Volume 6, Issue 2
May 2013
ISSN: 1946-1836

JOURNAL OF INFORMATION SYSTEMS APPLIED RESEARCH

In this issue:

The **Journal of Information Systems Applied Research** (JISAR) is a double-blind peer-reviewed academic journal published by **EDSIG,** the Education Special Interest Group of AITP, the Association of Information Technology Professionals (Chicago, Illinois). Publishing frequency is currently quarterly. The first date of publication is December 1, 2008.

JISAR is published online (http://jisar.org) in connection with CONISAR, the Conference on Information Systems Applied Research, which is also double-blind peer reviewed. Our sister publication, the Proceedings of CONISAR, features all papers, panels, workshops, and presentations from the conference. (http://conisar.org)

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the JISAR journal. Currently the target acceptance rate for the journal is about 45%.

Questions should be addressed to the editor at editor@jisar.org or the publisher at publisher@jisar.org.

JOURNAL OF INFORMATION SYSTEMS APPLIED RESEARCH

Editors

JISAR Editorial Board

Creating an Audio Conferencing Application on Android Smart Phones

Jui Sun
Computer Science and Information Systems

Ron Vetter
vetterr@uncw.edu
Department of Computer Science

Bryan Reinicke
reinickeb@uncw.edu
Information Systems and Operations Management

University of North Carolina Wilmington
Wilmington, NC 28403, USA

Abstract

This paper describes an approach to building an audio conferencing application for Android smart phones. As the need for audio conferencing systems grows and smart phone market penetration increases, the smart phone becomes a viable platform for developing conferencing applications. We have implemented a centralized audio conferencing model and a client application which was deployed on Android-based smart phones. Experiments for battery consumption and packet delay were carried out to evaluate the usability of the application. The smart phones were not affected by the application under low traffic conditions; however, the application did consume twice as much battery life under heavy traffic conditions. The results for delay testing showed that increasing the number of participants also resulted in longer average packet delays. Throughout the development process, problems involving software/hardware diversification and audio signal processing were uncovered and potential solutions were proposed. The paper provides valuable information for developing VOIP applications on smart phones, specifically on the Android platform, and can help to direct future development of audio conferencing systems.

Keywords: Android Development, Mobile Development, VOIP, client-server architecture

1. INTRODUCTION

Voice over IP (VOIP) audio conferencing systems are increasingly becoming an important application on the Internet (Freese, 2005). VOIP introduces a possible low cost solution for long distance multi-people communication problems (Jaiswal and Raghav, 2004). As the need for voice conferencing systems continues to grow, these systems are being applied to many areas of business, as well as in academic and social circles (Gilson and Xia, 2007). VOIP systems are gaining more acceptance as the software and the quality of service of the network environment improves (Park, 2010). A highly attractive scenario combines VOIP with the expanding use of smart phones (ComScore, 2012), and allows users to participate in a

conference meeting, without having to physically be present or incurring charges for the minutes used on their cell phones.

A smart phone is a portable handheld device with the capability of a personal computer and traditional cell phone rolled into one. Smart phones are now technically capable of delivering sufficient performance for rich multimedia applications and audio communication; therefore deploying a high quality VOIP conferencing system in smart phones is now possible. Deploying a VOIP audio conferencing system in smart phones provides a new opportunity for making life more convenient for people all over the world. Although there are many products available in the marketplace, only a few of these products provide an audio conferencing service on smart phones. The lack of hardware and software resources on many older cell phone models is the primary reason for the limited availability of high quality audio conferencing systems on mobile phones.

The purpose of this paper is to explore how a simple and extensible audio conferencing system for smart phones can be designed and implemented. The paper includes all of the fundamental components of how to construct an audio conferencing system for Android-based mobile phones. In addition, two experiments were designed to examine the usability of the system. The experiments examined battery/energy use and measured application quality of service via delay testing.

The rest of the paper is organized as follows. Section 2 introduces related work and the design principles for a smart phone based audio conferencing systems. The overall methodology and system architecture is discussed in section 3. Section 4 discusses the experimental design, and section 5 discusses the results and lessons learned. Finally, section 6 provides conclusions, and discusses future work that could improve audio conferencing systems on smart phones.

2. BACKGROUND AND RELATED WORK

Voice over IP (VOIP) was first introduced in 1991 when Speak Freely developed internet-based telephony software for the personal computer (Tech-Pro, 2012). In 1996, the ITU Telecommunication Standardization Sector (ITU-T) defined the first version of the H.323 standard (International Telecommunications Union, 1996). Because the Internet was a bandwidth constrained environment, few companies invested in the VOIP industry. In 2001, Yahoo Japan integrated the public switched telephone network (PSTN) and VOIP services, thereby providing a communication link between traditional telephone service and the Internet. In 2003, Skype was released and proved the reliability and quality of VoIP services in the marketplace, which convinced users of the capability and possibility of internet telephony (Jia, 2008).

VOIP uses two types of Internet protocols in order to achieve end-to-end communication functionality: Signaling Control Protocol and Media Transport Protocol. Signaling Control Protocol, or Call Signaling Protocol, is used to establish and manage building and terminating connections between users. This protocol regulates the approach of searching for the correct target user, building connections, and processing data based on each user's processing capabilities. SIP (Session Initiation Protocol), H.323, and MGCP (Media Gateway Control Protocol) are instances of a Signaling Control Protocol. The Media Transport Protocol (e.g. RTP and RTSP) is used to facilitate the transfer of digitalized media data after the connection is built (Jia, 2008). In addition, management protocols and other types of support protocols are also used in VOIP applications.

The Session Initiation Protocol (SIP) is an ASCII-based, application-layer control protocol that can be used to establish, maintain, and terminate calls between endpoints (CISCO, 2012) using HTTP and SMTP concepts. It transfers users' information by text, such as IP address, ports, media ability, and codec type. The message is in plaintext; hence the receiver can realize the sender's message without decoding it (Jia, 2008). SIP allows call information to be carried across networks, and provides the ability to manage connections between users.

In general, a SIP application should possess the following capabilities (CISCO, 2012):

- Name translation and user location.
- Feature negotiation.
- Establish a session between the originating and target end point.
- Handle the transfer and termination of calls.

Once the connection is established, the software implements other protocols in order to achieve the desired functionality.

User Agent Clients and Servers

A peer in a session is called the user agent. From a functionality standpoint, a user agent can be classified as either user agent client (UAC) or user agent server (UAS). A UAC, or Caller, initiates the request. A UAS, or Callee, receives the request and returns the user's information. A SIP's endpoint is typically able to act as either a UAC or UAS (Jia, 2008). From an architecture standpoint, SIP is composed of two components: clients and a server. The clients includes phone and gateway, and based on different responsibilities, the server can be a proxy server, a redirect server, a register server, a location server, a media server, a media delay server, and a Back-to-Back user agent (Jia, 2008; CISCO, 2012).

Figure 1 in Appendix A introduces a simple direct call peer-to-peer SIP model. It establishes a session without any proxy server. In this case, John wants to call Mary. John's machine is a UAC and Mary's machine is a UAS. John's machine calls the target via Universal Resource Identifier (URI). The machine then sends an "INVITE" plaintext to Mary's machine (UAS). Mary's machine returns messages appropriately ("100 Trying" and "100 Ringing"). After John's machine sends an ACK back to Mary's, the two machines transfer data through RTP/RTCP protocol. If any user agent knows other SIP device's IP address or domain name, it can process a SIP direct call.

Codecs

A Codec is the method used to encode and decode a digital stream or signal, and there are several types in widespread use (Isnardi, Fielder, Goldman and Todd, 2006). One of the first things that needed to be determined is which Codec should be used to encode and transmit voice data. In general, Codecs can be defined as lossless or lossy (Wikipedia, 2012a; Wikipedia, 2012b). Lossless codecs try to maintain the original audio information, while lossy codecs trade some information to achieve other requirements. There are a number of different Codecs available which can provide toll quality speech under real-time transmission (Light, 2006), which can be seen in table 1 in Appendix A. All of these were available prior to the release of Android OS version 2.3. It should be noted that while, in many respects, codecs for speech present a simpler signal than other audio codecs (Kroon, 1995), this does not necessarily make them simple.

Audio Conferencing

Audio conferencing software in the marketplace commonly uses the client-server architecture. Most server products run as a dedicated server, rather than as peer-to-peer. The TeamSpeak product allows users to install the server on their own machine. The service provider provides a location server for IP and DNS lookup. Raidcall manages servers by itself, but provides user client software. The user does not need to know detailed information, such as the server address. In addition, it extends its capability with social networking. It brings entertainment elements into a classic audio conferencing system.

According to the connection approach, conferences can be grouped as "Centralized Conferencing" and "Distributed Conferencing" (Jia, 2008). Centralized conferencing (Figure 2, Appendix A) require a focus server. The focus server connects with clients independently, and upon receiving data from one client, it delivers the information to the remaining clients.

Energy Management

One of the most critical issues in smart phone application design is the management of energy consumption. Smart phones integrate the functionality of computer and mobile phone into one device; however, whereas a personal computer requires a continuous energy supply, a smart phone relies on its battery. If an application is a burden on the phone's battery, it decreases the time for voice calling.

While a VOIP system communicates through a Wi-Fi network, the associate interface is active. Energy is consumed even if no data is transferred. When an application is running, program size, algorithms, and other programming factors influence battery consumption. As a software developer, it is impossible to increase the battery size on a mobile device, thus requiring the battery to be managed via software. To address this problem, Agarwal, Chandra et al. presented a wakeup mechanism to solve the waste of energy by system idling (Agarwal, Chandra, Wolman, Bahl, Chin, Gupta, 2007) while Naeem et al proposed an adaptive algorithm to switch codecs based on remaining battery life (Naeem, Namboodiri, Pensi, 2010).

Generally speaking, longer operating hours represents higher usability. Because audio conferencing requires that the software be continually active, an experiment was designed to determine how long the developed software can be run.

3. METHODS AND SYSTEMS ARCHITECTURE

The VOIP application was developed using the Eclipse IDE and included the development of two software components.

The first component was the user client, which was deployed on several Android based smart phones (see table 2 in Appendix A). The program acts as a caller, or UAC. It established the connection by SIP and can:

- Send the SIP request.
- Establish and maintain a connection to the server.
- Send and receive the audio stream.
- Terminate the connection with server.

In practice, the project uses SIP stack's interface and classes under the Android system. The program does not permit any incoming SIP request and, as a result, cannot act a UAS.

The second component was the focus server. A focus server is the central node of the conferencing network. All clients transfer the audio streams through this central node. The server has the following capabilities:

- Receive and respond to SIP requests.
- Establish and maintain the connection between clients and itself.
- Manage participants.
- Clarify the incoming audio stream.
- Send an audio stream to the correct endpoints.
- Receive the client termination request, and disconnect clients.

We did not implement the focus server on a smart phone due to energy consumption concerns. Rather, the focus server was deployed on a personal computer running in a Java environment.

System Architecture

This system implemented a two-tier client-server architecture (shown in Figure 3, Appendix A). Clients communicate with the server through the Internet and the server's IP address is the intended location of every client.

The server includes the following modules:

- ChatServer: the server's main module which initiates the SIPEngine module and ChatHandler module.
- SIPEngine: listens to client calls. Once it receives a client SIP call, it creates a specific SIPListener instances for the client and waits for the next SIP call.
- SIPListener: handles SIP massages with a specific client target. The listener will close if it receives a BYE message from its related client. This module also adds the client to or removes the client from the member list.
- Member: a class which stores all conference participants' information.
- ChatHandler: receives audio data from the connected clients list and determines which target it should forward to.

The client includes the following modules:

- AConPortableMain: this generates the graphical user interface (GUI). This GUI asks users for the server's information and then creates a SIPEngineClient instance for further actions.
- SIPEngineClient: sends a SIP call, terminating request, and interacts with all other SIP events. Once it builds the session between itself and the server, it then connects to the server's audio port by calling UDPSocket.
- UDPSocket: handles audio data transferring. It includes the method to communicate audio streams with the server.
- ChatHandler: the part of the GUI which allows users to talk to the server. The user can turn on and off the talking threads. The model initiates InComePacket and OutComePacket and creates threads, respectively. Additional functionalities include volume adjustments and the method called from the SIPEngineClient module to leave the conference room.
- InComePacket: a listener which listens to the incoming packets. Once it receives a packet it will push that data into the buffer to await play back.
- OutComePacket: a thread class which reads the data from the microphone's buffer and sends the data to the server.

From a model standpoint, the system implements a centralized conferencing model,

which is comprised of a server and all clients (participants). Figure 4 (Appendix A) introduces the model and the possible message flow. The numbers label the outgoing flow and possible incoming flow for every smart phone.

System Operating Mechanism

The system uses a simplified SIP message to establish, manage, and terminate sessions. Figures 5, 6, and 7 (Appendix A) introduce the actual mechanism for joining and leaving the conference system.

Figure 5 presents a user client which intends to participate in the system. In this case, no other participants are currently in the system. Client A sends an INVITE request to the server. The server checks the register information, and sends 200 OK back to the client. After the client sends ACK to the Server, these two endpoints can start transferring audio streams.

Figure 6 gives an instance of another user who wants to attend the conference. It sends the same request, and the server sends the same response back to build the connection. Once the connection is built, it can transfer the audio stream between the client and server.

Figure 7 shows an example of terminating a connection. The Client user first sends a BYE message. Once the server receives the message, it sends back an ACK message and closes the connection. It also manages the member list and notifies conference members about the leaving client's message.

Figure 8 shows the architecture of the system's modules. The SIP message is transferred through the TCP port, and the server creates different SIPListener objects for every connected client. Audio streams are transferred through the UDP port. One object of the server's ChatHandler is created to handle all audio stream traffic. The Server's ChatHandler uses a First-in-First-out (FIFO) algorithm to forward incoming packets. Figure 8 also shows how the modules associate with one another.

4. EXPERIMENTAL DESIGN

In order to test the application two experiments were carried out. One to test the application's ability to transmit data and the time lag associated with this transmission and a second to test the impact of the application on the mobile devices' batteries.

Table 2 and Figure 4 illustrate the clients and server for the experiment. The "Device" column in Table 2 distinguishes the different smart phones, which are labeled correspondingly in figure 3. This representation will be used when describing both the experiments and the results.

For the experiment we used the same audio settings in both power consumption and delay testing. The client program read 1024 Bytes from the microphone's buffer and sent it to the server. The audio data was configured as follows:

- Audio format: PCM 16 bit
- Channel configuration: Mono
- Sample rate: 8000 Hertz

Power Consumption

In order to evaluate the usability of the system, an experiment evaluating power consumption was designed. Measurements were taken to determine how long it took the system to decrease the smart phones' battery life from 90% to 85%. This increment was chosen as a sample as a matter of practicality to meet time constraints. In order to make every experiment more consistent, smart phone screens were turned on while testing.

Experiments included three conditions:

- Without system running: measure the power consumption duration without running the audio conferencing system.
- Without data transferring: connect the client program to the server without audio data transferring.
- Heavy data transferring: connect to the server and keep transferring data during measurements. Speakers and microphones are turned on for all devices.

Smart phones B, C, D, and E from Table 2 were used as experimental devices. Each smart phone measure was taken three times in the three different conditions. Smart phone A was not used in these tests, as it was the primary phone for one of the authors.

Delay Testing

A conferencing system can be classified as a real-time multimedia system, and as such the perceived quality of the system depends in part

on audience perception of audio delay. An audio event with a huge delay would result in low system accessibility and usability. In order to test the system's performance, we carried out a delay testing experiment to determine the delay time for a specific audio packet.

In the system implementation, an analog signal is captured by the smart phone's microphone. The resulting audio data is stored in a buffer and the system waits for the application to read the data. The program reads a specific amount of audio data from the buffer (in this case 1024 bytes), places the data in a packet, and sends the packet to the server. The server forwards the packet to the target smart phone. Once the target smart phone receives the data packet, it extracts the data from the buffer and writes it to the audio track. The digital signal is then converted to analog sound and played through the phone's speaker.

The experiment measured the elapsed time of the packet between two events: the data read from the buffer of the microphone and the same data being received by the server. A specific 5 byte header was added to every packet for testing, including 1 byte for the device number and 4 bytes for a time stamp value. The elapsed time was calculated as the current system time minus the time stamp's value.

Single vs. Multiple Streams

All tests were measured under two different environments: single source audio stream and multi-sources audio streams. The single source audio stream transfers only the tested subject's audio stream. In contrast, all devices try to transfer audio data at the same time in a multi-sources audio stream. In general, once the server receives a data packet, it forwards the packet to every participating client, excluding the packet sender. The delay testing experiment sent every packet to the tested smart phone, whether it was the packet's owner or not. In addition, the tested subject was last in order of the server's forwarding targets.

Smart phone A was the test subject in this experiment. Smart phones B, C, D, and E were participants in the testing network. For the single source audio stream network, 100 and 1000 packets were collected and measured, respectively. For the multi-source audio stream network, 100 packets were collected and measured. Each condition was tested 5 times.

5. RESULTS AND LESSONS LEARNED

Two separate tests were conducted to determine the impact of the system on battery life and latency of the audio traffic over a network. These results are presented separately, followed by the lessons learned in the process of building the system.

Battery Consumption

Figure 9 in Appendix A provides a graphical representation of the results from the battery consumption test. As the figure shows, when the application is running, but not transferring data, there is very little impact on battery performance. This is not unexpected, as the client application only maintains a TCP connection with the server, and an open UDP port for incoming packets.

Under conditions of heavy data, there is a noticeable impact on battery life. Based on the results shown in figure 9, the application requires roughly half the time to reduce the available battery life from 90% to 85%. Based on this, we can project that a phone running this application would drain it's battery it a little less than 5.3 hours. This assumes constant traffic levels, and could obviously vary based on other factors.

Delay Testing

The purpose of this experiment was to test the system's multi-user processing capability. The test measured the additional delay when a new participant joins the conference room, and was run under three conditions.

The first condition was with a single audio stream and a sample size of 100 packets for the audio stream. As expected, the delay time to add a participant grew as the number of conference call participants grew, though the average was still very low. One other factor illustrated by this test was that network traffic at different times appears to have an impact on the delays. As this was not part of the experiment, data was not collected on this factor, though it would likely explain the variation in the results. Results from this test are shown in figure 10 in Appendix A.

In the second condition, the sample size was increased from 100 to 1,000 packets of data. Once the sample size was increased to 1,000

packets, the delay time grows more than twenty times the 100 packet cases. This significant change is caused by system data processing speed. In the system, both the client and server program implemented FIFO as the audio data processing policy. If the packet receiver's reading speed is too slow, more and more incoming data will remain in the buffer. Once again, the congestion on the network itself likely played a role in the test results. The results from this test can be seen in figure 11 in Appendix A.

The final test condition was with a sample size of 100 packets, but with multiple sources of the audio stream, rather than a single source. Under these conditions, the average delay ballooned to over 2.5 seconds, and went as high as 3.9 seconds. This delay would be noticeable to participants in the audio conference. Complete results are shown in figure 12.

A design defect caused a reading speed problem, particularly in the final test condition. The program is designed to receive a packet and write that data into the audio track. These two events occur sequentially, so in order to receive a new packet, the thread has to wait for the program to call the write() method to write the buffer to the track. Calling this method causes additional overhead and decreases the speed of reading packets from the socket. The design of the program should call these two methods in different threads. By implementing two threads, receiving data and writing tracks become two independent events which should increase the speed of reading data.

Lessons Learned

One of the primary issues with the development of this system revolved around issues with the Android OS itself. To delve into this area, some explanation is required.

One of the primary difficulties in the development for the client application came from implementing the SIP functionality. These could have been resolved by using either the library provided in the android.net sip package, or the library provided in the android.net.rtp package. Using these libraries would have eased development, as this would have negated the need to design the low level audio I/O.

The reason these were not implemented is that they require different API levels (9 and 12 respectively). Different API levels are not available to every version of Android. The API levels available to different versions of Android are presented in table 3 in Appendix A. One of the goals for this project was to implement a system that would be available on multiple versions of Android, which required working at a significantly lower API level. This is because the majority of Android devices do not, and cannot, run the latest version of the Android OS. In fact, when the discovery of the libraries was made, an attempt was made to upgrade the OS on the devices used in this experiment. However, this was not possible – the devices would not support newer versions of the operating system.

There were also problems resulting from using open source libraries for the server and client software developed. Specifically, some of the open source projects are not well documented, which makes them difficult to implement, especially when interoperability is required. A great deal of the open source software was created in C/C++, which requires additional time to research how this native code operates, and how it must be embedded using the Android NDK toolset.

Another problem was hardware diversification. This project used multiple devices to reflect the fact that there are multiple hardware vendors that produce Android handsets. However, each hardware vender has different settings for their audio devices. Also, the minimum buffer size required for the relative Android audio record object (android.media.AudioRecord) may differ. Even if the configurations are all the same, the sound quality on different smart phones varies. Because of this, a usable configuration for all devices is very limited. Testing the audio parameters on different devices is required, and therefore increases the amount of effort spent developing and testing code.

On the server side, the biggest hurdle was audio mixing for multiple streams. If a server does not have any mechanism for audio mixing, then the audio stream cannot be sent concurrently. For instance, assume three different audio stream sends from different clients, with each of the streams comprised of five packets (A_1~A_5, B_1~B_5, and C_1~C_5). Each packet includes audio data which can be played for n milliseconds. The server then determines client D to be these three streams final destination. In this case, client D will receive 5

×3 =15 packets. As a result, client D needs at least 15n milliseconds to play all the data it received. If each stream was sent from its source concurrently, then in theory, client D should only need 5n milliseconds to play the received audio data.

As more and more clients participate in transferring audio streams, this phenomenon grows more severe. From the Delay Testing, we saw that if multi-source packets are transferred concurrently, the client program will have a significant delay. If the server can mix different packets and reduce the overall number of sending packets, the delay can be considerably improved. Careful design for audio mixing is necessary to improve overall system performance.

6. CONCLUSIONS AND FUTURE WORK

This project undertook the development of an audio conferencing solution based on a server and a client program built on the Android platform. While there could certainly be demand for applications such as this, the development of such a system was more difficult than anticipated. When developing audio solutions for multiple streams, it is necessary to implement a server side solution for audio mixing. Also, when implementing an Android application the developer must make a choice. Either they can develop an application that will work on the majority of Android devices by working at a low API level, or they can simplify the problem by working at a higher level API. One problem of course is that to develop for a higher level API, the developer must have access to devices capable of running more recent versions of Android, and must accept that if they wish to sell their application, they will be selling to a more limited market.

A possible future study would also involve looking more carefully at the impact of other network traffic on VOIP calls. While there has certainly been research done on this area (Vaiapury, Nagarajan, & Jain, 2009; Mahani, Kavian, Naderi, & Rashvand, 2011), it was not the intent of this study to examine this in particular. It would likely be worth testing the impact of delays on VOIP over smart phones to see if there are quality differences, or if network congestion impacts any of the other measured factors mentioned in this study.

While there are challenges to developing this type of application, we believe that the demand for this type of application will continue to expand. As the use of smart phones becomes more and more widespread, and users become more comfortable with the idea of using VOIP, there will be increased pressure on the development community to develop this type of solution.

7. REFERENCES

Agarwal, Y., Chandra, R., Wolman, A., Bahl, P., Chin, K., & Gupta, R. (2007). Wireless wakeups revisited: energy management for voip over wi-fi smartphones, *Proceedings of the 5th international conference on Mobile systems, applications and services*, June 11-13, San Juan, Puerto Rico.

Android Developers, Android API Levels, Retrieved June 18, 2012, from http://developer.android.com/guide/appendix/api-levels.html#level12.

CISCO, Overview of SIP. Retrieved June 18, 2012, from http://www.cisco.com/en/US/docs/ios/12_3/sip/configuration/guide/chapter0.html.

comScore (2012). comScore Reports July 2012 U.S. Mobile Subscriber Market Share. Retrieved September 14, 2012 from http://www.comscore.com/Press_Events/Press_Releases/2012/9/comScore_Reports_July_2012_U.S._Mobile_Subscriber_Market_Share

Freese, D.D. (2005). Voice over Internet Protocol. *Saturday Evening Post*, (277:1), 52-55.

Gilson, C.L., Xia, R. (2007). Spanning the Pacific Ocean through Voice-over Internet Protocol Chat with the Hadley School for the Blind. *Journal of Visual Impairment and Blindness*, (101:4), 232-236.

International Telecommunication Union (1996). Visual telephone systems and equipment for local area networks which provide a non-guaranteed quality of service. Recommendation H.323 (11/96). Retrieved September 14, 2012 from http://www.itu.int/rec/T-REC-H.323/.

Isnardi, M.A, Fielder, L.D., Goldman, M.S., Todd, C.C. (2006). ATSC Video and Audio Coding. *Proceedings of the IEEE,* (94:1), 60-76.

Jaiswal, M.P., Raghav, B. (2004). Cost-Quality based consumer perception analysis of voice over internet protocol (VoIP) in India. *Internet Research*, (14:1), 95-102.

Jia, W.K. (2008). Session-Initiation Protocol Methodology Handbook Second Edition, Kings Information, Taipei, TW.

Kroon, P. (1995). Evaluation of speech coders, in Speech Coding and Synthesis. Elsevier Science, Amsterdam.

Light, J. (2006, June). Performance Analysis of Audio Codecs over Real-Time Transmission Protocol (RTP) for Voice Services over Internet Protocol. Computing & Informatics, 2006. ICOCI '06. International Conference, pages 1-8, IEEE.

Mahani, A., Kavian, Y.S., Naderi, M., Rashvand, H.F. (2011). Heavy-tail and voice over internet protocol traffic: queuing analysis for performance evaluation. *IET Communications,* (5:18), 2736-2743.

Naeem, M., Namboodiri, V., & Pendse, R., (2010). Energy implication of various VOIP codecs in portable devices, Local Computer Networks (LCN), 2010 IEEE 35th Conference on, pages 196-199, Oct. 2010.

Park, Namkee (2010). Adoption and Use of Computer-Based Voice Over Internet Protocol Phone Service: Toward and Integrated Model. *Journal of Communications*, (60:1), 40-72.

Tech-Pro.net, 2012 Introduction to Voice over IP (VOIP), Retrieved June 18, 2012, from http://www.tech-pro.net/voice-over-ip.html.

Vaiapury, K., Nagarajan, M., Malmurugan, J., Kumar, S. (2009). Ambience-based voice over internet protocol quality testing model. *IETE Journal of Research*, (55:5), 212-217.

Wikipedia (2012a), Codec, Retrieved June 18, 2012, from http://en.wikipedia.org/wiki/Codec.

Wikipedia (2012b), Audio Codec, Retrieved June 18, 2012, from http://en.wikipedia.org/wiki/Audio_codec.

Editor's Note:

This paper was selected for inclusion in the journal as a CONISAR 2012 Distinguished Paper. The acceptance rate is typically 7% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2012.

Appendix A: Figures and Tables

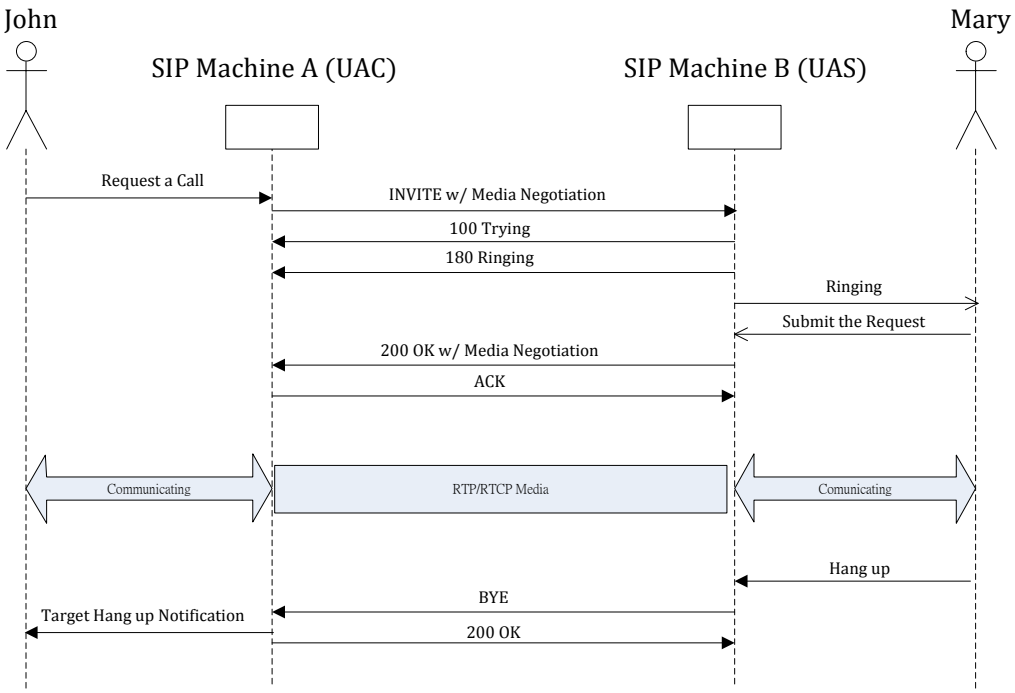

Figure 1: SIP Direct Call Model.

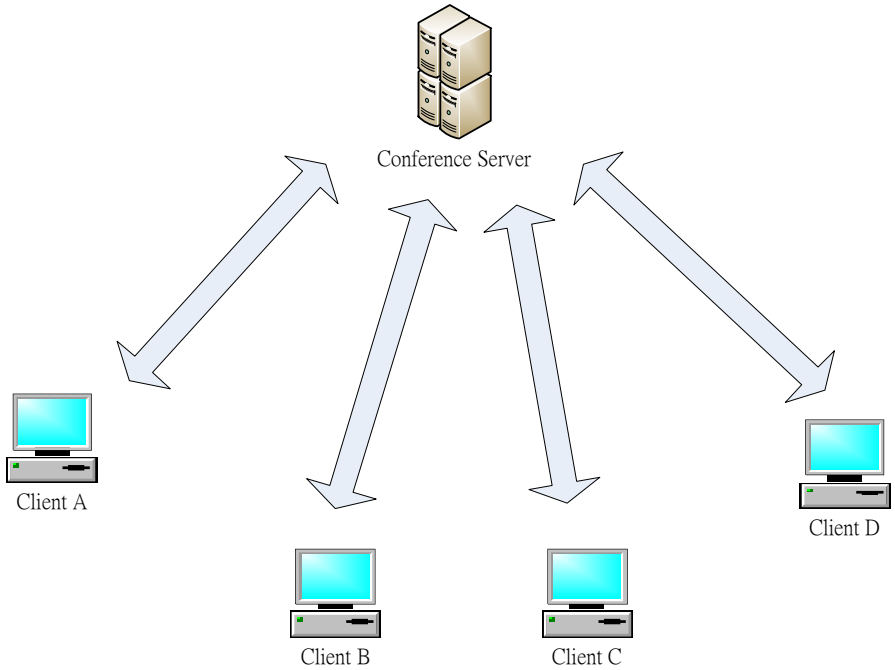

Figure 2: Centralized Conferencing Model

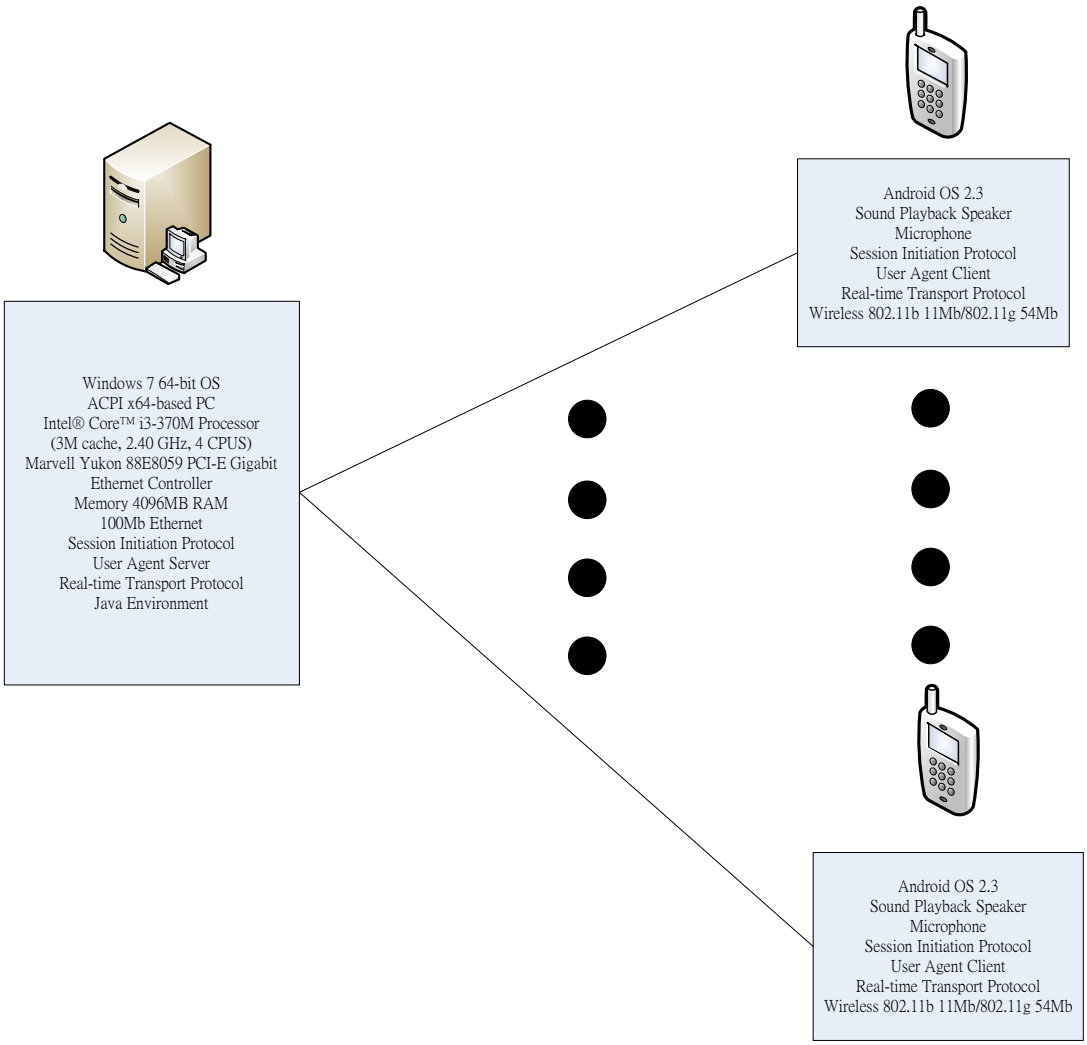

Figure 3: Two-Tier Client Server Architecture

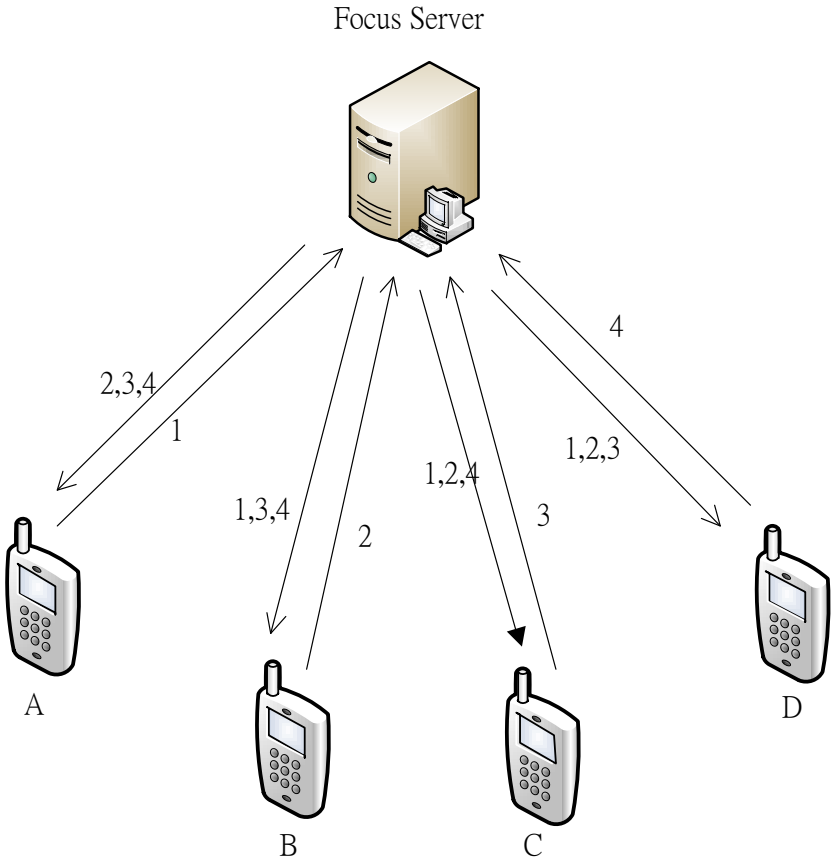

Figure 4: Centralized Message Flow Model

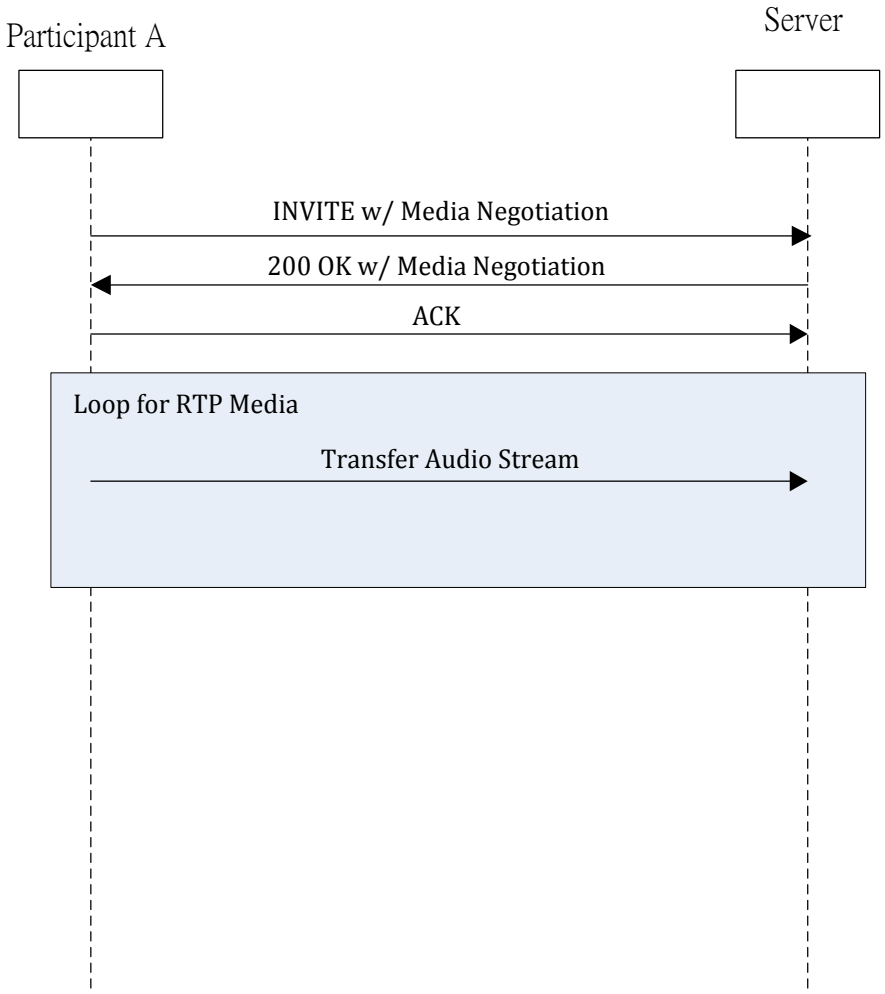

Figure 5: First client joining the conference

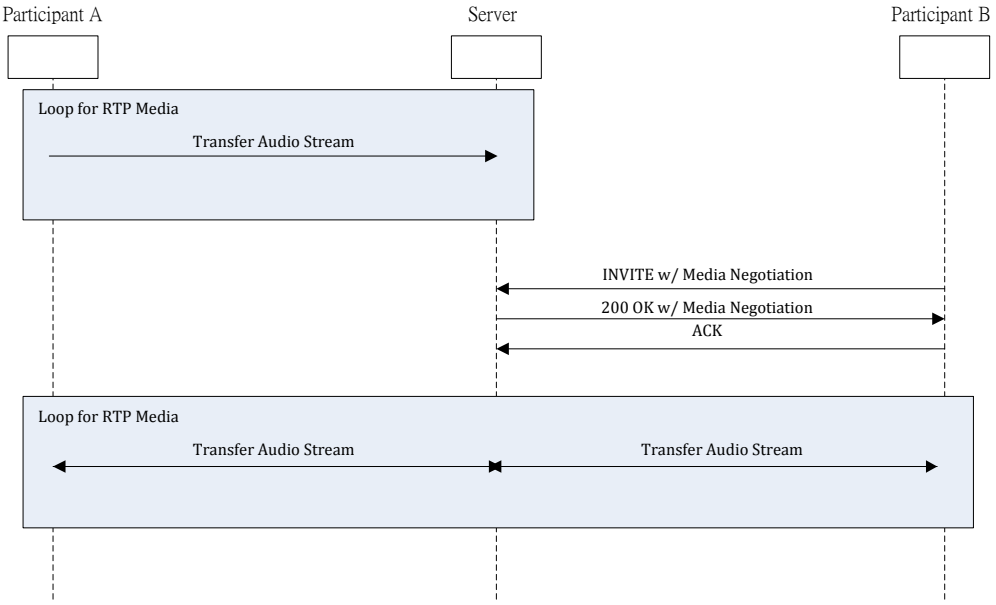

Figure 6: Second client joining the conference

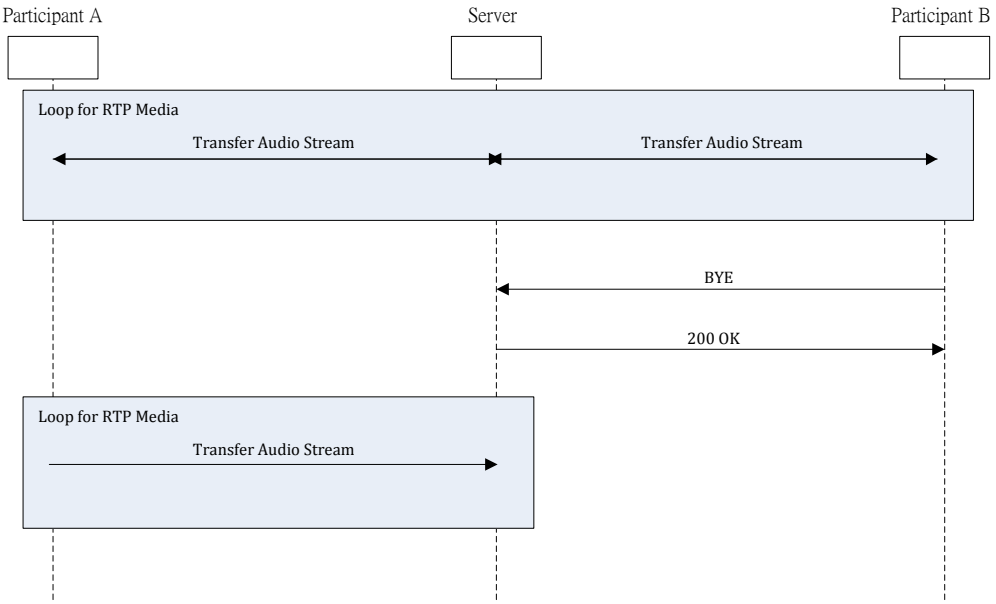

Figure 7: Client leaving the conference

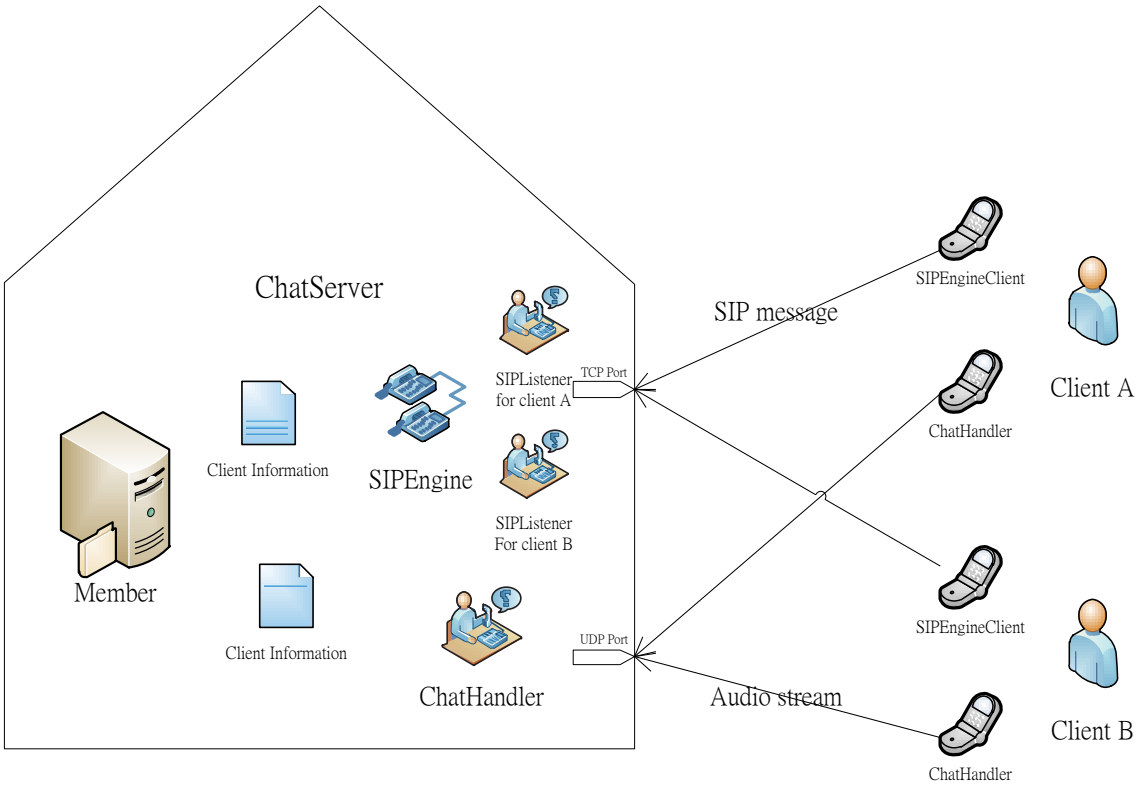

Figure 8: System architecture

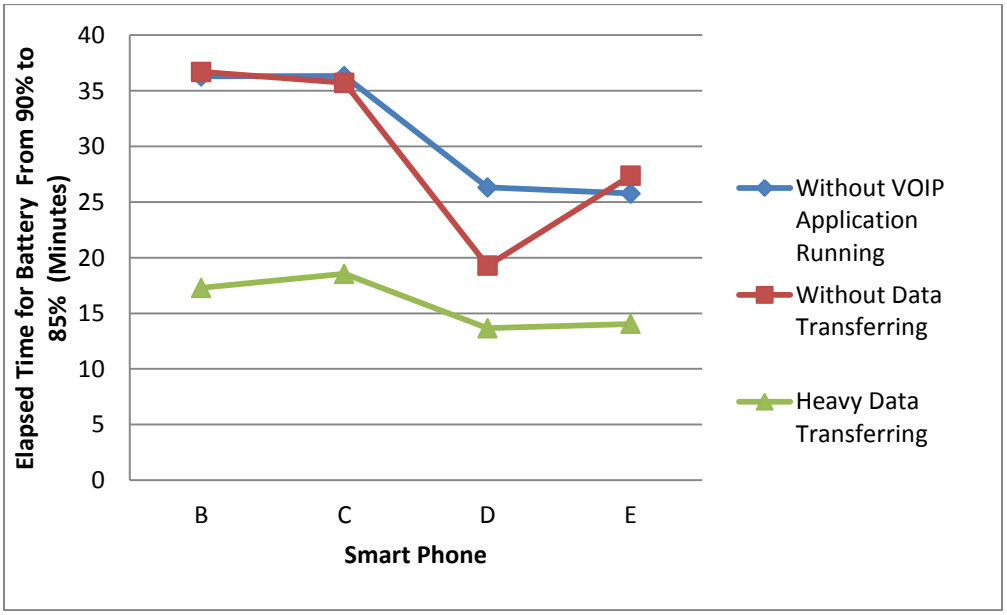

Figure 9: Elapsed time to consume 5% of battery life

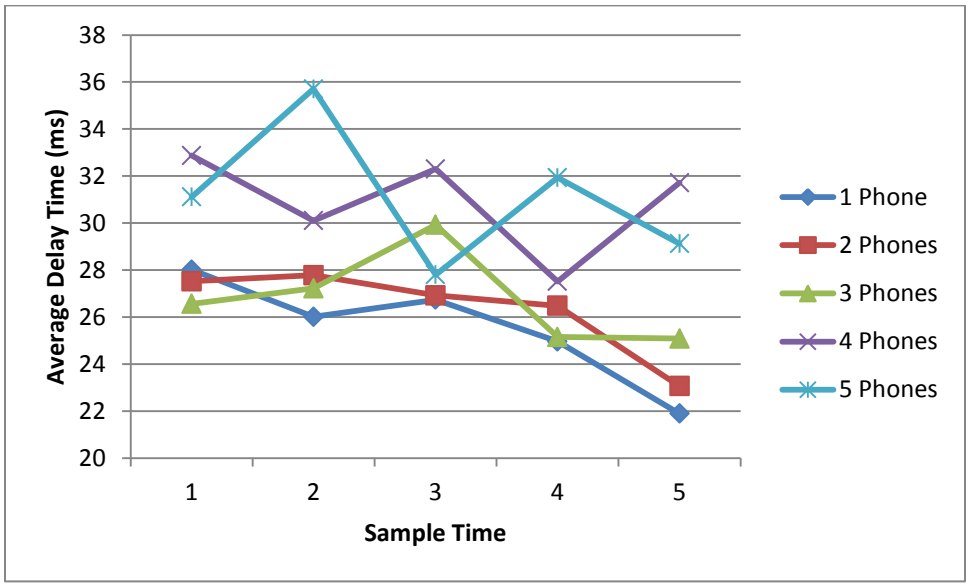

Figure 10: Average delay time of single source audio stream network (100 Packets)

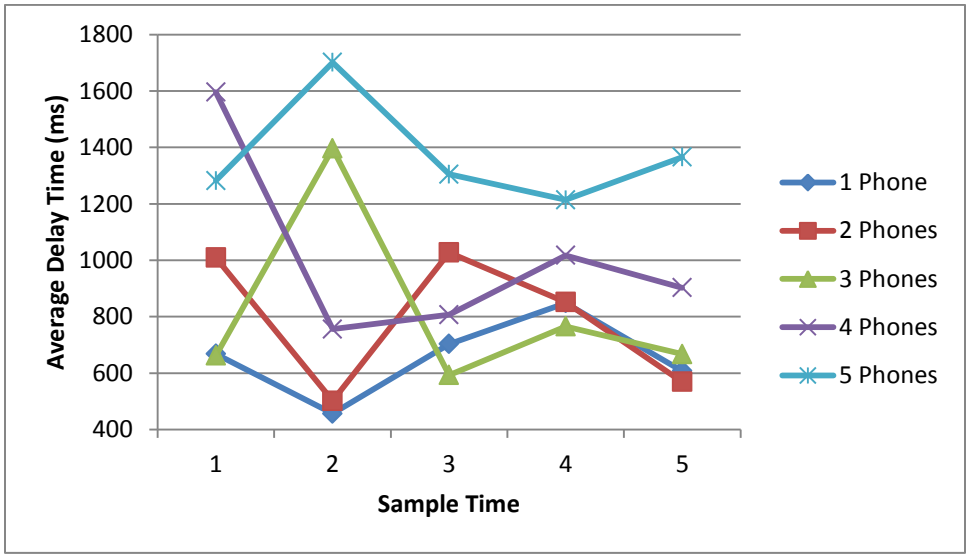

Figure 11: Average delay time of single source audio stream network (1000 Packets)

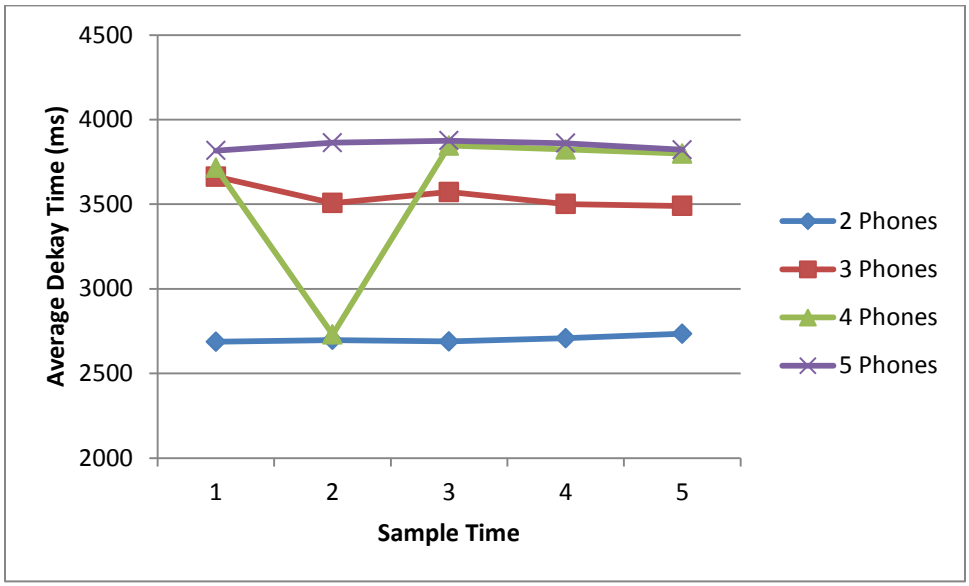

Figure 12: Average delay time of multi-sources audio stream network

Application Name	Support Codec
Sipdroid	Speex, G722, G711, GSM
LinPhone	Speex, G711, GSM, ILBC
SipAgent	Speex, G711, GSM
Kapanga	Speex, G.711, G.722, G.733, G.726, G.728, G.729, AMR, GSM, iLBC
fring	G.711, GSM
aSip	G.711, GSM

Table 1*: Third-party VOIP Application Supported Codec*

Device	**Model Number**	**CPU**	**RAM**	**Android Version**
A	Xperia Play	1 GHz Scorpion ARMv7 processor	512MB	2.3.4
B	DROID2	ARMv7 Processor rev 2 (V7I)	512MB	2.3.7
C	DROID2	ARMv7 Processor rev 2 (V7I)	512MB	2.3.7
D	SCHI500	Samsung-Intrinsity S5PC110 RISC Application Processor	512MB	2.3.4
E	SCHI500	Samsung-Intrinsity S5PC110 RISC Application Processor	512MB	2.1-update1

Table 2: *Experimental Smart Phones Specification*

Platform Version	API Level	VERSION_CODE
Android 4.0.3	15	ICE_CREAM_SANDWICH_MR1
Android 4.0, 4.0.1, 4.0.2	14	ICE_CREAM_SANDWICH
Android 3.2	13	HONEYCOMB_MR2
Android 3.1.x	12	HONEYCOMB_MR1
Android 3.0.x	11	HONEYCOMB
Android 2.3.4 Android 2.3.3	10	GINGERBREAD_MR1
Android 2.3.2 Android 2.3.1 Android 2.3	9	GINGERBREAD
Android 2.2.x	8	FROYO
Android 2.1.x	7	ECLAIR_MR1
Android 2.0.1	6	ECLAIR_0_1
Android 2.0	5	ÉCLAIR
Android 1.6	4	DONUT
Android 1.5	3	CUPCAKE
Android 1.1	2	BASE_1_1
Android 1.0	1	BASE

Table 3: API Level Supported by each version of the Android OS (Android.com, 2012)

Open Source Software Volunteerism vs. Motivating Potential of Primary Employment: Suggestions for a Research Agenda

Donald A. Carpenter
dcarpent@coloradomesa.edu
Colorado Mesa University
Grand Junction, CO 81506 USA

Abstract

The open source software phenomenon can be studied from a variety of perspectives. Given that much of the work on open source project has been accomplished by volunteers, one popular thread has been to examine the motivation of those who engage in open source software development. If motivation wanes, predictably there would be fewer contributions to open source projects. This paper reviews relevant literature on open source software, highlighting a recently published study that offers a prescription for future research based on social practice theory. The literature review herein also examines research on the motivation of computer programmers, the motivation of volunteers, and the personality of programmers. Next this paper weaves together those various research threads. This paper concludes with recommendations for a different direction for expanding the research on open source software developers' motivation by blending the extensive findings on job design of computer programming positions, research on motivation of volunteers, and research on programmer personalities. The paper ends with recommendations for future research hypotheses.

Keywords: open source software, programmer, motivation, work design, volunteerism

1. INTRODUCTION

The open source software (OSS) movement has grown over three decades—longer if free software (FS) is included—to the point where OSS resulted in $60 billion annual consumer savings in 2008 (OSS, 2012). Nearly 32,000 developers had contributed over five billion bytes of OSS code by 2002, with 74% of the code contributed by 10% of the developers (OSS, 2012). The advent of the Open Source Initiative (OSI) in 1998 formalized structure of the OSS community and quality of the software produced through standards and licensing (OSI, 2012). A few years ago, it was assumed that most of the OSS code is contributed by volunteers on their own time (OSS, 2012).

In the past few years, the situation has changed. A 2004 InformationWeek survey "found that 67% of companies use OSS/FS products, with another 16% expecting to use OSS/FS in 2005" (Wheeler, 2012). That has caught the attention of companies such as Google, Red Hat, IBM, Intel, Apple, Oracle, and Sony, all of which have released their own OSS products.

Such an economically important activity certainly warrants examination from multiple perspectives. One such recent angle has been to examine the motivation of the programmers who collaborate to create OSS primarily on their own time and without remuneration, i.e., as volunteers. At one time those accounted for the majority of OSS code. However, the playing field has changed. There are those instances of major corporations (as listed in previous paragraph) paying programmers to write OSS code. Another example relates to the popularity of Linux as the primary operating system for

servers on the Internet: 75% of the code of Linux is contributed by paid programmers. There remains a substantial portion of the total OSS developers who contribute to OSS voluntarily (Wheeler, 2007).

Many scholars have written on the topic of motivation of OSS contributors. Many of those seem to consider OSS contribution as a completely or predominately voluntary activity. However, no instances were found of any examination similar to that done herein, which addressed both paid and voluntary contributions to OSS.

This paper examines the literature on motivation of OSS developers, on motivation of programmers in general, and on motivation of volunteers. Given that motivation is individualized, this paper also examines the literature on research into the personalities of computer programmers. Then, it discusses the implications of the findings in the literature regarding directions for future research. It draws conclusions that previously recommended research paths might not get at the question thoroughly. This paper then suggests other opportunities for future research to get to the bottom of the question "What really motivates volunteer OSS developers?"

2. LITERATURE REVIEW

Motivation to Create OSS (OSS web sites)

Before turning to the scholarly literature, some enlightenment can be gained from considering what OSS developers state as their motivation to engage in OSS development. OSS developers' opinions are more likely to be found on blogs or on their own web sites rather than in the scholarly literature. Four sample web sites pertaining to OSS development provide the insights presented below. It should be noted that "motivation" in lay terms may be used to mean extrinsic "goals," "aims," or even "philosophies," rather than intrinsic "psychological motivators" used in scholarly research (Luthans, 2011, p. 157).

Hughes (2004) emphasizes that "The open source software I have seen is too good to be done by people who aren't very talented, experienced engineers" and that many "are in fact people who have a regular full time job." He emphasizes that there exists a gatekeeper on each OSS project to enforce quality, plus quality is enforced by standards and licensing rules. Hughes also explains the following as "a few of the better known reasons" why talented, employed programmers moonlight in OSS, conditions not necessarily present in day jobs.

- "plaudits from peers,"
- "technological challenges,"
- opportunity to be one's own boss,
- sharing common interests with others,
- making an important contribution to something that will endure over time,
- building additional work experiences.

Rhodes (2008) takes a different perspective, noting that "Most open source software projects were created by a programmer who needed a piece of software to accomplish a certain task. … the programmer decided to share it with the world by publishing it under an open source license." Subsequent programmers discover that software and make modifications to fit their specific needs, giving rise to open source communities and even to open source businesses.

A third explanation blends elements of Hughes and of Rhodes. Open Advocacy (2012) emphasizes that "By sharing the source, people are able to create a better product working together than they could working apart." That sharing builds camaraderie desired by the participants, encouraging a common opinion that all software source code should be shared, i.e., open source.

Wheeler (2007) reports a 2002 survey found that the top OSS developer motivations were

- intellectually stimulating (44.9%)
- improves skill (41.3%)
- work functionality (33.8%)
- code should be open (33.1%)
- non-work functionality (29.7%)
- obligation from use (28.5%)

from which he derived four groups:

- Believers (19%): believe source code should be open,
- Learning and Fun (29%): for non-work needs and intellectual stimulation,
- Hobbyists (27%): need the code for a non-work reason,
- Professionals (25%): for work needs and professional status.

Motivation to Create OSS (scholarly)

This paper now turns to the scholarly literature on the motivation to write OSS. Many scholars have published on the topic. Von Krogh, et al. (2012) have done a yeoman's job compiling the results of dozens of those OSS studies. Hence, there is no need to duplicate their efforts herein. Rather, highlights of their findings are in order.

Von Krogh, et al. explain that of the dozens of research frameworks in the scholarly works they examined, "the most frequent framework by far has been the distinction between intrinsic and extrinsic motivation in self-determination theory (SDT)," a mainstay psychological and organization behavior theory posited by Deci & Ryan (1985). Subsequently, von Krogh, et al. arbitrarily group the outcomes of forty research studies they examined into ten motivational categories under three headings, as illustrated in by following list.

- Intrinsic motivators
  - Ideology
  - Altruism
  - Kinship
  - Fun
- Internalized Extrinsic
  - Reputation
  - Reciprocity
  - Learning
  - Own-use
- Extrinsic
  - Career
  - Pay (p. 654)

Von Krogh, et al. state that "existing literature does not provide satisfactory answers to three differentiated questions as to why this [OSS] phenomenon exists." (p. 650). Those three questions are framed in the context of the values of social practice theory advocated by philosopher Alasdair MacIntyre (1981). Each question is well supported by theory and research propositions. The three questions, which are the basis for further research advocated by von Krogh, et al. (pp. 666-8) are:

- "How and why do OSS developers produce high-quality software (goods) when they do?"
- "Why do OSS developers change institutions?"
- "Why do developers sustain the social practice of OSS development?"

In Von Krogh, et al.'s concluding remarks is: "there is ample room to investigate motivation from a multitude of perspectives and methodological approaches." (p. 671)

A 2002 survey (Knosh, et al.) of 2,784 open source and free software developers indicated a median developer age of 26 years and a median starting age of those developers of 22 years (p. 8). Nearly 70% of the contributes spend ten or fewer hours per week on such work. The following list presents reasons (and respective percent of responses) to join an OSS community and to stay in an OSS community (p. 45).

- Make money (4.4, 12.3)
- Software should be open (30.1, 37.9)
- Limit power of big software firms (19, 28.9)
- Solve a specific problem (29.7, 29.6)
- Get help w/good software product (23.8, 27)
- Distribute software products (8.9, 10)
- Build self reputation (9.1, 12)
- Improve others' sw products (33.7, 39.8)
- Improve job opportunities (23.9, 29.8)
- Participate in OSS/FS scene (30.6, 35.5)
- Share knowledge and skills (49.8, 67.2)
- Learn and develop new skills (78.9, 70.5)
- Participate in cooperative form (34.5, 37.2)

Motivation of Computer Programmers

While Van Krogh, et al. present a literature review that supports their research agenda, they miss a large portion of research on motivation of computer programmers. Hackman & Oldham (1980) applied their job characteristics model (JCM) and job diagnostic survey (JDS) to 6930 workers in 56 organizations and in dozens of jobs in nine general job categories (p. 316). Their findings were reported as Growth Need Strengths (GNS) of the workers and Motivating Potential Score (MPS) of jobs. The study has been replicated many times in many settings with similar results each time. JCM concepts are components of other more complex, more individualized motivational theories such as Deci and Ryan (Carpenter, 2003).

JCM research indicates that high intrinsic work motivation (as well as high quality work performance, high work satisfaction, low turnover, and low absenteeism) result from three critical psychological states: (1) experienced meaningfulness of work, (2) experienced responsibility for outcomes of work, and (3) knowledge of the results of activities. Those three critical psychological states result from five

core job dimensions: (1) skill variety, (2) task identity, (3) task significance, (4) autonomy, and (5) feedback. There is no inherently good or bad level of MPS or GNS. Rather, it is good when an employee's GNS matches the MPS of his job and it is bad when there's a mismatch (Hackman & Oldham, p. 90).

Cougar & Zawacki (1980) expanded Hackman & Oldham's research to include computer professionals (i.e. programmers and systems analysts) and computer operators. Their original study and many subsequent studies demonstrate that programmers/analysts have significantly higher GNS than the next highest category (i.e., other professionals) but their jobs have nearly identical MPS as other professionals' jobs. (p. 21)

An interesting subset of Cougar & Zawacki's findings pertains to the component of satisfaction levels of programmers. While programmers have a higher level of general satisfaction than do other professionals, their levels of satisfaction with their co-workers and with their supervisors tends to be significantly less than those of other professionals. (p. 17)

Carpenter, et al. (2004) administered the JDS to 43 information technology (IT) employees in 13 different IT job categories. Only one programmer and one analyst were included in the sample. There was no statistically significant difference between MPS of the subjects' jobs and that of computer professionals in Cougar & Zawacki. However, subjects' GNS was statistical equivalent to GNS of general white collar workers, which is below that of general professionals, which in turn in below that of computer professionals. This indicates a greater mismatch for IT jobs in general than there is for analysts and programmers, between their motivational needs and the motivation that their jobs provide.

Motivation of Volunteers

Given that the majority of OSS is developed via a volunteerism model, it is important to look at the literature on motivating volunteers. Bang & Ross (2009, p. 61) studied responses from 254 volunteers at sporting events using a specialized instrument for measuring motivation of sporting event volunteers. Their findings indicate sporting event volunteers' motivations are grouped into seven factors: (1) expression of values, (2) community involvement, (3) interpersonal contact, (4) career orientation, (5) personal growth, (6) extrinsic rewards, and (7) love of sports.

Millette & Gagné (2008) analyze the motivation of 230 volunteers in a community clinic by applying Hackman & Oldham's JCM model. Among other findings, they note that MPS of volunteer positions is positively correlated with workers' levels of satisfaction (p. 17). That indicates that the JCM concepts of motivation apply similarly to volunteers as they do to paid workers and that the JDS is usable in volunteer settings.

Broekmeier (2010) warns against romanticizing the volunteer aspect of free and open source software (F/OSS or FLOSS), suggesting it is a norm that contributions begin during college when the volunteers have time and falls off if the graduate does not get a job writing FLOSS as a result of his contributions. He also seems to doubt whether volunteers are any differently motivated than paid contributors.

Berdou (2006) suggests that paid FLOSS contributors drive the major free and open source (F/OS) projects, while volunteers operate on the periphery. She notes that the paid FLOSS contributions have "been largely overlooked n the F/OS literature." (p. 201) The notion that motivations of paid versus volunteer programmers differ "has been challenged on several fronts by researchers who draw attention to the interconnections between the two spheres of economic activity." (p. 202)

Personality of Programmers

The Myers-Briggs Type Indicator (MBTI) has been applied in hundreds of studies. By means of a subject-answered survey, MBTI categorizes a subject's personality into one of sixteen types that are measured on four dimensions, as indicated by the pairs of terms in the list below. Studies show occurrences in the United States adult population to be distributed as follows, per CAPT (2012):

- Extroversion (E) 45-53%
- Introversion (I) 47-55%

- Sensing (S) 66-74%
- Intuiting (N) 26-34%

- Thinking (T) 40-50%
- Feeling (F) 50-60%

- Judging (J) 54-60%
- Perceiving (P) 40-46%

It is important to note that these ranges are probably different in other cultures, as some traits are valued and nurtured differently in various cultures. It is also important to recognize that people often behave differently than would be predicted by their personality types. (E&I, 2012)

Because they will come in handy later, the author inserts at this point in the literature review, four definitions. "Introversion" is "the act of directing one's attention toward or getting gratification from one's own interests, thoughts, and feelings" (Introversion, 2012). "Sensing" is "paying attention to physical reality" rather than intuition (Sensing, 2012). "Thinking" is "the action of using one's mind to produce thoughts" (Thinking, 2012). "Judging" is "forming an opinion about through careful weighing of evidence and testing of premises (Judging, 2012). The other four MBTI anchor terms can be derived as functional opposites of the ones just presented.

Many web sites provide information as to which jobs are best suited to which personality types. One such source (Career Matches, 2012) indicates that "computer programmer" is a good match for four of the eight personality types that include introversion but for only one of the eight personality types that include extroversion. Also, computer programmer is a good match for five of the eight personality types that include thinking, but none that include feeling; three that include judging but two that include perceiving; three that include intuiting, but two that include sensing. It is unknown as to whether those listings are supported by data and statistical analysis. Therefore, it would be irresponsible to conclude from that rundown that computer programming aligns very strongly with thinking and strongly with introversion. Nonetheless, it is an interesting set of information.

Similarly, one should be cautious about statements not supported by sound research. For example, in one essay is found: "I also observed that most really good software engineers were ENTJ, INTJ, ESTJ, ISTJ, ISFJ, and ENTP (Hardiman 1997, p. 10). Questionable also is an approach such listing software programmer job requirements, mapping those to a list of soft skills, and in turn arbitrarily mapping those to the eight anchors on the four MBTI scales in order to determine that "Most programmers are introvert (I), sensing (S), thinking (T) types." (Capretz & Ahmed, 2010, p.10) While the latter holds merit as a hypothesis, there is no quantitative data in that paper to support the claim.

Bentley (2005) cites McConnell (1999) as estimating that 20-40% of programmers are ISTJ or INTJ and that 80-90% of programmers have personalities that include the thinking element, but that programmers are evenly split between sensing and intuiting. That would imply that thinking and intuiting are overrepresented in programmers by comparison to the general U.S. population, according to the CAPT data on the previous page, above. Bentley himself concludes that the three great virtues of programmers are laziness, impatience, and hubris (2005, p. 7). However, per Bentley's definition, "efficient" and "quality-oriented" might fit equally well.

In Capretz (2003), the literature review discusses the results of four studies that used MBTI to measure programmer personality types. Capretz summarizes: "The common thread running through the results of these studies in the prevalence of introverts, thinking, judging, and almost as many sensing as intuitive among software professionals." (p. 209) When this author examines the data in those four studies in light of the CAPT ranges, thinking and intuiting are overrepresented in the programmers studied. On the other hand, introversion and judging percentages appear to be within CAPT's ranges.

Capretz's own study of 100 software engineers revealed 57% I, 67% S, 81% T, and 58% J (p. 211). By visual comparison to the CAPT ranges for the general U.S. population, introversion is slightly over-represented and thinking is hugely over-represented, while sensing and judging are within the CAPT ranges, although Capretz states otherwise. Capretz notes that 24% of his sample was scored as ISTJ, compared to 11.6% of the general U.S. adult population (p. 212).

In addition to reporting on GNS of programmers and MPS of their jobs, Cougar & Zawacki (1980) also reported on a personality aspect of programmers and system analysts, specifically their social needs strengths (SNS). Subsequent research with other programmers in other settings confirmed their findings. SNS is defined as "an individual's need to interact with others" (p. 23). Programmers and analysts have a substantially and statistically lower SNS than all

other professionals. It would be a logical leap to connect low SNS with introversion. However, the research did not explore that connection.

In addition to considering personality types, some researchers consider two-letter combinations of the MBTI variables as definitions of "temperaments." Keirsey & Bates (1978) present four temperaments: SP, SJ, NT, and NF, which they label as artisans, guardians, rationals, and idealists, respectively. Temperament letter combinations can be identified in the findings of several of the research studied discussed above. However, Keirsey (2012) does not give potential employment connections for the temperaments. Rather, it specifies programming or related positions as possibilities for a few of the MBTI types in guardian and rational groups. Here also, it would be dangerous to assume. Yet, in other studies, the SJ and NT letters are associated with programmers.

An aspect related to personality is behavior, the difference being that behavior is observable whereas personality is internal (Merrill & Reed, 1981, p. 7). Patterns of behavior can be grouped by four social styles (p. 2). While there is no stated correlation between social styles and preferred employment type, the thinking orientation of the analytical style (TRACOM, 2012) aligns with the overrepresentation of the thinking (T) variable in some of the above stated research. It is this author's observation over the years that most programmers also tend to exhibit most of the other characteristics of the analytical style.

3. DISCUSSION

Pursuing questions of OSS programmer motivation by applying the concepts of social practice theory (as advocated by von Krogh, et al.) might indeed yield significant results. Replication of that research with multiple populations in multiple settings will be a long term endeavor. It will be interesting to follow the process through the scholarly publications that result.

One challenge with treating the OSS community from a sociological perspective is the rapid rate of change occurring in that community. It is no longer the predominately-volunteer set of activities that it was a few short years ago. Major corporate players are shaping the game as much if not more so than are volunteers. Moreover, some of recommended research points about quality suggested by von Krogh, et al. might be answered by simply looking to the standards set by the Open Source Initiative. Other suggested research points about how developers shape organizations might be answered by examining the reported recent heavy involvement of large corporations which might be the real shapers.

This author thinks important answers about volunteer OSS developers' motivation lie elsewhere. Specifically, one can readily build upon a large body of research on motivation and personalities of programmers. The reason for drawing on literature related to personality is that motivation is individualized. Personality-oriented research gives clues to the individualistic nature of motivation. OSS programmers are a subset of all programmers. Measurement of OSS developers' GNS, SNS, MBTI types, and social styles, and MPS of their jobs would provide linkages back to extensive prior research.

If MPS was measured for both the programmers' paid jobs and for their volunteer OSS activities, differences might be seen that explain that they engage in OSS development because doing so provides important elements that are lacking in their paid jobs. Such measurement could also be tied to the literature on volunteers as it has been show that JDS can be appropriately applied to volunteer settings.

In particular, conducting such research could show OSS involvement is the preferred form of volunteerism for certain MBTI types who have the necessary skills. Those with lower SNS might find creating OSS to be a more desirable outlet than volunteer work that requires more face-to-face social interaction. The same could be said for the thinking or any other personality component in regard to OSS development. Perhaps programmers volunteer in OSS communities because they are stimulated by the additional thinking required of them, or the additional sensing, or the additional judging. Perhaps they do not get enough of those in their paid jobs.

Ultimately, the analysis could point out what needs to be improved in programmers' paid jobs. That would allow the application of job design to improve those jobs' motivating potential. That is not to say that employers would improve jobs solely in order to reduce the amount of time their

employees spend moonlighting on OSS. Rather, the goal would be to improve productivity, satisfaction, absenteeism, and turnover for those paid jobs. Employers should no more object to their employees' volunteer involvement in OSS than to other employees volunteering for other community projects.

Despite the reality that motivation is highly individualized, averages can be established to explain groups of typical subjects. As those analyses are performed, there are two groupings that should be established. As noted above, a growing portion of OSS is built as part of programmers' job assignments. The data for those should be segregated from the data for those who engage in OSS development voluntarily. A third category would be those who create OSS both as part of a paid job and as a volunteer.

Another demographic breakdown to consider is based on the off-used 80/20 rule of work. Above it is reported that 10% of the OSS developers have done 74% of the work. Perhaps those most heavily involved in developing OSS do not have paid jobs. It is unknown as to whether that ratio still holds. If it does hold, and if those in that minority are paid workers, perhaps they are less motivated by their paid jobs than are those in the majority. Or perhaps they are doing the OSS work because it is part of their paid jobs.

Other worker, job, and employer demographics should be collected as well as data about the type of OSS and the amount of time expended. It would be interesting to learn if there are age, gender, education-level, or other considerations that correlate with involvement in OSS. Given today's prevalence of offshore outsourcing of IT jobs, including programming, one imperative of such research should be to measure whether programmer and job characteristics as well as OSS involvement vary by geographic location.

One set of data that would greatly interest this author is regarding those programmers who are not involved in OSS development activities. Some say that OSS engagement is the new normal among programmers. If so, it might be more revealing to learn why some (most?) programmers do not become involved in OSS creation. Ultimately, that could prove to be more revealing than answers to why programmers do engage in OSS development.

4. CONCLUSIONS

This paper was written partially in response to a recent publication (Von Krogh, et. al, 2012) that recommends a social practice theory-based research agenda to determine why programmers engage in OSS development. This author thinks that it would be more productive to tie the OSS question to the large body of prior research on programmer motivation, programmer personality, and volunteer motivation. Those instruments are readily available. Hence, those results could be generated more quickly, which is important in a rapidly changing environment as the OSS movement. Those findings might be more widely received as more scholars have been exposed to those concepts than to social practice theory.

This author anticipates such research to yield several specific results that align with the notion that OSS involvement is the norm for younger programmer and that the OSS movement now is driven by corporations. They could be used as hypotheses by those who might engage in that research. It is expected such research will show:

- Extensive involvement in OSS development will vary with certain demographics, especially age, education level, time on the job, time proximity from college graduation, and geographic location.
- Demographics of OSS developers will show that a majority of OSS contributions are now being made by paid developers.
- By comparison to those who are not engaged or only tangentially engaged, those programmers who are engaged extensively in OSS development will have marginally higher growth need strengths and different personality types and social styles. However those measures of all OSS contributors will not be much different than all programmers.
- The paid jobs of those who generate OSS on their own time are deficient in terms of their motivating potential, thereby falling short of the employees' motivational needs. However, those jobs are no less deficient than the jobs of programmers in general.
- Outside OSS engagement fills some motivational void in those insufficient paid jobs.

- Those who spend extensive amount of time engaged in OSS development will have personality types, social styles, and social needs strengths that are substantially different from the general programmer population.

- OSS development allows programmers who possess certain personality types to engage in volunteerism that is not people-oriented.

- Developers' goals for their engagement in OSS will vary depending on whether they are volunteers or paid contributors.

The caveat to consider as such a research project begins is to pay attention to the rapid change in the OSS community. Both the above set of opinions and the hypotheses that result are subject to change with change in the nature of the OSS phenomenon. A thorough review of then-current literature generated by the OSS community might serve the researchers well as they deal with those ongoing inevitable changes.

5. REFERENCES

Bang, H., & Ross, S. D. (2009). Volunteer motivation and satisfaction. *Journal of Venue & Event Management, 1*(2). 61-77.

Bentley, J. E. (2005). Laziness, impatience, hubris: personality traits of a great programmer. Proceedings of Southeast SAS Users Group (SESUG). Retrieved July 5, 2012 from http://analytics.ncsu.edu/sesug/2005/SERO^_05.pdf.

Berdou, E., 2006, Insiders and outsiders: paid contributors and the dynamics of cooperation in community led F/OS projects. In Damiani, E., Fitzgerald, B., Scacchi, W., Scotto, M., & Succi, G., Eds. *International Federation for Information Processing, 203, Open Source Systems.* Boston: Springer. 201-208.

Brockmeier, J. (2010) Motivation & contributions in open source: stop romanticizing unpaid contributions. Ostatic Blog. Retrieved July 10, 2012 from http://ostatic.com/blog/motivation-and-contributions-in-open-source-stop-romanticizing-unpaid-contributions.

Carpenter, D.A. (2003). Connecting Logotherapy with motivational theories: Plan for research & consulting regarding meaningfulness. Proceedings of *World Congress Logotherapy.*

Carpenter, D. A., Slauson, G.J., Snyder, J., & Winniford, M.A. (2007). Motivating & managing computer personnel in 21st century: expanding pilot study. *Proceedings of Mountain Plains Management Conference*. 270-279.

Capretz, L. F. (2003) Personality types in software engineering. *International Journal of Human-Computer Studies, 58*(2). 207-214.

Capretz, L.F., & Ahmed, F. (2010). Making sense of software development and personality types. *IT Professional, 12*(1). 6-13.

Career matches by personality type. (2012). Retrieved July 5, 2012 from http:///www.mypersonality.info/personality-types/careers/.

Center for Applications of Psychological Types (CAPT). (2012). Estimated frequencies of the types in the United States population. Retrieved Jul 7, 2012 from http://www.capt.org /mbti-assessment/estimated-requencies.htm.

Couger, J. D., & Zawacki, R. A. (1980). *Motivating and Managing Computer Personnel*. New York, John Wiley and Sons.

Deci, E. L., & Ryan, R. M. (1985). *Intrinsic Motivation and Self-Determination in Human Behavior.* New York: Plenum Press.

Extraversion and introversion (E&I). (2012). In *Wikipedia, The Free Encyclopedia*. Retrieved on July 8, 2012, from http://en.wikipedia.org/w/index.php?title=Extraversion_and_introversion&oldid=500919123.

Ghosh, R. A., Glott, R., Krieger, B., & Robles, G. (2002). Free/libre & open source software: survey and study, Part 4. International Institute of Infonomics, University of Maastricht, Netherlands.

Hackman, J. R., & Oldham, G. R. (1980). *Work Redesign*. Reading, MA: Addison-Wesley.

Hardiman, L. T. (1997). Personality types and software engineers. *Computer, 30*(10). 10.

Open Source Initiative (OSI). (2012). History of Open Source Initiative, Retrieved Jul 9, 2012 from http://www.opensource.org/.

Hughes, J. (2004). Why do people write open source software? *OpenXtra*. Retrieved June 14, 2012 from http://www.openxtra.co.uk/articles/why-do-people-write-open-source-software.

Introversion. (2012). In Merriam-Webster online Dictionary. Retrieved July 7, 2012 from http://www.merriam-webster.com/medical/introversion.

Judging. (2012). In Merriam-Webster online Dictionary. Retrieved Jul 7, 2012 from http://merriam-webster.com/dictionary/judging.

Keirsey, D. (2012). Overview of the four temperaments. Retrieved on July 8, 2012 from http://www.keirsey.com/4temps/overview_temperaments.asp

Keirsey, D., & Bates, M. (1978). *Please Understand Me: Character & Temperament Types.* Del Mar, CA: Prometheus Nemesis Books.

Luthans, F. (2011). *Organizational Behavior, An Evidence-Based Approach (12th ed.).* New York: McGraw Hill Irwin.

MacIntyre, A. C. (1981). *After Virture: A Study in Moral Theory (1st ed.)*. Notre Dame, IN: University of Notre Dame Press.

McConnell, S. (1999). *After the Gold Rush: Creating a True Profession of Software Engineering*. Redmond, WA: Microsoft Press.

Merrill, D. W., & Reid, R. H. (1981). *Personal Styles and Effective Performance.* Radnor, PA: Chilton Book Co.

Millettte, V., & Gagné, M. (2008). Designing volunteers' tasks to maximize motivation, satisfaction and performance: the impact of job characteristics on volunteer engagement. *Motivation & Emotion, 32*(1). 11-22.

Open Advocacy (2012). Why do people create open source software? *How Open Source Works*. Retrieved June 14, 2012 from http://www.openadvocacy.org/answers/why-do-people-create-open-source-software/.

Open-source software (OSS). (2012, June 8). In *Wikipedia, The Free Encyclopedia*. Retrieved July 2, 2012, from http://en.wikipedia.org/w/index.php?title=Open-source_software&oldid=496613745.

Rhodes, T. (2008). Why do people make software for free? *The Open Source Advocate*. Retrieved June 14, 2012 from http://www.useopensource.blogspot.com/2007/08/why-do-people-make-software-for-free.html.

Sensing or Intuition. (2012). The Myers & Briggs Foundation. Retrieved July 7, 2012 from http://www.myersbriggs.org/my-mbti-personality-type/mbti-basics/sensing-or-intuition.asp

Thinking. (2012). In Merriam-Webster online Dictionary. Retrieved July 7, 2012 from http://www.merriam-webster.com/dictionary/thinking.

TRACOMM Group. (2012). The four social styles —analytical style. Retrieved July 10, 2012 from http://www.tracomcorp.com/training-products/model/Analytical-Style.html

Von Krogh, G., Haefliger, S., Spaeth, S., & Wallin, M.W. (2012). Carrots and rainbows: motivation and Social development in open source software practices. *MIS Quarterly, 36*(2). 649-676.

Wheeler, D. A. (2007). Why Open Source Software/Free Software (OSS/FS, FLOSS, FOSS)? Look at the numbers! Retrieved July 9, 2012 from http://www.dwheeler.com/oss_fs_why. html #challenges.

Rocky Relationships: Enterprise Resource Planning and Supply Chain Management

Jack Crumbly
crumblyj@mytu.tuskegee.edu
Supply Chain Management
Tuskegee University
Tuskegee, AL 36088, USA

Meg Fryling
mfryling@siena.edu
Computer Science
Siena College
Loudonville, NY 12211, USA

Abstract

In the past decade, supply chain management (SCM) and logistics leaders have increasingly looked to information technology tools to enhance performance. Implementation of enterprise resource planning (ERP) information systems has been the common choice for organizations to integrate enterprise-wide processes. ERPs have been advertised as an application that improves business processes and develops visibility across the organization. However, there is concern about the effectiveness of ERP capabilities in improving supply chain processes. This exploratory study examines perceptions regarding the success and effectiveness of ERP installations on supply chain management, with a focus on the post-implementation phase. Five main ideas emerged from this study: 1) ERP systems lack the functionality to effectively manage the entire supply chain. 2) There is a "black hole" in the supply chain, which ERP systems are currently unable to handle, that prevents organizations from achieving the transparency and improved logistics they desire. 3) Supply chain management is often given little consideration during the ERP implementation phase, leaving functionality gaps and supply chain inefficiencies. 4) The inability for ERP systems to effectively manage the supply chain is a source of frustration for supply chain management but has little impact on perceptions of ERP success at the organizational level. 5) The most successful implementations involve trust, communication, and collaboration between individuals involved in the implementation, including supply chain management.

Keywords: ERP, enterprise resource planning, SCM, supply chain management, post-implementation, IS Success

1. INTRODUCTION

As organizations adapt to changing environments, including globalization, economic variability, natural and human-aided disasters, they have relied upon supply chain management strategies to improve delivery of products and services, reduce costs within the organization and improve overall business strategies (Gunasekaren, Patel, & Tirttiroglu, 2001). In the

past decade, supply chain and logistics leaders have increasingly looked to information technology tools to enhance performance (Fawcett, Wallin, Allred, Fawcett, & Magnan, 2011). Implementation of enterprise resource planning (ERP) information systems has been the common choice for organizations to integrate enterprise-wide processes (Soh, Kien, & Tay-Yap, 2000). ERPs have been advertised as an application that improves business processes and develops visibility across the organization (Gargeya & Brady, 2005).

Organizations use IT solutions such as Enterprise Resource Planning (ERP) systems to improve productivity (Sabherwal & Chan, 2001) and gain competitive advantage (Byrd & Turner, 2001). Information exchange is a key component for successful supply chains (Bartlett, Julien, & Baines, 2007; Cooper, Lambert, & Pagh, 1997; Derocher & Kilpatrick, 2000; Evans & Wurster, 1997; Forslund, 2007; Griffith & Myers, 2005; Myers & Cheung, 2008; Thatte, 2007). Sharing information in the supply chain greatly improves its performance (Bilek, 2010; Hsu, Kannan, Tan, & Leong, 2008). ERP installations are intended to reduce the challenge of managing multiple information systems and interfaces by introducing one continuous application that is continuously updated (Allen, 2011). This process is meant to provide supply chain visibility within an organization (Allen, 2011), improving knowledge integration. Proper knowledge integration connects functions within organizations to improve core competencies (Wadhwa, Saxena, & Chan, 2008). Research has found that an increase in knowledge integration within the organization increases knowledge sharing with supply chain partners (Wadhwa et al., 2008). Patnayakuni et al. (2006) found that collaborative exchange and integration of knowledge across phases of development has a positive influence on development performance. Myers and Cheung (2008) postulate that when buyers and suppliers share information, suppliers improve their ability to adapt to buyers' needs.

ERP systems attempt to reduce costs and increase efficiencies by integrating business processes and improving information access across an organization (Umble, Haft, & Umble, 2003). Similarly, effective supply chain management aims to increase margin by sharing information inter-organizationally (Chen, Yen, Rajkumar, & Tomochko, 2010). ERP systems have the ability to improve information exchange but are traditionally intended to manage information within a single organization (Li, Chaudhry, & Zhao, 2006). Since supply chain management (SCM) consists of multiple organizations needing to work together as one organism, some would argue that ERP systems are insufficient for modern SCM (Akkermans, Bogerd, Yucesan, & van Wassenhove, 2003). One reason for the challenge is the lack of trust between organizations (Galaskiewicz, 2011). The multiple relationships within supply chain management to deliver supplies to manufacturers and deliver products to customers takes a substantial level of trust (Galaskiewicz, 2011). Trust can be a significant challenge in any ERP implementation (Gefen, 2004). Successful supply chain integration and long-term performance not only requires trust between parties in one organization but between supply chain partners in multiple organizations (Bowersox, Closs, & Stank, 2000; Dyer & Singh, 1998; Kwon & Suh, 2005; Vijayasarathy, 2010). Sharing information between organizations and supply chain partners through ERPs not only requires trust but a high level of commitment between all parties involved (Gefen, 2004).

ERP systems impose standardized business practices that may conflict with existing supply chain processes. Research has found that ERP systems provide little improvement to the supply chain and often introduce new supply chain management issues (Akkermans et al., 2003). ERP systems have found to be lacking critical functionality and are inflexible to the ongoing changes in the supply chain (Akkermans et al., 2003). Additionally, the business process reengineering typically required to implement ERP systems introduces a variety of challenges (Lee et al., 2003; Robey et al., 2002). ERP installations potentially threaten an organization's ability to sustain a competitive advantage by introducing generic software that is used by competing organizations (Carr, 2003). According to the resource-based view of organizations, a firm's resources, which include knowledge, are a key determinant of organizational performance (Hofer & Schendel, 1978; Wernerfelt, 1984, 1995). A competitive advantage is gained by resources that are not easily replicated by another organization (Menor & Roth, 2008). Technology resources can offer competitive advantages only if they remain proprietary (Carr, 2003).

Organizations are left with this dichotomy where it appears illogical to replicate software in-house that can be purchased off-the-shelf and the challenge that this type of software with embedded best practices leaves the organization vulnerable. As Carr (2003, p. 11) mentions, "when a resource becomes essential to competition but inconsequential to strategy, the risks it creates become more important than the advantage it provides." Organizational procedures may be reengineered in order to match the "best practice" processes an ERP imposes. While in some cases this may be advantageous, "the best practices underlying the ERP system have a highly integrative nature and consequences of changes in one aspect may "ripple through" the organization in unforeseen and even unseen ways" (van Stijn & Wensley, 2005a, p. 11). Changing existing business practices to match ERP best practices is typically considered the best way to increase efficiencies, reduces costs, and improve competitive advantages (Yu, 1996, p. 613). However, ERP systems can be customized in order to resolve functionality gaps and provide organizations with a way to differentiate themselves from their competitors (Bearda & Sumner, 2004; Davenport, 1998).

The success of ERP implementations is often measured at implementation time, leaving post-implementation completely out of the equation. However, post-implementation is a critical phase that should be taken into consideration in order to fully evaluate the success of an ERP installation (Fryling, 2010; King & Burgess, 2006). In fact, it is not unusual for much of the business process improvement to occur during the post implementation phase (Willis & Willis-Brown, 2002). While research on ERP post-implementation is limited (Ifinedo & Nahar, 2006; Santhanam, Seligman, & Kang, 2007; Wagner & Newell, 2007), there has been an increased interest in recent years in examining post-implementation critical success factors (Bai & Mao, 2010; Ifinedo, Rapp, Ifinedo, & Sundberg, 2010) and total cost of ownership (Fryling, 2010). This manuscript examines the post-implementation effectiveness of ERP installations on supply chain management through semi-structured interviews.

2. METHODOLOGY

This study aims to gain dynamic insight as to the relationship between ERP systems and supply change management, with a focus on the post-implementation environment. The objectives of this study are to uncover the impact of ERP system implementations on supply chain processes and logistics as well as identify areas for future research. Primary questions to be addressed in this study are:

1. Are institutions meeting the objectives, goals and expectations of their ERP implementation?
2. Is supply chain management considered when setting the objectives, goals and expectations of the ERP implementation?
3. Has the ERP improved supply chain transparency?
4. Has the ERP improved supply chain functionality?
5. Has the ERP improved the logistics process?

Data were collected via interviews with eight representatives from six organizations in ERP post-implementation phase and a consultant with approximately 15 years of experience on ERP implementations and post-implementations at several major companies involving multiple ERP vendors. Interviews consisted of 12 open-ended questions related to the interview's experience with ERP systems and their impact on the supply chain (see Appendix A). Organization demographics (size, revenue, ERP vendor) were collected in the interviews (see Appendix B). All interviews took place between 2010 and 2011 and were conducted by the same interviewer to improve consistency between interviews. As needed, follow-up questions were solicited via the original interviewer.

3. RESPONSES

Overall interview responses indicated that institutions were struggling to meet the objectives, goals and expectations of their ERP implementation. Interviewee G explained that the ERP met their objectives for simple shipping but critical SCM functionality was lacking. Manual processes and external systems were needed to satisfy SCM needs. The company found that inbound and outbound logistics struggled and the ERP forced certain business practices that were not ideal for the organization. Company G reported that the ERP did not provide the transparency promised by the vendor and supply chain improvement was

only realized when a third party transportation management system (TMS) was implemented to supplement the functionality gaps in the ERP. While Organization F reported that the ERP did help with transparency, it simply was not designed to enhance inbound/outbound logistics.

While most of the interviewees acknowledged that the ERP implementation improved some business processes, the supply chain functionality was deficient. Organization A found that the ERP simply did not manage the entire supply chain, leaving significant functionality gaps and information "black holes". They were also frustrated by the time and cost associated with implementing the ERP. Respondent D explained that their organization had to hire consultants to fix issues discovered during implementation. Interviewee B saw the implementation as having both positive and negative implications. The company was able to eliminate its paper-based system so orders could be processed much more quickly. However, the respondent expressed concern regarding sacrificing people-oriented customer service for automation efficiency. Interviewee F stated he felt the organization simply did not meet the goals and objectives of the ERP implementation. The respondent stated that the applications did not adequately meet supply chain functions. Even after spending $40 million dollars on the ERP project, there were major functionality gaps and the implementation occurred five years later than expected. Functionality workarounds were necessary and required hiring subject matter experts to customize the software. Nonetheless, the organization perceived the implementation as a success because the ERP did improve other processes.

Interviewees A, D, and G all reported that SCM was not adequately considered during the ERP procurement process. However, Interviewee E stated that their organization had good communication and collaboration regarding functionality gaps. It was understood and accepted from the beginning that the ERP system would need to be customized in order to meet their needs. It seems they were able to understand the functionality limitations upfront and set realistic expectations. Overall interviewees reported that SCM has little impact in determining ERP success at the organizational level. When respondents reported difficulties with SCM functionality, they still indicated that the organization viewed the implementation as successful. Even Company E, that reported the most successful implementation experience, mentioned that ERP systems need to be able to adapt to changing technologies. Interviewee G echoed this by stating that the ERP inflexibility is a barrier to SCM success. Interviewee F added that "…the ERP does not perform warehouse functions well; it is very primitive." He felt that supply chain execution is a misnomer in ERP applications. Significant changes were required to adapt the ERP to critical functions such as the bill of lading.

Trust and user resistance was found to be a barrier to achieving transparency and taking advantage of the full functionality of the ERP system. Interviewee C stated that while the ERP did provide the organization with improved functionality and transparency, not all members of the organization were willing to use the software. In addition, the organization did not solicit consulting support from the vendor to improve the ERP functionality, which the interviewee felt would have beneficial to overall process improvement. Interviewee C also shared that administration seemed to lack trust in its workforce, giving very limited access to users of the system. Interviewee B felt that his organization did get improved functionality and transparency from the ERP but found that employees were unclear about why they needed to change their business practices and were left defending their existing methods for conducting business. Interviewee F stated that even after the ERP implementation employees continued to work in silos. He felt there is minimal collaboration between functional units and lack of focus on improving the supply chain process.

4. DISCUSSION

ERP systems offer some operational efficiencies and help improve parts of the supply chain but fall short of effectively managing the entire supply chain network. There is a "black hole" in the supply chain when the merchandise is between companies, which ERPs are currently unable to handle. ERP systems were intended to manage the entire enterprise of one organization, not multiple organizations. A generic system lacks the ability to adequately manage this "black hole" because the relationship between each company is complex and unique. Additionally, ERP systems simply lack the agility to adapt to changes in SCM logistic processes or organizational changes. Organizations must rely on software vendors and consultants to modify and/or extend

capabilities as needed. There are potential long-term negative implications of implementing generic ERP systems, which warrant additional investigation. Can organizations maintain a sustainable competitive advantage fitting their business practices to the ERP's delivered blueprint best practice processes? Should customization of the software be an accepted and embraced part of the implementation process? What impact does customization versus business process reengineering have on ERP success? Does the inflexibility of ERP systems simply make them inadequate solutions for supply chain management? Will the inflexibility of ERP systems put companies at risk as they lack the agility to efficiently adjust to business model changes?

This research supports prior research on the importance of trust, communication, and collaboration between organizational departments in the evaluation and implementation of ERP systems. When supply chain management and functional representatives were not invited to interact directly with ERP vendors, trust and user expectation issues suffered. For the cases interviewed as part of this study, the supply chain seems to have little impact in determining ERP success at the organizational level. Even when respondents reported difficulties with SCM functionality, they indicated that the organization viewed the implementation as successful. While ERP systems provide some of the expected benefits, not all goals and objectives are met. Interview responses suggest that ERP improves some supply chain and logistics, but not all. Organizations were still left with functionality gaps, which required workarounds and additional software solutions.

One area for further research is the impact of SCM leadership in the ERP decision-making and implementation process. Many of the organizations interviewed acknowledged that SCM personnel were left out of the ERP evaluation and selection process. Would these companies have enjoyed a more successful implementation, particularly with supply chain functionality, had SCM employees been more heavily involved in project planning or is ERP software simply inadequate for SCM? One interviewee, who indicated their organization had ongoing communication and collaboration with the SCM staff during the pre-implementation stages, reported the most favorable post-implementation supply chain management experience. Is this simply because the organization set more appropriate user expectations?

5. CONCLUSION

The results of the study indicate ERP installations improve some processes within the organization, but often supply chain functionality is lacking. The ERP does not provide functions that improve transportation and warehouse processes that are performed by supply chain partners. ERP processes do not have the flexibility to advance the organization's supply chain without high costs and long-term commitments to ERP vendors, third-party vendors, consultants, and software customizations. Applications such as transportation management systems (TMS) have been used as a solution for supply chain partners to exchange information with each other. Applications such as TMS provide transparency, but as with ERP they do not provide all the features needed to support the supply chain.

Organizations are spending a considerable amount of resources implementing ERP systems in an effort to improve efficiencies and reduce operational costs. This research provides practical information regarding the limitations of ERP for supply chain management and offers areas for future investigation. If organizations better understand both the benefits and limitations of ERP systems they may have a better experience and realize more organizational benefits. ERP vendors are setting unrealistic expectations regarding software capabilities in order to "make the sale", leaving customers frustrated. From this study we postulate that even with the shortcomings of ERP systems, customer frustration can be mitigated and post-implementation success can be elevated through proper preparation and training. Organizations which are better educated regarding ERP limitations and prepared to manage functionality gaps, experience a more successful implementation and enjoy an improved post-implementation environment.

6. REFERENCES

Akkermans, H. A., Bogerd, P., Yucesan, E., & van Wassenhove, L. N. (2003). The impact of ERP on supply chain management: exploratory findings from a European Delphi

study. *European Journal of Operational Research, 146*(2), 284-301.

Allen, T. (2011). Improve Your Business. *Strategic Finance, May,* 54-59.

Bai, H., & Mao, J.-Y. (2010). *Exploring the Path to ERP Implementation Success: In Retrospect after Rollout.* Paper presented at the Pacic Asia Conference on Information Systems.

Bartlett, P. A., Julien, D. M., & Baines, T. S. (2007). Improving supply chain performance through improved visibility. *The International Journal of Logistics Management, 18*(2), 294-313.

Bearda, J. W., & Sumner, M. (2004). Seeking strategic advantage in the post-net era: viewing ERP systems from the resource-based perspective. *Journal of Strategic Information Systems, 13*(2), 129-150.

Bilek, G. M. (2010). The Value of Information Sharing in a Build-to-Order Supply Chain. *The Business Review, Cambridge, 15*(1).

Bowersox, D. J., Closs, D. J., & Stank, T. P. (2000). Ten mega-trends that will revolutionize supply chain logistics. *Journal of Business Logistics, 21*(2), 1-16.

Byrd, T., & Turner, D. (2001). An exploratory analysis of the value of the skills of IT personnel: Their relationship to IS infrastructure and competitive advantage. *Decision Sciences, 32*(1), 21-54.

Carr, N. G. (2003). IT Doesn't Matter. *Harvard Business Review, May*, 41-49.

Chen, J. V., Yen, D. C., Rajkumar, T. M., & Tomochko, N. A. (2010). The antecedent factors on trust and commitment in supply chain relationships. *Computer Standards & Interfaces*. doi: 10.1016/j.csi.2010.05.003

Cooper, M. C., Lambert, D. M., & Pagh, J. D. (1997). Supply chain management: More than a new name for logistics. *The International Journal of Logistics Management, 8*(1), 1-14.

Davenport, T. H. (1998). Putting the Enterprise into the Enterprise System. *Harvard Business Review, 76*(4), 121-131.

Derocher, R. P., & Kilpatrick, J. (2000). Six supply chain lessons for the new millennium. *Supply Chain Management Review, 3*(4), 34-41.

Dyer, J. H., & Singh, H. (1998). The relational view: cooperative strategy and sources of interorganizational competitive advantage. *Academy of Management Review, 23*(4), 660-679.

Evans, P. B., & Wurster, T. S. (1997). Strategy and the new economics of information. *Harvard Business Review, 72*(2), 75-82.

Fawcett, S., Wallin, D., Allred, C., Fawcett, A., & Magnan, G. (2011). Information technology as an enabler of supply chain collaboration: A dynamic-capabilities perspective. *Journal of Supply Chain Mangement, 47*(1), 38-59.

Forslund, H. (2007). The impact of performance management on customers' expected performance. *International Journal of Operations and Production Management, 27*(8), 901-918.

Fryling, M. (2010). Estimating the impact of enterprise resource planning project management decisions on post-implementation maintenance costs: a case study using simulation modelling. *Enterprise Information Systems, 4*(4), 391–421.

Galaskiewicz, J. (2011). Studying Supply Chains from a Social Network Perspective. *Journal of Supply Chain Management, 47*(1), 4-8.

Gargeya, V. B., & Brady, C. (2005). Success and failure factors of adopting SAP in ERP system implementation. *Business Process Management Journal, 11*(5), 501-516.

Gefen, D. (2004). What Makes an ERP Implementation Relationship Worthwhile: Linking Trust Mechanisms and ERP Usefulness. *Journal of Management Information Systems, 21*(1), 263-288.

Griffith, D. A., & Myers, M. B. (2005). The performance outcomes of strategic fit of relational norm inter-organizational governance strategies. *Journal of International Business Studies, 36*(3), 254-269.

Gunasekaren, A., Patel, C., & Tirttiroglu, E. (2001). Performance measures in a supply chain environment. *International Journal of*

Operations and Production Management, 21(1/2), 71-87.

Hofer, C. W., & Schendel, D. (1978). *Strategy formulation: Analysis concepts*. St. Paul, MN: West.

Hsu, C.-C., Kannan, V. R., Tan, K.-C., & Leong, G. K. (2008). Information sharing, buyer-supplier relationships, and firm performance. *International Journal of Physical Distribution & Logistics Management, 38*(4), 296-310.

Ifinedo, P., & Nahar, N. (2006, April 23 - 27, 2006). *Prioritization of Enterprise Resource Planning (ERP) Systems Success Measures: Viewpoints of Two Organizational Stakeholder Groups.* Paper presented at the ACM Symposium on Applied Computing, Dijon, France.

Ifinedo, P., Rapp, B., Ifinedo, A., & Sundberg, K. (2010). Relationships among ERP post-implementation success constructs: An analysis at the organizational level. *Computers in Human Behavior, 26*, 1136-1148.

King, S. F., & Burgess, T. F. (2006). Beyond critical success factors: A dynamic model of enterprise system innovation. *International Journal of Information Management, 26*(1), 59-69.

Kwon, I.-W. G., & Suh, T. (2005). Trust, commitment and relationships in supply chain management; a path analysis. *Supply Chain Management: An International Journal, 10*(1), 26-33.

Lee, J., Siau, K., Hong, S. (2003). Enterprise integration with ERP and EAI. *Communications of the ACM*, 46(2), 54-60.

Li, Z., Chaudhry, S., & Zhao, S. (2006). Designing ERP systems with knowledge management capacity. *Systems Research and Behavioral Science, 23*(2), 191-200.

Menor, L. J., & Roth, A. V. (2008). New Service Development Competence and Performance: An Empirical Investigation in Retail Banking. *Production and Operations Management, 17*(3), 267-284.

Myers, M., & Cheung, M. (2008). Sharing global supply chain knowledge. *Sloan Management Review, 49*, 67-73.

Patnayakuni, F., Rai, A., & Seth, N. (2006). Relational antecedents of information flow integration for supply chain coordination. *Journal of Management Information Systems, 23*(1), 13-49.

Robey, D., Ross, J.W., Boudreau, M. (2002). Learning to implement enterprise systems: An exploratory study of the dialectics of change. *Journal of Management Information Systems*, 19(1), 17-46.

Sabherwal, R., & Chan, Y. E. (2001). Alignment Between Business and IS Strategies: A Study of Prospectors, Analyzers, and Defenders. [Journal Article]. *Information Systems Research, 12*(1), 11-33.

Santhanam, R., Seligman, L., & Kang, D. (2007). Postimplementation Knowledge Transfers to Users and Information Technology Professionals. *Journal of Management Information Systems, 24*(1), 171-199.

Soh, C., Kien, S. S., & Tay-Yap, J. (2000). Cultural Fits and Misfits: Is ERP a Universal Solution? *Communications of the ACM, 43*(4), 47-51.

Thatte, A. A. (2007). *Competitive advantage of a firm through supply chain responsiveness and SCM practices.* Dissertation, University of Toledo, Toledo, Ohio.

Umble, E., Haft, R., & Umble, M. (2003). Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operations Research, 146*, 241-247.

van Stijn, E., & Wensley, A., J.L. (2005a). *ERP's Best Practices and Change: An Organizational Memory Mismatch Approach.* Paper presented at the Thirteenth European Conference on Information Systems, Regensburg, Germany.

Vijayasarathy, L. R. (2010). Supply integration: an investigation of its multi-dimensionality and relational antecedents. *International Journal of Production Economics, 124*(2), 489-505.

Wadhwa, S., Saxena, A., & Chan, F. T. S. (2008). Framework for flexibility in dynamic supply chain management. [Article]. *International Journal of Production Research,*

46(6), 1373-1404. doi: 10.1080/00207540600570432

Wagner, E. L., & Newell, S. (2007). Exploring the Importance of Participation in the Post-Implementation Period of an ES Project: A Neglected Area. . *Journal of the Association for Information Systems, 8*(10), 508-524.

Wernerfelt, B. (1984). A resource-based view of the firm. *Strategic Management Journal, 5*(2), 1-180.

Wernerfelt, B. (1995). The Resource-Based View of the Firm: Ten Years After. *Strategic Management Journal, 16*, 171-174.

Willis, T. H., & Willis-Brown, A. H. (2002). Extending the value of ERP. *Industrial Management & Data Systems, 102*(1), 35-38.

Yu, L. (1996). *Guided by Learning Organization to Practice a Process Reengineering for an Enterprise Business.* Paper presented at the 1996 International System Dynamics Conference, Cambridge, Massachusetts.

Appendix A – Interview Questions

1. What is your position? Were you with the company when the ERP was implemented? If so, do you feel the ERP has been successful for the supply chain?
2. Describe the role the ERP has in the supply chain process of the organization?
3. What were your objectives, goals, and expectations in implementing the ERP? Follow up questions: functionality versus transparency – did they want both? Which is more important to your org? Which did you get?
4. Did you meet your objectives, goals, and expectations? If not, which objectives, goals, and objectives did you not meet? Why do you believe they were not met?
5. Is your organization able to use the ERP to complete all SCM tasks? If not, what workarounds have you established?
6. What has been the outcome of the implementation? Does your organization perceive the implementation as successful?
7. What is your definition of organization transparency?
8. Has the ERP provided transparency throughout the organization?
9. Did you think the ERP would provide the supply chain with improved efficiency? If so, did the ERP meet your expectations after it was successfully implemented?
10. Did you think the ERP would transform the supply chain into a process for the organization rather than a function? If so, did the ERP transform the supply chain from a functional capability into an efficient process in the organization? If not, why did it not meet your expectations?
11. Did the organization include the supply chain leadership in the ERP implementation process? If so, did this improve the transformation of the supply chain to a process within the organization rather than a function?
12. Was there interdepartmental communication and collaboration during implementation? Was there interdepartmental communication and collaboration after the ERP was successfully implemented within the organization?

Appendix B – Interview Data Summary

Interviewee	Corporation Type	Corporate Size	ERP
A	Public	18,000-20,000	SAP
B	Private	8,000-10,000	SAP
C	Private	500	Unknown (Organic)
D	Public	100,000-150,000	SAP
E	Private	10,0001+	SAP, Oracle, JD Edwards
F	Public	250,000-300,000	SAP
G	Public	40,000-50,000	SAP

The Impact of Intra-Organizational Social Networking Sites on Impression Formation

Jeff Cummings
cummingsj@uncw.edu
Dept. of Information Systems and Operations Management
University of North Carolina Wilmington
Wilmington, NC 28403, USA

Abstract

The introduction of social media has changed the way individuals communicate and collaborate both within and outside the organization. This paper examines a specific social media, Social Networking Sites. Organizational use of social networking sites (both public and enterprise) is discussed followed by a closer examination of intra-organizational social networking sites and how these sites have the potential to change the impression formation process. The results suggest that the information available through this technology can impact the impression formation process. However, these impressions may vary depending upon the information provided through an individual's online profile. Specifically, the information available can impact perception of another individual's social capital, which plays a significant role in intragroup relationship formation and performance. Implications and future directions are suggested.

Keywords: intra-organizational social networking sites, impression formation, Web 2.0, social capital

1. INTRODUCTION

Organizational use of social media technologies continues to increase. The popularity of sites such as Facebook, Twitter and Wikipedia has led many organizations to not only have a presence in the public domain but also invest in these technologies for intra-organizational use. Companies are scrambling to provide social networking capabilities within an organizational environment to meet the increasing demands of many young employees ("digital natives" (Palfrey and Gasser 2010)) who see contemporary technologies such as social networking software as necessary for a productive work life. McAfee (2006) has termed this phenomenon of moving traditionally "public" technology into the organization as Enterprise 2.0, the use of social media technologies within an organization for increased performance or benefit to the employees and business.

While social media packages often entail a range of tools, companies have tended to focus on the a few specific technologies such as wikis (e.g. a knowledge repository) or blogs (e.g. a tool for internal, company-wide communications) and strategies of bringing these into the organization (Cummings et al. 2009). One social media technology, social networking sites (SNSs), is often overlooked or undervalued due to the stigma generated by popular public sites (e.g. Facebook, Twitter, Google+, etc.) as being a socially oriented, unproductive tool. However, organizations have begun to include SNSs in their social media adoption including IBM, Dell and Cisco with many companies planning to in the future. They are finding the use of intra-organizational social networking sites has the potential to provide benefits beyond a tool traditionally thought of as means of only "socializing". The current study focuses on how intra-organizational social networking sites (IOSNS) can be beneficial for employees when

forming impressions of others. These impressions have the potential to alleviate issues often experienced with the increased emphasis on "virtual" work. For example, a common issue that could benefit from impressions formed in IOSNS is the interpersonal relationship development process of virtual teams.

Virtual teams are often formed to leverage differing backgrounds with team members having little to no prior knowledge of one another who teams may never meet in person or work together again in the future (Hung, Dennis et al. 2004). This has raised questions of how relationships form in this non-traditional work environment. Research suggests that in the development of "virtual" relationships, an initial face-to-face meeting is needed to form impressions of team members and develop rapport for effective future collaboration (Ramesh and Dennis 2002). However, the diverse (and dispersed) workforce of today's organization limits the likelihood of face to face meetings as coordination of such meetings is often prohibitive. This limitation leads to the reduction of the socio-emotional processes of relationship development among team members (Martins, Gilson et al. 2004).

The introduction of social media technologies in the enterprise, such as IOSNS, has provided a new set of resources that may change the limitation often found in virtual team relationship building. The increase of the availability of information provides individuals with significant insights about others not previously available. A typical SNS allows individuals to create an extensive "online profile" which includes, but is not limited to a profile summary (e.g. self-description), education background, experience, group affiliations, contacts/friends and the ability of contacts to post information about the profile owner (through wall posts and recommendations) (Dwyer, Hiltz et al. 2008). The abundant amount of information provided allows anyone to know more about the person than ever before.

The goal of the current paper is to provide an overview of intra-organizational social networking sites and examine the benefits of these sites on the relationship development process. Specifically, we are interested in how individuals use these technologies to supplement the traditional impression formation process used in virtual environments. In the subsequent sections, an overview of social networking sites is provided, including the difference between organizational use of public sites as well as enterprise only sites. This is followed by an examination of the traditional impression formation process and how social networking sites impact this process. Finally, results from an exploratory study examining intra-organizational social networking sites are discussed including implications to organizations and future directions.

2. SOCIAL NETWORKING SITES

SNSs have been broadly defined as "*a web-based service that allows individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system*" (boyd and Ellison 2008, p.211). Within the public domain, social network sites offer individuals an outlet to present themselves in a digital format by allowing them a means to provide details concerning themselves as well as establish/maintain their network of relationships to fellow members (Ellison, Steinfield et al. 2007). However, profile owners are not the only ones who are able to create information on their sites. Most sites (e.g. Facebook and LinkedIn) allow for the creation of information by their connections, for example, through the use of wall posts or recommendations. The primary features of these sites include the ability to connect with others, share personal information (including photos and videos), send/receive messages, provide "status" updates, post comments on a friend's site, and numerous other tools that can be tailored by the user.

Uses of public sites by individuals range from simply a tool to connect/maintain contact with friends (e.g. Facebook) to more work-related sites used for connecting to companies and professionals (e.g. LinkedIn). SNSs have gained immense popularity, often becoming integrated into our daily lives. In fact, Facebook, one of the leading SNSs, has over 1 billion monthly active users worldwide as of October 2012 (http://newsroom.fb.com/Key-Facts). Due to this immense popularity, the primary interest in social networking site research has been in the public domain, focused on popular sites such as Facebook and MySpace (Lampe, Ellison et al. 2006). However, there has been a shift with organizations focusing efforts on taking advantage of these sites for internal benefits.

There are two strategies organizations have taken to leverage social networking sites for internal purposes. One potential approach is to leverage existing, publically available SNSs (e.g. Facebook or LinkedIn) for employee interaction purposes. However, as discussed in the next section, the use of such sites have had limited success, especially when intended to build relationships internal to the organization. Alternatively, organizations can choose to implement an intra-organizational social networking site that limits use to employees only. Each strategy is discussed in the subsequent sections including reasons for using an internal site as opposed to public SNSs.

Public SNSs

A number of studies have examined the use of public SNS as a resource to maintain a company network and build an internal network of relationships. Because of their immense popularity within the public realm, two of the more commonly used sites by organizations are Facebook and LinkedIn. Both sites offer functionality such as Groups (in Facebook) and Company Sites (in LinkedIn) that allows employees to join and maintain connections with fellow employees throughout the organization. However, studies show mixed results for using these to build internal relationship. Skeels et al. (2009) found employees rank internal networking lowest for usefulness of such sites. Instead, many employees engage in these sites (e.g. LinkedIn) as a tool for maintaining relationships with former colleagues or external clients.

Additionally, these sites are not being used for their intended purposes. In a study examining Facebook use within IBM, DiMicco and Millen (2007) found most employees who joined the Facebook IBM network were not using the SNS as way of building internal, company relationships. Instead, a majority of users were categorized as "Reliving the College Days," socially using Facebook as a tool to maintain personal relationships. This study points to one of the pitfalls of using these sites for internal, organizational uses: making a distinction between personal and professional use. Thus, while public sites may be useful from an external relationship building perspective, these sites are not being used to build a network of relationships at the company where the user is employed.

Enterprise SNSs

As previously mentioned, the reach of social networking sites is beginning to gain ground in the organizational domain. Forrester Research estimates spending on enterprise social media to reach $4.6 billion by 2013 (Young, Brown et al. 2008). This has driven many companies to provide social networking software specifically designed for enterprise use. Microsoft (Sharepoint), Cisco (Quad) and IBM (LotusConnections) are just a few of the companies now offering intra-organization social networking sites (IOSNSs) as part of their enterprise social media packages. This increase in IOSNSs is due to the emphasis on the adoption and usage of enterprise social media (c.f. Young, Brown et al. 2008). Organizations are finding the utility that social and interactive technologies bring through sharing of expertise and employee support.

The primary benefit of these sites over publically available SNSs is the increased creation and maintenance of organizational relationships (Dwyer, Hilts et al. 2008). Employees have appropriated these technologies to not only interact with known employees but to search and create connections with new ones (DiMicco, Geyer et al. 2009). DiMicco et al (2008) has examined the types of users within IOSNSs suggesting that many employees use these sites as a self-promotion outlet for advancement or for campaigning for new ideas. They also suggest that employees are using the site for "social browsing" (i.e. discovering and connecting with employees they may not know). While it is known that employees are using these sites to "socially" understand others, little is known about how employees use this information when forming impressions.

As more organizations explore the use of IOSNSs, the amount of information made available to users and how that information is evaluated should be explored further. Large organizations (e.g. IBM, Deloitte, Best Buy, etc.) are already promoting the use of IOSNSs, allowing employees to provide information not traditionally found within a company. For example, IBM launched their internal SNS, BeeHive, as a tool to find, collaborate and maintain contact with fellow employees from around the world (DiMicco and Millen 2008). The popularity of IOSNS has led IBM to create LotusConnections which has been made available to any organization wishing to

implement an IOSNS. The primary reason these sites enable collaboration and collection is the ability to share a variety of information not traditionally found on company intranets of the past.

One might expect that the information employees provide would be limited in an organizational setting (e.g. omitting personal information). However, DiMicco et al (2008) found that employees actually supplied more personal information on IOSNS than on public SNS. They found that users of IOSNS felt that the information was more secure compared to public SNS. Furthermore, because profiles serve as a mechanism for one's social self, people consciously put effort toward crafting a presentation that is aimed at influencing others within the network (Dwyer, Hiltz et al. 2008). This emphasizes a distinct characteristic entering the organizational landscape where both professional as well as personal information is now being provided. Thus, IOSNS provide an outlet for information that goes beyond that traditionally found in organizational technologies (i.e. the type of information you would find through the intranet or email system about fellow colleagues).

The information now available to individuals in organizations allows additional insights into coworkers previously unavailable. Because SNSs originally started in the public domain, IOSNSs are often based upon their public counterparts, providing information such as past experience, education, location, and personal characteristics similar to those found in sites like LinkedIn. Many IOSNSs allow additional information beyond work life, encouraging individuals to list hobbies, post photos, and interact with other individuals within the network. Additionally, these sites allow individuals the ability to manage relationships with coworkers. IOSNSs incorporate new aspects to relationships management such as providing visual components (e.g. pictures of contacts), enabling active engagement (e.g. status of contacts) and maintaining fringe relationships (e.g. staying in contact with former project team members) (Shih 2009).

Previous research tends to focus on the potentially positive implications of social media use in organizations (McAfee 2006). While there are definite benefits to incorporating social media into the organization, little is known concerning how the amount of information now publically available can impact perceptions of others during the impression formation process. In the following section, a closer examination of the implications an IOSNS may have on impressions is conducted using virtual teams as an example.

3. IOSNS AND IMPRESSION FORMATION

Impressions can affect numerous facets of an individual's career from the initial job interview to subsequent career stages such as advancement, project decisions and organizational citizenship behaviors (c.f. Bolino, Kacmar et al. 2008). An initial impression sets the stage for future interpersonal interaction by signaling a number of factors, among them trustworthiness (Donath 2007). More importantly, these initial impressions have been found to have an anchoring effect on individuals in which they are less likely to search for disconfirming information during future observations and interactions (Good and Gambetti 2000). Thus, impression formation becomes an important component within team interaction especially when these interactions occur in a dispersed, virtual environment.

Impression formation is referred to as an interpersonal process occurring as an individual uses the information available to them to make general judgments concerning another individual's personal characteristics (Switzer 2008). Traditionally, individuals form impressions through an initial, face-to-face (FtF) encounter occurring prior to the formation of a project team. From these direct encounters, individuals are able to interpret "signals" into attributes of the person they will interact with in the future (Donath 2007). These interpretations often set the basis for future team interactions, affecting team cohesiveness, the decision making process and the overall success of the project or task undertaken (Tidwell and Walther 2002).

The abundant signals available in FtF are reduced dramatically as these initial, FtF interactions are replaced by a mediated setting. Computer mediated communication has become commonplace in today's business environment for their usefulness in reducing cost and time but at the expense of initial face to face interaction that help with impression formation of fellow team members. Nevertheless, individuals do form a consistent impression based on whatever information is available to

them at that time (Walther, Anderson et al. 1994). There is still a cognitive need for individuals to form an impression despite being in a mediated setting (Walther 2005). Thus, when faced with a mediated environment, individuals rely on alternative approaches, leveraging available signals when assessing the interpersonal characteristics of their communication partners.

Impression Formation and Computer Mediated Communication

As organizational dependence on Computer Mediated Communication (CMC) continues to increase, the need to understand how individuals process information presented electronically has become important. Individuals have an inherent, cognitive need to form an impression of others, with impression formation in CMC being no exception in which individuals use any type of information source to form an initial impression (Walther 2005). Early research examining social behavior attempted to explain impression formation through the amount of cues (social) available given a specific communication medium (Short et al 1976). Social information processing argues that those media low in bandwidth (i.e. media with limited ability to provide rich cues) leads to low social presence causing the development of relationships to be stifled.

Alternatively, Walther (1992) takes a different perspective by examining how relationships build over time through continued group interaction. He suggested that individuals pursue the need for an interpersonal connection with their team and as time progresses the effects of diminished relational communication through CMC is reduced. Thus, individuals using CMC will compensate for the lack of traditional cues by examining the cues available in the medium being used (Tidwell and Walther 2002). For example, an individual using email may use spelling ability as a cue to assess the cognitive ability of their communication partner. This research, like most to date, focuses on relationship building as a process-level behavior occurring overtime, often comparing online to the offline relationship building processes (Ramirez et al 2006). Limited attention has been placed on how individuals process social information.

With the advent of socially enabled technologies, individuals no longer have to rely solely on interpretation of cues available through traditional CMC (e.g. email) to garner impressions of others. SNSs have enabled users to construct more detailed impressions that go beyond examining (if available) personal web pages or the common practice of "googling" an individual to find out more details (Tong, Van Der Heide et al. 2008). Research has shown this to be true as individuals use SNSs to acquire interpersonal impressions of not only acquainted but unacquainted targets as well (Walther, Van Der Heide et al. 2008).

Users treat profile information provided on these sites in the same manner as cues obtained through ongoing interactions; using this during the interpersonal relationship formation process (Lampe, Ellison et al. 2007). Thus, the same principle exists in IOSNS where information provides a signal about the individual being judged. When actual performance measures are absent, people will rely on available information that may be imperfectly correlated to features (e.g. social capital) that signal future performance within the team (Donath 2007). For example, an individual cannot know for sure if he/she will trust another team member he/she has yet to interact with. That individual relies on signals available through the profile information of the future counterpart to form judgments about trust. These signals may be in the form of a former or current coworker's comments about this individual or other general profile information signaling that this person can be trusted.

4. EXPLORATORY STUDY & HYPOTHESES

To understand if information available in IOSNS can impact impression formation, an exploratory study was conducted manipulating the amount of information available to see if this effects impressions. That is, would the amount of profile information available (increasing cues) impact the individual impressions of profile owners? The impressions that were examined in this study were based on perceptions of social capital, a commonly cited construct related to virtual team success.

Social capital is defined as those resources (actual and potential) derived through an individual or social unit's network of relationships, comprising both the network and assets that can be mobilized through that network (Burt 1992; Nahapiet and Ghoshal 1998). Adler & Kwon (2002:93) extend this

definition by stating that "social capital is a resource for individual and collective actors created by the configuration and content of the network of their more or less durable social relations."

Within organizational literature, social capital continues to gain popularity as a way of capturing specific social elements and their contributions within various contexts, both individual and collectively as a team (Adler and Kwon 2002). At the individual level, social capital has been examined as a way to facilitate a person's actions and reflect their access to a variety of networking resources (Coleman 1990). Wasko & Faraj (2005) show individuals and relationships with others are a primary source of social capital, influencing how individuals behave in a collective to promote both creation and contribution of knowledge within their community.

The underlying argument surrounding social capital is the idea that social ties of one kind may be leveraged or used for different purposes (e.g. work related objectives) (Adler and Kwon 2002). Social capital has been linked to numerous benefits including increased information exchange, product innovation, cooperative behavior, knowledge contribution and team effectiveness (Coleman 1990; Adler and Kwon 2002; Wasko and Faraj 2005). Researchers have argued that social networking sites are used for creating and maintaining social relationships within their social community or unit (Dwyer, Hiltz et al. 2008). Thus, while users may not directly intend an IOSNS profile for this purpose, the use of these sites can be seen as a tool for displaying and/or influencing impressions of social capital. Within the current study's context, the focus is concerned with an individual's impression of the social capital dimensions of another. Social capital can be examined across the dimensions of relational (i.e. trust and identification), cognitive (shared meaning) and structural (connectedness).

Given our previous discussion on cues, increasing the amount of information present in a profile (i.e. cues) should elicit a stronger impression of an individual's social capital. Therefore,

H1: As the amount of profile information made available increases, impressions of social capital will significantly increase across all dimensions.

Additionally, the type of information available in the profiles will also have varying impact on these impressions. For example, information concerning educational background or hometown may have a greater impact on identification or trust when compared to the number of connections which would impact impressions of connectedness. Thus, the type of information available will also impact impressions.

H2: Education information in the profile will significantly impact impressions of relational social capital more than structural or cognitive capital.

H3: Connection information in the profile will significantly impact impressions of structural social capital more than relational or cognitive capital.

H4: Experience information in the profile will significantly impact impressions of cognitive social capital more than cognitive or structural capital

5. RESEARCH METHODOLOGY AND RESULTS

An empirical study was conducted to examine the impact of IOSNS information on impressions of social capital. Data was collected using an experimental lab setting at a large, state university with participants drawn from an undergraduate business course. 340 participants were used. Participants were placed in a hypothetical situation (via a vignette) in which they were to evaluate members of a future virtual team based upon the information available in an IOSNS. Profiles were created online that would be similar to the type and way information would be displayed within a social networking site. The goal of creating online profiles was to simulate an environment similar to an internal social networking site. Furthermore, to incorporate a familiar work environment that students may have previously been associated with, the setting of the team was an internship which many students participating had prior experience in this setting.

There were two different manipulations of profiles. The first manipulation (assessing hypothesis 1) had two different profiles: one contained a minimal amount of information consisting of basic demographics while another contained the same basic demographics with the addition of information about education,

experience and connections. The second manipulation (assessing hypotheses 2-4) had three different profiles, one with demographic/education information (relational), one with demographic/experience (cognitive) and one with demographic/connections (structural). These were presented in an social network environment to simulate similar information presentation, layout and emphasis on keywords that are readily picked up by users.

Vignettes were chosen to place all subjects in the same scenario with the only change being the manipulation of the data presented via the IOSNS. This provides increased control over the information presented (i.e. manipulation of IOSNS data) to ensure that the participants' judgments or perceptions (i.e. social capital dimensions) are less biased or contaminated as may be the case in traditional experimental settings (Greenberg and Eskew 1993).

Measures

Relational capital was assessed based on prior scales, adapted for the current study to assess both individual trust in others (Jarvenpaa and Leidner 1999) and identification with their team members (Brown, Condor et al. 1986; Henry, Arrow et al. 1999). Both are based on established scales and were measured as one variable.

For cognitive social capital, a scale was adapted to measure relevant expertise, knowledge expectations and values. Based on Wasko and Faraj's (2005) operationalization of cognitive capital, items were adapted to asses an individual's tenure in the field and expertise in the given situation of the experimental task (Kirsch, Ko et al. 2009). Other items for cognitive capital were developed based on the conceptual definition and scales developed in prior questionnaires to measure shared meaning and values (Nahapiet and Ghoshal 1998).

Structural capital items were developed using both a prior scale that assessed overall structural social capital within a team (van den Hooff and Huysman 2009) and the conceptual definition of structural capital (Nahapiet and Ghoshal 1998)

All measures were captured using a seven point Likert scale of 1 (strongly disagree) to 7 (strongly agree). Table 1 below provides the correlation matrix including means and standard deviations. Table 2 provides the general demographics of participants. Trustworthiness, age and gender were used as controls across all manipulations. Each of the controls were found to be insignificant for the manipulation used in this experiment.

Table 1. Construct Descriptive Statistics and Correlations

	Mean	Std Dev	SC	CC	RC
Structural Capital	4.65	1.22	-		
Cognitive Capital	4.89	1.06	.314**	-	
Relational Capital	4.54	1.06	.342**	.553**	-

** Correlation is significant at the 0.01 level (2-tailed).

Table 2. General Demographics and Usage Information

Gender	60 % Male 40 % Female
Age	20 yrs
Avg. Length of Computer Use	10.5 yrs
Avg. Length of SNS Use	6.5 yrs

6. FINDINGS

Multivariate analysis of variance (MANOVA) was used to first compare the impact of the amount of information on social capital impressions (H1) followed by mean comparisons to evaluate the impact of the type of information (H2, H3, H4).

Hypothesis 1 states that impressions of social capital will increase (stronger perceptions of relational, structural and cognitive dimensions) as the amount of information provided in profiles increase. The results indicate that there are significant differences for structural, cognitive and relational social capital perceptions as information increases (Wilks' λ=0.56, $F(6,176) = 9.90$, $p < 0.01$). This provides partial support for Hypothesis 1, however to understand these differences, follow-up ANOVAs were conducted to understand the mean difference across these groups (amount of information). Table 3 contains the Mean Square and F-Ratios from the follow-up ANOVAs. From these results, we see that each social capital dimension (F-Ratios) was significant. Thus, Hypothesis 1 is supported.

Table 3. ANOVA Results

Relational Capital		*Cognitive Capital*		*Structural Capital*	
MS	*F*	*MS*	*F*	*MS*	*F*
13.55	**17.29****	1.88	**3.15***	20.07	**20.85****

* p < 0.05, ** p < 0.01.

Hypotheses 2-4 examine the type of information presented and the impact to impressions of specific social capital dimensions. For each social capital dimension manipulated, measures for all dimensions were captured (i.e. when structural was manipulated, relational and cognitive constructs were also measured). Table 4 provides the means across manipulated dimensions.

Table 4. Means and Standard Deviations Across Manipulated Dimensions

Relational Social Capital[1]		
Relational	Cognitive	Structural
4.84 (0.99)	4.10 (0.98)	4.27 (1.1.6)

Cognitive Social Capital[1]		
Relational	**Cognitive**	Structural
4.95 (0.84)	**5.37 (0.86)**	4.48 (1.11)

Structural Social Capital[1]		
Relational	Cognitive	**Structural**
3.82 (0.96)	4.24 (1.01)	**5.21 (1.16)**

[1] These represent the manipulated dimension (i.e. the profile information that is changed to induce perceptions of these dimensions).

Note: Below each manipulated dimension are the scores for each construct measured in the model.

Mean comparisons suggest that the manipulated features were significantly different (and higher) for the manipulated dimension compared to the other dimensions. For example, the manipulation for structural social capital resulted in a mean of 5.21 for the structural capital. This was significantly greater (i.e. pairwise comparisons) than the mean for the resulting relational and cognitive constructs when structural social capital is manipulated in the profile. Thus, Hypotheses 2-4 were supported.

7. CONCLUSIONS

The study presented here shows the impact information available in a SNS can impact factors important in an organizational setting (i.e. social capital impressions of team members). While this only focused on a few specific SNS features, it provides preliminary results which suggest that specific information found in profiles can have a direct impact on impression formation. More studies are needed to understand the how these impressions may change overtime and extent they may have on future interactions with team members. However, some implications for management and employees can be drawn from the previous discussion of enterprise social networking sites and these preliminary results.

First, in general, organizations should be aware that employees are indeed engaging in these technologies when made available in the organizations and are using them as a way of presenting themselves in the best light possible. This is based on numerous studies evaluating SNS in organizations (DiMicco and Millen 2008, Shih 2009). A majority of the information located on these sites is self-generated meaning the profile owner creates and maintains their own profile information. Management should be aware that these practices are occurring and understand the validity (or potentially, lack thereof) of information provided by the employee.

Second, the results of our study show that individuals can and do use information available in profiles to form impressions of factors that could impact team and organizational performance. This could have both positive and negative implications. From a positive perspective, SNSs appear to provide a bridge to building interpersonal relationships with employees who may be dispersed or virtual. This overcomes many of the limitations previously seen with CMC. However, these impressions are not formed overtime through interaction. Individuals may form impressions based purely on information provided in a SNS profile. As previously mentioned this information is self-generated and may not be a true representation of the employee. Thus, employees using these systems need to be

aware of both the advantages and disadvantages accompanying SNS use.

Finally, the impact of intra-organizational social networking sites is still being researched. To date, most studies have focused on how users participate in these sites without considering how they use this information to make judgments. The current study provides initial insights into how this information can be used to create impressions. This study presents a snapshot in time (i.e. the initial formation of a virtual team). Prior research suggests the potential for anchoring effects to occur during the impression formation process including formation within an SNS (Ellison et. al 2007). While some research suggests that individuals are less likely to search out disconfirming information depending upon how the impression is formed (Petty and Cacioppo 1986), additional research is needed to understand the ongoing impact of these impressions especially in a technology mediated environment. Furthermore, research is needed to understand the impacts these site may have on the productivity and collaboration of employees.

8. ACKNOWLEDGEMENTS

I would like to thank Alan Dennis for his continual support throughout this research including all the advice and direction he has provided.

9. REFERENCES

Adler, P. S., & Kwon, S.-W. (2002). Social Capital: Prospects for a New Concept. *The Academy of Management Review, 27*, 17.

Bolino, M. C., Kacmar, K. M., Turnley, W. H., & Gilstrap, J. B. (2008). A Multi-Level Review of Impression Management Motives and Behaviors. *Journal of Management, 34*(6), 1080-1109.

boyd, d., & Ellison, N. (2007). Social Network Sites: Definition, History, and Scholarship. *Journal of Computer-Mediated Communication, 13*(1), 210-230.

Brown, R., Condor, S., Mathews, A., Wade, G., & Williams, J. (1986). Explaining intergroup differentiation in an industrial organization. [Article]. *Journal of Occupational Psychology, 59*(4), 273-286.

Burt, R. (1992). *Structural Holes: The Social Structure of Competition*. Cambridge, MA: Harvard University Press.

Coleman, J. (1990). *Foundations of social theory*. Cambridge, Mass.: Belknap Press of Harvard University Press.

Cummings, J., Massey, A.P. & Ramesh, V. (2009) "Web 2.0 Proclivity: Understanding How Personal Use Influences Organizational Adoption." Proceedings of the 27th ACM International Conference on Design of Communication, Bloomington, IN.

DiMicco, J. M., & Millen, D. R. (2007). *Identity management: multiple presentations of self in facebook*. Paper presented at the Proceedings of the 2007 international ACM conference on Supporting group work, Sanibel Island, Florida, USA.

DiMicco, J., Millen, D. R., Geyer, W., Dugan, C., Brownholtz, B., & Muller, M. (2008). *Motivations for social networking at work*. Paper presented at the Proceedings of the 2008 ACM conference on Computer supported cooperative work, San Diego, CA, USA.

DiMicco, J. M., Geyer, W., Millen, D. R., Dugan, C., & Brownholtz, B. (2009). *People Sensemaking and Relationship Building on an Enterprise Social Network Site.* Paper presented at the System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on.

Donath, J. (2007). Signals, cues and meaning *Signals, Truth and Design*. Cambridge, MA: MIT Press.

Dwyer, C., Hiltz, S. R., & Widmeyer, G. (2008). *Understanding Development and Usage of Social Networking Sites: The Social Software Performance Model*. Paper presented at the Proceedings of the Proceedings of the 41st Annual Hawaii International Conference on System Sciences.

Ellison, N. B., Steinfield, C., & Lampe, C. (2007). The Benefits of Facebook "Friends:" Social Capital and College Students' Use of Online Social Network Sites. *Journal of Computer-Mediated Communication, 12*, 1143-1168.

Greenberg, J., & Eskew, D. E. (1993). The Role of Role Playing in Organizational Research. [Article]. *Journal of Management, 19*(2), 221.

Good, D., & Gambetti, D. (2000). Individuals, Interpersonal Relations and Trust. In D. Gambetti (Ed.), *Trust: Making and Breaking Cooperative Relationships* (pp. 31-48). New York: Blackwell.

Henry, K. B., Arrow, H., & Carini, B. (1999). A Tripartite Model of Group Identification. *Small Group Research, 30*(5), 558-581.

Hung, C., Dennis, A. R., & Robert, L. P. (2004). *Trust in Virtual Teams: Towards an Integrative Model of Trust Formation.* Paper presented at the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04)

Jarvenpaa, S. L., Knoll, K., & Leidner, D. E. (1998). Is Anybody Out There? *Journal of Management Information Systems, 14*(4), 29-64.

Kirsch, L. J., Ko, D.-G., & Haney, M. H. (2009). Investigating the Antecedents of Team-Based Clan Control: Adding Social Capital as a Predictor. *Organization Science, 21*, 469-489.

Lampe, C., Ellison, N. B., & Steinfield, C. (2008). *Changes in use and perception of facebook*. Paper presented at the Proceedings of the 2008 ACM conference on Computer supported cooperative work, San Diego, CA, USA.

Martins, L., Gilson, L., & Maynard, M. T. (2004). Virtual Teams: What Do We Know and Where Do We Go From Here? *Journal of Management, 30*(6), 805-835.

McAfee, A. P. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. *MIT Sloan Management Review 47*(3), 21-28.

Nahapiet, J., & Ghoshal, S. (1998). Social Capital, Intellectual Capital, and the Organizational Advantage. *Academy of Management Review, 23*, 242-266.

Palfrey, J., & Gasser, U. (2010). *Born Digital: Understanding the First Generation of Digital Natives*: Basic Books.

Petty, R. E. and J. T. Cacioppo (1986). Elaboration Likelihood Model of Persuasion. *Advances in Experimental Social Psychology* 19, 123-181.

Ramesh, V., & Dennis, A. (2002). *The Objectoriented Team: Lessons for Virtual Teams from Global Software Development*. Paper presented at the Proceedings of the Thirty-Fifth Annual Hawaii International Conference on System Sciences, Hawaii.

Ramirez, A. (2007). The Effect of Anticipated Future Interaction and Initial Impression Valence on Relational Communication in Computer-Mediated Interaction. *Communication Studies, 58*(1), 53-70.

Skeels, M. M., & Grudin, J. (2009). *When social networks cross boundaries: a case study of workplace use of facebook and linkedin*. Paper presented at the Proceedings of the ACM 2009 international conference on Supporting group work, Sanibel Island, Florida, USA.

Switzer, J. (2008). Impression Formation in Computer Mediated Communities & Making a Good (Virtual) Impression In P. Zemliansky & K. Amant (Eds.), *Handbook of research on virtual workplaces and the new nature of business practices*. Hershey, PA: Information Science Reference.

Tidwell, L. C., & Walther, J. B. (2002). Computer-Mediated Communication Effects on Disclosure, Impressions, and Interpersonal Evaluations: Getting to Know One Another a Bit at a Time. *Human Communication Research, 28*(3), 317-348.

Tong, S. T., Van Der Heide, B., Langwell, L., & Walther, J. B. (2008). Too Much of a Good Thing? The Relationship Between Number of Friends and Interpersonal Impressions on Facebook. [Article]. *Journal of Computer-Mediated Communication, 13*(3), 531-549.

van den Hooff, B., de Ridder, J., & Aukema, E. (2004). Exploring the eagernes to share knowledge: the role of social capital and ICT in knowledge sharing. In M. Huysman & V. Wulf (Eds.), *Social capital and information technology*. Boston: MIT Press.

Walther, J. B. (1992). Interpersonal Effects in Computer-Mediated Interaction - a Relational Perspective. *Communication Research, 19*(1), 52-90.

Walther, J. B., Anderson, J. F., & Park, D. W. (1994). Interpersonal Effects in Computer-Mediated Interaction - a Metaanalysis of Social and Antisocial Communication. *Communication Research, 21*(4), 460-487.

Walther, J. (2005). *The rules of virtual groups.* Paper presented at the Proceedings of the 38th Hawaii International Conference on System Sciences, Hawaii.

Walther, J., Van Der Heide, B., Kim, S., Westerman, D., & Tong, S. (2008). The Role of Friends' Appearance and Behavior on Evaluations of Individuals on Facebook: Are We Known by the Company We Keep? *Human Communication Research, 34*(1), 28-49.

Wasko, M., & Faraj, S. (2005). Why Should I Share? Examining Social Capital and Knowledge Contribution in Electronic Networks of Practice. *MIS Quarterly, 29*, 35-57.

Young, G. O., Brown, E. G., Keitt, T., Owyang, J. K., Koplowitz, R., & Shey, H. (2008). Global Enterprise Web 2.0 Market Forecast: 2007 To 2013. *Forrester Research*. Retrieved from Forrester website: www.forrester.com.