Об информационных системах для групп мобильных пользователей

Д.Е. Намиот, М.А. Шнепс-Шнеппе

Аннотация – В статье рассматриваются вопросы создания информационных систем для групп мобильных пользователей. Речь идет о выборе и использовании механизмов информирования мобильных пользователей (пользователей мобильных устройств), находящихся в некоторой географически ограниченной области. К рассматриваемым системам предъявлялись два основных требования. Во-первых, нас интересовали мобильные пользователи в некотором помещении (помещениях), так что в работе рассматриваются решения, которые не ориентированы на использование гео-локационных возможностей современных смартфонов. Другим требованием к рассматриваемым системам была их доступность для сторонних разработчиков, так что предлагаемые в данной работе решения не зависят от возможностей мобильных операторов. Решение для подобной задачи требует ответа на две группы вопросов: что можно использовать для непосредственной рассылки сообщений (доставки сообщений на мобильные терминалы), а также, каким образом проводить локализацию мобильных пользователей (как определить, что мобильный пользователь находится в заданной области). Именно ответ на последний вопрос и требует решений, которые будут работоспособны в помещениях (то есть, не будут использовать спутниковые системы гео-позиционирования). Для непосредственной доставки мобильных сообщений предлагается использовать так называемые push-уведомления. Этот механизм, с практически одинаковыми функционалом и программными интерфейсами, присутствует во всех современных мобильных операционных системах. Для локализации мобильных пользователей предложено несколько механизмов. Во-первых, это использование беспроводных тегов (iBeacons). Теги используют протокол Bluetooth (Bluetooth Low Energy) и потенциально позволяют определить местоположение конкретного мобильного телефона с точность до нескольких метров. В качестве замены (функционального эквивалента) специализированным тегам предложено использовать существующие узлы беспроводных сетей: точки доступа Wi-Fi и Bluetooth узлы. Также рассмотрены две формы пассивного мониторинга. В первом случае анализируются информационные фреймы, которые Wi-Fi устройства рассылают существующим точкам доступа. Во втором случае, анализируется информация о подключении мобильного устройства к точкам доступа (получение адреса). В обоих случаях удается получить идентификацию мобильного устройства (его адрес). Такой же адрес предлагается запоминать при осуществлении подписки на уведомления. Сравнение двух массивов адресов (подписчиков и локально определенных пользователей) и позволит в каждый момент времени определять круг пользователей для рассылки уведомлений.

Ключевые слова: мобильные пользователи, уведомления, определение местоположения

Введение

В данной статье рассматриваются вопросы создания информационных систем для групп мобильных пользователей. В данном случае речь идет об архитектуре и реализации информационных систем, которые позволяли бы доносить данные до мобильных пользователей (пользователей с мобильными устройствами), находящихся в некоторой географически ограниченной области. При этом, говоря о географически ограниченной области, мы имеем в виду, в первую очередь, мобильных пользователей в некотором помещении (помещениях). В соответствии с этим, речь не идет об использовании гео-локационных возможностей современных смартфонов. Также, речь не идет об использовании возможностей мобильных операторов, и в работе рассматриваются модели и решения, которые доступны сторонним разработчикам. Но при указанных ограничениях мы, тем не менее, предполагаем наличие некоторой сетевой инфраструктуры. Так называемые мэш-сети (мобильные мэш-сети) и их использование в информационных системах, работающих без сетевой инфраструктуры, будет являться предметом отдельного рассмотрения.

Понятие мобильные пользователи в данном случае шире понятия мобильные абоненты. В число пользователей входят, например, и пользователи планшетов (мобильных компьютеров), которые могут быть и не подключены к сетям связи (к телекоммуникационным операторам). Если бы речь шла только о мобильных операторах, то задача могла бы быть сформулирована (описана) как организация уведомлений для мобильных абонентов в заданной области. Это, безусловно, технически решаемая задача, мобильный оператор знает нахождение своих абонентов (по крайней мере, с точностью до базовой станции) и уведомления посредством SMS (и/или USSD), безусловно, могут быть организованы. Но этот подход не работает для сторонних производителей программного обеспечения в силу отсутствия на стороне оператора подходящих программных интерфейсов (API) и разного рода ограничений по безопасности, также присутствующих в сетях мобильной связи. Отметим, что под программным обеспечением в данном случае понимаются не только транспортные (в сетевом смысле) механизмы, то есть организация доставки сообщений, но и механизмы управления такого рода посылками (определение подписчиков, времени рассылки и т.п.). То есть именно то, что и отличает информационную систему от сетевых механизмов рассылки. Кроме того, такое возможное решение должно еще быть и меж-операторским, поскольку в группе пользователей могут, конечно, оказаться абоненты разных мобильных операторов. Поэтому мы рассмотрим решения, которые не связаны с мобильными операторами и доступны сторонним разработчикам.

Мобильные уведомления

Задача уведомления пользователей в мобильных операционных системах решается с помощью так-называемых push-уведомлений. Это механизм уведомления, поддерживаемый непосредственно мобильной операционной системой и инфраструктурой ее производителя. Мы можем говорить именно о мобильных операционных системах во множественном числе, поскольку практически идентичные по характеристикам и схожие по свои интерфейсам решения предлагаются всеми производителями мобильных операционных систем. Например, GCM (Google Cloud Messaging) для мобильных устройств под Android [1], APNS (Apple Push Notification Service) для мобильных устройств под iOS [2], MPNS (Microsoft Push Notification Service) для мобильных устройств под Windows [3] и так далее.

В основе данной технологии лежат две модели – клиент-сервер (client/server) и публикатор/подписчик (publisher/subscriber). Субъектом подписки является не мобильный абонент (как в операторских сервисах типа SMS), а приложение (рисунок 1).

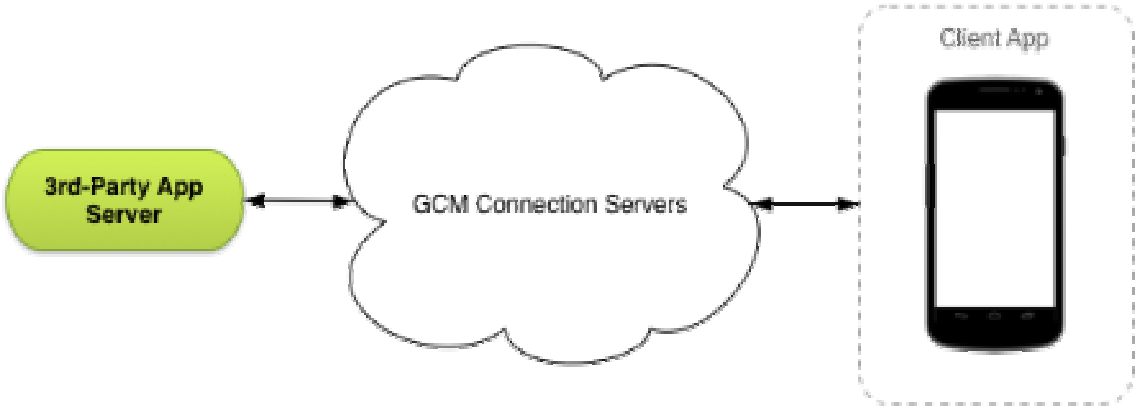

Рис. 1 Google Cloud Messaging [1]

Мобильный клиент (subscriber) при помощи установленного приложения подписывается на рассылку push-уведомлений, а сервер публикаций (publication server), по наступлению определенного события или по инициативе публикатора (publisher), осуществляет рассылку подписавшимся клиентам. Данная технология применяется для рассылки разного рода информации (сообщения, купоны, новости, прогнозы погоды, реклама, обновление программного обеспечения и т. д.) [4].

Важно, что push–уведомления позволяют удаленным серверам уведомлять пользователей о наступлениях событий, даже если приложение неактивно. Приложение, осуществившее подписку, должно быть лишь установлено на мобильном устройстве.

Push–уведомления экономят заряд батареи, не требуют постоянного соединения с удаленным сервером; они дешевле SMS, но для их доставки необходимо соединение с интернетом. Экономичность push–уведомлений создается за счет введения промежуточного звена между сервером - отправителем и приложением – получателем. Для программирования прикладных систем можно использовать и многочисленные фреймворки (например, Parse [5]). Все они содержат поддержку мобильных уведомлений. Подробный обзор инструментальных систем для поддержки push-уведомлений можно найти в работе [4].

Наличие программных интерфейсов (непосредственно в OS и/или в сопутствующих фреймворках) позволяет говорить о создании специализированных систем управления контентом (CMS), ориентированных на организацию рассылок push-уведомлений без программирования. Пример такой системы можно найти в работе [6]

Таким образом, push-уведомления вполне могут выступать транспортным уровнем для доставки информационных сообщений мобильным пользователям (и не только абонентам мобильных операторов). Но этот механизм сам по себе не решает задачу локализации этих сообщений. Без решения проблемы определения текущего местоположения подписчиков рассылка уведомлений не будет сильно отличаться от массовой рассылки почтовых уведомлений, чаще всего ассоциируемой со словом спам.

Использование беспроводных тегов для определения местоположения

Поскольку в работе речь идет о группах мобильных пользователей в помещениях, то для определения местоположения таких пользователей не получится использовать глобальные возможности гео-позиционирования мобильных устройств.

Одним из наиболее прозрачных (с точки зрения архитектуры) способов определения местоположения мобильного пользователя в некотором замкнутом пространстве является использование беспроводных тегов. Мобильное устройство определяет наличие (“видимость”) тега и по идентификации этого тега (места установки тегов известны) определяет свое местоположение. Все, конечно, базируется на предположении, что радиус действия такого тега ограничен. Системы на базе RFID [7] имеют уже достаточно длинную историю. Но первый вопрос, который здесь встает – это совместимость. Мобильные устройства должны понимать протокол, используемый тегом. В этом плане технология iBeacons [8] от компании Apple имеет, на сегодняшний день, пожалуй, наибольшие шансы стать стандартом. Теги iBeacons используют стандарт Bluetooth Low Energy (BLE) [9] для обмена данными. А поддержка BLE присутствует в мобильных устройствах Apple. Схема работы проиллюстрирована на рисунке 2:

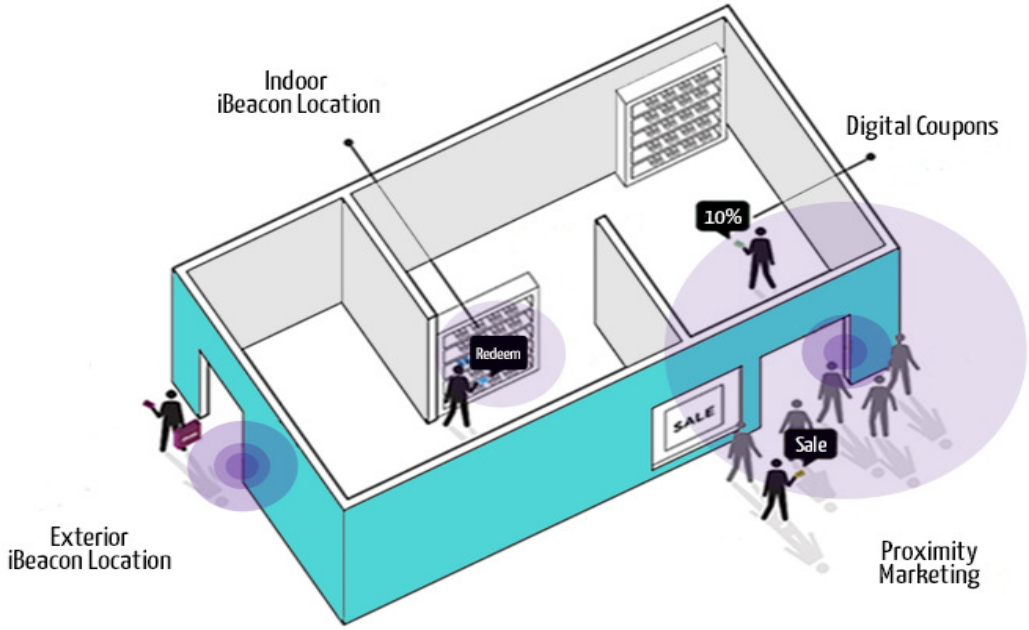

Рис. 2 Proximity Marketing [10]

Как это можно связать с описанной выше транспортной подсистемой? Решение может быть следующим. Приложение, для которого существует подписка на push-уведомление должно быть вызвано (выполнено), по крайней мере, один раз. При этом вызове пользователь и подтверждает свое согласие получать уведомления в данном приложении. Далее для общения с данным приложением (на программном уровне) будет использовать некоторый уникальный идентификатор, который и создается в процессе подписки (подтверждения прав). Так устроены все системы уведомлений. Приложение, которое будет осуществлять отправку сообщений, хранит у себя эти уникальные идентификаторы подписчиков и использует их при рассылке сообщений. Идея состоит в том, что при осуществлении подписки сохранять не только идентификатор приложения, но и идентификацию мобильного устройства (MAC-адрес). Далее, когда сканирующее приложение на телефоне определит наличие тега, приложение отметит это факт в сетевой базе данных, указав тег и адрес устройства. При рассылке же сообщений для подписчиков в конкретной области (область теперь определяется набором тегов) можно проверить нахождение MAC-адреса подписчика в текущем списке “присутствующих” адресов. Сам список присутствия обновляется при переходе пользователя к другому тегу, либо по времени (например, по истечении 30 минут запись автоматически удаляется) [11].

Естественно, что при наличии тегов должно существовать и какое-то запущенное приложение мобильном устройстве, которое эти теги сканирует. Если для приема сообщений не нужно запускать какое-то приложение (достаточно, чтобы оно присутствовало на телефоне), то для сканирования активная компонента обязательна. И, конечно, такой подход предполагает некоторую начальную процедуру (инвестиции) по установке тегов. Из недостатков iBeacons (мы не касаемся сейчас навигации с помощью тегов, что есть отдельная задача) можно выделить следующие моменты. Во-первых, эта технология пока слабо представлена вне устройств, базирующихся на операционной системе iOS. Только старшие версии Android (с очень малой долей распространения) имеют поддержку BLE. iPhone. Второй момент заключается в том, что для конкретного приложения (это только в iOS) необходимо специфицировать все доступные для него теги. То есть, по факту, для каждого набора тегов (помещения) иметь отдельное приложение.

Другой подход (network proximity [12]) состоит в том, что в качестве тегов можно использовать существующие узлы беспроводных сетей. Описанные выше теги iBeacons не хранят никакой информации. Сканирующая программа получает только идентификацию тега. И уже этот идентификатор может использоваться как ключ в последующих информационных запросах. Но точка доступа Wi-Fi, например, также позволяет “сканировать” собственную идентификацию. И распространение сигнала от такого сетевого узла ограничено по определению (в силу свойств самого протокола Wi-Fi). Соответственно, факт “видимости” приложением на мобильном устройстве заданной точки доступа Wi-Fi (узла Bluetooth) является основанием для заключения о том, что данное мобильное устройство находится поблизости от указанного сетевого узла. Географическая область будет описываться набором сетевых узлов. Дальнейший алгоритм работы идентичен описанному выше алгоритму работы с iBeacons. Достоинством этого подхода является то, что можно использовать существующую сетевую инфраструктуру. Другой положительный момент состоит в том, что беспроводные узлы могут создаваться специально для позиционирования. Например, точка доступа Wi-Fi может быть открыта непосредственно на мобильном телефоне не для организации доступа, а для предоставления своего идентификатора, который будет использоваться для определения располагающихся в данный момент поблизости мобильных пользователей.

Пассивный мониторинг

Использование тегов требует явной активности от мобильного клиента. Это может быть постоянно работающее приложение (процесс), которое определяет доступные (“видимые”) теги или явный запуск пользователем какого-либо приложения, которое начнет свою работу с определения тегов. В любом случае, требуется активность со стороны мобильного устройства. Альтернативой может быть решение, которое определяет наличие мобильных устройств в пассивном режиме, без явно определяемой активности со стороны последних.

Пассивный мониторинг может осуществляться путем анализа сигналов Wi-Fi модуля мобильного устройства. Основой для этого служат специальные широковещательные рассылки (так называемые Probe Requests), описанные в спецификации Wi-Fi [13]. Это рассылка, которую осуществляют Wi-Fi клиенты. Согласно спецификации, одна станция (сетевой узел) может посылать такой запрос при необходимости получения информации от другой станции (сетевого узла). Например, Wi-Fi клиент посылает такой запрос, чтобы определить, какие точки доступа существуют поблизости. Информационные потоки Wi-Fi запроса показаны на рисунке 3.

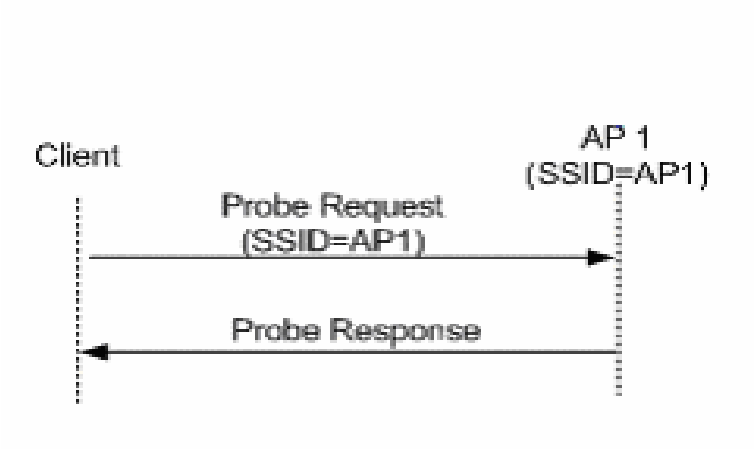

Рис. 3. Запрос Wi-Fi Probe Request.

Wi-Fi клиент посылает (точнее – может посылать) такой запрос без присоединения к какой-либо точке доступа Wi-Fi. Достаточно лишь включенного Wi-Fi интерфейса на телефоне [14]. Запросы Probe Request содержат MAC-адрес мобильного клиента. Анализ (расшифровка) такого рода пакетов может выполняться либо специализированными устройствами [15,16], либо (что и было в наших проектах) Wi-Fi маршрутизатором, который в качестве дополнительной опции обрабатывает Probe Requests пакеты [17]. Маршрутизатор собирает текущих “посетителей” (MAC-адреса) в отдельной таблице базы данных (MySQL). Следовательно, модель использования этой информации остается такой же, как она была описана выше. Мобильное приложение в данном случае вызывается только один раз и служит для “оформления” подписки. При этом серверная часть приложения сохраняет как идентификатор подписки приложения, так и MAC-адрес мобильного устройства. При рассылке, адреса возможных получателей проверяются на присутствие в базе текущих “посетителей” и сообщения отправляются только тем подписчикам, которые физически находятся в зоне действия Wi-Fi маршрутизатора.

Из недостатков этого подхода можно отметить два основных момента. Во-первых, спецификация Wi-Fi не гарантирует, что мобильный клиент будет рассылать требуемые пакеты. На практике, реальное выполнение зависит от многих факторов, включая текущую загрузку телефона. В целом, из наших собственных экспериментов мы можем подтвердить оценку в 70% обнаруживаемых таким образом мобильных пользователей [18]. Другая проблема связана с политикой компании Apple. С недавнего времени, iOS использует случайный MAC-адрес в пакетах Probe Request. Соответственно, для предложенной модели не удастся определить посетителя среди подписчиков.

Другая вариация пассивного мониторинга возможна в случае наличия в помещении реальной точки доступа Wi-Fi. Тогда для мобильных пользователей их мобильные устройства могут автоматически присоединяться к такой точке доступа (Preferred Network List [19]). Никакого обмена данными (активности пользователя) здесь не нужно, нужен именно сам факт соединения. Наличие соединения даст возможность определить MAC-адрес присоединившегося устройства. В нашем случае использовался разбор данных DHCP фильтра в WireShark (рисунок 4).

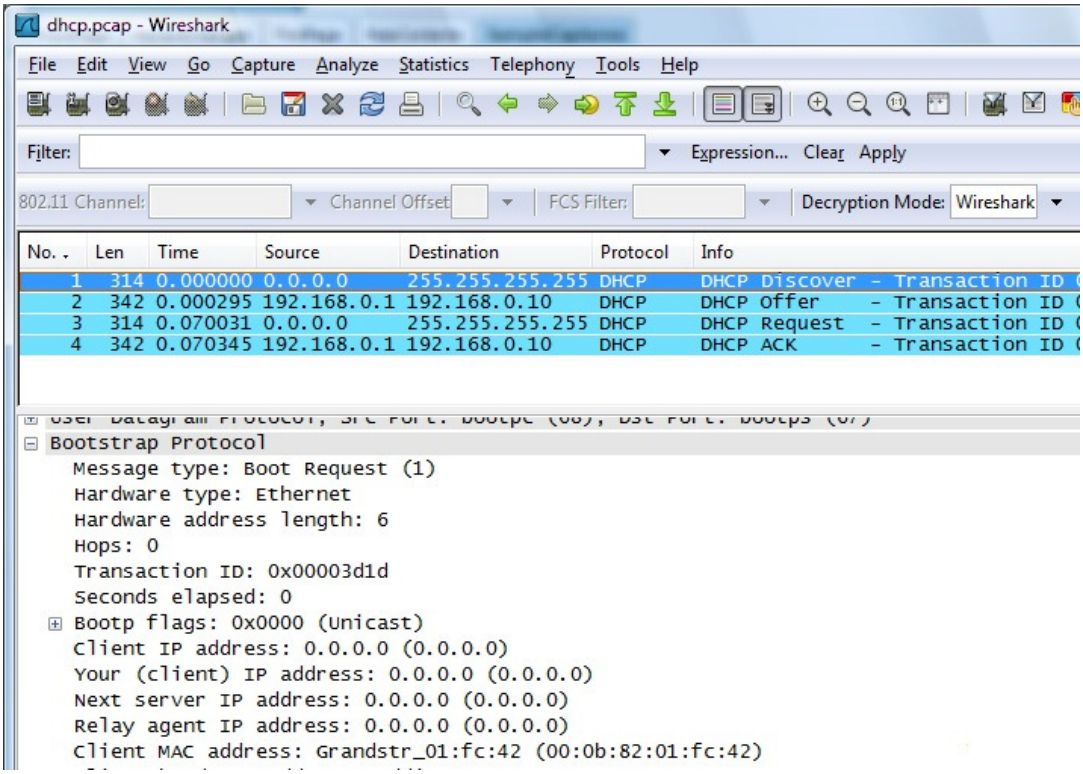

Рис. 4 WireShark [20]

Заключение

В работе рассмотрены различные модели программных систем, которые могут использоваться для информирования мобильных пользователей в привязке к произвольным помещениям. Такого рода модели состоят из двух основных компонент. Во-первых, это собственно механизм доставки сообщений мобильным пользователям, а во-вторых, механизм (механизмы) локализации мобильных пользователей. В качестве транспортного уровня (непосредственной доставки сообщений) использовались инструменты push-уведомлений мобильных операционных систем. Для персонализации подписок и возможной будущей локализации предложен механизм дополнения идентификационной информации адресом мобильного клиента. Для локализации (оценки местоположения) мобильных пользователей использовались несколько возможных подходов. Во-первых, это активный мониторинг, связанный с использованием беспроводных тегов или их эквивалентов. В качестве эквивалентов беспроводным тегам предложена модель использования элементов сетевой инфраструктуры (точек доступа Wi-Fi и Bluetooth узлов). Также, были рассмотрены несколько форм пассивного мониторинга, использующего служебный трафик мобильных устройств.

ЛИТЕРАТУРА

1. Google Cloud Messaging https://developer.android.com/google/gcm/index.html.
2. Apple Push Notification Service https://developer.apple.com/library/ios/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/Chapters/ApplePushService.html
3. Push notifications for Windows Phone 8. https://msdn.microsoft.com/en-us/library/windows/apps/ff402558%28v=vs.105%29.aspx
4. Павлов А. Д., Намиот Д. Е. Системы для поддержки push-уведомлений //International Journal of Open Information Technologies. – 2014. – Т. 2. – №. 7. – С. 37-44.
5. Parse mobile app platform https://parse.com/
6. Павлов А. Д., Намиот Д. Е. Информационные системы на основе push-уведомлений //International Journal of Open Information Technologies. – 2014. – Т. 2. – №. 8. – С. 11-19.
7. Bouet M., Dos Santos A. L. RFID tags: Positioning principles and localization techniques //Wireless Days, 2008. WD'08. 1st IFIP. – IEEE, 2008. – С. 1-5.
8. Newman N. Apple iBeacon technology briefing //Journal of Direct, Data and Digital Marketing Practice. – 2014. – Т. 15. – №. 3. – С. 222-225.
9. Gomez C., Oller J., Paradells J. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology //Sensors. – 2012. – Т. 12. – №. 9. – С. 11734-11753.
10. Wave a lot http://wave-a-lot.com/app/ibeacons
11. Sneps-Sneppe M., Namiot D. Spotique: A new approach to local messaging //Wired/Wireless Internet Communication. – Springer Berlin Heidelberg, 2013. – С. 192-203.

12. Namiot D. Network Proximity on Practice: Context-aware Applications and Wi-Fi Proximity //International Journal of Open Information Technologies. – 2013. – Т. 1. – №. 3. – С. 1-4.
13. Chandra R. et al. A Beacon-Stuffing: Wi-Fi without Associations Mobile Computing Systems and Applications, 2007. In HotMobile 2007. 8th IEEE Workshop on, pp.53-57.
14. Gast M. 802.11 Wireless Networks: The Definitive Guide. O'Reilly Media, Inc., 2005, 654 p.
15. Navizon ITS. http://its.navizon.com/doc/index.html
16. Cisco MSE http://www.cisco.com/en/US/products/ps9742/index.html
17. Sneps-Sneppe M., Namiot D. Smart cities software: customized messages for mobile subscribers //Wireless Access Flexibility. – Springer Berlin Heidelberg, 2013. – С. 25-36.
18. Musa A. B. M., Eriksson J. Tracking unmodified smartphones using wi-fi monitors //Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems. – ACM, 2012. – С. 281-294.
19. Klasnja P. et al. When i am on wi-fi, i am fearless: privacy concerns & practices in eeryday wi-fi use //Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. – ACM, 2009. – С. 1993-2002.
20. WireShark DHCP http://wiki.wireshark.org/DHCP.