A Mobile Gaming Platform for the IMS

Amjad Akkawi
IBR, TU Braunschweig
Mühlenpfordtstraße 23
38106 Braunschweig, Germany
a.akkawi@tu-bs.de

Sibylle Schaller
NEC Europe Ltd.
Kurfürstenanlage 36
69115 Heidelberg, Germany
schaller@netlab.nec.de

Oliver Wellnitz
IBR, TU Braunschweig
Mühlenpfordtstraße 23
38106 Braunschweig, Germany
wellnitz@ibr.cs.tu-bs.de

Lars Wolf
IBR, TU Braunschweig
Mühlenpfordtstraße 23
38106 Braunschweig, Germany
wolf@ibr.cs.tu-bs.de

ABSTRACT
Mobile devices offer the opportunity to play games nearly everywhere. Moreover, networked games allow individual players to interact with other people and to participate in a larger gaming world, which also provides for new business opportunities. Hence, we currently see an increased interest from game developers, providers and players in mobile games. In this paper we propose a novel architecture and platform for games on the IMS. This allows games to utilize the features and capabilities that are inherent to the IMS. At the same time existing games can be flexibly adapted to this new type of network and have the possibility to reserve network resources for game data transmission, thus improving the experience of players.

Categories and Subject Descriptors
K.8.0 [**General**]: Games, C.2.1 [**Network Architecture and Design**]: Wireless communication

General Terms
Design, Standardization.

Keywords
IMS, Mobile Networked Games, Platform Architecture.

1. INTRODUCTION
Until recently, most games were stand-alone applications designed for only a single player. This has changed lately, where many games are directed towards a multi-player scenario where people can interact and compete. Today, most of these networked multiplayer games are developed for PCs connected to fixed networks. Especially in the early days games had to cope with severe bandwidth limitations, e.g. due to narrowband modem links. Nowadays, many computers have access to broadband services (e.g. via ADSL).

Multiplayer games can be distinguished by their flexibility with regard to location and network connectivity. In online multiplayer computer games and console games the players are usually connected via a fixed line network and thus quite restricted for choosing their location. Players on mobile handheld devices are more flexible: they may play in different locations and/or networks, and they may even move while playing.

Currently, game developers, providers and players get more and more interested in games that can be played everywhere. Thus, games must be available on mobile devices

Standards organizations and other related bodies have agreed to co-operate for the production of a complete set of globally applicable technical specifications for a 3rd Generation Mobile System based on the evolved GSM core networks and the radio access technologies supported by 3GPP (Generation Partnership Project [1]) partners. The technical specifications will be developed in view of global roaming and circulation of terminals. The first release of the 3GPP 3G standard has stabilized, and the first 3GPP compliant networks are going into operation. Work is currently ongoing for 3GPP Release 6. [2] From 3GPP specifications, a complete solution for the support of IP multimedia applications (including voice communications) shall be available. The solution consists of terminals, GERAN (GSM EDGE Radio Access Network) or UTRAN (UMTS Terrestrial Radio Access Network) radio access networks and GPRS evolved core network. [3] UTRAN is a new radio network architecture. It provides transmission capabilities that allow for more bandwidth for data and better quality for voice as compared to GSM and GPRS. The 3GPP IP Multimedia Subsystem (IMS) enables a "platform" with capabilities like presence, multimedia conferencing, messaging, and support for QoS (Quality of Service) for data traffic, etc.

Wireless LAN (WLAN) is another type of access network supported by the 3GPP IMS. WLAN – according to the IEEE 802.11 standards family – provides bandwidth from 2 up to 54 Mbit/s. The actual bandwidth available per device decreases proportional to the number of the connected, active devices.

By use of IMS service capabilities and standards protocols, a new QoS for games as well as better performance and scalability can be achieved. Complex services can be created and integrated into a game based on simple IMS services, such as Instant Messaging and/or special services, such as a scoring service. The availability of information like location, terminal capabilities and presence status can significantly ease the development and feature enhancement of games.

In this paper we propose a novel architecture and platform for games on the IMS. This allows games to utilize the features and capabilities that are inherent to the IMS. At the same time existing games can be flexibly adapted to this new type of network. Another advantage is the possibility to reserve network resources for game data transmission, thus improving the experience of players.

NetGames'04, Month 1–2, 2004, City, State, Country.

2. Related Work

It seems that there is no other work directly related with the topic of the paper: a gaming service platform for mobile, networked games over the IMS.

The term Game Platform usually seems to be bound to some hardware as e.g. the Nintendo Gamboy [4], Sony Playstation [5], or Nokia N-Gage [6]. Connected with those platforms is a development kit that enables game developers to integrate their games via usually proprietary interfaces.

The game platform described in this paper is not tied to any hardware platform. Also we took the approach to base the platform architecture on standardized components and interfaces.

3. A General Game Service

After a brief introduction to aspects of multiplayer games, this section explains the components that make up game services.

3.1 Aspects of Multiplayer Games

Multiplayer games allow two or more people to play together or against each other in the same game. Networked multiplayer games are playable over a network (e.g. the Internet). Since the first video game for computer (Space War) in 1961 and the first multiplayer game in 1969, the multiplayer game world has faced many challenges.

Despite both the network and hardware limitations, a variety of multiplayer games were produced. The communication range, speed, network coverage, bandwidth and latency, as well as parameters of the game client devices (processor, memory, graphics, etc.) have an influence on what kinds of multiplayer games can be developed. These technical issues can have a significant impact on the applicable techniques for specific types of games.

Independent of the type of network that connects the players with the game, the physical limitation of the network cannot be ignored. Important limitations are the scarcity of resources, interferences, etc. on the radio link, leading to small bandwidth and high latency. Several communication models and design technologies deal with these limitations. The communication model can be Peer-to-Peer, Client-Server based, or Hybrid. [7][8][9]

3.2 Components in a General Game Service

Figure 1 shows the components used generally for game services. The Game Service is the sum of the contributions of all these components.

The central component is the *Gaming Service*, a server-side platform providing network connectivity and general support for gaming. A *Game Provider* is a human, an organization, or a collection of humans and/or organizations that own a game application, or have the right to use and publish a game application (e.g. Tetris, Quake, Age Of Empires). The Game Provider publishes and distributes games via the Gaming Service.

Potential game players must install the game client(s) in their terminals. For mobile terminals, for example, this could be done with Over-the-Air provisioning. [10] An authenticated and authorized player can either join a (running) game session of his choice or create a new game session. A game session is a single gathering of players to participate in a particular instance of a game. A game session lasts for some period of time.

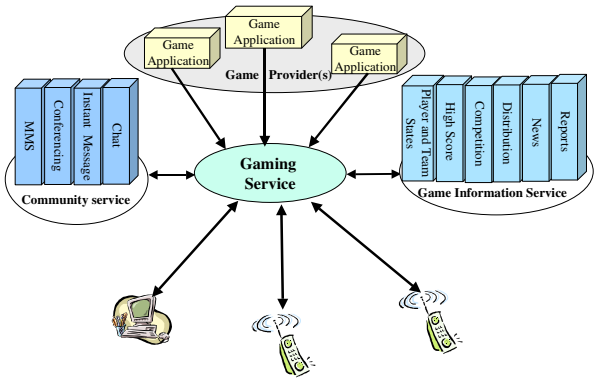

Figure 1: Components in a General Game Service

The *Community Service* consists of different services that enable the players to communicate. Examples for Community Services are: Instant/Voice/Multimedia Messaging, Chat or Conferencing. One or more of these services may be integrated in a game and the player has the choice to use, join or leave them without having to interrupt the game.

The *Game Information Service* contains functionalities for the management of game related information. It may contain information as listed in Table 1.

Table 1: Game Related Information

Type of Information	Description
Scores and Competition	Past and current scores, and competition information about past, current and future games. Each game may have one or more Score and Competition Information tables.
Players and Teams	Contains information about the game players and their teams. It may also contain information about a player's profile.
Downloading and Distribution	Information about new releases, new updates, demonstrations and levels. This information may also contain a download link.
News	Headlines, messages-of-the-day, previews, screenshots, advertisement, etc.
Reports	A report may contain different information collected about games, e.g. during game launch. They give a game provider the possibility to make development and marketing decisions. Reports may contain statistical information about the players' requests and interests.

For the management, collection and distribution of game information the Gaming Service may use a Presence Service. Game presence information is a set of attributes characterizing the

current status of a game such as score(s), player(s) and team(s), competition information, etc. The Game Information Service manipulates game presence information. Players and watchers of games can subscribe to all or parts of this information and may also be notified of any changes. Game watchers are users that do not (yet) actively participate in a game, but are interested in obtaining game related information and/or using game community services.

4. Architecture for the Integration of Games over the IMS

In this section we first motivate the utilization of the 3GPP IMS for game services. Then we explain how the architecture looks for game services over the IMS.

4.1 Why Games over 3GPP IMS

3G combines two of the world's most powerful innovations, the Internet and wireless communications. Having access to the Internet in mobile devices will be of great value to many users.

The 3GPP IP Multimedia Subsystem (IMS) is a standardized infrastructure, able to run services of all categories while allowing ease of inter-working between (mobile) operators. [11] It will allow:

- Peer to peer, real-time and non-real-time services delivered over the packet switched domain.
- Seamlessly combined services with different QoS categories.
- Only one infrastructure is required for the support all types of services, keeping operators' cost of the network solution, infrastructure and terminals (CAPEX) and cost of keeping the network up and running (OPEX) low.

From a technical point of view, the IMS provides a horizontal approach to integrated, real-time, multiparty services for mobile networks. The IMS services could be integrated or combined in the gaming service and so providing an advanced game experience (see also section 6). The Session Initiation Protocol (SIP) [12] was chosen as the signaling protocol. It allows determining capabilities of user terminals, to negotiate QoS parameters, and to use and switch between media components as needed.

4.2 The IMS

The first release of 3GPP architecture, Release 99 (R'99), defines the basic architecture of the network. R'99 was designed to be backward compatible with the existing GSM circuit switched infrastructure. The IMS is not part of this and the following release. The so-called Phase 1 IMS is specified in 3GPP Release 5 and contains the basic mechanisms for multimedia session management. The 3GPP Release 6 adds many new capabilities to the system and the IMS. Examples are: Presence, Conferencing, Messaging, WLAN-Interworking, and Group Management.

The IP Multimedia Subsystem (IMS) is an extension of the 3GPP packet-switched core network. It uses SIP to setup, maintain and terminate voice and multimedia sessions. The IMS network architecture is specified in [13].

The Call Session Control Function (CSCF) in IMS is needed for session management and support for QoS provisioning in the core network. The CSCF can act as Proxy CSCF (P-CSCF), Serving CSCF (S-CSCF) or Interrogating CSCF (I-CSCF). The P-CSCF is the first contact point for the UE (User Equipment = terminal) within the IMS; the S-CSCF actually handles the session states in the network; the I-CSCF is mainly the contact point within an operator's network for all IMS connections destined to a subscriber of that network operator, or a roaming subscriber currently located within that network operator's service area.

The Home Subscriber Server (HSS) is the master database for a given user. It is the entity containing the subscription-related information to support the network entities actually handling calls/sessions. The HSS also generates User Security information for mutual authentication, communication integrity check and ciphering.

The Media Resource Function (MRF) is split into the MRF controller (MRFC) and MRF processor (MRFP). The MRFC controls the media streams resources in the Media Resource Function Processor (MRFP). The MRFC interprets information coming from the Application Server (AS) and the S-CSCF (e.g. session identifier) and control the MRFP accordingly. The MRFP controls and provides resources to be controlled by the MRFC. It mixes the incoming media streams (e.g. multiple parties), is also a source of media streams (for multimedia announcements), and processes media streams (e.g. audio trans-coding, media analysis).

5. The Gaming Platform

In this chapter we introduce our gaming platform for game services over the IMS.

The Game Platform proposed in this paper "sits" between the games and the 3G network (see Figure 2). It allows using the IMS capabilities (e.g. presence, messaging, QoS), but also may offer additional functionalities that every game needs (e.g. player and game management). The platform API is used on both client and server side of the game, but with different functionality. For example, when a game server does *Create_Application*, a new game session is created. On the client side *Create_Application* either lets a player join a game session or create a new one. The following section describes more details about the Gaming Platform.

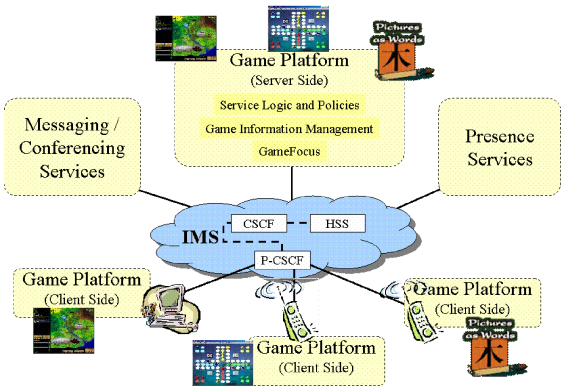

Figure 2: Game Service over the IMS and Using IMS Capabilities and Functionalities

5.1 A Standard API from the Gaming Platform towards the Games

The Gaming Platform utilizes the OMA Game Services API to interface with both the Game Server(s) and Game Clients (Figure

3). [14] The OMA API provides session management functionalities, connectivity, metering, scores and competition managements, logging as well as timers.

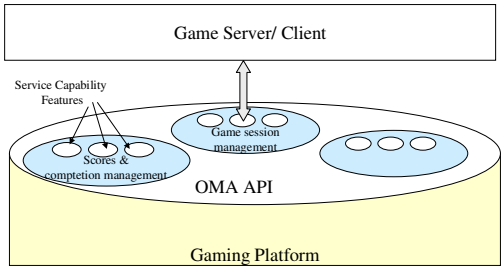

Figure 3: Gaming Platform utilizing the OMA Game Services

Session management provides the identifiers that bind the user interactions into the single concept of a game. It provides access to the other API functions, like, for example, for retrieval and storage of player scores. Additionally it is the interface through which the lifecycle of a game can be managed (including Game Information, Game Sessions, Community Services, etc.).

By providing connectivity functionalities the OMA Game Services API makes the game logic transparent from the network layer and protects the developer from the low-level implementation details of the transport mechanism.

Metering provides a standard interface through which the game can inform the gaming platform of game specific billable events.

Scores and Competition Management provides the mechanisms by which the game can report and retrieve scores from the gaming platform, thus allowing competitions to be run in a unified manner.

Logging provides a standard reporting mechanism by which a game informs the gaming platform of its status. This insulates against specific formatting requirements and through the implementation of variable logging levels, assists in the troubleshooting process.

Timers provide a mechanism by which a game and/or other activities can be scheduled or delayed.

5.2 The Game Application

The game application is not part of the Gaming Platform. Usually the game application consists of two sides: client and server. They are located on top of the game platform using the Gaming Platform functionalities via the OMA Game Services API. The game server makes sure that the actual game data flows are available to the player on the client side in the game.

The Game Server and the Game Focus communicate with each other to setup a new game session and to add or remove a player from a game session. Each Game Server is associated at least with one game session. A Game Server may be capable to provide more than one game session. A Game Server can initiate a request towards the Game Focus to a setup game session and to be associated with it, or the Game Focus can initiate a request to some pre-configured Game Server to start and associate a game server with a game session.

The Game Client(s) may learn about games and game session by using game presence information. A Game Client manages its participation in game sessions via the Game Focus using the SIP protocol. By using a presence service the client could, e.g., subscribe to updates about changes of the highest score in a certain game. Or, a user may be able to watch a game, to wait for a certain state to be reached, and only then to join the game.

5.3 The Platform Components

In this sub-section we describe the components of the gaming platform: Game Focus, Game Logic and Policies including Game Information Management, APIs inside the platform. Please refer also to Figure 2 on page 3.

5.3.1 The Focus

The central component of Gaming Service architecture is the Game Focus. The Focus maintains a SIP signaling relationship with each player in the game.

The Game Focus is responsible for the game session management including creating new a game session, removing a game session and updating a game session by joining/leaving of a game player to/from the game session. The focus serves the user's request, and forwards it to the "Service logic and policies" component.

The Focus has access to the game policy (composed of membership policy, messaging policy and game presence policy, proxy policy), an instance of which exists for each game. The game policy can be thought of as a database that contains policy information about how the focus should operate and may also contain player authorization information. It is the responsibility of the focus to enforce those policies. The focus must know about policy changes and eventually react to it. Such changes might result, for example, in the termination of a dialog with a player by sending a SIP BYE message. Most policy changes will require a notification to subscriber(s) via the game notification service.

The game Focus may function as a proxy and forward a received SIP-Request to another Game Focus. This may occur according to the game server state, game client capabilities and/or the user location, which may be retrieved through a query to the HSS by the Game Focus.

A game Focus may provide proxy functionalities for access to IMS Messaging capabilities that are supported in a game. A user who wishes to access to those services initiates a SIP request with an indicator for the messaging service targeted to the game Focus. The Game Focus will enforce the messaging and proxy policy installed for that user and forwards the request accordingly.

Similarly, the Game Focus may provide proxy functionalities for access to game presence information. Game presence information may be provided by the IMS Presence services. Whenever there is a need to inter-work with the IMS Presence Service for game and player related presence information, the focus takes the role of the so-called presence user agent. The IETF SIMPLE protocol is used to publish and obtain presence information. [15][16]

A user, who wishes game presence information, initiates a SIP-SUBSCRIBE request targeted to the game Focus. After authorizing the user (watcher) the Game Focus forwards the request to the respective Presence Service. Forwarding the request is based on game presence policies and proxy policies.

The proxy function of the Game Focus will be used in the following situations:

- Forwarding SIP-requests for joining a game to another game Focus, e.g. depending on user location, game

client capabilities, or other game performance considerations.

- Forwarding SIP-requests for joining an IMS Messaging Service to the IMS Messaging Service, e.g. in case the Messaging Service supported in the game must be joined through the game Focus.
- Forwarding SIP-SUBSCRIBE requests to the IMS Presence Service, e.g. in case a subscription for game presence information must be done through the Game Service.

Another functionality of the Game Focus is the Presence User Agent functionality. Here the Game Focus collects and sends user related Game Presence Information to a presence server on behalf of a game. The Presence User Agent shall provide the following functionality:

- Collecting Game Presence Information associated with a game.
- Assembling Game Presence Information in the correct format. For presence information the game platform uses the SIMPLE protocol [16] to publish and retrieve Game Presence Information.
- Sending Game Presence Information to the IMS Presence Server (so-called presence information publication).
- Managing subscription authorization policies.

5.3.2 Service Logic and Policies

Figure 4 shows elements of the Services and Policies component in the Game Platform.

The game policy can be thought of as a database that describes the way that the Focus should operate. It is the responsibility of the Focus to enforce those policies. The Focus needs also to know when some policy changes, because such changes might have to result in SIP-signaling (for example, terminating of a dialog with a player, by sending SIP-BYE). Most changes will require a notification to be sent to subscribers using the game notification service.

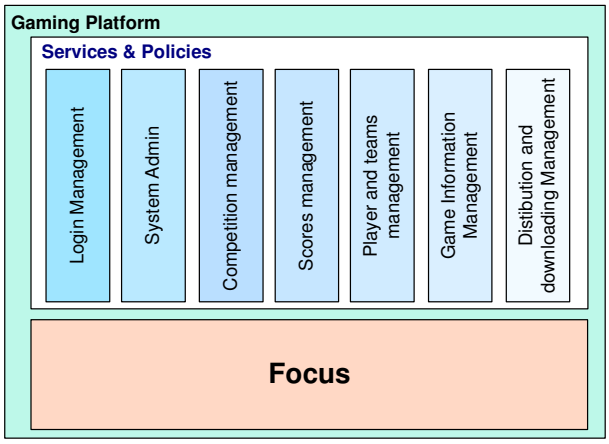

Figure 4: Services and Policy based on Game Information Management

The login management service has the task to validate requests and to authorize users, who want to join a game. Also, the login management is responsible to route the users to the appropriate game application server. This decision will be based on the proxy policy and/or some other algorithm (e.g. load-balancing, location, based).

The system admin service manages game server tasks such as managing game policies, start and stop of community service. It is also responsible for process control and performance monitoring for all servers like game servers, dispatching server, and so on.

The distribution and downloading management service is responsible for the distribution and download of game related information and software. Information about how to download game software may be part of the game presence information, e.g. a download link. For the actual download an Over-the-Air mechanism (OTA) could be used.

5.3.3 Game Information Management

Game Information Management collects game presence information from the Game Servers (such as updates on game player score) and from the game Focus (e.g. which player is involved in which game session). The Game Information Management can be thought of as database that collects different kinds of information. This information can be used, for example, for generating reports about the game servers or game sessions that can be useful for the game providers.

Game information can be divided in two categories: private game presence information and public game presence information. The public game presence information contains all the game presence information that a user (watcher) can request and obtain without being in the game. The private game presence information contains all the game presence information that just the player in game can request. The Game Information Management has access to the game presence policy and determines the watcher who can access to this information.

The game information can be collected in different components and services of the game platform. These may be score management, player and team management and competition management, etc.

5.4 Distribution of Platform Elements

The gaming platform is distributed over the three different components Gaming User Agent, Gaming Server and Gaming Server Agent as shown in Figure 5.

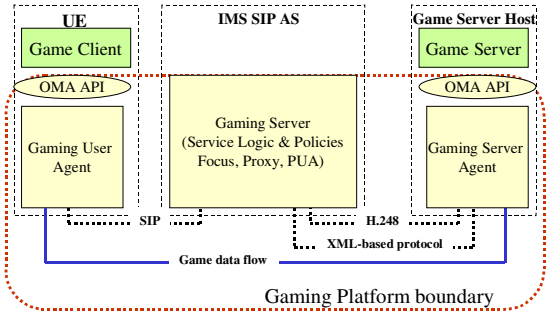

Figure 5: Distribution of the gaming platform

5.4.1 Gaming User Agent

The Gaming User Agent on the user's equipment (UE, terminal) provides the game client with network connectivity and access to the gaming platform via the OMA API. It hides the underlying SIP protocol from the game client. In addition, the OMA API provides connectivity also to the game server, making the game client transparent from the network layer of the equipment.

5.4.2 Gaming Server

The Gaming Server implements the main part of the Gaming Platform. It provides the functionalities of the Game Focus, Service Logic and Policies, and Game Information Management. The Gaming Server communicates with the game client through the Gaming User Agent based on SIP. With the SIP based communication between the Gaming Server and the Gaming User Agent, game sessions and game data flows are managed, i.e. players can join games, create a chat or video session in parallel to an ongoing game, obtain information about games, etc. Towards the Game Server Application (Game Server Host) the Gaming server interfaces with the Gaming Server Agent using H.248 (Megaco) [17] and a XML-based protocol (see also sub-section 5.4.4).

5.4.3 Gaming Server Agent

The server side part of the gaming platform is separated into 2 components: Gaming Server and Gaming Server Agent. This separation was done to allow for both flexibility with regard to the integration of different games into the platform, and scalability of the platform and the amount and structure of game servers.

The Gaming Server Agent provides the game server with network connectivity as well as access to gaming platform. Like the Gaming User Agent, the interaction between the game server and gaming platform is based on the OMA API.

5.4.4 API's Internal to the Game Platform

On the game server side the platform uses two protocols for the communication between the Gaming Server and the Gaming Server Agent.

5.4.4.1 The Megaco Protocol

In the IMS the Megaco protocol is specified for the control of media resource processing (please refer also to sub-section 4.2). The Megaco protocol is used between elements of a physically decomposed multimedia gateway, i.e. a Media Gateway and a Media Gateway Controller. The IETF specification document RFC 3015 [17] is common text with ITU-T Recommendation H.248. The latest, most up-to-date IETF document describing the protocol is RFC 3525 [18]. It contains corrections and clarifications of RFC 3015 that are based on the continued cooperation between IETF and ITU-T.

In our platform architecture the Game Focus takes a part of the role of Media Resource Controller, and the Game Server takes part of the role of Media Resource Processor. For that reason the Megaco protocol is used between the Gaming Server and the Gaming Server Agent.

5.4.4.2 The XML-based Protocol

Megaco is limited for multimedia session control. Because the Game Server also provides application-based information via the OMA API an XML-based protocol was chosen in addition. This XML-based protocol transfers between the Gaming Server Agent and the Gaming Server components of the Gaming Platform specific game related information like scores, in-game presence information, etc.

6. Using IMS Services and Capabilities via the Gaming Platform

6.1.1 IMS Presence Service

The IMS Presence Service is interesting for games. First of all, it enables publishing of games related information based on a standard protocol. It can also be interesting for the dynamic management of a group of users and players that have the same interest, and for or a dynamic grouping of several players according to their presence status, their location obtained from the HSS and/or their capabilities. The IMS Presence Service can be extended to support other additional information than already specified for supporting game presence information, such as scores and competitions. Using standard protocols for providing and supporting presence information brings several benefits for the game developers, service providers as well as the users.

6.1.2 IMS Messaging Services

Interactivities between the players during a game or players interested in a game added a significant factor for success of the game. 3GPP defines three types of IMS Messaging: immediate, deferred delivery and session-based messaging. While for the first one the message delivery is (near) real time fashion, for the second type it is expected that the message delivery will take place as soon as the receiver becomes available. The third messaging type requires that the sender and the receiver have to join a session in order to send and receive messages, e.g. chat and Conference. Different games scenarios may prefer one message type rather than the other.

IMS release 6 works also on the specification of conferencing. Conferencing Service is the most powerful interactivity that can be supported for the game players. A Game player can combine different communication techniques according to his terminal and network capabilities to communicate with others. Such communication techniques can be audio, video and text.

6.1.3 QoS

Having a high-speed network access is not enough to provide a high quality of game service as discussed in 3.1. Several games have high requirements for the network, such as the maximum delay. QoS provisioning is supported by the IMS. Users can specify the QoS they would like to be guarantied for a particular game. The IMS authorizes the QoS requirements. This is done in SIP signaling messages. After QoS authorization the game application can do resource reservation in the network. The QoS enforcement of the 3G network may significantly improve the game experience of the users.

7. Example for a User Joining a Game Session

In this section we present an example scenario, where a user joins and leaves a game session, and subscribes to a game information notification service.

Let's assume that the user knows the game session URI of a game he is interested in. The advertising of game sessions can take place through different mechanisms. Examples are a public known

SIP URI, the IMS Presence Service or a message service that distributes information about the game session, including the URI.

Figure 6 shows an example signaling flow that will be explained below.

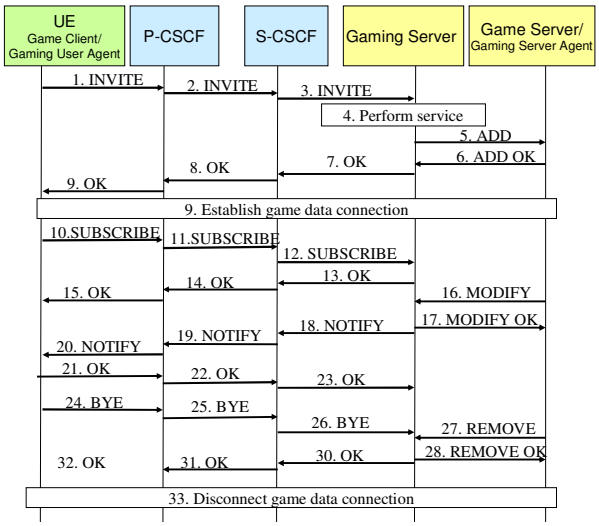

Figure 6: Example Signaling Flow

The user sends a SIP INVITE message containing the SIP URI of the game session to the Gaming Server (Game Focus). After receiving the SIP INVITE message, the Gaming Server performs certain tasks such as authorization and authentication of the user, and enforcement of different policies. Then the Gaming Server sends an ADD message to the Gaming Server Agent (Game Server). The Game Server adds the new player to the game and allocates resources according to the session description that was sent with the INVITE message. Afterwards the Gaming Server responses to the user with a SIP OK message together with the Game Server's session description. Then user establishes the game data connection with the game server based on the received session description of the game server. Now the user can start playing.

Because the user is interested to obtain the latest game related information such as the top score, he issues a SIP SUBSCRIBE message to the Gaming Server (Game Focus). The Gaming Server accepts his subscription and responds with a SIP OK message. Some time later, the Game Server modifies some game information. The Gaming Server immediately notifies the user about the change by sending a SIP NOTIFY message. The User responds with a SIP OK message.

The IMS Presence Service may be involved in the subscription and notification service for the user. Then the Gaming Server publishes the respective game information on behalf of the Game Server.

The user leaves the game session by sending a SIP BYE message to the Gaming Service. The Gaming Server removes the user from the game session and informs the Game Server by sending REMOVE. After a successful removing allocated resource for that user, the game data connection is closed.

8. An Example Implementation

For our architecture two games were integrated: TetriNet and GNU Arcade Volleyball. TetriNet was tested over GPRS as well as WLAN networks. TetriNet was playable over both types of networks. Because Volleyball has most stringent requirements in regards to delay, the game was playable over WLAN, but not over the current GPRS network. In the GPRS network a delay of about one second was experienced.

We implemented a prototype of the architecture described in the previous section and integrated the real-time multiplayer game TetriNet, a Tetris game (see also Figure 7). The original TetriNet supports up to 6 players in each game session. For the client device a PDA (HP IPAQ) was chosen. Due to the small screen size of the PDA, we limited the number of possible players to 4. For our implementation we used the NIST SIP stack implementation, which required JVM 1.2. In order to run on the PDA, we modified the SIP stack to run on JVM 1.1, which is running on the PDA.

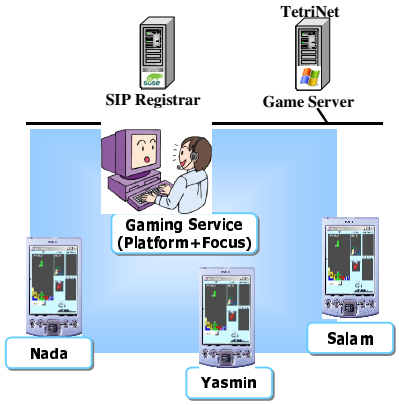

Figure 7: A Prototype Implementation of the Gaming Service

The following entities were implemented:

1. A game client manager, the Gaming User Agent (as described in section 5.4), and the TetriNet game client on the PDA with the Windows PocketPC 2002 operating system.

2. A graphical user interface (GUI) belonging to the Gaming User Agent manages all games installed in the PDA. XML files are used for the configuration. With this GUI the user (player) can register himself at the SIP Registrar and join/leave game sessions. Our game client manager on the PDA automatically starts the respective game client after the user successfully joined a game session. The Focus of the Gaming Service on a PC running SUSE Linux 8.2. The Focus is part of the Gaming Server (as described in section 5.4).

   This component implements the functionalities of the Focus in the games architecture described above. The focus manages the game sessions via SIP signaling with the game clients and with the game servers via the game platform API, enforcing the policies of the game provider.

3. A Gaming Server Agent and the TetriNet game server on a PC running Windows XP.

The Gaming Server Agent (as described in section 5.4) supports the game server to register at the game focus. The Gaming Server Agent may run as standalone application and serve all game servers on the host It is able to trigger the initialization of game servers.

4. Additionally a SIP proxy and registrar were needed.

   For this server we used the NIST SIP proxy/registrar on a PC running SUSE Linux version 8.2.

In our testbed, the TetriNet game server registers itself with the gaming service (Gaming Server) using the Gaming Server Agent. The Gaming Server allocates the required resources, creates a SIP URI for the game session and registers it at the SIP Registrar. Several TetriNet game servers may publish game sessions at the gaming service at the same time.

The Gaming User Agent GUI on the IPAQ lists the published game session URIs, and then the user selects and joins a game session. (Figure 8 shows a screenshot of the GUI.) Afterwards, the TetriNet game client (Figure 9) starts automatically. It should be noted that the Focus is only involved with the game session management. Our current prototype only implements the OMA session management (as described in section 5.1). Therefore, the data path for the game in the current prototype is still set up directly between the game client and the game server. This may be changed in the future.

Figure 8: Screenshot for the GUI running on the PDA

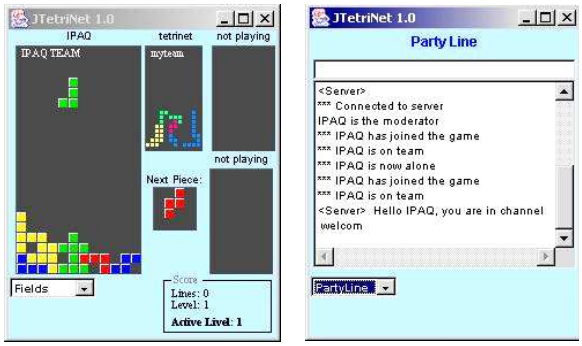

Figure 9: Screenshots from the TetriNet Game Client

9. Conclusions

The prototype implementation of our architecture for Games over the IMS shows that games can be integrated relatively easily onto the 3GPP and IMS platform. The minimal changes that existing games will have to undergo are to enable them for SIP signaling to set up the game session before the game data path is established.

Overall, the IMS is capable to support mobile, networked games. Of course, mobile devices may have restricted capabilities for running game applications. Such limitations cannot be solved completely by the IMS but it can help to overcome critical issues.

The current prototype was evaluated on a SIP based network. The next step is to integrate the prototype with a real IMS system and to extend the prototype for using all available IMS functionalities (e.g. Presence Service, Conferencing). Security and charging issues are other topics that need more investigation.

10. REFERENCES

[1] 3rd Generation Partnership Project; www.3gpp.org.

[2] 3GPP TR 21.902; *3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Evolution of 3GPP System.*

[3] 3GPP TS 22.228; *3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Service requirements for the Internet Protocol (IP) multimedia core network subsystem; Stage 1.*

[4] http://www.nlgaming.com/nl/asp/id_1453/nl/newsDisp.htm.

[5] http://www.gamespot.com/all/news/news_6072659.html.

[6] http://www.heise.de/newsticker/data/daa-11.07.01-003

[7] Yu-Chen; *Internet Game Design*; www.gamasutra.com.

[8] Joseph D. Pellegrino, Constantinos Dovrolis; *Bandwidth requirements and state consistency in three multiplayer game architectures*; NetGames 2003.

[9] Jouni Smed, Timo Kaukoranta, Harri Hakonne; *Aspects of Networking in Multiplayer Computer Games;* http://staff.cs.utu.fi/~jounsmed/papers/AspectsOfMCGs.pdf

[10] *Generic Content Download Over The Air Specification*, OMA-Download-OTA-v1_0-20030221-C; www.openmobilealliance.org/tech/release.html.

[11] *Mobile Evolution Shaping the Future*; A White Paper from the UMTS Forum; August 2003; www.umts-forum.org.

[12] *SIP: Session Initiation Protocol*; IETF RFC 3261.

[13] 3GPP TS 23.002; *3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects; Network architecture (Release 6)*

[14] *OMA Games Services version 1.0, Candidate Enabler Releases*; www.openmobilealliance.org/tech/release.html.

[15] 3GPP TS 23.141; *3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Presence Service; Architecture and functional description (Release 6).*

[16] *SIP for Instant Messaging and Presence Leveraging Extensions (SIMPLE)*; www.ietf.org/html.charters/simple-charter.html.

[17] *Megaco Protocol Version 1.0;* IETF RFC 3015

[18] *Gateway Control Protocol Version 1;* IETF RFC 3525.