CloudBooks: An Infrastructure for Reading on Multiple Devices

Jennifer Pearson and George Buchanan

FIT Lab, Swansea University
{j.pearson,g.r.buchanan}@swan.ac.uk

Abstract. The use of light, portable devices such as iPads whose reading angle is readily changed is radically different to reading on a desktop or laptop. However, it would be naive to view this as mere evolution. Rather, such devices permit reading activity to more closely mirror paper. A light, keyboardless device can be used in many different locations and orientations. This paper reports an infrastructure for supporting reading on multiple slate devices using a single cloud-based system to provide for numerous configurations.

Keywords: Slate PCs, Collaboration, Digital Reading, Annotation.

1 Introduction

The central role of slate PC devices such as the iPad is "media consumption", including the reading of electronic books. Superficially, this simply transfers existing software from the desktop (or laptop) PC onto a new form factor. Apps such as GoodReader and iAnnotate primarily replicate the interaction design found in Adobe Acrobat and other desktop document reading applications. In this paper, we introduce a basic infrastructure, CloudBooks, that supports multiple reading devices (primarily, slate PCs), which may be connected in co-located or remote, synchronous or asynchronous, single- or multi-user configurations.

By placing key features in a network-based 'cloud' infrastructure, the CloudBooks services become ubiquitous: e.g. mark-up can be transferred and communicated quickly between multiple devices. Previous systems such as Polar [4] and DiLAS [1] have demonstrated the viability of providing an extensive annotation function which established the value of providing an external service that integrates with a central DL server. There is, in short, an established corpus for separating annotations from the document itself, using some form of continuous annotation service, that serves as a repository for the notes applied to documents by the library's users. CloudBooks builds on these existing principles and adapts them to a user's own, informal collection rather than a DL.

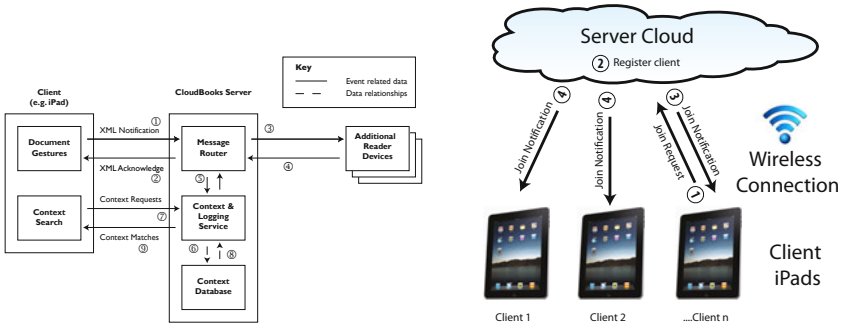
2 CloudBooks Architecture

The general CloudBooks Architecture is shown in Fig 1a. On the left appears a single iPad (or slate) device; further devices are depicted on the right. In the centre appears the CloudBooks server, and its associated components.

S. Gradmann et al. (Eds.): TPDL 2011, LNCS 6966, pp. 488–492, 2011.

To connect to the server, an iPad must first register with the message router component using its IP address as an identifier. The message router is later used to forward messages received from one device to another. Each registered device belongs to one (or more) groups, and a message is normally forwarded through the message router to all members of the group. To take a concrete example, when a device wishes to send an event to its group, it first forwards an XML message to the router ①; the router sends a response to confirm its receipt of the message ② and immediately forwards the message to the other members of the group ③, which also send acknowledgements of receipt ④. Optionally, messages can be saved to the log service ⑤ and its database ⑥.

The communication support provided by the message router provides no long-term storage, and simply provides the same messaging capability as the Greenstone Alerting Service [3]. CloudBooks can store contextual data using its logging components (centre, middle and bottom). This can be achieved through the log service (steps ⑤ and ⑥). This service can also be used directly ⑦; either to retrieve content or history (Fig 1a), or for logging. When contextual information from the log service is needed, a request can be sent ⑦, requesting ⑥ and retrieving ⑧ content that is then returned to the client ⑨.



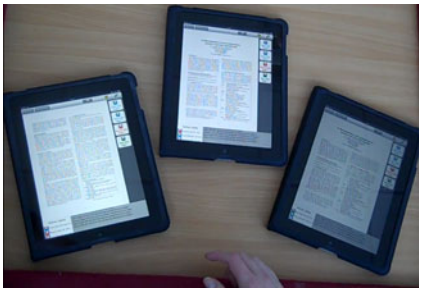
(a) The CloudBooks Architecture **(b)** The gsBooks Architecture

Fig. 1. CloudBooks and gsBooks Architectures

3 Reading in a Group: Multiple Devices, Multiple Users

The act of reading is typically solitary, e.g. reading a newspaper. However, there are occasions where reading is performed interactively in groups, e.g. in study groups, where there are several ways of reading a document: single paper copy, multiple paper copies, single computer, multiple computers, single large display. Each method has its own strengths and weaknesses [2]. Paper is easily manipulated and does little to impede interaction between people; readers can sit wherever there is space without technological constraints. A single paper copy of a document (or indeed a single computer) can cause space and 'group leader' problems whereas multiple paper copies can be tricky to reference.

(a) gsBooks Interface



(b) gsBooks Collaboration

Fig. 2. gsBooks

To better support collaborative reading, we have implemented a subset of CloudBooks called gsBooks; a system that makes use of multiple iPads to facilitate real-time communication between multiple users. The device's form factor allows users to sit around the same desk while also giving them a personal document view. The iPad's WiFi allows users to contribute easily via real-time changes to all iPads in the group. This not only facilitates simultaneous annotation between group members, but also raises user awareness of the notes made within the session. It supports both local and remote collaboration via our cloud–based infrastructure.

The gsBooks architecture is shown in Fig 1b. To connect to a group, a client iPad sends an XML join request ①. CloudBooks then registers the user ②: allocating a unique colour[1] to the client, and adds the client IP address and nickname to the active client list. Following a successful join, the server then distributes a join confirmation back to the client ③ and then to all other clients within the group ④. When an action (see below) is made by a client, it sends a new XML 'action' to CloudBooks. The server updates its own list of the annotations on the document before distributing the change to all active clients. By keeping its own complete copy of the annotations, CloudBooks can supply new or returning clients with a complete history of the document.

The gsBooks system supports three main document activities: annotation, bookmarking which are permanent user notes, and 'point outs' which are temporary markers that coordinate the reading of group members. When reading in a group, users need to indicate content to other members, e.g."look at this figure on the right of page 45". When working on a single document this process is straightforward: the user can physically point to the section. This problem is more difficult when each user has their own copy of the text, particularly if there are many users or if group members are sitting far apart. To aid in this process then, we have implemented the 'look at this queue' (shown in Fig 2a on the left of the screen): a tool that allows users to quickly point out specific sections of the

[1] For consistency, users that disconnect from the working group, then later re-connect will be assigned their original colour.

working document. When a point out is made, a new entry which includes the nickname and colour of its creator, is added to the top of every group member's 'look at this queue'. Clicking on this entry then takes the user directly to the page and points out the exact area the point out was made.

4 Reading Alone: Multiple Devices, One User

Cloudbooks can also capture reading behaviours over a significant span of time, so a user's reading history is enriched by recording details of what was read on different devices or locations. We have developed another subset of CloudBooks, called xBooks, that captures details of a reader's history (e.g. when a document was opened) which provides a unified log of the user's reading behaviour. If such a log is maintained on a single device, then under most circumstances that information cannot be retrieved remotely. However, a variety of options are made available when using CloudBooks. One configuration stores a log of the reading on each copy of xBooks within the CloudBooks service. A user can then query what they read, or when they read it. Adding contextual information such as GPS (when available) can permit location to be inferred from that data. Hence, one can (at least approximately) retrieve what one read at a given place.

The first prototype for single users provides the ability to distribute reading histories across multiple devices, building on the basic configuration just described. This prototype mirrors the functionality available of the Kindle - i.e., reading history across *all* reading devices are synced. However, unlike the functionality provided by Amazon, we support both local and global histories. A user can search against their history using the location, device, time or title.

The second prototype uses CloudBooks to control the view presented on multiple iPads via a single document reader interface. The iPad views can thus be used as a supplementary space for the main document reader, without the user having to control its display directly through their individual displays. This localised "remote control" allows for a user to, ultimately, place reading displays around their environment (as an academic may do in their office) and read from multiple sources as they compose a document. The user controls the secondary displays through a widget that then passes a message through CloudBooks to the paired reader devices. The copy of xBooks running on the reader then adjusts its display (e.g. turning a page) in response to the received message.

5 Conclusions

This paper presents CloudBooks, an architecture for reading from multiple slate PC devices. We have described systems for reading with multiple users on multiple devices in a collaborative environment, and provided support for single users allowing multiple displays to be co-ordinated through one document reader interface. The current data-model of Cloudbooks encapsulates support for location-aware and context-aware computing, allowing users to query *where* they read a text, or, conversely, what they did read in a particular time or place.

Acknowledgements. We would like to acknowledge Microsoft Research and EPSRC grant EP/F041217.

References

1. Agosti, M., Albrechtsen, H., Ferro, N., Frommholz, I., Hansen, P., Orio, N., Panizzi, E., Pejtersen, A., Thiel, U.: Dilas: a digital library annotation service. In: IWAC, pp. 91–101. CNRS (2005)
2. Amershi, S., Morris, M.: Cosearch: a system for co-located collaborative web search. In: CHI 2008, pp. 1647–1656. ACM, New York (2008)
3. Buchanan, G., Hinze, A.: A generic alerting service for digital libraries. In: JCDL 2005, pp. 131–140. ACM, New York (2005)
4. Frommholz, I., Fuhr, N.: Probabilistic, object-oriented logics for annotation-based retrieval in digital libraries. In: JCDL 2006, pp. 55–64. ACM, New York (2006)