Reducing Traffic of Messages in Distributed Editing Systems*

Huaihai Hui[1, 2], Gongqian Liang[1], Jiancheng Ma[1], Jianxv Yang[2]
[1]College of Management,
Northwestern Polytechnical University, Xi'an, P.R. China, 710072
[2]College of Computer Science,
Zhejiang University, Hangzhou, P. R. China, 310027
E-mail: pekingperson@yahoo.com.cn, yangjianxv@yahoo.com.cn

Abstract

With the embedded mobile devices greatly proliferating, the communications under ubiquitous environment are welcomed in terms of collaborative work. For the limited bandwidth of wireless network and the limited size of screen on palmtop computer, tailored communication and data exchanges between desktop and mobile sites was studied in our research. We proposed a grouping scheme together with Sending on Demand Policy to reduce traffic of messages in distributed collaborative editing systems implemented by various types of devices. Consistency maintenance relevant to grouping was studied out with a Linked History Buffer. The scheme had been implemented in our prototype system that can work well in heterogeneous environments.

1. Introduction

Most of the existing collaborative graphics editors [1] based on desktop only deal only with elementary operations such as create object, delete object, move object, change object color. Claudia-Lavinia Ignat's work based on desktop deals with grouping and ungrouping operations in a real-time environment where the intention of the users involved in the concurrent editing are preserved [2]. Although consistency has been widely studied in interactive groupware systems and a variety of types of algorithms or schemes were designed to maintain consistency [3], grouping and corresponding consistency in the collaborative graphics editors based on the combination of desktop and palmtop computers has not been researched.

Research on collaborative editing system on mobile became a focus in recent years. Raymond K. Wong [4] studied collaborative hypertext editing in mobile environment. M Suleiman el al [5] researched on concurrent operations in mobile environment. Li Yushun et al [6] and Joseph S et al [7] focused more on the whole mobile environment

There are also plenty of studies on the communication under wireless network environment [8, 9, 10, 11]. Upkar Varshney and Ron Vetteras studied on the relationship between mobile and wireless networks [12]. B. Zenel and D. Duchamp studied on the intelligent communication filtering for limited bandwidth environments [13]. The communication and exchanges between desktop and mobile sites was studied in our research, for the limited bandwidth of wireless network and the limited size of screen on palmtop computer.

For the bandwidth of wireless network is limited, messages generated on desktop which sent to mobile ends would be reduced as many as possible. We will partly adopt Sending on Demand Policy to reduce the traffic of message flow. For the limited size of screen on palmtop, we should provide user with less objects on showing. Because the objects in our system was organized in a tree structure, the objects in document could collapse to a

* This paper is supported by National Natural Science Foundation of China (60573176)

backbone as simple as possible by taking the advantage of the existing of the structure tree, with each non-leaf node representing a group, each leaf node representing an object.

The structure of the rest of this paper is as follows: Section 2 gives an introduction of the definition of grouping, and depicts the mapping relationship between single nodes and group node. Section 3 proposes the sending on demand scheme which is basic of traffic reducing, while the inconsistency problem in Undo/Redo coming with the proposed scheme is discussed. Section 4 presents the test results of the effectiveness to the scheme. Section 5 presents comparison to Related Work. Finally, Section 6 concludes the paper.

2. Grouping

The objects in our system was organized in a tree structure, a single object was represented by a leaf node, while a group was represented by a non-leaf node. Each object has a unique ID number, as shown in Fig.1.

Fig. 1. Document structure tree

2.1 The Definition

While a group node on desktop, named as Desktop Node (DN), or a group node on mobile site, named as Mobile Node (MN), was transmitted to the mobile sites; the group, composing of a number of objects, could be interpreted to a mere group on mobile, that is, the objects in the group would collapse to a node on the screen of mobile in term of the small screen.

To illustrate, there is a graphics composing of four flowers with each one taking one pistil and seven petals. It is possibility for system to implement a collapse on a group node like “petals” or “Flower X”, as shown in Fig.2.

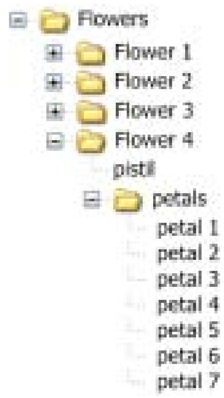

Fig. 2. Flowers structure tree (a)

Suppose we deploy every non-leaf node, leaving each object disappeared in the structure tree, we would get a 100% deploy-degree or 0% collapse-degree.

$$\text{degree}_{\text{collapse}} = 1 - \text{degree}_{\text{deploy}} = \frac{\text{collapsed objects}}{\text{total objects}}$$

Suppose we collapse every single petal to its father node “petals”, deploy every child node of root, leaving every pistil disappeared, we would get a 12.5 deploy-degree or 87.5% collapse-degree. The structure tree is shown in Fig.3.

2.2 Mapping Relation

The mapping relation between a DN and the corresponding MN is interpreted as following when the document on desktop was mapped into the mobile.

Fig. 3. Flowers structure tree (b)

To illustrate the mapping:

[desktop] Group (id_1, id_2, id_3, ...)
-> [mobile] idg_1.
[desktop] Group (id_1, id_2, Group (id_3, id_4, ...), ...)
-> [mobile] Group (id_1, id_2, idg_2, ...)
-> [mobile] idg_1.

Where,

id indicates single object ID;
idg indicates group ID.

Given a document structure tree on the desktop site:

$$T_{desk} \text{ (Group } (id_1, id_2, id_3, ...), id_4, id_5, ...)$$

Then we can get the mobile-end mapping result:

$$T_{mobile} \text{ } (idg_1, id_4, id_5, ...)$$

Where,

T is structure tree.

Given a document structure tree on desktop:

$$T_{desk} \text{ (Group } (id_1, id_2, ... , \text{Group } (id_3, id_4, ...)), \text{ Group } (id_5, id_6, ...), id7, ...)$$

The mobile-end mapping result:

$$T_{mobile} \text{ } (idg_1, idg_2, id_7, ...)$$

Any MN can be opened by user through clicking. After a MN is opened, user can view the details of the group, including the sub-groups and sub-objects within the MN.

After the mobile user opened the group idg1, the structure tree changed to:

$$T_{mobile} \text{ (Group } (id_1, id_2, ... , idg_3), idg_2, id_7)$$

After the user opened the group idg3 farther, the structure tree changed to:

$$T_{mobile} \text{ (Group } (id_1, id_2, ... , \text{Group } (id_3, id_4, ...)), idg_2, id_7)$$

2.3 Location Mapping

Since there are a number of objects in a DN, while these objects collapsed to a group, where is the position of the group and where is the position of MN?

We calculate the center of each object on desktop respectively by the following formula:

$$x_j^c = \frac{\int_s xds}{s}, \quad y_j^c = \frac{\int_s yds}{s}$$

Where,

x_j^c, y_j^c are central x-coordinate and y-coordinate of object j;

s is the area of object j;

ds is the area of differential object;

x, y are coordinates of differential object.

The position of DN is calculated as the following formula:

$$Xc = \frac{\sum_{j=1}^{n} s_j x_j^c}{\sum_{j=1}^{n} s_j}, \quad Yc = \frac{\sum_{j=1}^{n} s_j y_j^c}{\sum_{j=1}^{n} s_j}$$

Where,

Xc, Yc are central coordinates of DN;

sj is the area of object j;

n is the number of objects in group.

Then the position of DN will be mapped to the position of MN.

3. Sending on Demand Scheme

We discuss here how to send the modification to the mobile site when object within group is modified by an operation on desktop.

As we known, the wireless network is unstable some times; farther more, network congestion may happen if the number of concurrent users is large enough. Therefore, the bandwidth cost, to a degree, usually needs to be considered in the development of wireless network application. When a group has not been opened by the user, there is no immediate need to see the details inside that group collapsed into a node.

Therefore, we would not send the modification inside the group to the mobile site immediately.

Since all operation would be recorded in the local history buffer in local site, any “not sending” may make confusion of operation order in the local history buffer.

3.1 Policy

Suppose the policy “Sending on Demand” is adopted, that is, any editing operation inside a group on desktop will not be transmitted to mobile site until needed. Whenever the mobile user opens a group node to view the detailed sub-groups and sub-objects on his user interface, the operations in the group, related to graphics object, image object and text object, are needed on demand. These operations are pulled from corresponding document on desktop to the mobile end immediately.

3.2 Inconsistency in Undo/Redo

On arriving at the mobile end, operations get to the end of the local history buffer, may lead to the history buffer out of order.

To illustrate, operations (Op2, Op3), say, move object and change object color respectively, are operations inside a group. As shown in Fig.4.a, Op2 and Op3 stayed between Op1 and Op4. Op2 and Op3 were not sent immediately, because they were inside group and where were no demands that let them sent. Therefore, Op4 was sent consequently. At the moment, user would like to open the group node to have a look at the details inside the group. Demand to get details was transferred to the desktop end, and then Op2 and Op3 were pulled to the mobile end, as shown in Fig.4.b. Sending on Demand Policy leads to an obvious inconsistency between the two history buffers.

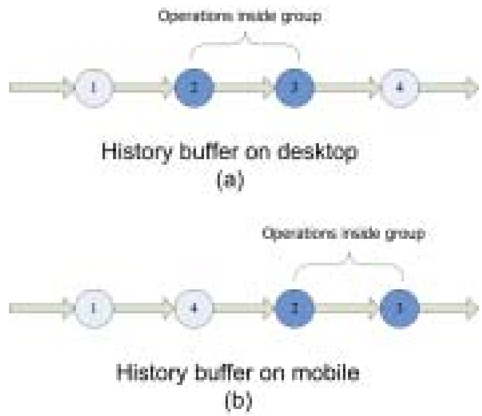

Fig. 4. Example of inconsistency between history buffers

Although the inconsistency in history buffer will not bring about confusion to our eyes, for there are no essential conflictions among disordered nodes in history buffer, either inside of group or outside, there are some other problems existing any way.

In the case that there were Undo/Redo operations in the editing system, what happened if user on mobile did an Undo operation? If we hold the old history buffer and the old undo policy, the Undo operation could not work any more, for the end node of the history buffer might not the last operation in logic.

To solve the problem, we changed the history buffer to a linked one. The head of the linked list was the end node in logic, the next of the head was the one before the end node in logic. The rest may be deduced by analogy and a linked history buffer was built as shown in Fig.5. After a user on mobile did an Undo action, editing system found the head of the list; did a reverse operation to implement the Undo effect. The editing system on desktop finished the same work then. The next node of the head became the new head of the list. Vice versa, user on desktop did an Undo action, and then the mobile followed.

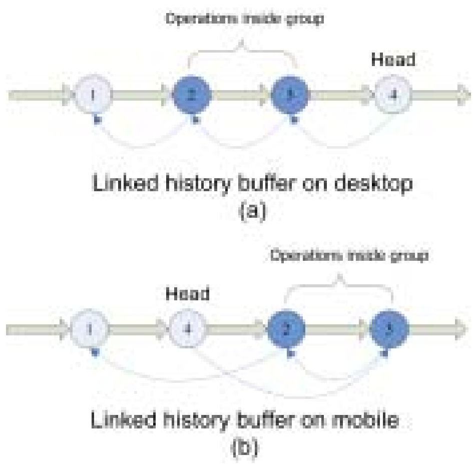

Fig. 5. Linked history buffer

4. Effectiveness

To test the effectiveness of the algorithm, an experiment was carried out. Ten volunteers (5 male, 5 female) from local University were invited to our lab. The experimenters were divided into 5 groups (2 persons for each group). For each couple, one person operated a document on desktop PC and the other operated the same document on Microsoft Pocket PC emulator. The desktop used in our experiment was Dell PC running our application system, using a 17-inch monitor set to 1024x768 resolution, 512M memory and 2.4G CPU.

While each couple was operating on PCs or the emulators, they grouped the objects in the document piece by piece. At the very beginning, there are not any groups in the shared workspace; with the action "grouping" operated by users, more and more objects were collapsed into groups. An applet running on PC recorded the network flow between the couple. Fig.6 shows the testing results.

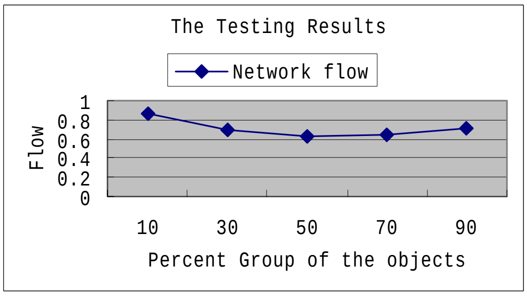

Fig. 6. The testing results

X-axis represents the percent of objects which have been collapsed into group. Before they started the experiment, null objects have been grouped. If all of objects have been grouped, 100 percent collapse-degree is gotten.

Y-axis is the mean network flow between each couple. We recorded the value of largest network flow as 1, the other value would compare with the highest value, getting a percent of flow value. The highest value was gotten at the process of the experiment.

We could find that the flow value changed with the collapse-degree (percent group of objects). From the beginning, with more and more objects were grouped, the flow value got down accordingly. It got down to the lowest valley when the collapse-degree is about 60%. Then it counter got up to about 70~80% at the 90~100 percent collapse-degree.

Generally speaking, sending on demand based on grouping could help reduce the network value, but highest percent group may not get the best performance on wireless network.

5. Comparison to Related Work

Some work has been done in reducing the network flow in wire and wireless environments. Grouping and ungrouping proposed by Claudia-Lavinia Ignat received its attention in CSCW'04. However, grouping has not been used for saving bandwidth of wireless network.

In our work, the communication and exchanges between desktop and mobile sites was studied in our research, for the limited bandwidth of wireless network and the limited size of screen on handheld computer. Sending on Demand Policy together with grouping was proposed to save bandwidth of network in heterogeneous environments. Experiments showed that the proposed scheme got an accepted performance in the real-time collaborative communication.

6. Conclusion

Ubiquitous computing is rapidly developing, especially in small portable devices, such as handheld computers, smart phones and PDAs. However, the limitation of screen size, bandwidth of network, CPU speed and memory makes it impossible for embedded mobile devices to communicate with desktop by traditional methods [14].

We proposed a grouping scheme together with Sending on Demand Policy to save bandwidth of network in heterogeneous collaborative editing systems. A Linked History Buffer is adopted to solve consistency problem caused by grouping. The grouping scheme has been realized in our prototype system that seemed to be able to improve the flexibility and usability of mobile embedded devices.

7. References

[1] Bo Jiang, Chun Chen, Jiajun Bu: CoDesign-A collaborative pattern design system based on agent. Proceedings of the Sixth International Conference on Computer Supported Cooperative Work in Design, Canada, (2001) 319-323.

[2] Claudia-Lavinia Ignat and Moira C. Norrie, Grouping in Collaborative Graphical Editors, Proc. of the CSCW'04, November 6-10, 2004, Chicago, Illinois, USA.

[3] C. Sun and D. Chen: Consistency Maintenance in Real-Time Collaborative Graphics Editing Systems. ACM Transactions on Computer-Human Interaction. (2002) 1-41.

[4] Raymond K. Wong, Collaborative Hypertext Editing in Mobile Environment, Proceedings of the 10th International Multimedia Modeling Conference (MMM'04).

[5] M Suleiman, M Cart, J Ferrie, Concurrent Operations in a Distributed and Mobile Collaborative Environment, Proceedings of the Fourteenth International Conference on Data Engineering, Pages: 36 – 45.

[6] Li Yushun, Shi Meilin, Mobile Collaboration Writing, The 14th IEEE 2003 International Symposium on Personal, Indoor and Mobile Radio Communication Proceedings.

[7] Joseph S. Valacich, Clayton A. Looney, Mobile Technologies and Collaboration, Proceedings of the 38th Hawaii International Conference on System Sciences – 2005.

[8] Mirhakkak M., Schult N., Thomson D., Dynamic bandwidth management and adaptive applications for avariable bandwidth wireless environment, Selected Areas in Communications, IEEE, Volume: 19, Issue: 10, page(s): 1984-1997.

[9] M. Stemm and R. Katz, Vertical handoffs in wireless overlay networks, ACM Mobile Networks Applicat., vol. 3, no. 4, pp. 335-350, 1998.

[10] C. Lin and J. Liu, "QoS routing in ad-hoc wireless networks," IEEE J. Select. Areas Commun., vol. 17, pp. 1426-1438, Aug. 1999.

[11] Bender, P. Black, P., et al., CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users, Communications Magazine, IEEE Volume: 38, Issue: 7, page(s): 70-77.

[12] Upkar Varshney, Ron Vetter, Emerging mobile and wireless networks, Communications of the ACM, Volume 43, Issue 6 (June 2000), Pages: 73 – 81.

[13] B. Zenel, D. Duchamp, Intelligent communication filtering for limited bandwidth environments, Fifth Workshop on Hot Topics in Operating Systems (HotOS-V), p. 28.

[14] Allan Meng Krebs and Ivan Marsic: Adaptive Applications for Ubiquitous Collaboration in Mobile Environments. Proceedings of the 37th Hawaii International Conference on System Sciences.