A Framework for Long-Term Archiving of Pervasive Device Information

Reza Rawassizadeh
Vienna University of Technology, Information and Software Engineering Group
Favoritenstraße 9-11/188, A/1040 Vienna, Austria
+43 (1) 588 01 - 18840
rrawassizadeh@acm.org

Amin Anjomshoaa
Vienna University of Technology, Information and Software Engineering Group
Favoritenstraße 9-11/188, A/1040 Vienna, Austria
+43 (1) 588 01 - 18854
anjomshoaa@ifs.tuwien.ac.at

Martin Tomitsch
The University of Sydney
2006 NSW, Australia
Faculty of Architecture, Design and Planning
+61 (2) 9351 - 3030
martin.tomitsch@sydney.edu.au

ABSTRACT

In this paper we describe an approach for preserving personal information, stored on pervasive devices, using a reliable platform. Nowadays, pervasive devices such as mobile phones or personal multimedia players, provide software and hardware capabilities that allow individuals to read and create digital contents. Therefore it is important to provide the ability to archive these contents for later retrieval and use on other platforms. To address this need we introduce an archiving framework and present a working prototype based on this framework. The prototype runs on the Android operating system and enables individuals to archive their pervasive device information for long-term preservation.

Categories and Subject Descriptors

H3.2. Information Storage and Retrieval: Information Storage.

General Terms

Algorithms, Design, Experimentation.

Keywords

Digital preservation; pervasive computing; personal archive; personal information.

1. INTRODUCTION

In the last decade mobile phones have progressed beyond their initial purpose of being portable telephones. Today they feature advanced computing capabilities, which allows individuals to carry a computer with them around-the-clock. Furthermore, people are using a variety of mobile devices, such as personal multimedia players or personal digital assistants. The ubiquity and computing capabilities of these pervasive devices make them capable to be a container of important personal information; however such pervasive devices are more vulnerable to loss or damage than desktop computers [9]. The basic reason for this is the fact that the portable devices are always carried with their owners which increases the likelihood to lose the device or its sensitive data. Personal information created and stored on pervasive devices includes contact information, audio recordings, digital images, meeting notes, etc. It is therefore crucial to provide the capability for archiving personal information stored on these devices.

Furthermore the data created via a pervasive device might have a specific data structure that cannot be readable by other pervasive devices which makes the information portability very difficult. In this context there is a growing need to store and preserve the digital user information on pervasive devices in a sustainable way.

Digital preservation is the act of maintaining information in a correct and independently understandable form for a long-term period of time.

Typically most of the information which are stored on pervasive devices are represented in an appropriate format for long-term preservations. However, in some cases, information such as binary data, are only accessible through specific applications. Therefore, we would need a mechanism to archive this type of information in an application and device independent format, which makes it accessible in the long-term.

In this paper we introduce a prototype application based on the Android platform [1] which enables users to archive their pervasive device data by changing the data schema and mapping it to a uniform schema which is appropriate for long-term preservation. We use the term "pervasive device" as described above to cover all relevant devices; however, the focus of this paper is on mobile phones, because currently they are the most common type of pervasive devices.

The next section discusses related work. This is followed by some design considerations that build the foundation for the proposed solution. The remainder of the paper presents and discusses the architecture of the framework and our prototype implementation based on the proposed framework. At the end we conclude this paper.

2. DESIGN CONSIDERATIONS

We identified a number of design considerations that need to be addressed by tools for digital preservation of pervasive device information. These considerations should be taken into the account in our proposed framework and thus constitute the foundation for the implementation of our prototype.

2.1 A Backup Software is not enough

There is a variety of backup applications for different pervasive devices (mostly for mobile phones), which enable users to backup their information stored on the device. In our belief the available backup solutions have some restrictions in the context of long-term preservation. For example, to our knowledge these applications are not considering the long-term preservation of the information objects where the major challenge for digital preservation in terms of software is the data format obsolescence.

Furthermore, the backup action in most of such applications needs to be performed manually, which puts the device information on risk. Some of these backup applications store data on the software provider's server which might be against the user's privacy policies, since the backed up data may include some personal information that are highly sensitive. Users might intend to keep such data only on their local storage and not to disclose it on the net.

Another problem is that backup applications mostly focus on backing up general information on a device such as calendar contents and contact lists. Mobile (smart) phones are increasingly becoming a portable computer, which implies that the device can host other applications that produce valuable information. For instance, a mobile phone may host a health monitoring application, which produces information that users want to archive for a long-term period of time. This need is currently not addressed by standard backup applications since they do not provide backup solutions for third party applications.

2.2 User Intervention

According to valence instrumentality expectancy theory [12], performing administration tasks is not desirable by users. Users prefer to use devices with more capabilities and features [11]. Usually they are not keen to perform additional administration tasks unless they are motivated enough to do them. In this framework we try to reduce the user intervention as much as possible. When a user installs the application he/she should be able to specify some archiving parameters and there after everything should run in the background eliminating the requirement to perform any additional administration tasks. It is therefore desirable to run the archiving application 24/7 in the background without any need for user intervention during the archiving process.

2.3 Application Performance

Pervasive devices by their very nature are not designed to host large computing applications. Client thickness [10] is one of the major challenges in designing applications for pervasive devices. The archiving application should always run in the background and take care of the archiving process of newly created content. On the other hand due to the resource restriction of mobile devices, the background backup application should not consume too many resources. Additionally it should not affect the performance of applications running in the foreground.

2.4 Long-term Preservation of the Information

The main goal of this research is to archive pervasive device information in a way that is accessible for a long period of time. Gordon Bell [3] noted that personal information is worth to be kept at least during the owner's life. To solve this problem an archiving application should check the file format of the binary data. In case the data is not in a long-term archive-able file format, the archiving application should send the binary file to an appropriate migration service based on the source file format. For instance, if the information is stored in a 3GP file then it would choose the "3GP to MP4" conversion service. In this example MP4 is assumed to be a long-term archive-able format, while 3GP is not. More about this feature will be discussed in the framework architecture section.

3. RELATED WORK

To our knowledge there is no archiving solution for pervasive devices available that considers the long-term preservation of personal information. Currently digital preservation is mainly being investigated for digital libraries and news media [7]. Only few projects exist that address the preservation of personal information.

MyLifeBits [6], which has been proposed by Microsoft, is the first scientific effort toward archiving personal information. MyLifeBits is a life-log application, which archives users' activates via a desktop application and a ubiquitous camera. A Life-log tool produces a dataset, which contains valuable information. Their information is worth to preserve at least during the user's life. Therefore, long-term preservation is a necessary requirement in life-log tools.

iMemex [4] is another research work, which provides a desktop-based PIM (Personal Information Management) system. iMemex introduces a data model, which does not require any specific application for browsing data and data is stored in long-term accessible formats such as XML.

4. FRAMEWORK ARCHITECTURE

In order to implement a prototype of the described framework we use the Android SDK 2.0 [1] emulator on a Mac Book Pro with 2.4 GHz CPU. Android is an open source and well-known operating system for pervasive devices. It provides useful features such as network connectivity, optimized virtual machine for mobile devices, etc.

Figure 1. shows the current architecture of the proposed framework. The architecture was inspired by similar efforts in context-aware applications such ContextPhone [8] and MyExperience [5]. Context-aware applications gather information from different sensors and aggregate them based on their usage and functionalities. Here we substitute sensors with resources and refine the acquired information based on the digital preservation requirements.

Most pervasive devices (especially smart phones) are able to connect to a network which allows users to connect their devices to a larger and more reliable storage platform. Further, connecting a pervasive device to an external resource can also extend the limited computing capabilities of mobile phones and let them calling external Web Service that are running on powerful computers or clouds.

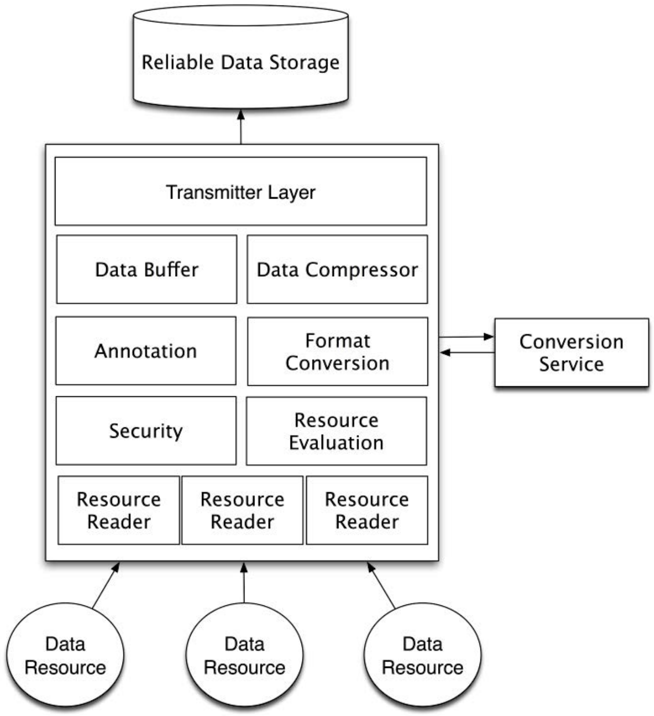

Figure 1. Architecture of the Framework

In the proposed framework the "Resource Reader" will read the information, which is worth to be archived for a long-term period of time. The "Data Resource" node, is the source of information, e.g. an audio file of a call content or a contact in a contact list are considered as data resources. In our approach each resource type is assigned to a specific resource reader. Moreover the resource readers work independent from each other. As a result in case a resource failed and the associated resource reader cannot read information from it, this will not affect the functionality of other resources. Another advantage of using multiple resource readers is that it facilitates the framework extension (by adding a new resource reader for new resource). This means when a new resource has been discovered which contains valuable information to archive, an appropriate resource reader can be plugged into this framework in order to handle that data resource.

This framework relies on the network connection of the device. It gathers information from different data resources annotates them and sends them to a "Reliable Storage Media" (RSM). The RSM might be the user's personal computer or an online server. The sending process is executed via the "Transmitter Layer".

The "Security" component is used for securing the connection between the device and the RSM. For instance, if the security option is enabled, HTTPS will be used instead of HTTP. On the other hand we propose a log in mechanism that creates an access key for the authenticated device and each time an HTTP post request will be sent to the server the access key and user id of the request will be checked. If the request is authenticated the file can get uploaded to the conversion service. After the file format has been changed, the conversion service checks if the client has access key or not. If it has access key then the service sends back the new file to the client and removes the old one from the server storage.

We introduce the "Data Buffer" for temporary storage of information. Based on the pervasive device capabilities the network connection between the pervasive device and the RSM is not always available, hence, a data buffer will be used to keep information objects temporarily on the device. When the connection becomes available, the f5ramework will hand the data over to the Transmitter Layer to send it to the RSM. A basic drawback of pervasive devices is the limited processing and storage capabilities. In order to reduce disk space utilization, information objects that reside within the Data Buffer will be compressed via the "Data Compressor". As compression technique our prototype creates a zip file from the source files.

Further, we consider the annotation of the personal information objects via an annotation component. Annotating personal information is one of the main challenges in using lifetime personal information stores [3]. Metadata maintains information about a binary file in order to ease browsing and information retrieval. In the current implementation of the prototype annotation is not addressed and we are planning to implement it in the next implementation phase.

In order to overcome the limited processing power of mobile devices, the framework sends large processes, which are hard for a mobile device to perform, to an external service. This could be achieved via using Web Services. For instance we may use Web Services to perform file format conversion for JPEG images and convert them to TIFF format which is a long-term archive-able data format for image files.

The archive unit in this framework is file. In other words, and information object which is represented as a file will be sent to the RSM, format of the file will be changed to a long-term archive-able format, and other steps will be followed. In the implemented prototype, the file conversion is assumed to be a resource intensive process, therefore we use an external service to convert the file format. "Conversion Service" could be a web service, which gets the file and converts it to an archive-able format, and then return it to the application. Based on the file type there could be different conversion services. "Format Conversion" contains information about the conversion services and enables the framework to choose the appropriate service.

One drawback of the current model is the networking issue. Sending a file to an external service or receiving a file from an external service consumes lots of network bandwidth. Although the bandwidth is an expensive resource in pervasive devices, but the alternative solution (i.e. local processing of files) is very process intensive for mobile phones.

After the file gets converted to the desired format, "Resource Evaluation" will be used to evaluate if the file conversion has been successful or not. Some evaluation methods might be resource intensive as well, e.g. if Resource Evaluation checks only some properties of the received file this can be done locally and there is no need for external service invocation. In the implemented prototype Resource Evaluation checks the file format and based on the file format, it reports, whether the conversion has been successful or not.

In the described framework we tried to explain the core components of the framework and do not consider the details of implementation such as GUI, Application Log, etc. Figure 2 shows some of the GUI screens of the prototype.

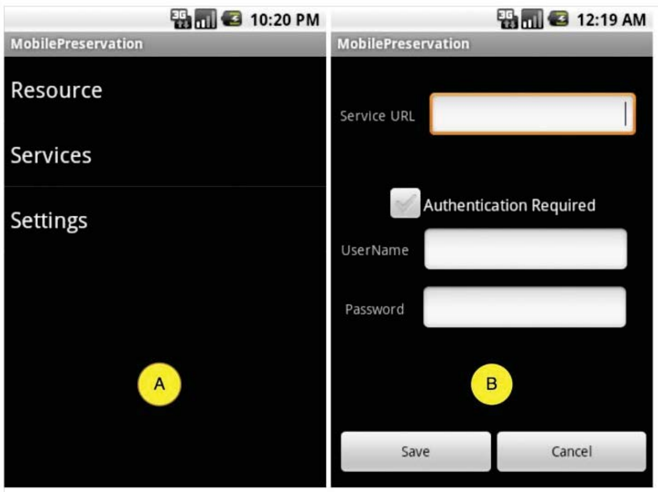

Figure 2. Frame A depicts the main UI of implemented framework and Frame B shows the Web Service connection page

5. PROTOTYPE IMPLEMENTATION

To evaluate this framework, some parts of the framework have been implemented as a prototype. Resource readers for Images, Calls and SMS have been implemented. SMS and Calls will be annotated with name of the senders or receivers. Images will be annotated with time and location. Each Data Resource can have one associated annotation class.

Now, the most challenging part is the service discovery mechanism and how to substitute a service when it is not available. This task is not in the scope of current research but this remains as an important requirement to make the prototype implementation an operational tool. In order to overcome this problem temporarily, we have used our local services for file conversion but there must be reliable service providers to handle conversion services.

The RSM is implemented as Java Servlet, which enables the framework to upload archive-able information via HTTP protocol. We use HTTP Client APIs of the Android for establishing HTTP connection and sending messages. The prototype currently uses the Internet connection of the device to send files, but in the future we intend to manipulate the Transmitter Layer to add support for Bluetooth connections. This means, whenever a user is located in the Bluetooth coverage area of his RSM, data will be sent automatically to the RSM.

6. CONCLUSION

In this paper we introduced our ongoing research work toward establishing a software model for digital preservation of pervasive personal information. . The proposed framework employs external Web Services to archive pervasive device information for long-term preservation. A prototype has been implemented based on his architecture, which users can use it for archiving the pervasive device information in long-term. This framework aims to provide a tool for archiving the valuable contents for long-term access. These contents have been generated or stored on the pervasive device. There are many projects focusing on the long-term preservation of digital documents in libraries, news media and related institutions, but to our knowledge only few researches has been done on long-term archiving of personal information. One important container of personal information is the mobile phone or in a more general term pervasive devices. Thus the contribution of this paper is to propose a framework which employ external Web Services to archive pervasive device information in long-term.

REFERENCES

1. Android Developers Home Page. http://developer.android.com/sdk.
2. Bell, G. A Personal Digital Store. Comm. ACM, 44,1 (2001), 86-91.
3. Bell, G., Gemmell, J. and Lueder, R. Challenges in using lifetime personal information stores. In SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (2004), 1-1..
4. Dittrich J.P, iMemex: A Platform for personal dataspace management. In SIGIR PIM 2nd Invitational Workshop for Personal Information Management. (2006)
5. Froehlich, J., Chen, M.Y., Consolvo, S., Harrison, B. and Landay, J. A.. MyExperience: a system for in situ tracing and capturing of user feedback on mobile phones. In MobiSys: The Annual International Conference on Mobile Systems (2007), 57–70.
6. Gemmell J., Bell G., Lueder R., Drucker S., and Wong C. MyLifeBits: fulfilling the Memex vision. In MULTIMEDIA '02: Proceedings of the tenth ACM international conference on Multimedia. (2002), 235-238.
7. Jones W., Teevan J., Personal Information Management. Seattle: University of Washington Press (2007), 59-60
8. Raento, M., Oulasvirta A., Petit, R., and Toivonen, H. ContextPhone: A Prototyping Platform for Context-Aware Mobile Applications. IEEE Pervasive Computing, 4, 2 (2005), 51-59.
9. Satyanarayana, M. Fundamental challenges in mobile computing. In PODC '96: Proceedings of the fifteenth annual ACM symposium on principles of distributed computing, (1996), 1-7.
10. Satyanaraynana, M. Pervasive computing: Vision and challenges. IEEE Personal Communications. 8,4, (2001), 10-17
11. Thompson, D. V., Hamilton, R. W., & Rust, R. T. Feature Fatigue: When Product Capabilities Become Too Much of a Good Thing. JMR, Journal of Marketing Research, 42,4, (2005) 431–442
12. Vroom V.H., Work and Motivation. Jossey Bass (1994)