

# Social Proximity Networks on Cruise Ships

Guang Yang, Zhigang Liu, Karim Seada, Hawk-Yin Pang, August Joki, Jun Yang, Daniela Rosner, Manish Anand, Péter Pál Boda<sup>§</sup>

## ABSTRACT

Social interaction is an essential element of our daily lives. One could argue that such interaction is even more important while on vacation. To showcase certain technology enablers, we implement a cruise ship scenario with a few advanced applications. The cruise ship scenario serves not only as a tangible goal but also as a metaphor: these applications can be adapted easily to other environments such as office, classroom, conference, exhibition, museum, malls, etc. Cruise ships represent a unique environment for social life; passengers live in the same time and space and attend the same social activities together, with frequent physical encounters. They also produce/consume large amounts of media content and heavily interact with each other. To meet people's social networking needs we propose a new paradigm called Social Proximity Network (SPN). SPN applications are built on our connectivity and indoor positioning infrastructure, as well as on advanced device-based utilities. By relying on the sensing power of today's mobile devices and mashing up digital content with physical context, SPN services are able to provide rich and unique experiences to cruise passengers, both during and after the trip.

## 1. INTRODUCTION

Cruise vacations are a popular choice for many people with different default social relations (families, couples, singles, friends, colleagues). The length of stay can vary from a few days to several weeks. In a rather closed setting, a cruise trip provides a unique environment and experience. It is a society in miniature in which life is much condensed – dining, shopping, sightseeing, socializing, entertainments, sports and romance – with all the fun moments you have long dreamed for and none of the dullness.

Technology has evolved such that social network services are an integral part of our everyday lives, and cruise trips ought to be no exception. However, the way social networks work on a cruise ship should be fundamentally different from what we see on the Internet today. In one respect, cruise passengers are going to share the space on the ship and attend the same events such as a cocktail party; their cyber-social activities are more tied to the physical interactions. In another respect, instead of spending time staring at computer displays, passengers may be moving around; mobile devices and wireless connectivity are at the center stage of the services. Therefore, although the investment one has made in building circles of friends and sharing media content on social networking sites is very valuable, bringing them simply “as-is” to the cruiser scenario is not the best solution.

In this paper we propose a set of Social Proximity Network (SPN) services developed for the social needs of cruise passengers – a community of people with common interest in a relatively closed environment. Our approach is mobile centric. The goal is to use a mobile device as a phone, browser, camera, music player, game

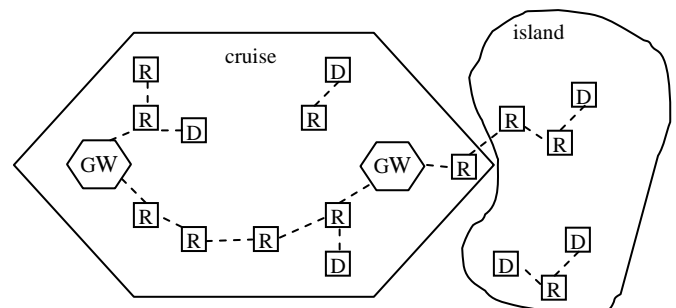
console, location detector, sensor, diary book, ID – the only digital device needed for the trip. To achieve this the following issues are to be addressed: 1) freeing users from wireless network coverage problems; 2) providing robust indoor positioning; 3) enhancing one's social life by mashing up digital content with physical context; 4) utilizing sensing power on devices; and 5) preserving privacy and security in content sharing. Although we present the utilization of these applications in a cruise ship scenario, we feel that SPN services can be easily generalized and applied to conferences and sports events, among others.

The rest of the paper is organized as follows. Section 2 describes the connectivity and positioning infrastructure in the system, as well as software architecture. Section 3 presents several showcase services, i.e. mapping and messaging, content sharing, WebCall, Digital Montage and Reflective Media, followed by related work in Section 4. Section 5 concludes the paper.

## 2. INFRASTRUCTURE

### 2.1 Connectivity

Today's passengers cannot afford to stay disconnected for a long time, and cruises aware of the modern traveler's needs are trying to accommodate that. In addition, local connectivity can support activities within the cruiser and provide the base for the new experiences to be shown later in this paper.



**Figure 1: The communication architecture consists of gateways (GW) and devices acting as relays (R) that transfer packets and provide access to other end-devices (D).**

Regular cellular coverage cannot always be expected onboard as cruise ships wander away from ports. Current typical solutions are to provide “Internet Cafes” with terminals connected to Internet via a satellite link. A step further is providing Wi-Fi hotspots at some locations. Going beyond a few hotspots and providing coverage for the whole ship is challenging for several reasons: the structure of the ship makes cabling and wiring difficult and costly; wireless radio environment is also rough due to the unique ship architecture and operation. In short, conventional communication solutions do not necessarily work well for a cruise ship.

<sup>§</sup> Nokia Research Center. 955 Page Mill Road, Palo Alto, CA, USA. +1-650-496-4400. {guang.g.yang, zhigang.c.liu, karim.seada, hawk-yin.pang, august.joki, jun.8.yang, daniela.rosner, manish.anand, peter.boda}@nokia.com

Our approach is to use passengers' mobile devices as relays to extend the coverage beyond access points. This is technically related to the work on wireless ad-hoc and mesh networks. The idea is to make mobile devices form a multi-hop network and become "moving access points" that relay packets and provide access to other devices not in the range of any real access points.

Figure 1 shows the architecture consisting of gateways (GW), relays (R) and end-devices (D). Relays are also end-devices in addition to acting as relays. The gateways connect to both the infrastructure and the ad-hoc networks and transfer packets in between. The relays belong to the ad-hoc network; they transfer packets between other relays and the gateway and act as access points to other end-devices. For an end-device to become a relay, it needs to install special software provided by our system. We do not expect all devices to become relays, but it is beneficial for the cruise to provide incentives for as many devices to act as relays.

Relays are not only for Internet access, but also to provide access to local servers and even form pure device-to-device networks when the rest of the network is not reachable. For example, our architecture will still allow devices nearby to communicate locally even when they are not connected to any gateways or access points. This kind of local ad-hoc connectivity is also valuable off the ship, e.g. when passengers visit islands or ports temporarily with no convenient Internet access – people can stay connected to one another until they get back to the ship.

## 2.2 Indoor Positioning

Location based services are emerging rapidly due to availability of radio infrastructures, e.g. GPS, GSM, Wi-Fi, etc., as well as the increasing computing power of portable mobile devices. These technologies allow for various applications and services to be built such as guide and tracking systems to location specific advertising. However, some of the radio systems for indoor location based systems are limited due to poor indoor coverage, granularity in terms of location accuracy and widely enabled radios in mobile devices. One method for indoor location is radio tagging, where short range radio can be used as location beacons.

It is predicted that 80% of mobile devices in 2009 will have Bluetooth [2], an enabler for indoor positioning together with the existence of low cost Bluetooth tags as location beacons up to 10 meters. However, for stricter control of radio propagation confinement from the Bluetooth tags for particular services that require finer location granularity Bluetooth discovery in public space, room level or even object level, an ability to adjust the Bluetooth tags' transmitted power level is needed. Such off-the-shelf products are already available on the market[4].

One main disadvantage of Bluetooth tagging (without AC power) is the battery life of the tags, only lasting from days to weeks. To achieve low maintenance of the indoor location system, Bluetooth power consumption needs to be significantly better. It is anticipated that in the near future ultra low power (ULP) Bluetooth will be available. ULP Bluetooth in many cases makes it possible to operate low cost sensor-type devices, namely radio tagging, for more than a year without recharging. Hence, future standalone ULP Bluetooth as tags will be able to operate without intervention for long periods of time.

## 2.3 Software Architecture

Our SPN software system architecture is client-server based, with peer-to-peer (P2P) adopted when connectivity to the server is unavailable, P2P is more efficient and/or is preferred by the user for privacy concerns. Figure 2 shows the overall software system architecture. Below are a few rationales behind our approach:

- The client-server architecture meets the basic requirements of authentication and authorization. A centralized repository also simplifies contextual data mashup, new service deployment and universal user access anytime from any device.
- Easy "plug-ins" of third party services, e.g. advertisement or feeds to/from other social websites. The SPN server is the center of intelligence and handles data filtering/aggregation.
- Persistent data storage. This is particularly valuable as a user may easily lose or damage a mobile device in the environment of a cruise ship.

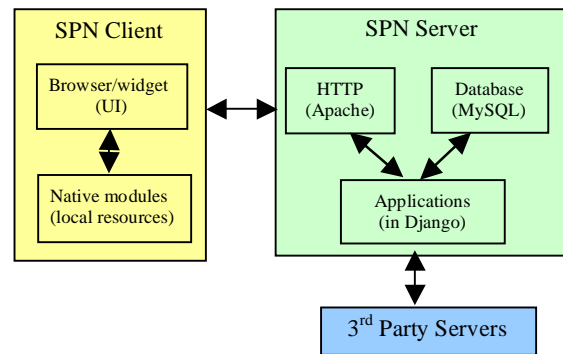


Figure 2: SPN system architecture.

Server clustering and load balancing can be applied to scale the architecture. Trends indicate that newest cruise ships are taking nearly as many passengers as inhabitants in smaller cities [13]. Our software architecture should be able to handle this load with good system performance. The server software is implemented in Linux; the client is implemented as a combination of the web browser, widgets and OS native modules in Python and C++.

There are also application-specific server modules not shown in Figure 2. For instance, we choose to use VoIP for external and internal communications between cruise passengers, as we cannot rely on regular cellular coverage. We plan to run our own local servers, e.g. SIP [10] and XMPP [11], such that communications are provided even when the Internet is not reachable. SIP and XMPP clients are available for most common desktop and mobile device-based operating systems.

## 3. SERVICES

### 3.1 Mapping and Messaging

As in any tourism activities, mapping and messaging are two fundamental day-to-day services. The map service allows a user to easily find locations of herself, family/friends, and points of interest (POI). The service includes both indoor (e.g. on the ship) and outdoor (e.g. an island tour) services. Although the concept is straightforward, its realization is not, particularly an indoor positioning mechanism that is precise and fast at the scale of 1000 feet long and 100 feet high. Another challenge is a good user

interface given the small screens on mobile devices – not only the map itself needs to be presented, but also contextual information. Filtering out irrelevant and less interesting data is a key first step.

Our main idea in the messaging is location-based services similar to [5][8][9], while focusing on cruise ships. A user can attach a message to a specific location, e.g. a cabin, with the help of the indoor positioning system. The intended recipient will receive the message on her mobile device when coming to the vicinity of that location. Location-based messaging can be triggered in a number of ways, e.g. by remote and/or virtual objects, through on-device sensors such as the accelerometer and compass. This not only provides a new way of communication and information retrieval, but also serves as a social gaming platform. Messages can be further filtered with contextual information such as time and social relationship. All these features are under implementation on top of the software architecture aforementioned.

### 3.2 Proximity Content Sharing

Content sharing is the keystone of the Web 2.0 mansion. All social network services provide sharing, usually among a group of users under certain social relationships, which are often defined in a separate and asynchronous step. For example, one may create a list of friends on a social site and later upload and share pictures with them.

Proximity content sharing is very different in two ways. First, the social relationship among sharing parties is often as ad-hoc as “people in the same room with me right now”. This type of encounter-based relationship is dynamic, temporary and hard to anticipate in advance. Another difference is that in proximity sharing, content is often stored on mobile devices rather than on dedicated servers. Asking users to upload content before sharing is an inconvenient step that will turn many away.

We believe proximity sharing must be simple and fast – people may share directly from their mobile devices. They should also have control on what and to whom to share. Our solution is a simple sequence of the following steps, in a hypothetical scenario where Alice meets Bob and wants to share a picture with him:

- Alice browses and selects content to be shared on her device;
- Alice scans Bluetooth devices nearby and finds Bob’s;
- Alice sends Bob via Bluetooth an auto-generated text message invitation with a link to the shared content on her device;
- Bob receives the message and clicks on the link, launching the web browser to connect to Alice’s device;
- Alice’s device, running PAMP [12], serves content to Bob’s.



Figure 3: Screenshots of Proximity Sharing.

The mobile web server and PHP in PAMP are used to serve content via HTTP to a web browser, widely available on mobiles.

The link generated by Alice contains a unique authentication component that can be valid for one time only, a period of time, so on, which is managed using MySQL.

Figure 3 shows several screenshots of the proximity sharing client prototype for Symbian S60. The leftmost is from Alice’s device where she can choose to share via a hotspot or an ad-hoc network. In the middle is the “invite” screen showing Bluetooth scanning, also from Alice. The rightmost is from Bob’s device, showing the text message he just received from Alice via Bluetooth.

### 3.3 WebCall

Often the intent of a phone call is to find out a specific piece of information. For example, calling a co-worker to find out when they have time for a meeting, or calling a friend to find out their locations at this moment, or calling a loved one to chat about the weather when he/she is on a trip. Mobile devices are also more often used for access to the Internet, and with the onset of 3G networks, data channels do not conflict with voice channels.

We propose to make use of the data channel while making normal voice calls to display the commonly queried information on the display. If the caller wants, she can completely ignore the screen and use the voice call functionality normally. However, if she wants the extra information, it is there on the screen for her to peruse.

For our initial implementation, we use a client application on the mobile device to extend the address book and query an external server for the extra pieces of information that a contact can or is willing to share. This information is gathered from the mobile device using external data collection programs or scraped from the internet, and stored on the server for later retrieval. When the caller picks a contact to call, she gets a list of information options to choose from, and corresponding data is displayed at the same time the voice call is placed to the contact.

### 3.4 Digital Montage

Digital Montage extends WebCall to allow cruise passengers to connect with each other using their digital photos and video clips captured on mobile devices. For example, if Carol takes a photo of her friends on her mobile phone, the phone detects the Bluetooth IDs of other surrounding phones at the moment, and stores this contextual information (relevant only to this photo) in the SPN server, building up her contextually-tagged personal photo collection. Carol can later decide to connect with a friend Dave by selecting him in a photo in which he appears, and if Dave’s mobile phone was detected when the photo was taken, his contact information will pop up on Carol’s phone. Carol can then contact him through voice call, email, SMS, etc. She can also send the photo or a collection of photos that triggered the connection. Such automatic tagging of digital media enables a range of services, including the transfer of digital postcards, photo-montages and interactive media relevant to both the producer and the receiver.

### 3.5 Reflective Media

Privacy is a serious concern in the age of digital media and online sharing. A popular privacy control provided by social websites is based on the social relationship among users. This may be fairly effective for the first degree of relationship, but no matter how

strictly you control access to your content, whoever obtained such access can copy and redistribute it freely without you knowing a bit. Once the content is out of your hand, you lose track of it.

Reflective Media aims to solve this problem. Instead of “dumb” media objects, e.g. JPEG images, which contain content only, we extend their formats such that they can self-track and report back to the original owner of any events of interest that have just happened. One example scenario is business card exchange. Alice sends her vCard to Bob who later passes it to his friend Cindy. Although Alice may appreciate Bob sharing her card to new friends, she would still like to know who have her business card by now. If Alice encodes her vCard in a reflective format (e.g. hypothetical R-vCard), every time it is passed to a new person the card will automatically notify Alice of such an event.

A more collaborative scenario is photo sharing. With affordable digital cameras people take a number of photos on their trips, then send to friends or share on the Internet. Many photos can easily be improved, e.g. by red-eye removal, but the owner often does not have time. If these photos are encoded in a reflective format (e.g. hypothetical R-JPEG), a friend may remove the red eye and the owner automatically gets a copy of the new version. This is all without explicit human intervention.

Reflective Media may be further mashed up with social context to provide more sophisticated control on privacy and security. For instance, different levels of fidelity of the same content can be shown based on the social closeness between the viewer and the owner – family members see the full resolution; friends see the medium resolution; acquaintances see the thumbnail, so on. This is certainly more effective than a pure binary switch where either the viewer sees the full content or nothing. Moreover, modern source coding algorithms such as MPEG-4 FGS [6] have provided powerful methods to manipulate media content. A combination of Reflective Media and source coding may enable novel sharing patterns of social content not previously thought of.

## 4. RELATED WORK

Data mashups [14] refer to the techniques of merging information from disparate sources despite differing conceptual, contextual and typographical representations. The concept stays the same in the cruiser scenario except that mobility is added as a new feature. While mobility can cause device-captured data to vary in time and location, the inherent correlation between data sequences can represent the updating process of the trip experience.

SensorPlanet [3] is a research collaboration on fusion methods where different sources provide different types of data on various granularity and semantic levels. Both MetroSense [7] and Urban Sensing [1] utilize the sensing capabilities on modern mobile devices to develop people-centric, life-enhancing, mobile social applications. Our services share the same goal while specializing in the aspect of social interactions upon physical encounters.

## 5. CONCLUSION

In this paper we present our mobile-centric Social Proximity Network (SPN) services for cruise ship scenarios. The foundation of these services is the infrastructure of connectivity and indoor positioning, built specifically for the structural and architectural

characteristics of cruise ships. We choose the client-server model as our software architecture, complemented by P2P when the server is out of reach, inefficient, or not to be used due to various concerns. We then build four service modules, namely mapping and messaging, proximity content sharing, WebCall, and Reflective Media. They address a wide range of functionalities from location-based services and phone communications to social data mashups and privacy preserving. Most of the modules are currently under development; some are working demos. We plan to carry out demonstrations and trial experiments in the near future and demonstrate them in the workshop.

## 6. REFERENCES

- [1] Agapie, E., Chen, G., Houston, D., Howard, E., Kim, J., Mun, M. Y., Mondschein, A., Reddy, S., Rosario, R., Ryder, J., Steiner, A., Burke, J., Estrin, E., Hansen, M., Rahimi, M. 2008. “Seeing Out Signals: Combining Location Traces and Web-based Models for Personal Discovery”. Proc. HotMobile (Feb. 2008).
- [2] “Bluetooth wireless technology becoming standard in cars”. 2006. SIG Press Release (Feb. 2006). [http://www.bluetooth.com/Bluetooth/Press/SIG/Test\\_1.htm](http://www.bluetooth.com/Bluetooth/Press/SIG/Test_1.htm)
- [3] Boda, P. P. 2007. “SensorPlanet: an Open Global Research Framework for Mobile Device Centric Wireless Sensor Networks”. NSF MobileSensors (Jan. 2007).
- [4] BodyTag BT-002. Bluelon. <http://www.bluelon.com/>
- [5] Burrell, J., Gay, G. 2002. “E-graffiti: Evaluating Real-world Use of a Context-aware System”. Interacting with Computers. 14 (2002), 301-312.
- [6] Li, W. 2001. “Overview of Fine Granularity Scalability in MPEG-4 Video Standard”. IEEE Trans. on Circuits and Systems for Video Technology. 11, 3 (Mar. 2001), 301-317.
- [7] Miluzzo, E., Lane, N. D., Eisenman, S. B., Campbell, A. T. 2007. “CenceMe – Injecting Sensing Presence into Social Network Applications”. Springer LNCS 4793 (Oct. 2007).
- [8] Persson, P., Fagerberg, P. GeoNotes. “GeoNotes: a Real-use Study of a Public Location-aware Community System”. 2002. SICS Technical Report T2002:27 (Dec. 2002).
- [9] Rantanen, M., Oulasvirta, A., Blom, J., Tiitta, S., Mantyla, M. I. 2004. “InfoRadar: Group and Public Messaging in the Mobile Context”. Proc. NordiCHI (Oct. 2004). 131-140.
- [10] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E. 2002. “SIP: Session Initiation Protocol”. RFC 3261 (Jun. 2002).
- [11] Saint-Andre, P. 2004. “Extensible Messaging and Presence Protocol (XMPP): Core”. RFC 3920 (Oct. 2004).
- [12] Wikman, J., Nurminen, J. K., Kokkinen, H., Muilu, P., Heikela, M. 2008. “Mobile Web Application Development Stack”. Proc. IEEE CCNC (Jan. 2008), 1246-1247.
- [13] [http://en.wikipedia.org/wiki/Genesis\\_class](http://en.wikipedia.org/wiki/Genesis_class)
- [14] <http://www.mashupcamp.com>