ‘apply-template’ to demonstrate stations’ information. We set all each node a template to apply and define each template to extract the information. The main difference here is we added the information of history which is easy to show in list style.

After finish the two XSLT files above, these two forms are shown in two webpage which is presented above in figure 2 and 3. Additionally, two css files were built to adjust the two webpages in a solid way.

1. **Future Extension**

In the future, if anyone want to add some more stations the can just use the DTD file to produce more xml files and then fill the detail information into the tags. This is because DTD defines the document structure with a list of legal elements and attributes which means if someone wants to add some more new information they have to obey the rule defined in DTD so make the new information still in the same form of other information added before.

In addition, XSLT will go through all the information of xml to build webpage which means we can just need to use DTD to build a new xml file and fill in new information without any change to the code we write about DTD and XSLT. This is useful for some commercial sites doing business with underground stations to build up and extend a system which is used to demonstrate the detail information of these stations by just collect the new information and the run DTD and XSLT to implement that.