A More Precise Model for Web Retrieval

Junli Yuan
Institute for Infocomm Research
School of Computing, NUS
Singapore 119613
(+65) 6874-3121
junli@i2r.a-star.edu.sg

Chi-Hung Chi
School of Computing
National University of Singapore
Singapore 117543
(+65) 6874-2832
chich@comp.nus.edu.sg

Qibin Sun
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
(+65) 6874-6696
qibin@i2r.a-star.edu.sg

ABSTRACT
Most research works on web retrieval latency are object-level based, which we think is insufficient and sometimes inaccurate. In this paper, we propose a fine grained operation-level Web Retrieval Dependency Model (WRDM) to provide more precise capture of web retrieval process. Our model reveals some new factors in web retrieval which cannot be seen at object level but are very important to studies in the web retrieval area.

Categories and Subject Descriptors
C.2.5 [Computer-Communication Networks]: Local and Wide-Area Networks – Internet; C.4 [Performance of Systems]: Modeling techniques.

General Terms: Measurement, Performance

Keywords: Web retrieval, latency, performance, dependency, model

1. INTRODUCTION
Many works have been done to understand and improve web retrieval latency [1] [2]. However, most of them are object-level based. We would like to point out that this approach is insufficient and sometimes inaccurate. In current web system, web pages are often made up of multiple web objects. The relationship between the retrieval processes of objects in a page is actually quite complex. That prevents object retrieval latency from being mapped into page retrieval latency directly. To well understand and study the complex relationship between object and page latencies and the factors affecting them, we should investigate the process of web retrieval at more detailed levels. In this paper, we propose a fine grained operation-level Web Retrieval Dependency Model (WRDM) for studies of web retrieval performance. We show that the Definition Time (DT) and Waiting Time (WT) of objects are very important in web retrieval, yet they are largely ignored in object-level studies.

2. CONCEPTS
A web page is often made up of multiple objects. Among the objects in a page, there is one primary object corresponding to the URL of the page. This object is generally an HTML file (or .asp files etc.) which contains a number of URLs specifying some other objects needed by the page. We call this primary object *Container Object (CO)* and other objects the *Embedded Objects (EO)* of the page. In general, the retrieval process of a web page starts with the request for the CO. The reply data will be streamed to client in a sequence of *data chunks*. Only when a data chunk contains the definitions of some EOs are returned, will the retrieval processes for those EOs be *triggered*. On the other hand, current web system employs the mechanism of *parallelism* to fetch objects in a page. The default parallelism width in most common web browsers is four. The parallelism in web retrieval makes it possible for the retrieval of some objects to virtually have no effect on the final page latency because of the overlapping of multiple retrieval processes.

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

3. WEB RETRIEVAL DEPENDENCY MODEL (WRDM)
The basic idea of our *Web Retrieval Dependency Model (WRDM)* is to map the operations involved in web retrieval process and the relationship among them into a directed graph. We symbolize each operation by a vertex, and capture the relationship between two operations by an arc connecting them. Each arc carries a weight which represents the time spent in completing the operation represented by the target vertex. The resulting graph is called *Web Retrieval Dependency Graph (WRDG)*. Currently, we define six types of vertices representing six operations in our model:

(1) Request initiation operation *r*: submission of an object request
(2) Location resolution operation *l*: location resolution for the server address specified in the URL of a request
(3) Network connection operation *c*: establishment of network connection between client and web server
(4) Request sending operation *s*: sending out the request message of an object request from client to server
(5) Data chunk transfer operation *d*: transfer of a chunk of data from server to client. Note that there may be multiple occurrences of this operation in one object retrieval
(6) Ending operation *e*: releasing of resources (such as network connection) occupied by a request

Figure 1(a) gives an example WRDG graph demonstrating the retrieval process of an object. The retrieval latency of the object is represented by the distance of the path from the vertex *r* to the vertex *e*. Note that the distance of a path is the sum of the weights of the arcs along the path, not the number of the arcs. The whole object retrieval latency can be divided into five components: *Location Resolution Time (LRT)*, *Connection Time (CT)*, *Request Sending Time (RST)*, *Chunk Sequence Time (CST)*, and *Ending Time (ET)*. These five latency components are also shown in Figure 1(a). Note that the number of the latency components depends on the number of the types of vertices defined in WRDM model. Although we define six types of vertices for the WRDM model here, the model can easily be altered to include more or less types of vertices to cater for the needs in different situations.

Multiple individual object WRDG graphs can be connected together to capture the retrieval process of a page. Because the EOs of a page

are defined in the CO, so the retrieval processes of EOs depend on the retrieval process of CO. In WRDG graph, we capture this dependency between CO and EOs by using an arc to connect a data chunk vertex d_i of the CO to the request initiation vertex *r* of an EO, where the data chunk represented by d_i contains the definition of the EO. Furthermore, current web system uses certain parallelism width *N* to fetch multiple of objects of a page simultaneously. We capture this requirement by limiting the width of the WRDG graph to be not wider than *N*.

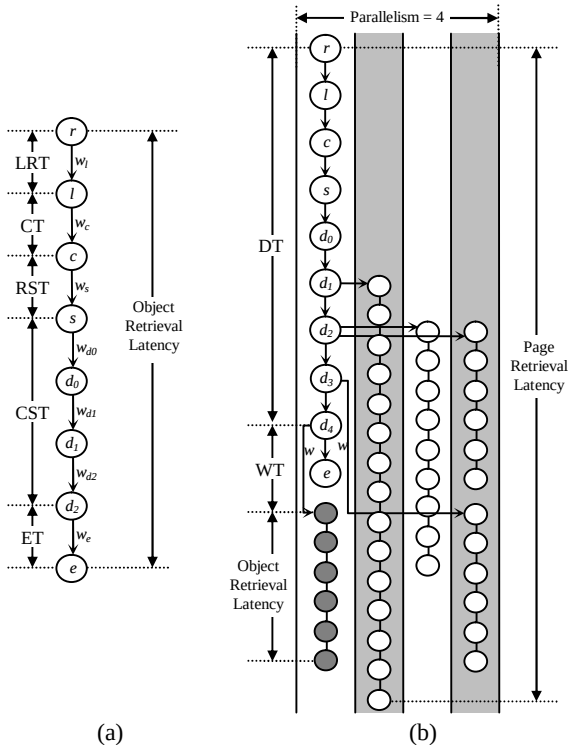

Figure 1. WRDG graphs for object and page retrieval.

Figure 1(b) gives an example WRDG graph showing the retrieval process of a page with five EOs. From it, we see that EOs will undergo two more latency components when objects are put together to form a page. The first latency component is called the *Definition Time (DT)* of EOs. The retrieval process of an EO can not be initiated until the CO's data chunk containing the definition of that EO has reached the client. However, user's perceived retrieval latency is counted from the time when he/she submits the page request, so the DT times of EOs should be considered as part of the EOs' total latency. This latency component can significantly postpone the finishing points of EOs' retrieval, which would in turn affect whole page latency. On the other hand, as most common web browsers use a limited parallelism width for the parallel retrieval of objects, so some EOs of a page may be held in waiting state due to the unavailability of parallelism when the number of objects contained in the page is larger than the parallelism width. The time spent by an object in waiting state is referred to as *Waiting Time (WT)*. Again, this latency should also be considered as part of the EOs' total latency, and it could also have important impact on the finishing points of EOs' retrieval and whole page latency. Due to space limitation, we only show the DT and WT times for one object in Figure 1(b). In the WRDG graph for a page retrieval, whole page retrieval latency is defined by distance of the longest-distance path from the vertex *r* of the CO to the last finished vertex *e* in the graph.

4. RESULTS AND ANALYSIS

We would like to investigate the importance of the two new components in page retrieval revealed by our WRDM model. Our experiments rely on very detailed information about web retrieval such as the number of objects in a page and the definition points of EOs etc. Such information is not available in existing traces. So we conducted real retrieval process for a large number of pages to obtain traces with detailed operation and chunk level information. The detailed traces are fed into various simulators to obtain the final results.

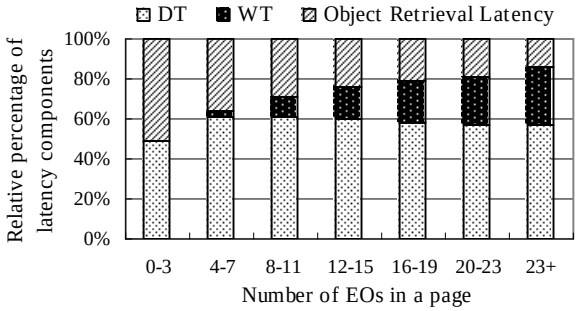

Figure 2. Relative distribution of latency components.

Figure 2 plots the relative distribution of the DT time, WT time and the actual object retrieval latency against the number of EOs in a page under the parallelism width of four. It is surprising to find that a great percentage of retrieval latency of objects in pages comes from DT, rather than the actual object retrieval latency which is often thought of as the dominating factors for page retrieval latency. DT often takes up more than 50% of the whole object latency. WT is also a major latency component when the number of EOs in a page is greater than 3. As the number of EOs per page increases, WT grows quickly and becomes even bigger than the actual object retrieval latency. As to the actual object retrieval latency, its relative percentage drops dramatically from about 50% to only 14%. From these results, we see that when objects are put together to form pages, the DT time and WT time, which are particularly found in pages, will become the dominating factors in determining page retrieval latency. Ignoring them in studies may cause inaccurate results. These findings also suggest a new way to web acceleration which is to improve DT time and WT time in web retrieval.

5. CONCLUSION

We propose a fine grained operation-level web retrieval dependency model for studies of web retrieval performance. By providing precise capture of web retrieval process at very detailed level, our model reveals two new important latency components in web retrieval. The DT time caused by dependency between objects and the WT time caused by limited parallelism contribute even more greatly to page retrieval latency than the actual object retrieval latency does. Taking proper consideration of these factors is very essential to studies in the web retrieval area.

6. REFERENCES

[1] Ruddle, A., Allison, C., Lindsay, P., Analysing the latency of WWW applications, Proceedings of the IEEE ICCCN, Phoenix, AZ, Oct., 2001.

[2] Habib, M.A., Abrams, M., Analysis of sources of latency in downloading web pages, Proceedings of WebNet 2000, San Antonio, Texas, November 2000.