Status of Hand-Held Interface to Garnet Collaborative Environment

Geoffrey Fox, Sung-Hoon Ko, Kangseok Kim, Sangmi Lee, Sangyoon Oh
Community Grids Laboratory, Indiana University
24 January 2002

1. Introduction

This short note describes in order: the basic PDA adaptor architecture, The text chat and Instant Messenger tools, and finally the shared display and shared SVG export applications. This system links via the adaptor to the JMS engine running the Garnet collaboration environment. The Garnet system is built on infrastructure from the commercial Anabas collaboration framework. The paper finishes with a summary.

2. Integration Architecture

The conventional Grid Message Service (GMS) [GMS] of the Garnet collaboration system is not able to support lightweight clients, such as Personal Digital Assistants (PDAs). For the integration, we propose a new universal collaboration and access architecture, which we call Grid Message Service Micro Edition (GMSME). This supports mobile devices and desktop computers.

GMSME consists of an Adaptor (personal server), the HHMS protocol (Hand Held Message Service), and an API for application processors, such as SVG or shared export. Adaptor sits between mobile clients and GMS for the user management, message conversion, connection management, and message optimization according to the user and device specification. It looks like a typical client to GMS, and adapts data to specified client specifications. For instance, Adaptor 'listens' to all the messages and events on the Garnet collaboration system by subscribing to GMS and it delivers all the messages from mobile clients to GMS by publishing regular GMS message.

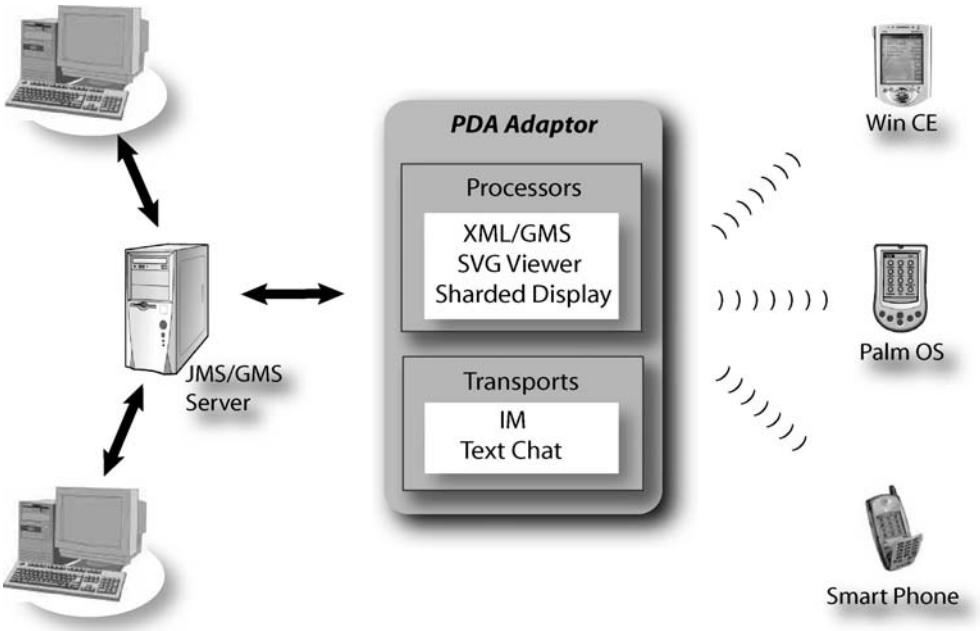
To provide universal access to the mobile devices, Adaptor has information about user and device specification. We put user profile and mobile device profile, such as screen size and device type in Adaptor. Like other objects in the Garnet collaboration system, user profile and device profile use the XML object metadata specification GXOS [GXOS]. As a result, we seamlessly exchange information from the conventional Garnet collaboration system with information on the mobile device. Also, these profiles make it possible to render messages differently for the mobile client (modified for efficiency, according to the specifications). For example, the shared export update of SVG, PDF, or HTML to a non-color and smaller-size display mobile device should be rescaled and do color modification to provide personalized and optimized look.

We consider various communication methods between mobile devices and Adaptor, which runs on a desktop. 802.11b wireless LAN [802.11b] is used as an initial communication media. Despite its security issues, 802.11b provides high-bandwidth, good range, and easy-to-apply without additional porting. But, even though 802.11b is a fast growing technology and has many benefits, wireless communication via cellular phone is dominant infrastructure because of its wide range cover and popularity. For cellular phones and most cellular communication using hand held devices, HTTP is the

most popular protocol. Thus, supporting HTTP protocol is essential for access to cellular communication using mobile device. Because of characteristic pull architecture of the HTTP protocol, we use a polling mechanism to maintain connection with the mobile device. Mobile devices make HTTP requests frequently to retrieve messages from Adaptor.

There is some basic design issue of GMSME to provide the mobile devices same level of service with desktop computer.

2.1 Optimized performance



One of the key design issues of GMSME is optimization. Since the performance of desktop computer exceeds mobile device's, optimized design is very critical. We accomplish this optimization by several design strategies.

- *Move process to server side:* To reduce process load of mobile device, we move the processing module for each application to Adaptor on desktop side. For example, before sending an event message, such as a shared export update, graphic processing is done on SVG processor or shared export processor. Because of this processing, the mobile client receives ready to use simple image from the Adaptor.
- O*ptimize protocol, HHMS:* HHMS is byte oriented message protocol between mobile client and Adaptor on desktop. Tag bytes in the byte array encode application types, event types and event message.

2.2 Extensibility

We consider two scalability requisites. The first one is standard interface (API) to processing module and to the mobile client. All information passing between Adaptor and processing modules is hashtable object of Java. Therefore, as far as the development of processing module follows specification and the module has a `getMessage(Hashtable h)` public method in the class, any additional processing

module can be added. For the communication interface to the mobile client, we have a java interface `ConnectionManager`. Handler class, which inherits `ConnectionManger` delivers byte array to and from Adaptor. As a result, byte array oriented HHMP protocol over TCP/IP or HTTP is implemented by proper handler class to the Adaptor.

2.3 Robustness

In contrast to GMS, which provides a robust message system, Adaptor must provide a recovery method. We assign log file to the personal server of each user. The log file is the list of major event. It makes Adaptor able to recover the state of each user from the initial connection or unexpected disconnection from collaboration system. And we introduce message queue to Adaptor. Message queue of Adaptor keeps the load on mobile device below the limit. Series of message, which is over the speed of process on mobile client, may cause jamming and client application on mobile device. And if so, it can not get possible emergency message. With a message queue, by putting the emergency message at the front of first message, mobile client is always open to the any emergency message.

The first step of a typical connection to Adaptor from mobile device is registering. With a unique userID, mobile device makes an initial connection to appropriate communication server: a TCP server or HTTP server, according to its major communication method type. Once Adaptor gets new user connection, it creates an instance of `UserInfoManager` class for each user. `UserInfoManager` has lightweight-process of processor modules and parser. Message communication between Mobile device and Adaptor (personal server) specified by HHMS protocols. Adaptor also publish/subscribe information of mobile client to Grid collaboration system. When leaving, the mobile client sends a log-out message to Adaptor and the instance of `UserInfoManager` is removed from the system.

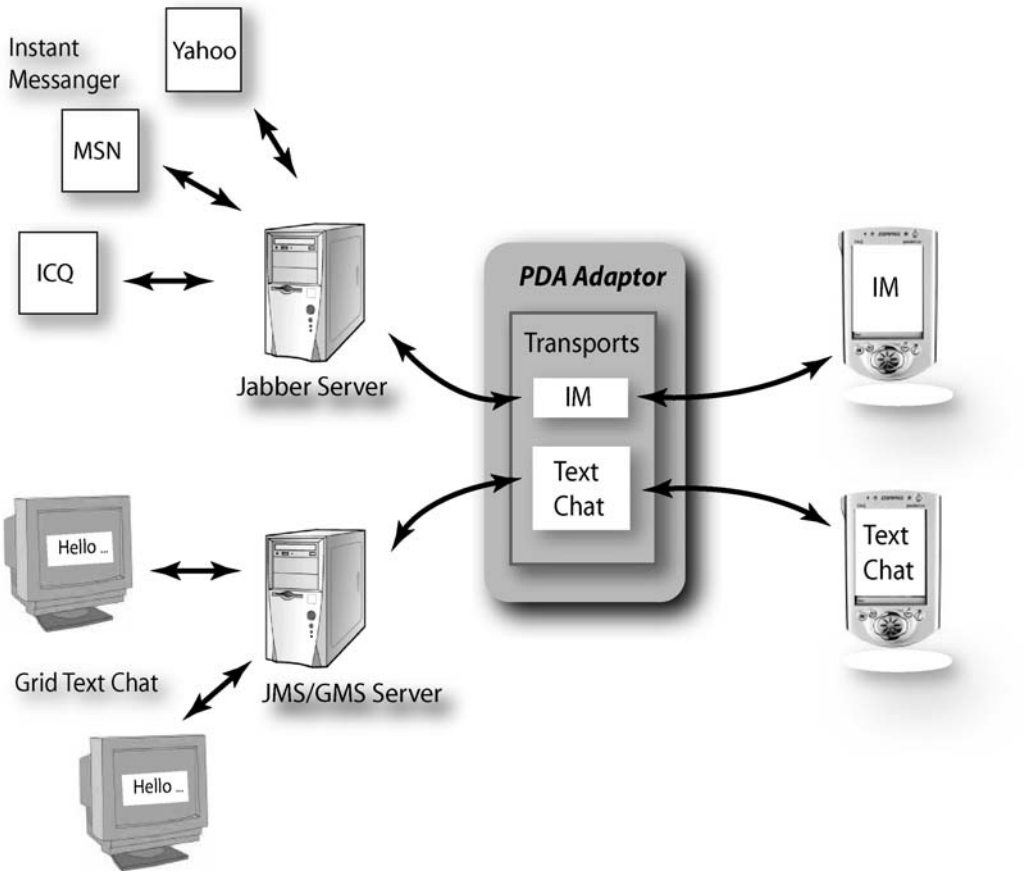
[802.11b] IEEE 802.11b Local Area Network http://grouper.ieee.org/groups/802/11/
[GXOS] http://aspen.ucs.indiana.edu/~balsoy/gxos/
[GMS] http://grids.ucs.indiana.edu/users/shrideep/narada/

3. Instant Messenger and Text Chat on iPaq

Universal access refers to the need that all users can access information systems (grids) independent of their access device and their physical capabilities. We address this for personal digital assistants (PDA's) and generalize it to universal collaboration – the capability of multiple users to link together over the web with disparate access modes. Mobile systems are typically slow, unreliable, and have unpredictable temporal characteristics. Further the user interface is clearly limited. The design of distributed

mobile applications needs to identify the practicalities, reliability and possibilities of continuous interaction and integrate synchronous and asynchronous collaboration.
A web-based collaboration is very simple but powerful asynchronous model. We can abstract the model as Publish/Subscribe mechanism and make it more useful by adding some mechanisms like Instant Messenger (IM) or Text Chat to tell the collaborating client in real time when new information is posted. Adding synchronous collaboration to the model involves providing real-time notification and automatic update for changed objects. We will present our results as proposed interfaces, portalML, which define XML based subscription and publication profiles. These are used by GMS (Grid Message Service) which generalizes the industry standard JMS (Java Message Service) to allow context sensitive publish/subscribe mechanisms. PortalML also defines an application interface on the PDA. Instant Messenger and Text Chat that will be an instance of universal collaboration applications are respectively based on the popular Jabber Open framework using XMPP (XML presence protocol) and GCF (Garnet Collaboration Framework) using GXOS. The benefits of using Jabber Open framework include presence management, message processing based on XML, transparent interoperability, structured information data, and open formats. With such an approach using open or commercial technology, in a modular fashion with appropriate interface for collaborative applications, we can build a sustainable high functionality system taking advantage of the latest technologies and enable multiple collaborative applications to re-use the same basic technologies.

The IM on iPaq is an instant messaging client on PDA's capable of interfacing with messenger services like ICQ, MSN, and Yahoo Messenger using Jabber Open framework that is Instant Messaging and Presence Managing platform based on XML, XMPP and open standards. It was developed as a "PersonalJava" ubiquitous application and allows you to send instant messages from anywhere and is interoperable with other IM services.
The IM on iPaq connects first to the Adaptor that is a personal server in Garnet Collaboration system using TCP connection and HHMS protocol, and IM transport that is one of components making up and is a program running on Adaptor establishes a connection to Jabber Open framework. Each IM user on iPaq can send messages to other IM users and check online or offline presences of IM users in real time.



The Text Chat on iPaq is also a messaging client on PDA's that was developed as a ubiquitous collaborative application in GCF. It connects to the Adapter using TCP connection and HHMS protocol and Text Chat transport that is one of components making up and is a program running on Adaptor establishes a connection to GMS (Grid Message System) server and publishes or subscribes messages between Text Chat and GCF.

IM and Text Chat Transports is respectively like processors for parsing XML messages between IM on iPaq and Jabber, and for translating messages into GMS messages between Text Chat on iPaq and GCF or agents that provide access to a non-Jabber framework and a non-Garnet Collaboration framework. The Transports are able to collaborate disparate platforms and to coordinate our works. More precisely, transports executing in Adaptor are modules (threads) providing procedures to exchange messages between disparate and diverse platforms. The Transports are composed of several internal objects and well-defined interfaces of interacting with Garnet and Jabber frameworks. Incoming messages from frameworks are immediately parsed or interpreted into HHMS messages by the transports. Outgoing messages into frameworks are converted to XML or GMS messages in the transports. The IM Transport is a server side transport that allows IM on iPaq to get roster and presence information, and performs chats with other users through Jabber Open framework. The IM Transport creates IM

session object to handle communication to Jabber Open framework. The IM Transport listens for request and reply between IM on iPaq and Jabber Open framework.
The Text Chat Transport is designed using a Sharedlet that is a collaborative application that works within GCF. The Text Chat on iPaq and GCF communicates through Sharedlet streams in the Text Chat Transport. GCF has its own private stream for accepting Text Chat requests on iPaq. Text Chat on iPaq receives GCF replies through a single public stream in Text Chat Transport. The Text Chat on iPaq and GCF communicate by sending GMS_TextMessage through Sharedlet Streams.

As a future plan, we are going to extend presently our works to mobile devices such as cellular phone using HTTP or TCP connection. It will be developed using J2ME (Java 2, Micro Edition). We can also add SMS (Short Message Service) on mobile devices for sending brief messages between devices like cell phones by using SMS-Jabber Open framework being currently in the planning phase and SMTP (Simple Mail Transfer Protocol) service on mobile devices for electronic mail used by most email services by using OKC (Online Knowledge Center) framework being in the developing phase to our universal collaborative applications. Putting SMS transport and SMTP transport to exchange messages between disparate platforms into the Adapter, we can support disparate mobile information devices for universal collaboration and access. The plans may move slower and be changed more different in the future technical directions than we expect.
Note that currently we have not integrated the Instant Messenger system into Garnet. This capability was removed from production version. One can either add IM capability to Garnet (as was present before) or modify text chat to interact with IM on PDA.

4. Shared applications on PDA

4.1 Shared Display

Shared Display is the simplest method for sharing documents with the frame-buffer corresponding to either a window or entire desktop replicated among the clients. Modest client dependence is possible with PDA's for example receiving a reduced size image. Some collaboration systems support remote manipulation with user interactions on one machine holding a replica frame buffer transmitted to the instance holding the original object. This is an important capability in help desk or consulting applications, similar to situations that occur frequently in the debugging of code. As this works for all applications without modifying them, this is the basic shared document mechanism in Pervasive Community Garnet System [1]. The public domain *Virtual Network Computing* (VNC) [3] from AT&T Laboratories Cambridge and Microsoft's *NetMeeting* [4] were two of the earliest popular collaboration systems to implement this capability. Currently, there are several collaborative systems supporting shared display including commercial products; WebEx communication Inc.'s *WebEx* [5], Centra Software Inc.'s *Centra* [6], PlaceWare Inc.'s *PlaceWare* [7], and *Latitude* [8] from Latitude Communication Inc.
Most popular approaches to the optimization for shared display are incremental update and encoding data. Pervasive Community Garnet system maintains the blocks of framebuffer and updates only blocks which are updated. The block is sent to other participants after encoded with various encoding methods. The encoding method in

Pervasive Community Garnet system is automatically selective. For instance, if *Huffman* and *LZ77* are requested but the Huffman compressor does not reduce the data, the Huffman request will be skipped and LZ77 will be just processed.
Pervasive Community Garnet system shared display is accessible from small wireless devices. The PDA Adaptor of Pervasive Community Garnet System processes the data from GMS with the protocol designed for its shared display system. The data from frame-buffer of PDA Adaptor is resized for small device, which has low resolution [2]. PDA Adaptor requests updated data periodically, and optimized data is transferred to PDA with HHMS as a Raw Bitmap data. We developed client application for Personal JAVA, runtime environment for Windows CE. This client application is particularly designed for Compaq Inc.'s *iPAQ* [9] which has resolution of 320 x 240. We communicate with high-speed IEEE 802.11 wireless connection, and application protocol of HHMS protocol based on TCP/IP. The PDA Adaptor and client application are developed with Java technology to support wide variety of current small wireless devices.
We are developing client application for Palm m505 from Palm Inc. It is developed based on Connected Limited Devices Configuration (CLDC) Java Platform [10] run able on small wireless devices such as Palm m505 [11] and smart phone. Optimized Raw Bitmap data is encoded to PNG graphic format and wrapped with HHMS protocol to communicate with Palm m505 device. It will use HTTP connection to communicate with PDA Adaptor, and support different resolution of 160 x 160. Moreover, we are investigating improving performance with optimization. Shared display for small device needs different type of optimization because of its limited hardware power and display system. As a traditional shared display system, VNC is using 16 blocks for their incremental update. However, the PDAs' low resolution display system will have fewer blocks to maintain its incremental update, and it will use light-weighted encoding technology.

4.2 Shared Export

Shared display does not allow significant flexibility; for instance different clients cannot examine separate viewpoints of a scientific visualization. More flexible sharing is possible by sharing object state updates among clients with each client able to choose how to realize the update on its version of the object [1]. This is very time consuming to develop if one must do this separately for each shared application. The shared export model filters the output of each application to one of a set of common formats and builds a custom shared event viewer for these formats. This allows a single collaborative viewer to be leveraged among several different applications. WebEx [5] uses a shared virtual printer which is achieved with shared Acrobat PDF export in Pervasive Community Garnet system.
The scalable formats W3C's Scalable Vector Graphics (SVG) [12] and Adobe, Inc.'s Portable Document Format (PDF) [13] are particularly interesting and support of

collaborative viewers for them is a major advantage of Pervasive Community Garnet System. Scalability implies that each client can resize and scroll while preserving correct positions of pointers and annotations for their various resolutions. SVG is useful as it is already available for Adobe Illustrator [14] and we can expect both PowerPoint and Macromedia Flash to be exportable to this syntax. Currently there is a Flash (which is a binary 2D vector graphics format) to SVG converter [15] from the University of Nottingham; OpenOffice.org's *OpenOffice* [16] exports PowerPoint to SVG.

We would recommend building SVG exports into tools like whiteboards and 2D scientific visualizations to allow convenient interchange among these different 2D presentation tools [1]. W3C's *Annotea* [17] allows similar general collaborative viewers to support collaborative annotation, which is anchoring to objects within the SVG document. Meanwhile, our system supports collaborative annotation at general pixel positions, which are then rendered at appropriate scaled position on each client synchronously.

Exported SVG document is shared by its URI in Pervasive Community Garnet System. Every event message from participant on Desktop is transferred via GMS service. Each PDA Adapter render image for various type of wireless devices. We define SVG elements for low-resolution devices with Cascading Style Sheets (CSS) on the PDA Adapter. Display types of small wireless devices are various: from 16bits color with 320x320 resolution [18] to 1 bit gray scale with 160x160 resolution [19]. Every image rendered by PDA Adapter is transferred to each device as a binary image format. We have developed the integrated client application for Compaq Inc.'s iPAQ, which displays the JPEG image from PDA Adaptor. All of delivered message between PDA Adaptor and client application on PDA is following the HHMS protocol.

Pervasive Community Garnet System provides the interactive framework for collaborative SVG export. As a basic function, our system supports collaborative annotation. Each event for the annotation is translated to SVG document and sent to other participants as a message object with GMS service. The received message is parsed and rendered dynamically. The SVG event processing is being developed based on the document model. Each exported SVG document creates DOM tree, any additional item is added to the DOM tree with DOM interface, and rendered dynamically. Every event message encoded with XML syntax is delivered to other participants via GMS.

Currently we developed the client application for SUN Microsystems Inc's *Personal JAVA* [20] which is runtime environment for Windows CE. We experienced that the small wireless devices have a wide variety of hardware powers and network environments. Although Java technology provides us the advantage of code mobility which is dubbed "Write Once, Run Anywhere™", we had to consider particular hardware environment such as network communication and image formats. We are developing the client applications for Connected Limited Devices Configuration (CLDC) Java Platform [10] run able on small wireless devices such as Palm m505 and smart phone.

4.3 References

[1] G. C. Fox. *Architecture and Implementation of a Collaborative Computing and EducationPortal.* ERDC Technical report, May 2001
http://aspen.ucs.indiana.edu/collabtools/

[2] G. C. Fox. *Ubiquitous Access for Computer Science and Education.* ERDC Technical report, May 2001, http://aspen.ucs.indiana.edu/collabtools/

[3] AT&T Laboratories, *VNC - Virtual Network Computing from AT&T Laboratories Cambridge*, http://www.uk.research.att.com/vnc/

[4] Microsoft, Inc. *NetMeeting: Microsoft commercial product description from http://www*.microsoft.com/netmeeting

[5] WebEx Communication, Inc. *WebEx*. http://www.webex.com

[6] Centra Software, Inc. *Centra*. http://www.centra.com

[7] PlaceWare, Inc. *PlaceWare*. http://www.placeware.com

[8] Latitude Communication, Inc. *Latitude*. http://www.latitude.com

[9] Compaq, Inc. *Compaq iPAQ Pocket PC 3765 Detailed Specifications*, http://athome.compaq.com/showroom/static/iPaq/3765.asp

[10] Sun Microsystems. Connected, *Limited Device Configuration: Java 2 Platform Micro Edition*. Sun Microsystems, version 1.0 edition, May 2000. Available at http://java.sun.com/ products/cldc/.

[11] Palm, Inc. *Palm m505 Handheld Product Details*, http://www.palm.com/products/palmm505/details.html

[12] J. Ferraiolo, ed., *Scalable Vector Graphics (SVG) 1.0 Specification*. August 2000. http://www.w3.org/TR/2000/CR-SVG-20000802/.

[13] Adobe Systems, Inc. *Portable Document Format*. http://www.adobe.com/products/acrobat/adobepdf.html

[14] Adobe Systems, Inc. *Adobe Illustrator*. http://www.adobe.com/products/illustrator/main.html

[15] SVG and the Electronic Research Group, *Flash to SVG*, http://broadway.cs.nott.ac.uk/projects/SVG/#project

[16] OpenOffice.org , *OpenOffice.* http://www.openoffice.org

[17] J. Kahan, M. Koivunen, E. Prud'Hommeaux, and R. Swick. *Annotea: An Open RDF Infrastructure for Shared Web Annotations*. In Proc. of the WWW10 International Conference. Hong Kong, 2001.

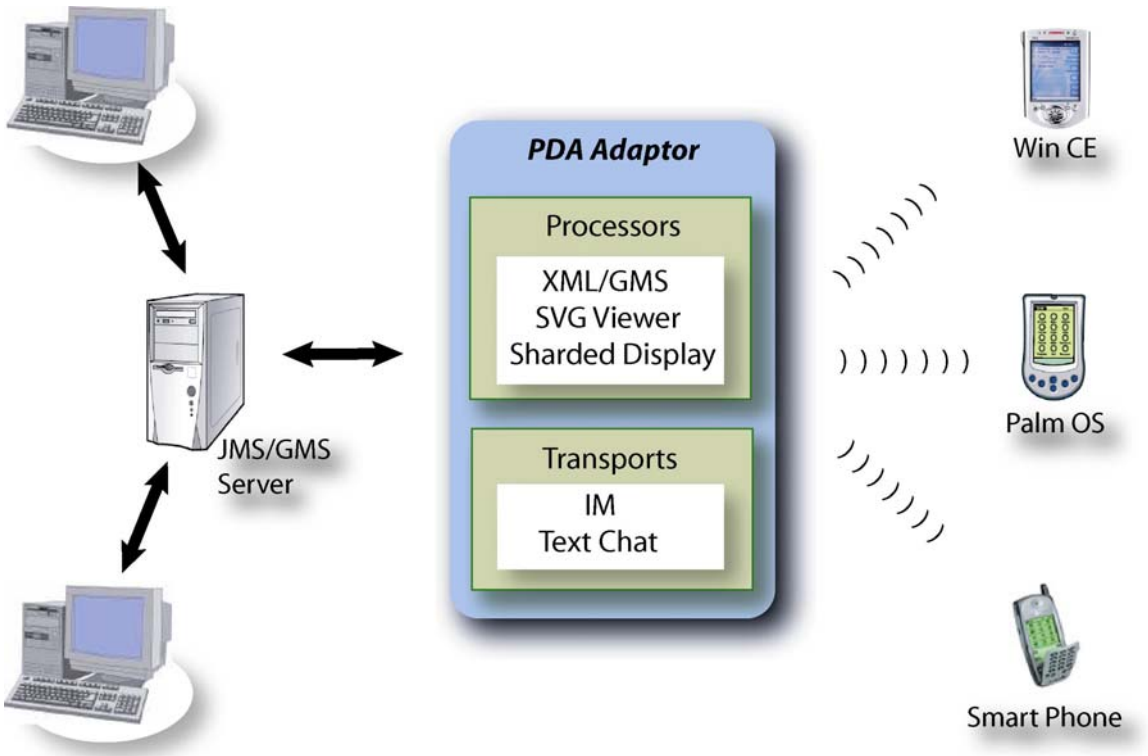
[18] Sony, Inc. CLIE Personal Entertainment Organizer Specifications, http://www.sonystyle.com/micros/clie/760specifications/PEG-N760C.html

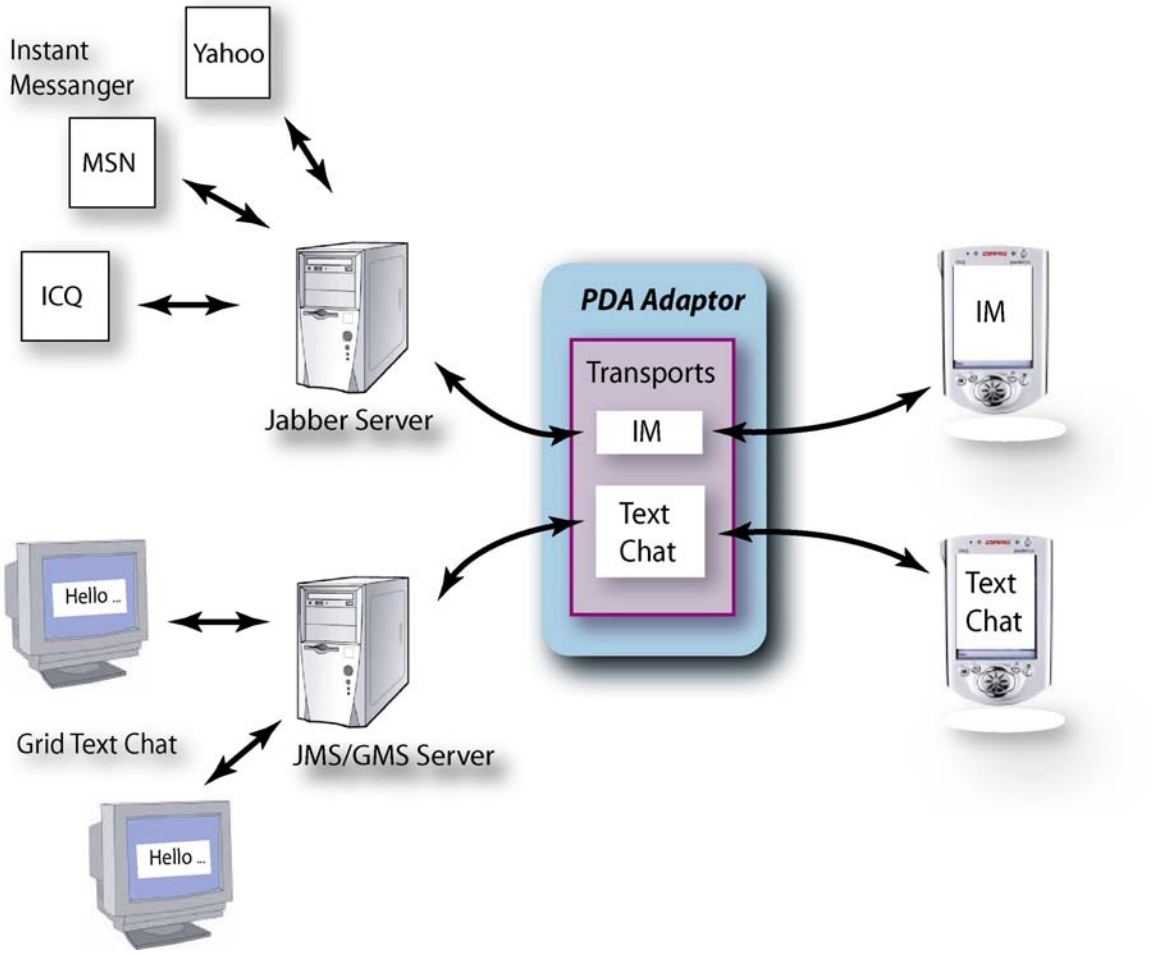
[19] Palm, Inc. Palm m500 Handheld Product Details, http://www.palm.com/products/palmm500/details.html

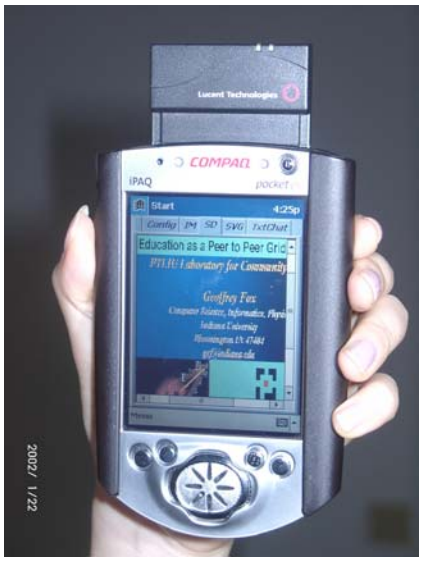
[20] Sun Microsystems Inc. *Personal Java 1.0 Specification*. http://java.sun.com/products/personaljava/ spec-1-0-0/personalJavaSpec.html


COMPAQ
pocket pc
Config
IM
SD
SVG
TxtChat
IP Address:
156.56.104.42
Port number:
7000
Meeting ID:
650
User Name:
kangskim
Nick Name:
kangskim
Password:
****
Jabber Server:
jabber.org
Machine ID:
1
OK
Messenger
- Friends
kdkim@jabber.org <Available>
sangycon
shko
sungko@jabber.org <Online>
- Gamet
gametjabber@jabber.com
gametjabber@jabber.org
- jabber.com
kangdkim@jabber.com
kdkim@jabber.com
- Professor
gametfox
kdkim@jabber.org
Time: Mon Jan 21 17:50:28 EST 2002
kdkim says: Hello!
kangskim says: Hi
kdkim says: How are you today?
kangskim says: very good
Send
Close
Chat With
550
Time: Mon Jan 21 17:48:14 EST 2002
sungko: How are you?
kangskim: Pretty good
send
Anatomy of arrows
Architecture of Collaboratio


COMPAQ
pocket pc
Config
IM
SD
SVG
TxtChat
IP Address:
156.56.104.42
Port number:
7000
Meeting ID:
650
User Name:
kangskim
Nick Name:
kangskim
Password:
****
Jabber Server:
jabber.org
Machine ID:
1
OK
Messenger
- Friends
kdkim@jabber.org <Available>
sangyeon
shkc
sungko@jabber.org <Online>
- Garnet
garnet.jabber@jabber.com
garnet.jabber@jabber.org
- Jabber.com
kangskim@jabber.com
kdkim@jabber.com
- Professor
garnetlee
Chat V
kdkim@jabber.org
Time: Mon Jan 21 17:50:28 EST 2002
kdkim says: Hello!
kangskim says: Hi
kdkim says: How are you today?
kangskim says: very good
Send
Close
Chat With
650
Time: Mon Jan 21 17:48:14 EST 2002
sungko: How are you?
kangskim: Pretty good
send
Anatomy of arrows
Architecture of Collaboratio
Transformed Object
Original Object
Renderer


PDA Adaptor
Processors
XML/GMS
SVG Viewer
Sharded Display
Transports
IM
Text Chat
JMS/GMS
Server
Win CE
Palm OS
Smart Phone


Instant
Messanger
Yahoo
MSN
ICQ
Jabber Server
PDA Adaptor
Transports
IM
Text
Chat
IM
Text
Chat
Hello ...
Grid Text Chat
JMS/GMS Server
Hello ...


Lucent Technologies
COMPAQ
iPAQ
pocket pc
Start
4:25p
Config
IM
SD
SVG
TxtChat
Education as a Peer to Peer Grid
PTLIU Laboratory for Community
Geoffrey Fox
2002/ 1/22


Lucent Technologies
Start
4:21p
Config
IM
SD
SVG
TxtChat
Menu


Lucent Technologies
COMPAQ
iPAQ
pocket pc
Start
4:20p
Config
IM
SD
SVG
TxtChat
Messenger
2002/ 1/22


Lucent Technologies
COMPAQ
iPAQ
pocket pc
Start
4:26p
Config
IM
SD
SVG
TxtChat
2002/ 1/22