

## Research Article

# Problem Solving of Low Data Throughput on Mobile Devices by Artefacts Prebuffering

**Ondrej Krejcar**

*Department of Measurement and Control, Centre for Applied Cybernetics, Faculty of Electrical Engineering and Computer Science, VSB Technical University of Ostrava, 17 Listopadu 15, 70833 Ostrava Poruba, Czech Republic*

Correspondence should be addressed to Ondrej Krejcar, [ondrej.krejcar@remoteworld.net](mailto:ondrej.krejcar@remoteworld.net)

Received 29 March 2009; Revised 29 July 2009; Accepted 11 November 2009

Recommended by Naveen Chilamkurti

The paper deals with a problem of low data throughput on wirelessly connected mobile devices and a possibility to solve this problem by prebuffering of selected artefacts. The basics are in determining the problem parts of a mobile device and solve the problem by a model of data prebuffering-based system enhancement for locating and tracking users inside the buildings. The framework uses a WiFi network infrastructure to allow the mobile device determine its indoor position. User location is used for data prebuffering and for pushing information from a server to PDAs. All server data are saved as artefacts with its indoor position information. Accessing prebuffered data on a mobile device can significantly improve a response time needed to view large multimedia data. The solution was tested on a facility management information system built on purpose with a testing collection of about hundred large size artefacts.

Copyright © 2009 Ondrej Krejcar. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

The mobile wireless devices (laptops, PDA devices, Smart phones, etc.) are commonly used with internet connection which is available almost everywhere and anytime these days. The connection speed of the two most common standards GPRS and WiFi varies from hundreds of kilobits to several megabits per second. In case of corporate information systems or some other types of facility management, zoological or botanical gardens, libraries, or museums information systems, the WiFi infrastructure network is often used to connect mobile device clients to a server. Unfortunately, the theoretical maximum connection speed is only achievable on laptops where high-performance components are used (in comparison to mobile devices). Other mobile devices like family PDAs or Smart phones have low-performance components due to a very limited space. The limited connection speed represents a problem for online systems using large artefacts data files. It is not possible to preload these artefacts before the mobile device is used in remote access state. This problem was found as a very important point. The rest of this paper specifies the problem and suggests a possible solution.

The goal is to complete the data networking capabilities of RF wireless LANs [1] with accurate user location and tracking capabilities for user needed data prebuffering. This property is used as an information base for an extension of existing information systems or to create a special new one.

An information about location is used to determine both an actual and future position of a user [2]. A number of experiments with the information system have been performed and their results suggest that determination of the location should be focused on. The following sections describe also the conceptual and technical details about Predictive Data Push Technology Framework (PDPT).

## 2. The Problem of Low Data Throughput

Why can we not use the classical model of user's requests and server's response for large data files? It is because some large amounts of data (artefacts) are impossible to download to PDA device and to be displayed in relatively short time.

Each data artefact has to go through an identical way. Starting at the server database, it follows the WiFi Access Point (AP) and finally reaches a PDA display (Figure 1).

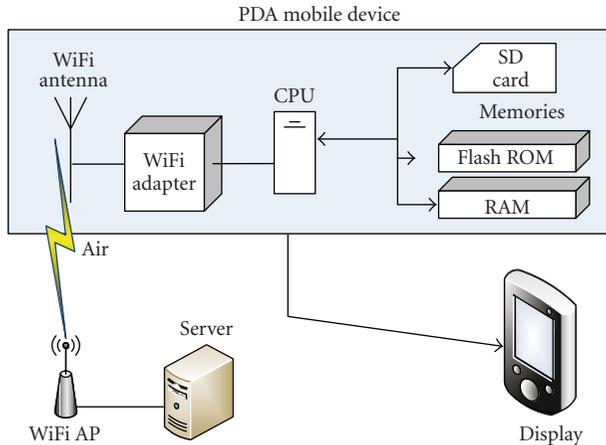


FIGURE 1: A data communication way—from server database to PDA display.

There are several important components:

- (1) an Ethernet network (Server to WiFi AP),
- (2) WiFi Access Points,
- (3) wireless communication between WiFi AP and WiFi antenna of the PDA device,
- (4) a PDA WiFi antenna,
- (5) a PDA WiFi adapter,
- (6) a PDA CPU,
- (7) a PDA memory,
- (8) a graphical unit—display data to user.

For large amount data artefacts, the most important parts are those listed under 5 to 8. The first and the second components have a relatively high throughput when compared to PDA device components and do not require testing. The Wireless communication between WiFi AP and PDA WiFi antenna cannot be improved.

The fourth component is a WiFi antenna of the PDA. This antenna has a standard of 2 dB gain and cannot be improved due to the absence of a connector to an external antenna. It is thus not necessary to test the antenna.

A number of tests with several types of PDA devices (HTC Athena, HTC Universal, HTC Blueangel and HTC Roadster) have been undertaken. These PDA devices were connected through CISCO WiFi APs (Signal Strength quality  $\geq 80\%$ ) to an FTP server. The FTP server holds 3 types of large artefacts (files) which were downloaded to the internal PDA memory.

The maximal theoretical transfer rates (max 687 kB/s for IEEE 802.11b standard [3]) have not been achieved (Table 1). The maximum transfer rate 350 kB/s has been obtained by HTC Athena, but this is not a standard PDA device. Athena is a mininotebook with Windows Mobile 6 operating system. All other standard mobile devices have reached only about 25% of this speed (max 160 kB/s). Significant part of transfer speed is taken by the protocol cost on physical layer (app. 30–40%), but the rest of difference between theoretical and

TABLE 1: Data transfer tests—PDA is connected through WiFi infrastructure—use of internal FLASH ROM memory.

Data size (MB)	PDA device			
	Athena	Universal	Blueangel	Roadster
10	347	123	160	106
20	344	121	157	79
30	314	123	58	43

TABLE 2: Data transfer tests—SPB Benchmark software—Internal flash memory of PDA devices.

Type of test	PDA device			
	Athena	Universal	Blueangel	Roadster
Write 1 MB	2268	667	359	133
Read 1 MB	20400	3659	2180	30395
Write 100 × 10 kB	298	229	128	27
Read 100 × 10 kB	1880	1637	1199	737

real transfer rate (687 kB/s versus 160 kB/s) is due to the low performance components of mobile devices.

In theory, it is possible to find the worst component of all components and try to improve it. With this idea in mind, components 5 to 8 have been tested. It is not possible to test the WiFi adapter directly. The only possible way is to update a driver. The hardware of the WiFi adapter was found as identical in most cases of classical PDA devices from HP, HTC, or Acer companies. State-of-the-art drivers were found for all test devices showing us exactly the same transfer rates. Unfortunately, no easier solution to improve this component is known at this stage.

A CPU component is of course one of the most important parts at all. The power of the CPU can be easily benchmarked. All tested devices have an Intel XScale CPU. The Blueangel is equipped with PXA 263 with 400 MHz, Universal has 520 MHz PXA 270, and Athena and Roadster have 624 MHz PXA 270. Let us try to compare the best devices—Athena and Roadster with same CPU unit. A single CPU has entirely different transfer rates. That is why the speed of CPU is not so important on this occasion.

The PDA memory is a very important part on the other side. There is a large room for improvement of every PDA because of an SD Card slot which is present in most cases. The test was performed with standard SD Cards (Table 3).

In the test, Athena outperformed other devices (Table 2). A writing operation of 1 MB file was achieved at a speed of 2268 kB/s. The remaining devices only achieve about 25% or less of this speed which is insufficient. A reading operation of 1 MB file provides two errors in case of Athena and Roadster. The testing was sufficient because of extremely high and quick RAM memory. On the other hand, Universal and Blueangel provide good and relevant data. The 100 × 10 kB test has been undertaken for comparative purposes only. The main objective is to focus on large data artefacts.

TABLE 3: Data transfer tests—SPB Benchmark software—SD Cards.

Type of test	SD Card			
	Kingston 1 GB 50×	Kingston 2 GB 50×	Kingston 2 GB 120×	Pretec 2 GB 133×
	Transfer Speed (kB/s)			
Write 1 MB	523	414	<b>929</b>	717
Read 1 MB	1008	1101	<b>1219</b>	1006
Write 100 × 10 kB	73	144	<b>263</b>	48
Read 100 × 10 kB	756	899	<b>1012</b>	822

The best tested SD Card (Table 3) is a Kingston 2 GB 120× (120× means the card is 120 times faster than standard single speed CD ROM (150 kB/s)). When compared to an internal flash ROM, the writing speed of internal flash ROM is lower the reading speed remains 2 or 3 times higher.

The problem of low data throughput begins exactly in writing speed of internal flash. All data files are first transferred to cache (use of Microsoft explorer or Opera browser) which is relatively slow. The use of an SD card could speed up the transfer.

The other problem particularly for large data artefacts is the free internal memory. All PDA devices have a very limited space and usually operate with a memory allowing for 20 to 50 MB of free space including Athena device. Therefore the use of SD Cards would be a solution.

*2.1. Maximum Application Response Time.* Nielsen [4] specified the time delay of application response to user request to 10 seconds [5]. “During this time the user was focused on the application and was willing to wait for an answer.” Nielsen’s book [4] was published in 1994, but it is a basic literature for this phenomenon. Another interesting source [6] suggests that “decreases in performance and behavioural intentions begin to flatten when the delays extend to 4 seconds or longer, and attitudes flatten when the delays extend to 8 seconds or longer.” Based on these sources the maximum application response time is set to 10 seconds.

During the set maximum response time, the requested data must be downloaded and showed to user on display (in case of remote request to server’s data). The time period of 10 seconds is used to calculate the maximum possible data size of a file transferred from server to client (during this period). To achieve the best transfer speed 160 kB/s, the calculated file size is 1600 kB.

The next step is to define an average artefact size. The network architecture building plan is currently used as a sample database, which contains 100 files of an average size of 470 kB. During the 10-second period, the client application can download 2 to 3 files (depending on the actual connection capabilities).

The second problem is the extremely long delay in displaying files in certain original file types (e.g., AutoCAD in case of vector graphic or MS Office in general cases). An AutoCAD file type is used in most cases of *facility management* of modern building [7]. In such cases the mobile user needs to view a selected building scheme (building area plan, gas line plan, etc.) immediately.

TABLE 4: Data files displaying tests for 500 kB files (13 iterations).

Data type	Universal	Blueangel	Roadster
	Data file displaying time (s)		
Jpeg	0,88	1,27	4,5
AutoCAD	45,1	56,8	81,6
Word	19,8	29,9	40,3

TABLE 5: Application starting times for selected data types (13 iterations).

Application	Universal	Blueangel	Roadster
	Application Start Speed (s)		
MS Internet Explorer	2,3	3,9	5,4
PPC CAD Viewer	1,3	2,4	3,1
MS Word Mobile	5,4	6,7	8,2

The time required to open a 500 kB file is summarized in Table 4. The data file open delay for both AutoCAD and MS Word files is significantly longer than the basic Jpeg data type. To avoid any doubt, the application start time was measured too (Table 5). The results are acceptable as the delays in all cases remain below the 10 seconds limit.

Unfortunately, the displaying time delays of files with nonbasic types are unacceptable. A basic data format must be used to display files by PDA natively (BMP, JPG, GIF) without any additional striking time consumption. The solution is a conversion from any format to these native formats (for PDA devices). In case of sound and video formats, it can also be recommended to use basic data format (wav, mp3, wmv, and mpg). In case of sample database, the display time of artefact is only about a half second per 500 kB artefact. This short time delay is not considered to 10 seconds response limitation. If other file types are used, the delay for presentation of file must be included.

The end result of several real tests and subsequent calculations give a definition of artefact size as an average value of 500 kB. The buffer size may differ from 50 to 100 MB in case of 100 to 200 artefacts.

In order to provide the reader with more information, the next chapter describes how a position can be obtained from wireless networks background.

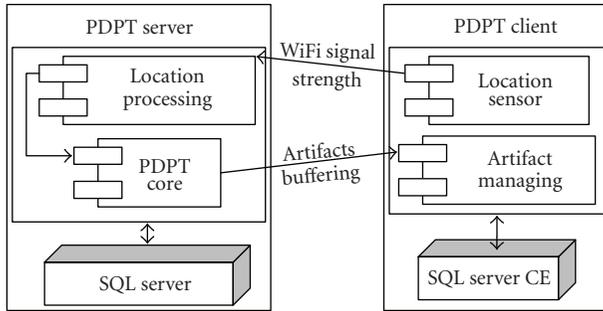


FIGURE 2: PDPT Framework architecture. Measured WiFi SS goes through the *Location sensor* to Location processing where the user position, current track, predicted track, and predicted user position are computed. PDPT Core makes a selection of artefacts to be prebuffered to mobile SQL Server CE.

### 3. Predictive Data Push Technology Framework

In most cases the low software level cache is used [8] or the residing of chips on system desk is recommended [9] to improve the performance of a system when operating with multimedia content. Such techniques are not allowed on existing mobile device where the operation system exists. Only a software solution added on top of the OS can achieve the objective.

A combination of a predicted user position with prebuffering of data associated with physical locations bears many advantages in increased throughput of mobile devices. An interesting solution (Microsoft US patent [9]) in this field needs to know all information (AP location, SS power, etc.) of all wireless base stations in mobile device before the localization process can be started (see the *Location Manager* module [10]). Moreover, the *Moving Direction Estimator* module is also situated in a mobile device application. These two facts present limitations to changing wireless base stations structure or to computing power consumption. Another solution (HP US patent [11]) represents a similar concept. A *Location Determination* and *Path Guide* modules are situated in mobile device side too.

The key difference between [10, 11] and PDPT solution is that the location processing, track prediction, and cache content management are situated at server side (Figure 2). This fact allows for managing many important parameters (e.g., AP info changes, position determination mechanism tuning, artefacts selection evaluation tuning, etc.) online at a PDPT Server.

The created PDPT Framework is based on a model of location-aware enhancement. This concept enables to increase the real dataflow from wireless AP (server side) to PDA (client side). The fact that the throughput (Table 1) is low on wireless connected mobile devices is very important with regards to the idea of using a prebuffered data for increasing transfer speed through WiFi connection on PDA mobile devices.

The general principle of localization states that if a WiFi-enabled mobile device is close to such a stationary device—base station, it may “ask” the provider’s location position by

setting up a WiFi connection. If the mobile device knows the position of the stationary device, it also knows that its own position is within a 100-meter range of this location provider. The location accuracy can be improved by triangulation of two or several visible WiFi APs [12, 13]. The PDA client will support the application in automatically retrieving location information from nearby location providers, and in interacting with the server. Naturally, this principle can be applied to other wireless technologies. The application is now implemented in C# using the MS Visual Studio .NET 2005 with .NET compact framework and a special OpenNETCF library enhancement. The information about the basic concept and technologies of user localization can be found in [7].

The current and predicted user positions are used for the PDPT framework to make decisions as to which data artefacts are needed in the PDA memory. The data prebuffering increases the primary dataflow from WiFi AP (server side) to PDA (client side). These techniques form the basis of the predictive data push technology (PDPT).

PDPTs push the data from an information server to the client’s PDA on the basis of the user’s location and user’s future predicted location. The prebuffered data will be helpful when the user comes to the location which was predicted by PDPT Framework. The benefit of the PDPT consists in the reduction of time needed to display a desired information requested by a user command on the PDA. This delay may vary from a few seconds to a number of minutes. It depends on two aspects.

The first aspect is the quality of wireless WiFi connection used by the client PDA. A theoretical speed of WiFi connection is maximum 687 kB/s. However, the test suggests a speed of only 43–160 kB/s (depending on file size and PDA device) (Table 1).

The second aspect is the size of copied data. The current application records just one set of WiFi signal strength (SS) values at a time (by Locator unit in PDPT Client). From this set of values the actual user position is determined by the PDPT Server side. PDPT Core responds to a location change by selecting the artefact to load to PDPT Client buffer. The data transfer speed is to a large extent influenced by the size of these artefacts. For larger artefact size the speed decreases.

**3.1. PDPT Framework Data Artifact Manager.** The PDPT Server SQL database manages the information (e.g., data about Ethernet hardware such as Ethernet switch UTP socket, CAT5 cable lead, etc.) in the context of their location in building environment. This contextual information is the same as location information about user track. The PDPT Core controls data, which are copied from the server to the PDA client by context information (position info). Each database artefacts must be saved in the database along the position information to which it belongs.

During the process of creating of a PDPT Framework the new software application called “Data Artefacts Manager” was developed (Figure 3). The manager is also described in [14]. This application manages the artefacts in WLA database (localization oriented database). The user can set the priority,

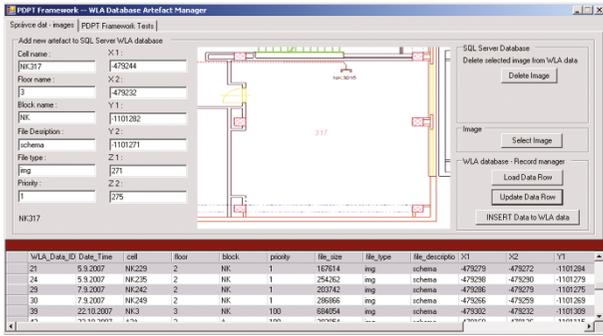


FIGURE 3: PDPT Framework Data Artefact Manager—this software substitutes a live connection to information system if it does not exist.

location, and other metadata of the artefact. This manager substitutes the online conversion mechanism, which can transform the real online information system data to WLA database data artefacts during the test phase of the project. This manager can be also used in case of offline version of PDPT Framework usage.

The Manager allows to the administrator to create a new artefacts from multimedia files (image, video, sound, etc.) and edit or delete the existing artefacts. The left side of the screen contains the text field of artefact metadata as a position in 3D space. This position is determined by the artefact size (in case of building plan) or by binding of the artefact to some part of a building in 3D space. It is possible to take the 3D axis from a building plan by a GIS software like Quantum GIS or by own implementation [15, 16]. The central part represents a multimedia file and the right side contains the buttons to create, edit, or delete the artefact. The lower part of the application screen shows the actual artefacts in WLA database located on MS SQL Server.

**3.2. The PDPT Framework Design.** The PDPT framework design is based on the most commonly used server-client architecture. To process data the server has an online connection to the information system. Technology data are continually saved to SQL Server database [17].

A part of this database (desired by user location or his demand) is replicated online to client’s PDA, where it is visualized on the screen. The user’s PDA has a location sensor component, which is continuously sending the information about nearby AP’s intensity to the framework kernel. The kernel processes this information and makes a decision as to which part of MS SQL Server database will be replicated (pushed) to client’s MS SQL Server CE database. The kernel decisions constitute the most important part of the whole framework as the kernel must continually compute the position of the user and track and predict the user’s future movement. After making this prediction, the appropriate data (part of MS SQL Server database) are prebuffered to the client’s database for the future possible requirements (Figure 4). The PDPT framework server is created as a Microsoft web service to act as bridge between the MS SQL Server and PDPT PDA Clients.

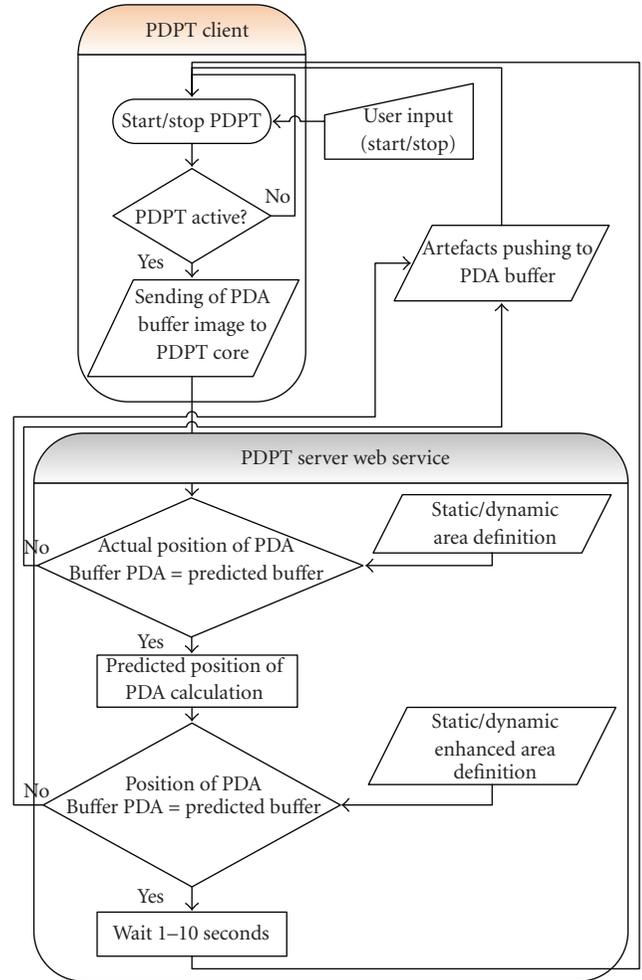


FIGURE 4: PDPT Framework—UML Design of flow diagram. The artefacts collections which belong to the actual user position are copied to PDA buffer firstly. The predicted position and new artefacts collection is used if all artefacts are already loaded to PDA buffer. The definition of the Static or Dynamic area depends on the specific case of artefacts size, area size of where the PDPT framework is equipped.

**3.3. PDPT Client.** The PDPT Client is a Windows Mobile 6.1-based application. The PDPT Client was developed for testing and tuning the PDPT Core. This client realizes a classic client to server side and an extension by PDPT and Locator module.

Figure 5 shows a screenshot from the mobile client. The figure shows the typical view of the data presentation from MS SQL CE database to the user (in this case the Ethernet plan of the current area). Each process running in a PDPT Client is measured in a millisecond resolution to provide a feedback from a real situation. The time window is in the upper right side of the screen (Figure 5). The user can select an artefact from prebuffered artefacts to view. Unfortunately, if the requested artefact does not exist in the PDA memory buffer, the online connection to the server must be used to select and download them online.

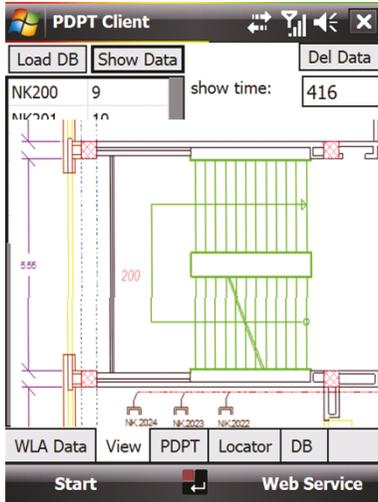


FIGURE 5: PDPT Client. View of prebuffered artifacts (building computer network plans) from Microsoft SQL Server CE database.

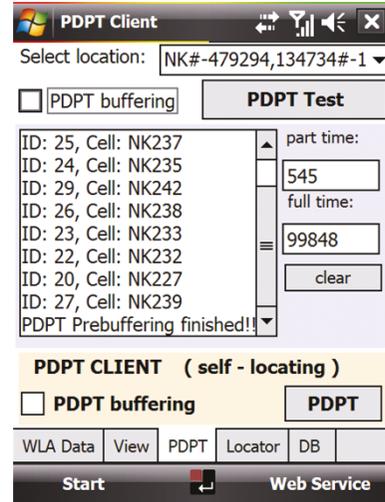


FIGURE 7: PDPT Client. On PDPT tab is a summary of prebuffered artefacts history and a part and full time of prebuffering. The upper part of screen is only for testing the functionality.

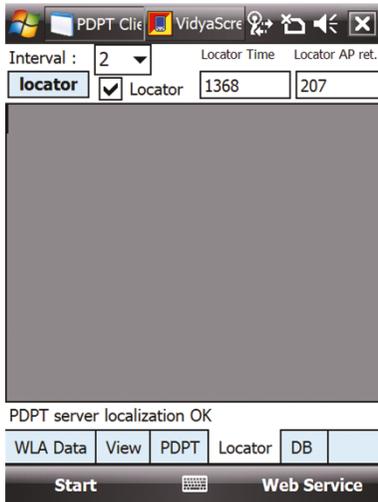


FIGURE 6: PDPT Client. On Locator tab is possible to set an interval for scanning of WiFi network neighbour. Locator time displays a real interval between a sending of neighbour WiFi AP summary and the reply from server.

Tabs Locator (Figure 6) and PDPT (Figure 7) presents a way to tune the settings of PDPT Framework. In the first case (Figure 5), the user must turn on Locator check-box which means that the continual measurement of WiFi signals of nearby APs (time of these operations is measured in Locator Time text window) will start.

The info about nearby APs is sent to the PDPT Server which responds with a number of recognized APs in the database (Locator AP ret. Text window). In the presented case, the 7 APs are in user neighbourhood, but only 2 APs are recognized by the PDPT Server database (info about 2 APs is in WLA database). The scanning interval is set to 2 seconds and finally the text “PDPT Server localization OK” means that the user PDA was localized in an environment and that

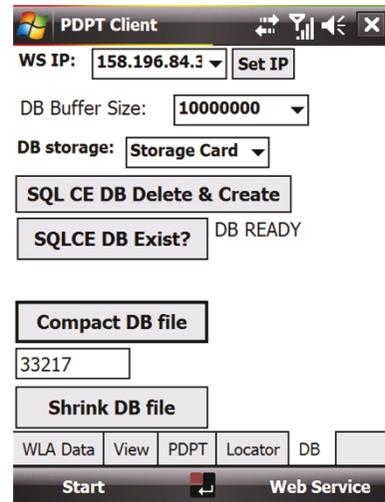


FIGURE 8: PDPT Client. The DB tab allows managing all necessary things from creation of database to Compact or Shrink.

this position can be used by the PDPT Core to prebuffer the data to the client device.

The middle section of the PDPT tab (Figure 7) shows logging info about the prebuffering process. The right side shows the time of artefact loading (part time) and the full time of prebuffering.

**3.4. PDPT Client—Microsoft SQL Server CE Database.** A DB manager for managing a database file on the PDA device was created (Figure 8). The first combo box menu on this tab deals with IP address settings of the PDPT Server. *DB Buffer size* follows on the second combo box. This size is important for maximum space taking by prebuffering database on selected data media. Data medium can be selected on *DB*

TABLE 6: PDPT Framework—Final Data transfer tests.

Test no.	Execution Time (min:s)	Move speed (m/s)	Quality of prebuffering (%)
1	1 : 40	1,32	25
2	1 : 48	1,22	18,18
3	3 : 37	0,61	75
4	3 : 24	0,65	36,36
5	5 : 12	0,42	97
6	5 : 26	0,40	54,55

Storage combo box. To check if a database exists, the *SQLCE DB Exist* button must be pressed. For example the *db is ready* means that the database file exists in a selected location. If such db file does not exist, the execution of *SQL CE DB Delete & Create* must be done. This button can be used for recreating of db file.

*Compact* and *shrink* of DB file means two options for manual database compression. The time in millisecond is measured in a text box in between the two buttons. Both of these mechanisms are used in prebuffering cycles when the large artefact is deleted from database table to release space of deleted artefact. The database file has occupied space of deleted artefact by default, because the standard operation of delete order does not include this technique. This is due to recovery possibilities in Microsoft SQL Server CE databases.

#### 4. Experiments

For a mobile device to determine its own position, it must have a WiFi adapter still alive. This fact provides a limitation of using of mobile devices. The complex test with several types of batteries is described in [18].

A number of indoor experiments were achieved with the PDPT framework using the PDPT Client application. The main result of the use of the PDPT framework is a reduction of data transfer speed. The tests focused on the real use of the developed PDPT Framework and its main impact on increased data transfer.

A realization of tests consists of a user movement from a sample location NK to C at a predefined direction. See the university campus map (Figure 9) where the tests were realized. For the purposes of the test, five mobile devices were selected with different hardware and software capabilities. Six types of test batches were executed in the test environment. Each test was between two points of the testing environment (building NK and C) with 132 meter distance. Every other test was in reversed direction. Five iterations (five devices used) were performed in each batch.

Results (Table 6) provide a good level of usability when user is moving slowly (less than 0,5 m/s). This is caused by a low number of visible WiFi APs in the test environment, where for 60% of total time only 1 AP was visible, 20% 2 visible, and 5% 3 or more visible WiFi APs. 15% of time represents a time without any WiFi connections. The values of prebuffering quality achieved in such case are very good.

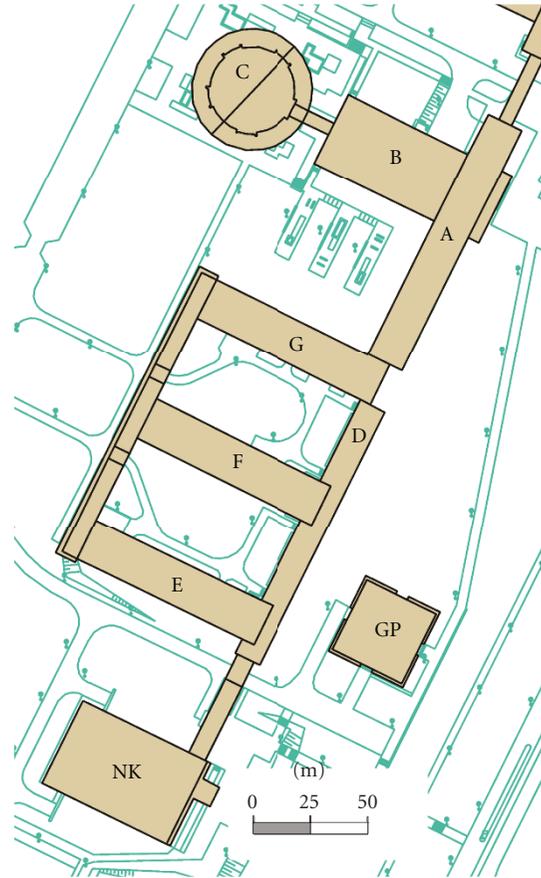


FIGURE 9: VSB Technical University of Ostrava—University Campus Map.

A special Biotelemetry system for patient monitoring is under development at our department. In this complex system the wide network of remote sensors is used to collect data. This system proved to be a useful platform for prebuffering the large data-artefacts [19, 20]. Localization module of PDPT framework is suitable for home security system [21]. For any kind of emergency cases, the special wireless network *MANET* can be suitable improvement of PDPT solution to avoid any problems in case the signal of preferred WiFi network is missing [22].

#### 5. Conclusions

The problem of low transfer rates in mobile devices was presented. Some suggestions have been put forward (e.g., to use a high-performance SD Cards for large data amount to get a higher transfer rate). The low transfer rates problem was considered also in the context of a maximum response time for user requests.

The PDPT Framework was described as one of the possible solutions. The indoor location of a mobile user is obtained through an infrastructure of WiFi APs. This mechanism measures the quality of the link of nearby location provider APs to determine the actual user position. User

location is used in the core of server application of the PDPT framework to data prebuffering and pushing information from the server to the user's PDA. Data prebuffering is the most important technique to reduce the time from a user request to system response. The experiments show that the location determination mechanism accurately and with sufficient quality determines the actual location of the user in most cases. Minor inaccuracies do not impact significantly on the PDPT Core decision making. The framework was evaluated in a real use experiment.

## Acknowledgment

This research has been carried out under the financial support of the research grant "Centre for Applied Cybernetics," Ministry of Education of the Czech Republic under Project 1M0567.

## References

- [1] F. Evrenou and F. Marx, "Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 86706, 11 pages, 2006.
- [2] V. M. Olivera, J. M. C. Plaza, and O. S. Serrano, "WiFi localization methods for autonomous robots," *Robotica*, vol. 24, no. 4, pp. 455–461, 2006.
- [3] A. Kostuch, K. Gierłowski, and J. Wozniak, "Performance analysis of multicast video streaming in IEEE 802.11 b/g/n testbed environment," in *Wireless and Mobile Networking*, vol. 308 of *IFIP Advances in Information and Communication Technology*, pp. 92–105, Springer, New York, NY, USA, 2009.
- [4] J. Nielsen, *Usability Engineering*, Morgan Kaufmann, San Francisco, Calif, USA, 1994.
- [5] M. Haklay and A. Zafiri, "Usability engineering for GIS: learning from a screenshot," *The Cartographic Journal*, vol. 45, no. 2, pp. 87–97, 2008.
- [6] D. F. Galletta, R. M. Henry, S. McCoy, and P. Polak, "When the wait isn't so bad: the interacting effects of website delay, familiarity, and breadth," *Information Systems Research*, vol. 17, no. 1, pp. 20–37, 2006.
- [7] O. Krejcar, "User localization for intelligent crisis management," in *Proceedings of the 3rd IFIP Conference on Artificial Intelligence Applications and Innovation (ALAI '06)*, pp. 221–227, Athens, Greece, June 2006.
- [8] A. Asaduzzaman, I. Mahgoub, P. Sanigepalli, H. Kalva, R. Shankar, and B. Furht, "Cache optimization for mobile devices running multimedia applications," in *Proceedings of the IEEE 6th International Symposium on Multimedia Software Engineering (ISMSE '04)*, pp. 499–505, December 2004.
- [9] S. Rosner, M. McClain, and E. Gershon, "System and method for improved memory performance in a mobile device," US patent, Spansion LLC, 20060095622, 2006.
- [10] G. P. Brasche, R. Fesl, W. Manousek, and I. W. Salmre, "Location-based caching for mobile devices," US patent, Microsoft Corporation, Redmond, Wash, USA, 20070219708, 2007.
- [11] R. F. Squibbs, "Cache management in a mobile device," US patent, Hewlett-Packard Development Company, L.P., 20040030832, 2004.
- [12] P. Brida, J. Duha, and M. Krasnovsky, "On the accuracy of weighted proximity based localization in wireless sensor networks," in *Personal Wireless Communications*, vol. 245 of *IFIP International Federation for Information Processing*, pp. 423–432, Springer, New York, NY, USA, 2007.
- [13] P. Brida, N. Majer, J. Duha, and P. Cepel, "A novel AoA positioning solution for wireless ad hoc networks based on six-port technology," in *Wireless and Mobile Networking*, vol. 308 of *IFIP International Federation for Information Processing*, pp. 208–219, Springer, New York, NY, USA, 2009.
- [14] O. Krejcar and J. Cernohorsky, "Database prebuffering as a way to create a mobile control and information system with better response time," in *Proceedings of the 8th International Conference on Computational Science (ICCS '08)*, vol. 5101 of *Lecture Notes in Computer Science*, pp. 489–498, Kraków, Poland, June 2008.
- [15] J. Horak, J. Unucka, J. Stromsky, V. Marsik, and A. Orlik, "TRANSCAT DSS architecture and modelling services," *Control and Cybernetics*, vol. 35, no. 1, pp. 47–71, 2006.
- [16] J. Horak, A. Orlik, and J. Stromsky, "Web services for distributed and interoperable hydro-information systems," *Hydrology and Earth System Sciences*, vol. 12, no. 2, pp. 635–644, 2008.
- [17] M. Jewett, S. Lasker, and S. Swigart, "SQL server everywhere: just another database? Developer focused from start to finish," *DR DOBBS Journal*, vol. 31, no. 12, 2006.
- [18] O. Krejcar, "PDPT framework—building information system with wireless connected mobile devices," in *Proceedings of the 3rd International Conference on Informatics in Control, Automation and Robotics (ICINCO '06)*, pp. 162–167, Setubal, Portugal, August 2006.
- [19] M. Penhaker, M. Cerny, L. Martinak, J. Spisak, and A. Valkova, "HomeCare—smart embedded biotelemetry system," in *Proceedings of the World Congress on Medical Physics and Biomedical Engineering (WC '06)*, vol. 14, pp. 711–714, Seoul, South Korea, August–September 2006.
- [20] M. Cerny and M. Penhaker, "Biotelemetry," in *Proceedings of the 14th Nordic-Baltic Conference on Biomedical Engineering and Medical Physics (NBC '08)*, vol. 20, pp. 405–408, Riga, Latvia, June 2008.
- [21] V. Kasik, "FPGA based security system with remote control functions," in *Proceedings of the 5th IFAC Workshop on Programmable Devices and Systems (PDS '01)*, pp. 277–280, Gliwice, Poland, November 2001.
- [22] T. A. Ramrekha and Ch. Politis, "An adaptive QoS routing solution for MANET based multimedia communications in emergency cases," in *Proceedings of the 1st ICST International Conference on Mobile Lightweight Wireless Systems (MOBI-LIGHT '09)*, vol. 13 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 74–84, Athens, Greece, May 2009.