PROJET MEGA

Mobile Gaming Services
Services pour les jeux multijoueurs sur mobiles

Rapport Bibliographique

Romain PELLERIN
Stagiaire CNAM-INT
Responsables : Eric GRESSIER et Michel SIMATIC

Résumé:

Depuis 2001, de nombreuses sociétés ont fourni aux opérateurs de téléphonie mobile des plateformes permettant de proposer le jeu multijoueur à leur client.
Ces plateformes, au-delà du fait d'accueillir le jeu et de le rendre accessible aux utilisateurs, se doivent de proposer tout une batterie de services non seulement à l'utilisateur dans un souci d'intéractivité et de confort d'utilisation, mais aussi des services d'administration du jeu tels que la configuration, l'organisation de compétition, et également des services d'administration système permettant notamment le contrôle d'accès ou encore le monitoring.
Ce travail constitue un état de l'art des plateformes disponibles à ce jour, des services qu'elles offrent, seront dégagés les problèmes que posent ces types d'archictectures en terme de recherche et les approches architecturales émergeantes.

TABLE DES MATIERES

1. Introduction 3
2. Plateforme pour le jeu multijoueur sur mobile 3
    2.1 Survol technologique 3
        2.1.1 Infrastructure du réseau mobile 3
        2.1.2 Technologies embarquées côté client 4
    2.2 Principe architecturaux 5
3. Présentation des services pour les jeux multijoueurs sur mobiles 6
    3.1 Services destinés à l'utilisateur 6
    3.2 Services en cours de jeu 7
    3.3 Services pour l'administration du jeu 7
    3.4 Services systèmes 7
4. Problèmes posés en terme de recherche 9
5. Plateformes de jeu et services existantes 11
    5.1 Terraplay MOVE 11
    5.2 MServer 13
    5.3 GeX 14
    5.4 Jive 16
    5.5 Deus+ 17
    5.6 KayaK Interactive 18
    5.7 Pecan Gaming 19
    5.8 Tableau comparatif 20
    5.9 Analyse 21
6. Tentative de standardisation: OMA-MGIF 22
    6.1 Présentation de l'OMA et du MGIF 22
    6.2 Architecture d'une plateforme de jeu MGIF compliant 23
    6.3 Spécification de la plateforme de jeu version 1.0 23
    6.4 Apports des futures plateformes version 2.0 et 3.0 24
7. Architectures émergeantes 25
    7.1 Approche agents mobiles pour les gaming services 25
    7.2 Plateforme sur GRID 27
8. Conclusion 28
9. Bibliographie 29

1. Introduction

Ces dernières années, le marché du mobile a littéralement explosé et constitue dorénavant un enjeu économique clé. Le mobile a pris une place prépondérante dans la société contemporaine, au-delà de ses fonctions de communication il fournit aujourd'hui des fonctionnalités de divertissement.
Le jeu sur mobile qui jusqu'à très récemment restait monojoueur, suit la tendance des jeux sur PC et sur consoles et tend à devenir multijoueur pour la notion d'ouverture et d'expérience de jeu qu'il peut offrir par opposition à l'isolement et aux limites de l'intelligence artificielle que pouvait faire ressentir le jeu monojoueur.
L'évolution technologique des réseaux pour mobiles avec dans un futur proche la popularisation des mobiles de 3ème génération, va permettre la création de jeux d'avant-garde necessitant la mise en place de services de pointe ayant pour objectif de transcender l'expérience du joueur sur mobile.
C'est dans ce contexte d'enjeu économique et de challenge technologique, que la recherche et le developpement de plateformes et de services de pointe s'imposent.
Dans ce rapport nous ferons tout d'abord un bref survol technologique des réseaux mobiles et de l'architecture pour les jeux multijoueur sur mobile. En deuxième lieu nous caractériserons les services requis par une plateforme de jeu multijoueur sur mobile. Suite à cela nous déterminerons quels sont les problèmes en termes de recherche que pose la mise en place de tels services. Nous aborderons ensuite les différentes offres propriétaires disponibles sur le marché actuellement en examinant les choix architecturaux effectués et les services proposés. Par opposition nous verrons la tentative de standardisation des services pour les jeux multijoueurs de l'Open Mobile Alliance. Nous terminerons en étudiant les axes de recherche émergeants.

2. Plateforme pour le jeu multijoueur sur mobiles

Les plateformes accueillant les jeux multijoueurs pour mobiles sont génériques c'est-à-dire qu'elles peuvent reposer sur tout type de couches protocolaires et sur tout type de réseaux, il convient tout d'abord de faire un bref état des lieux des technologies utilisées actuellement tant au niveau du réseau pour téléphones portables qu'au niveau des mobiles eux-mêmes.
En comparant différentes propositions, une architecture globale émerge et c'est ce que nous étudierons en second lieu.

2.1 Survol technologique

2.1.1 Infrastructure du réseau mobile

Actuellement, nous sommes dans une phase de transition technologique entre les technologies mobiles de 2ème génération (2G) et de 3ème génération (3G) [1].
La norme de communication mobile la plus répandue étant le **GSM** (Global System for Mobile communication) et son évolution orientée paquets le **GPRS** (General Packet Radio Services) considérée comme une technologie intermédiaire 2G/3G. En plus de permettre la communication par paquets, le GPRS permet d'occulter le processus d'établissement de liaison ou « dial-up » et de reprendre instantanément la communication en cas de rupture de liaison, constituant ainsi une surcouche GSM idéale pour les developpeurs de jeux, le débit théorique disponible étant de 13,4 kbps.
Une seconde évolution intermédiaire vers les technologies 3G a été élaboré, **EDGE** (Enhanced Data Rates for GSM Evolution) qui comme son nom l'indique fournit un débit plus élevé de l'ordre de 300 kbps théorique et dans la pratique l'équivalent d'une connection fixe ISDN, entre 64 et 128 kbps. EDGE permet donc le developpement et le déploiement de jeux plus gourmants en bande passante, tel que les jeux en temps réel par exemple.
Les capacités en bande passante du réseau GSM ont atteint leur limite avec EDGE, ainsi afin d'accroitre la bande passante entre le relai et le mobile il a fallu repenser l'infrastructure réseau en introduisant une toute nouvelle interface radio, le **WCDMA** (Wideband Code Division Multiple Access) plus communément appelée **UMTS** (Universal Mobile Telecommunication System).
L'UMTS permet donc non seulement d'offrir une bande passante accrue de 144 kbps à 384 kbps et même à faible distance du relai près de 2Mbps, mais est aussi bien mieux adapté à la communication IP et aux transferts de données multimédia.
Les zones rurales disposent généralement de couverture GPRS ou EDGE, tandis que les zones urbaines accueuilleront l'UMTS.
Certains jeux selon la bande passante qu'ils requièrent ne fonctionneront donc que dans certaines zones de couverture.

Bluetooth

De nombreux mobiles ont d'ores et déjà la possibilité d'accueillir des jeux multijoueurs sur réseau local bluetooth. Il convient donc d'introduire cette technologie dans l'infrastructure du réseau mobile car il est envisagé de pouvoir combiner les réseaux 2G/3G et les réseaux bluetooth. Cette nouvelle architecture réseau va amener à l'émergence de nouveaux types de jeu, par exemple des jeux sous formes de groupes interconnectés fonctionnant en temps réel localement combiné et au tour par tour via GPRS.
Les capacités techniques de la technologie bluetooth sont une distance maximale de 10m entre le mobile serveur et les mobiles clients, une bande passante de 1Mb/s et une connexion maximale de 7 clients sur un serveur.

2.1.2 Technologies embarquées côté client

Outre les fonctionnalités de communication vocale, les mobiles disposent de plusieurs technologies embarquées pour la communication par messages, la navigation web et des environnements permettant l'exécution de programmes tels que les jeux notamment.

La communication par messages de type texte est prise en charge par la technologie **SMS** (Short Message Service), tandis que l'**EMS** (Enhanced Message service) est apparu afin de permettre le transfert de logos, images ou sonneries. Plus récemment le **MMS** (Multimedia Message Service) a été developpé afin de permettre la combinaison de textes, sons, images et vidéos au sein de messages.

Le **WAP** a été élaboré afin de rendre accessible aux mobiles la navigation Web et le support des languages HTML, WML, XHTML, HDML, c-HTML et MML. Dans sa version 2.0, le WAP prendra en charge le WML2, l'UAProf qui permettra d'informer sur les capacités du mobile (résolution d'écran, technologies embarquées…) et sur les préférences utilisateur. Il fournira une interface pour le stockage persistant et la technologie MMS afin de pouvoir mixer navigation web et messages multimédia.

Ce qui nous interesse plus particulièrement pour le deploiement des jeux multijoueurs sur mobile est l'environnement d'execution du jeu, plusieurs environnements existent. Ils se divisent en deux catégories, la première repose sur la technologie Java de Sun Microsystems et la seconde sur un mélange de C++ et de Java.

- Environnements Java :

  J2ME (Java 2 Micro Edition) [2] est la plateforme de Sun Microsystems conçue pour fonctionner sur les terminaux embarqués tels que les mobiles mais aussi les PDA.
  Elle est constituéed'une machine virtuelle Java, KVM (KiloByte Virtual Machine) pouvant fonctionner dans un environnement limité en termes de mémoire, d'énergie et d'accès réseau, et d'une suite d'API orientée téléphonie mobile, telle que MIDP ou Doja combinée à l'API CLDC.

  MIDP (Mobile Information Device Profile) [3] est une API Java de haut niveau d'interaction avec le mobile permettant la gestion de l'interface utilisateur, de l'interface réseau, la gestion d'une base de données et dispose d'un module graphique de bas niveau. Une application basée sur MIDP est appelée « midlet ».
  Dans sa version 1.0, elle possède certaines limitations comme le support d'un seul format d'image (png), gestion réseau se limitant au protocole HTTP, pas d'accès à la carte SIM.
  La version 2.0, d'ores et déjà disponible, intègre de nouvelles fonctionnalités telles que la gestion du son dans les midlets, l'ajout d'une API spécifiquement dédiée au developpement de jeu sur mobile, une nouvelle gestion de l'interface utilisateur (pop-up, message d'alerte…), une connectivité évoluée (HTTP, HTTPS, socket, datagramme, port série), le deploiement dynamique et la mise à jour automatique d'applications (OTA), enfin la sécurisation point à point via HTTPS et SSL.

  CLDC (Connected Limited Device Configuration) [4] est constituée de l'API necessaire au langage Java ainsi que d'une API spécialisée dans l'accès au réseau mobile.

  DOJA (Docomo Java) [5] version MIDP propriétaire de la société NTT-Docomo [6] optimisée pour les terminaux i-Mode, par rapport à l'API Sun MIDP Doja introduit des améliorations au niveau de la gestion graphique.
  EXEN de la société In-Fusio [7] constitue un moteur spécifiquement dédié au jeu sur mobile, il fonctionne sans J2ME, fournit de nombreux outils avancés pour le développement de jeux

comme le calcul de matrice 3D, le rendering 3D, il permet également la prévention d'erreurs lors de l'exécution d'un jeu.

- Environnements C++/Java :

EPOC plus connu sous le nom de **Symbian OS** [8]**,** véritable OS descendant du Palm OS, intègre dans sa version 0.8 d'innombrables fonctionnalités multimédia, la gestion de nombreux protocol tels que TCP/IP, il est d'ores et déjà prêt pour le passage à l'UMTS, les mécanismes OTA, la sécurité via HTTPS et SSL, la gestion hardware du mobile via SyncML, des kits de developpements C++, J2ME/MIDP2.0. Destiné au jeu il dispose du moteur 3D OpenGL, fournit un framework pour les jeux et autres applications multimédia.
Symbian OS est intégré dans plus de 70% des mobiles sur le marché.

MOPHUN [9] plateforme dédiée au jeu multijoueur sur mobile, peut être soit préinstallée, soit téléchargée, elle constitue une API complète pour le developpement et l'exécution de jeu 2D ou 3D. Mophun combine 2 modules pour l'exécution, le moteur RTE (runtime engine) et l'API.
Pour le developpement du jeu un SDK est disponible. Le distributeur peut dédié son jeu à une plateforme ciblée grâce à l'outil VST (Vendor Signing Tool), le jeu ne pouvant s'executer sur les autres mobiles.

BREW de Qualcomm [10] est un système fournissant les fonctionnalités de messagerie instantanée, email, de navigation facilité, échange de données multimédia, et des fonctionnalités pour les jeux multijoueurs

Windows Mobile [11] successeur de Windows CE, dérivé du système d'exploitation bien connu, offre des fonctionnalités de messagerie instantanée et de mailing et des fonctionnalités pour les jeux.

2.2 Principes architecturaux

Une plateforme de jeu fait intervenir différents acteurs : l'opérateur de téléphonie tout d'abord est celui qui va décider de l'orientation stratégique vers le jeu multijoueur, ainsi il va déployer la plateforme ou confier ce déploiement à un hébergeur. Les éditeurs de jeux peuvent publier et gérer leurs jeux sur la plateforme.
L'utilisateur, abonné au réseau de l'opérateur et désirant jouer à un jeu donné doit s'il n'est pas au préalable déjà installé sur le mobile le télécharger, il faut donc que le mobile utilisé soit doté de la capacité de téléchargement sachant que c'est le cas dans la plupart des modèles récents.
Le téléchargement du client du jeu, va permettre d'alléger les communications entre le serveur et le client lors du déroulement de la partie. Le jeu désiré va être téléchargé via un serveur dédié, le serveur de téléchargement à partir d'une requête WAP. Le serveur de téléchargement va alors contacter la base de données des utilisateurs afin de déterminer si l'utilisateur dispose des droits requis pour ensuite facturer le telechargement.
Une fois l'installation effectuée, le joueur va tout d'abord s'authentifier, afin d'accéder à son compte le joueur, regroupant ses options de jeu, ses habitudes de jeu, ses serveurs favoris, ses contacts... cette personnalisation du jeu fait partie des services que peut proposer une plateforme de jeu. Ces informations sont hébergées par un serveur spécifique, le « lobby server ».
C'est ce serveur qui maintient également la liste des parties joignables, il détient l'ensemble des informations concernant le jeu comme les scores, les compétitions, etc …
Après avoir sélectionné la partie à rejoindre, le client va être dirigé vers le serveur hébergeant celle-ci, appelé communément « game server ».
Afin de fournir les services d'administration système, d'administration du jeu, les services de facturation ou encore d'authentification, des serveurs spécifiques et des bases de données se situent derrière le « lobby server » et le « game server », il n'y a pas d'architecture prédéterminée et chaque offre propose sa solution. Ces offres sont passées en revue dans la section 5.

3. Présentation exhaustive des services pour les jeux multijoueurs sur mobiles

Autour du jeu lui-même peuvent être déployés une série de services ayant pour objectif l'offre côté client de fonctionnalités améliorant interactivité, personnalisation et gestion avancée, mais aussi des services d'administration du jeu et d'administration système côté serveur.

3.1 Services destinés à l'utilisateur

La première catégorie de services utilisateur permet de créer une interface interactive entre l'utilisateur et le réseau « jeux » d'un opérateur :

Authentification et Ouverture de session : permet d'authentifier un utilisateur lors de sa connexion au réseau. Ainsi l'utilisateur se verra proposer une interface personnalisée correspondant à ses habitudes d'utilisation.

Téléchargement : fournit une interface de téléchargement de jeux ou encore de plug-ins tels que des cartes, avatars, sons supplémentaires mis à disposition. Ce service offre l'évolutivité du pool d'applications primordiale sur le mobile de l'utilisateur, bien sûr il va permettre de générer des revenus colossaux, l'opérateur facturant généralement le téléchargement.

Personnalisation : l'utilisateur peut personnaliser son interface en fonction de ses préférences, par exemple pour accéder directement à un certain type de jeu.

Informations : l'opérateur va pouvoir avertir l'utilisateur, par exemple s'il a téléchargé un jeu et qu'une mise à jour est dorénavant nécessaire, ou qu'un nouveau jeu correspondant à un type de jeux déjà téléchargé est disponible.

La seconde catégorie de services utilisateur est directement liée à un jeu en particulier. Ces services constituent l'interface du jeu:

Rejoindre une partie ou service de « lobby » : l'utilisateur peut accéder à une liste des parties en cours et sélectionner la partie à laquelle il désire participer en fonction de la carte jouée, du nombre de joueurs participant …

Créer une partie : il peut aussi initialiser une nouvelle partie, en choisissant les options de jeu comme la carte jouée par exemple.

Gestion de compétitions : des compétitions sont souvent organisées autour d'un jeu, ce service permet à un utilisateur de rejoindre une compétition, créer une équipe, consulter les classements, défier des adversaires, consulter les informations diverses disponibles sur la compétitions telles que les règlements, les nouvelles …

Gestion d'équipe : permet à l'utilisateur de gérer l'équipe s'il possède les droits de manager, comme la gestion et l'attribution de droits à ses coéquipiers, le recrutement, les statistiques de l'équipe…

Gestion de match : permet de mettre en place un match face à une autre équipe, par exemple le choix des cartes jouées, lorsque la partie s'est déroulée d'entrer les scores du match et commentaires.

Consultation des statistiques : l'utilisateur peut consulter toutes sortes de statistiques concernant le jeu, telles que le classement des meilleurs joueurs, ses statistiques personnelles.

Recherche d'adversaires/Recherche d'amis : service offrant la possibilité de rechercher des joueurs à partir de leur pseudo ou identifiant de compte.

3.2 Services en cours de jeu

Ces services sont disponibles entre les joueurs d'une même partie, que celle-ci soit déjà lancée, en cours d'exécution, ou non , en salle d'attente plus communément appelée « waiting room ».

Gestion de session : lorsqu'un utilisateur se connecte à une partie, une session représentant l'intéraction, le lien entre cette partie et l'utilisateur est initialisée, une session va donc être crée pour chaque participant. L'ensemble de ces sessions va permettre la synchronisation evênementielle entre les utilisateurs tout au long du déroulement de la partie.

Consultation des scores en cours de partie : au cours d'une partie, les scores sont maintenus en temps réels pour cela chaque participant met à jour son score en fonction du déroulement du jeu, des évênements du jeu.

Communication inter-joueurs : service établissant un mode de communication entre les joueurs d'une même partie, ces communications pouvent être vocales ou textuelles.

3.3 Services pour l'administration du jeu

Ensemble de services destinés à la gestion, à l'administration du jeu côté serveur.

Gestion des options - Configuration du jeu : l'administrateur peut ainsi gérer les paramètres du jeu, tels que les limites de temps d'une partie, le nombre maximum de joueurs, les cartes ou niveaux disponibles …

Gestion de contenu : permet à l'éditeur du jeu de gérer le contenu associé à ses jeux, d'effectuer des mises à jour …

Gestion/Organisation de compétitions : service permettant à l'administrateur de créer des tournois, en donner les règles et le modèle de fonctionnement. Ensuite il peut gérer les tournois, valideres matchs, regler les litiges, …

Gestion des joueurs : l'administateur peut gérer l'ensemble des données concernant un joueur, infliger des sanctions, …

Prix et récompenses : service de gestion des prix et récompenses suite liées aux tournois, récompenses pour les meilleures statistiques, …

Annonces - « Advertising » : service destiné à informer les clients sur les nouveaux jeux disponibles ou à leur communiquer des informations telles que l'annonce d'organisation d'un tournoi, de fin imminente d'inscription. Cette distribution d'informations peut être personnalisée, ciblée vers une catégorie de joueur dont les habitudes de jeux correspondent.

3.4 Services système

Contrôle d'accès : service permettant de contrôler les droits d'accès au réseau du jeu choisi, en consultant la base de données des utilisateurs et en vérifiant notamment si le jeu a été facturé afin d'interdire l'accès aux utilisateurs détenteurs de versions piratées du jeu.

Capacités du terminal : service permettant de récupérer les informations sur les capacités matérielles ou logicielles du terminal mobile de l'utilisateur, afin de pouvoir formater de manière adaptée les futures données à envoyer. Par exemple la résolution d'écran pour l'envoi d'images, les protocoles de communication supportés, les API disponibles …

Monitoring : service destiné à fournir des informations sur le taux d'utilisation courant des serveurs de jeu en termes de charge mémoire, charge processeur ou charge disque, et sur le taux de charge réseau en terme de bande passante utilisée.

Allocation de ressources : service destiné à allouer les tâches à exécuter aux différents serveurs présents sur le réseau, en fonction du taux d'utilisation de chaque serveur et afin de bien répartir la charge entre les serveurs pour éviter les surcharges. Par exemple pour l'initialisation d'une nouvelle partie, le service sélectionne un serveur approprié.

Facturation : ce service se charge de facturer tout service payant et d'effectuer la mise à jour de la base de données des utilisateurs, les services facturés peuvent être le téléchargement ou encore le temps de connexion au réseau, les abonnements aux services premium …

Journalisation – « Logging » : service dont la fonction est de faire l'historique de l'ensemble des opérations effectuées par les clients, l'ensemble de l'activité du réseau, afin de faire du Data Mining, ou de tracer d'éventuelles tentatives de piratage.

Etudes statistiques d'utilisation - « Data-Mining » : service qui à partir de l'historique généré par le service de journalisation, va établir les habitudes des utilisateurs afin de personnaliser ensuite les publicités ou informations qui leur seront distribués. Ce service évalue de manière statistique l'évolution de la demande afin de déterminer les évolutions matérielles ou logicielles à prévoir au sein du réseau.

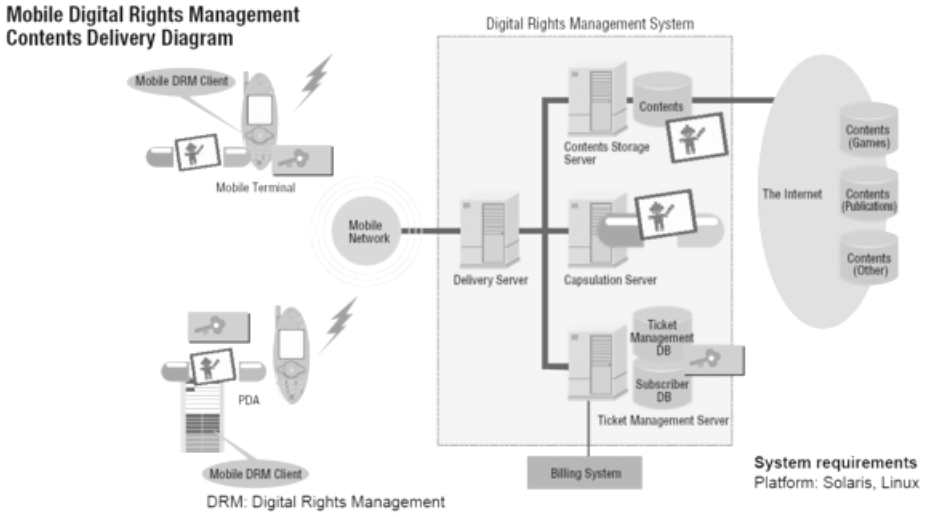
Persistance des données : service permettant à un utilisateur se deconnectant de reprendre le jeu là où il l'avait laissé au moment de sa déconnection, la persistance des données est un enjeu crucial pour les jeux sur teminaux portables soumis à des sessions fragmentés dûe à la mobilité de l'utilisateur. Les données survivent ainsi à l'execution des programmes.

Gestion des utilisateurs : service de gestion de la base de données des utilisateurs, gestion des droits d'accès des utilisateurs, gestion des informations personnelles, …

Gestion des droits digitaux – « Digital Rights Management » : service permettant d'acheminer du serveur de contenu au mobile, de manière sécurisée, les données soumises à des droits digitaux et cela afin d'éviter la prolifération de copies non autorisées notamment. Les données acheminées sont cryptées et nécessitent une clé pour être accessibles. Le serveur de contenu héberge donc les données à transférer qui vont être transférées à un serveur d'encapsulation/cryptage, les données cryptées étant ensuite fournies au serveur de téléchargement. Il est à noter que ce service sécurise également le service de paiement en lui transférant un ticket crypté contenant les informations du téléchargement, utilisateur et données téléchargées.

Géo-localisation : service permettant de localiser la position géographique de l'utilisateur afin par exemple de personnaliser les informations lui étant transmises ou encore pour pouvoir répondre aux besoins des jeux géo-localisés.

Localisation/Présence: service permettant à un utilisateur de se localiser dans un environnement virtuel tel que des salles de rencontres ou des lobbies, donnant donc à l'utilisateur la sensation de pouvoir se diriger. Le concept de présence se trouve dans le fait de rejoindre d'autres utilisateurs présents dans des salles, il y a donc présence virtuelle d'autres utilisateurs.



Ces services posent de multiples problèmes, difficultés, couramment rencontrés dans les environnements ou systèmes distribués, ils sont présentés et analysés en section 3

4. Problèmes posés en terme de recherche

Les services présentés précédemment constituent de réels challenges technologiques constituant d'ailleurs les plus grands thèmes de la recherche en systèmes distribués. Une plateforme de jeux multijoueur pour mobile est un système distribué à grande échelle possédant de fortes contraintes en terme de connectivité réseau, de synchronisation, de charge réseau ou machine …
Il convient de présenter quels sont les enjeux technologiques majeurs d'une telle plateforme, qui nous permettront ensuite d'évaluer les solutions existantes à ce jour.

Interopérabilité / Interfaçage de services / Assemblage

Avec la prolifération de services en tout genre qu'ils soient des services de la plateforme opérateur, de la plateforme de jeu ou encore de tout autre service externe, il est indispensable de les rendre interopérables, qu'ils puissent être interfacés avec un minimum d'effort voir sans.
L'interfaçage des services de metering de la plateforme de jeu et des services de facturation de la plateforme opérateur est un exemple caractéristique.
L'interopérabilité entre les différentes plateformes doit aller au-delà de l'interfaçage de service en autorisant le partage de ressources et l'échange d'informations.

Cohérence et synchronisation en environnement distribué multi-utilisateurs

Les plateformes de jeu tendent à offrir de plus en plus de services d'intéraction entre joueurs au-delà du jeu lui-même, ce type de services nécessite cohérence et synchronisation afin de satisfaire les utilisateurs.
Ces mécanismes doivent être robustes et efficaces, le volume d'intéractions étant proportionnel au nombre d'utilisateurs, généralement très élevé sur une plateforme de jeu multijoueur.

Détection d'erreurs/ Résistance aux fautes

Les machines supportant la plateforme de jeu, c'est-à-dire l'ensemble des routines systèmes, d'exécution des sessions de jeu, de mise à disposition des services ou encore de gestion du traffic des données, peuvent être amenées à tomber en panne. Cette plateforme se doit de fournir des fonctionnalités de qualité de services et de disponibilité à la hauteur du revenu qu'elle génère, un utilisateur qui paye à prix d'or son accès aux services de jeux multijoueurs sur mobile, ne peut tolérer des pannes systèmes en cours de partie et des indisponibilités de service fréquentes. C'est pourquoi il est primordial pour une plateforme d'être capable d'anticiper ces crashs et cela passe par des fonctionnalités de détection d'erreurs.
Toutes les erreurs ne peuvent pas être anticipées ainsi des mécanismes de résistance aux fautes se doivent d'être mis en place, un système amené à crasher va être capable de réamorcer son exécution à partir de points de reprise antérieurs au crash.
Ces mécanismes font l'objet de recherches actives au sein de la communauté scientifique et notamment dans le domaine des systèmes distribués.

Allocation de ressources/ Equilibrage de charge

Une plateforme de jeu est constituée d'un parc de serveurs qui peut être conséquent, ce parc informatique se doit d'être géré correctement et intelligemment. Il serait totalement abérent que certaines machines fonctionnent à pleine charge de manière permanente tandis que parallèlement d'autres machines soient perpétuellement désoeuvrées.
Ainsi une plateforme doit posséder des mécanismes d'allocation de ressources et d'équilibrage de charge avancés, pour que l'ensemble des traitements soit idéalement réparti sur le réseau de machines interne.

Persistance des données

La persistance des données est un enjeu crucial pour les plateformes de jeu, la mobilité de l'utilisateur crée des problèmes de déconnexions fréquentes, il est alors impératif de sauvegarder l'état courant de la session de jeu au moment de la déconnexion. Ce mécanisme de stockage d'état à l'échelle du système distribué à grande échelle que constitue une plateforme de jeu pour mobile en pleine charge avec ses milliers de connexions suceptibles d'être sujettes à des déconnexions devient un véritable défi technologique et de recherche de solutions satisfaisantes, solutions se devant d'être efficace tout en évitant d'être lourdes en termes de charge processeur ou encore de charge disque

Sécurité

Ce type de plateforme se doit d'être parfaitement sécurisée afin de protéger les droits digitaux, de limiter l'accès aux utilisateurs enregistrés ou encore afin d'éviter les attaques malveillantes.
C'est un axe de recherche privilégié et en constante évolution.

Routage des données

A l'instar des fonctionnalités d'allocation de ressources et d'équilibrage de charge, le traffic des données sur le réseau se doit d'être cohérent afin d'éviter les phénomènes de surcharge réseau plus connu sous le nom de congestion réseau.
Une plateforme doit donc posséder des mécanismes pointus de gestion du traffic réseau, en implémentant des algorithmes de routage efficaces.

Multicasting avancé et Streaming

Les services de communication entre joueurs, notamment vocale, repose sur les principes de multicasting voir de streaming, le serveur recevant l'émission de données vocales d'un joueur va devoir retransmettre ces données aux autres joueurs de la même session de jeu ou de la même salle de rencontre. Pour alléger le temps de retransmission et la charge nécessaires à cette retransmission, le serveur va utilisé le muticast ip voir ses extensions dans les travaux de recherche effectués dans le domaine du streaming audio.
Le multicasting va être également utilisé lors de la session de jeu pour transmettre les informations communes à chaque joueur…

Personnalisation : logging & data mining

Le concept de personnalisation de l'interface et des services proposés à l'utilisateur est un enjeu stratégique pour les opérateurs. Ainsi l'utilisateur se verra proposé des services correspondant à ses habitudes d'utilisation, pour cela la plateforme de jeu dispose de deux services clés : le logging et le data mining.
Le logging génère l'historique d'utilisation de la plateforme pour chaque joueurs, à partir de ses données le service de data mining va déterminer quels sont les besoins, les gôuts et habitudes de chaque joueur.
Ces informations concernant l'utilisateur vont permettre de personnaliser les publicités, les messages d'information, les téléchargements ou autres services proposés.

Mise à jour automatique et publication

Les fonctionnalités de mise à jour automatique ou de publication de contenu, qui peuvent dans certains cas se faire en tâche de fond, nécessitent des mécanismes complexes de distribution de données, de recherche des utilisateurs ciblés par la mise à jour, de gestion de points de reprise pour les mise à jour interrompues, d'avertissement de l'utilisateur via messagerie…

Presence/Localisation/Interactions

L'évolution actuelle des plateformes vers la mise en place de communautés virtuelles de joueurs, a posé le problème de la localisation personnelle de l'utilisateur dans ses environnements virtuels toujours plus vastes.
Ainsi le challenge actuel est de créer des environnements virtuels intuitifs telle que la mise en place de salle de rencontre, spécifique à un thème ou à un jeu, pour les joueurs. Ces espaces virtuels se doivent d'être bien pensés aussi bien visuellement que fonctionnellement avec l'intégration du chat textuel ou vocal.
Les concepts de presence, d'intéractions inter-utilisateurs et de localisation ont pour objectif de mettre à l'aise le joueur en décuplant ainsi son expérience virtuelle.

Découverte

Une plateforme de jeu propose des services de découverte notamment pour la recherche de parties joignables, de salles de rencontres, d'amis ou d'adversaires.
Ces fonctionnalités font appel à des mécanismes sous-jacents complexes car requiérant une vision globale de l'activité du serveur, ce qui constitue dans le domaine des systèmes distribués un véritable challenge technique.

Temps réel

L'objectif de recherche et developpement d'une grande majorité de services est de se rapprocher d'un fonctionnement en temps réel, et donc de mécanismes de synchronisation avancés.

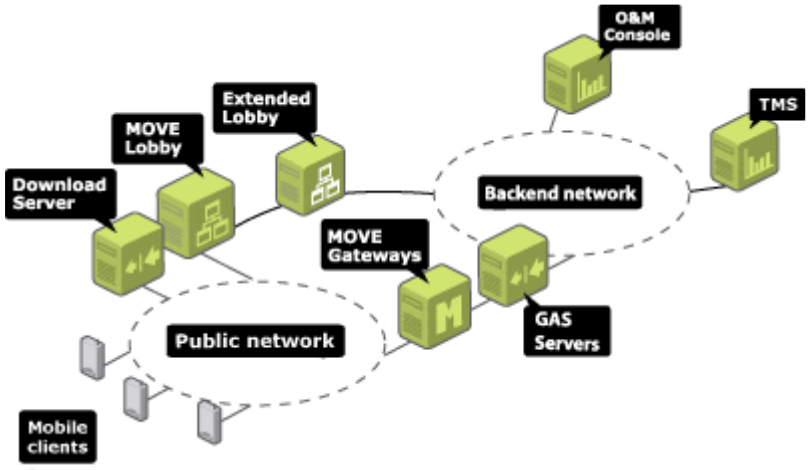
5. Plateformes de jeu et services

Il convient tout d'abord de préciser que cet état de l'art des plateformes de jeux multi-joueur pour mobile a été réalisé à partir des documents publiques ou white paper fournis par leurs éditeurs, ces plateformes peuvent effectivement proposer des services qui ne sont pas mentionnés dans ces publications.
En outre ces plateformes propriétaires n'étant pas accessibles pour des raisons commerciales, il est difficile de dégager clairement les choix architecturaux réalisés et de définir l'intégralité des services disponibles.

5.1 Terraplay Move

Terraplay Move de Terraplay Systems [12] constitue une solution complète incluant les fonctionnalités et processus nécessaires pour configurer, exécuter et déployer des services pour jeux multijoueurs sur mobile. Cette plateforme est générique, tout d'abord indépendante des terminaux et des réseaux via l'utilisation des protocoles TCP, UDP et HTTP. Un SDK est disponible pour le developpement de jeux pour cette plateforme.

Choix architecturaux



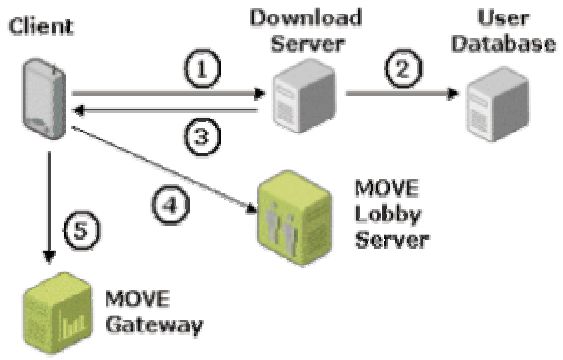
Cette plateforme est articulée autour de plusieurs nœuds offrant des services spécifiques : (cf [13])

- **Move Lobby**, serveur « lobby » fournissant les services de :
  - création/connexion à des parties privées, parties entre amis protégées par mot de passe
  - connexion à des parties publiques
  - gestion des scores
  - service d'informations personnalisées par messagerie via l'**Extended Lobby**
  - hébergement des tournois, ligues, compétitions
  - recherche d'adversaires
  - configuration des préférences de jeu
- **Move Gateways**, serveurs vers lesquels toutes les données en cours de partie vont être acheminées.
- **GAS Servers** (Game Access Server), serveurs se chargeant du routage intelligent des données vers les serveurs hébergeant les parties concernées
- **TMS** (Terraplay Management System), système de gestion des ressources du réseau interne ou « Backend network » se décomposant en 2 nœuds, **LAS** (Lobby Access Server) et **SM** (Site Manager) qui vont effectuer l'équilibrage de charge et l'allocation des ressources.
  TMS fournit également des statistiques (utilisateur/jeu/heure/durée), permet la gestion de session et la récupération des profils clients (caractéristiques du terminal).
- **O&M Console** (Operation&Maintenance Console), console permettant de configurer, lancer et stopper les nœuds du réseau interne de serveurs, de monitorer le système et d'alerter l'administrateur en cas de pannes.

Le protocole de communication utilisé entre le client et Terraplay Move est un protocole le **TWP** (Terraplay Wireless Protocol), il est utilisé à tous les niveaux de communication entre le client et le « Move lobby server », entre le client et les « Move Gateways » ainsi que dans l'ensemble du « Backend network ».

La plateforme est interfaçable avec le système de paiement de l'opérateur et prend en compte de nombreux modèle de facturation comme le paiement par session, par téléchargements. Cette interface possède un système de logging permettant de mettre en place tout type de modèle de paiement.

Stratégie de mise en place de la connexion client-plateforme



Phase de téléchargement du jeu (1 2 et 3) et phase de connexion à une partie (4 et 5), ses phases respectent les choix architecturaux traditionnels présentés en section 1.2, il est à noter que Terraplay Move se base sur l'authentification du client via la base de données utilisateurs.

Support systèmes

- Mophun
- Exen
- J2ME
- Symbian OS
- Windows Mobile
- Brew

Performance

- 4.000 joueurs simultanés sur un système
- 1.000 joueurs en une session
- 2.000 sessions sur un système

Récapitulatifs des services proposés par la plateforme Terraplay Move

- Authentification
- Téléchargement de jeux via un serveur de téléchargement
- Gestion/Organisation de matchs
- Gestion/Organisation de compétitions
- Communication en cours de jeu via TCP/IP
- Gestion de scores
- Paiement
- Allocation dynamique des ressources et equilibrage de charge
- Gestion de session
- Monitoring
- Logging/Statistiques
- Publicité et informations personnalisées
- Capacités du terminal
- Recherche d'adversaires
- Routage et multicast avancé

5.2 Mserver

Mserver [14,15] est une plateforme de services basée sur les technologies Java et XML pour applications sans fil développée par mForma, étant ainsi deployable sur tout type de terminal mobile. Mserver a été choisi par Intel/HP Solution Center pour son temps réponse et pour sa capacité à passer à l'échelle.

Un SDK est mis à disposition des developpeurs, dont la version courante est Mserver 3.0 et intègre les API offrant les fonctionnalités suivantes :

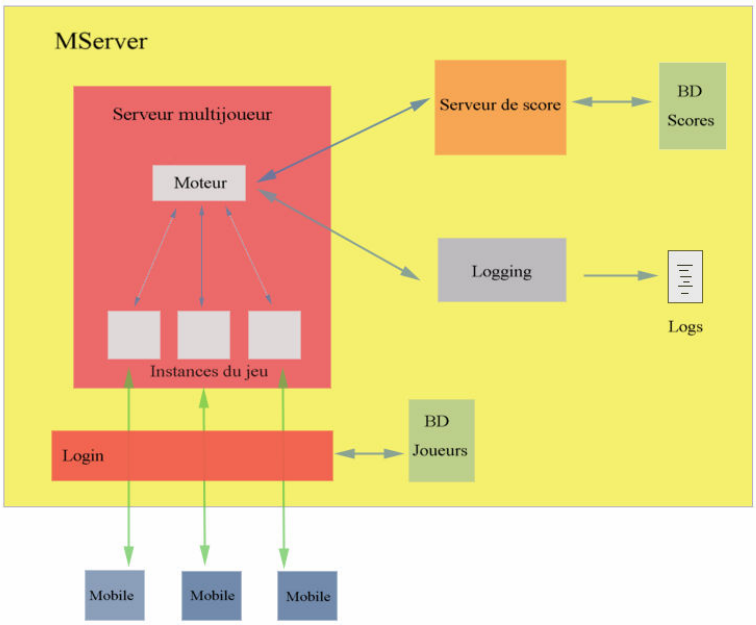
- Support de**MZML**, un langage d'affichage indépendant du terminal
- Processeur **XSLT**
- Logging
- Accès base de données
- Persistance des informations de session
- Gestion des scores et classement de l'utilisateur
- Imagerie dynamique 2D
- Géo-localisation
- Communication Peer-To-Peer
- Gestion SMS
- Gestion Multi-utilisateurs

Un moteur pour la gestion d'applications multijoueur est intégré au SDK, une console de test de jeu en environnement multijoueur simulé est également fournie ainsi qu'un IDE propriétaire permettant de créer son projet, et enfin **ant** pour tout ce qui est compilation, conversion XML, …
Toute application comme par exemple un jeu multijoueur crée à partir et pour Mserver porte le nom de **mlets**.

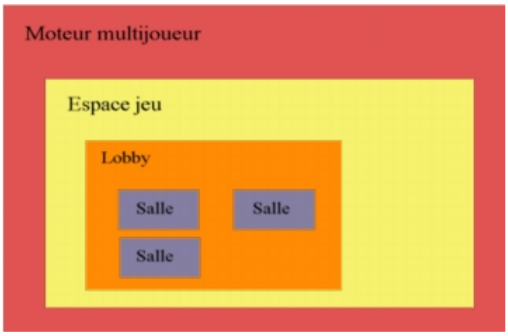
Services administratifs

- Authentification de l'utilisateur (login mlet)
- Base de données des utilisateurs (mForma) compatible JDBC (mysql, oracle,…)
- Accès à la base de données via transformation de schéma XML en schéma propre à la BD
- Logging via l'outil **Log4J** developpé par Jakarta project
- Interfaçage de SMS avec un mlet
- **Intellocate**, API de géo-localisation
- Gestion de scores, API **ServiceScoring,** un serveur de score est chargé de maintenir les scores dans la base de donneés des scores **dbScore**

Architecture de Mserver



Moteur multijoueur



La topologie du serveur multijoueur est représentée ci-dessus, chaque espace de jeu peut contenir « l » lobby, eux même peuvent contenir « s » salles, chaque salle accueuillant « j » joueurs.
L'application doit déterminer sa topologie puis la communiquer au moteur/serveur multijoueur, en donnant un identifiant à chaque lobby et à chaque salle.

L'utilisateur peut entrer dans un lobby puis dans une salle ou il va jouer avec les joueurs de la salle, ou bien sortir pour rejoindre une autre salle ou un autre lobby.

5.3 Gex

Gex [16] est une plateforme pour jeu multijoueur sur mobile orienté serveur basé sur la technologie Java et développée par GameFederation. L'application centrale de Gex est un navigateur de sessions de jeu permettant à l'utilisateur de visualiser les sessions, de les rejoindre ou de créer ses propres sessions.
Passons en revue les fonctionnalités de Gex :

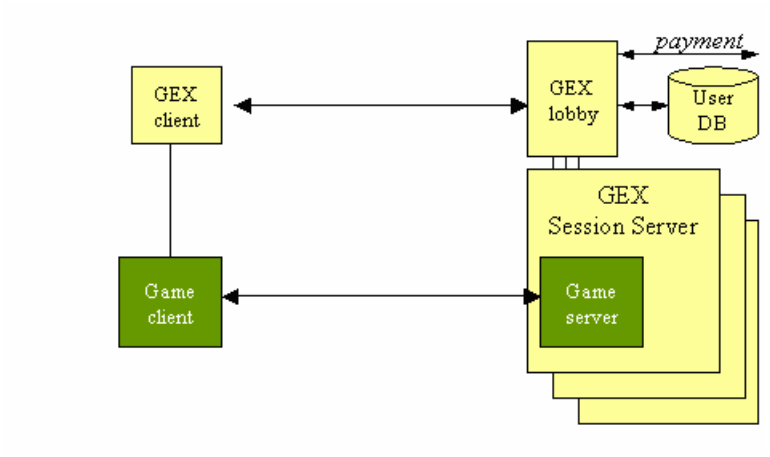
Services utilisateur

- Accès à un service de jeu
- Identification sécurisée
- Visualisation des sessions de jeu en cours
- Création d'une session de jeu
- Rejoindre une session en cours
- Visualisation des joueurs en ligne
- Participation à des compétitions
- Téléchargement ou mise à jour de jeux à la demande, c'est-à-dire lors de la sélection de la session
- Gestion des jeux enregistrer localement, gestion des préférences, desinstallation

Services opérateur et éditeur

- Interfaçage les services Gex avec le service de paiement
- Statistiques sur l'utilisation des services
- Configuration de l'outil embarqué pour la fixation de prix, définir en fonction du jeu et du terminal client quel type de souscription utiliser et en fixer le prix.
- Permettre la souscription en ligne
- Gestion des comptes utilisateurs
- Configuration des compétitions

Architecture de Gex



La plateforme côté serveur est composée de serveurs « lobby » chaque serveur lobby hébergeant les services d'un seul jeu, et de serveurs de sessions pouvant héberger plusieurs sessions de jeu qui chacune héberge leur serveur de jeu.
Un serveur lobby peut servir plusieurs serveurs de sessions pourvu qu'ils hébergent le même jeu.
La base de données des utilisateurs est destinée aux routines d'authentification et de stockage d'informations. Le serveur lobby est interfaçable avec le système de paiement du provider.

Cette architecture représente l'ossature principale de la plateforme Gex, la plateforme inclue également deux serveurs spécifiques offrant des services de communauté :

- le serveur **Yammo** implémentant un système « buddy » qui permet à un utilisateur de pouvoir visualiser ses amis en ligne et de les inviter à une session de jeu.
- le serveur de tournois offrant des services de gestion des compétitions

En raison des capacités limitées des mobiles, il ne peut y avoir d'installation du client Gex, ainsi Gex est prévu pour supporter J2ME via une API réduite Gex regroupant les fonctionnalités indispensables, les jeux Gex doivent donc implémenter l'API Gex.
Les services basiques de Gex sont accessibles via WAP.

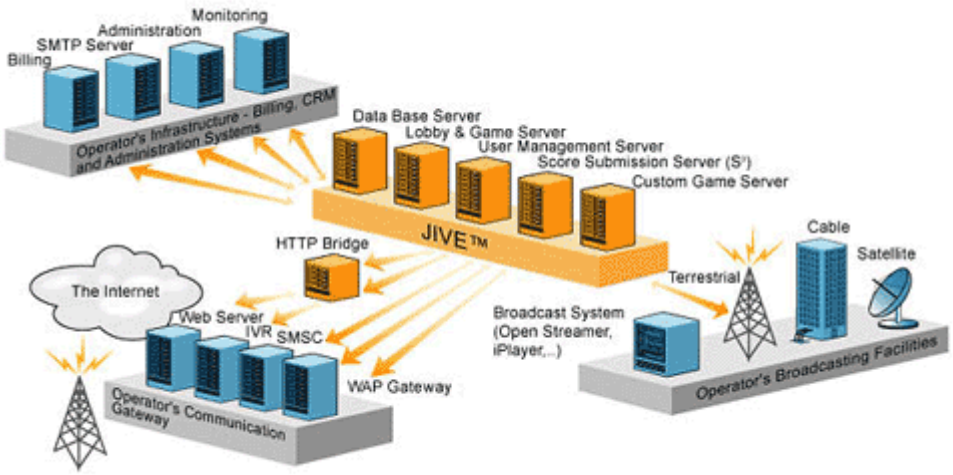
Autres services fournis par Gex

- Persistance des données, le serveur de jeu peut sauver et permettre de retrouver des données, ainsi une session de jeu peut être reprise ultérieurement
- Logging avancé, le serveur de jeu peut remonter certains évênements spécifiques du jeu vers le service de paiement, par exemple lors de changements de niveaux
- Gestion des scores, le serveur de jeu peut sauver les scores, les transférer au service de tournois.
- Téléchargement en tâche de fond, certaines parties du jeu peuvent être téléchargées en cours de session
- Mises à jour automatique

5.4 Jive

Jive [17] est une plateforme de jeu multijoueur pour mobile développée par la société Pixel Technologie disponible sous forme de SDK intégrant les outils requis pour l'intégration, la configuration et la maintenance des services pour les jeux.
Jive a obtenu la certification partenaire ASP (Application Service Provider) de Microsoft garant de ses performances lors de son déploiement sur un réseau opérateur réalisé autour des dernières technologies Microsoft.

Architecture de Jive



La plateforme Jive est composée de:
- serveurs de base de données
- serveurs lobby
- serveurs de jeu
- serveurs de gestion d'utilisateur
- serveurs de scores
- serveurs alloués à un jeu donné

Services utilisateur

- Gestion de scores
- Téléchargement des tableaux de scores
- Statistiques utilisateur
- Profil utilisateur
- Participation à des tournois
- Gestion de match
- Sauvegarde de jeu
- Chat
- Système « buddy », recherche d'amis
- Gestion du compte utilisateur

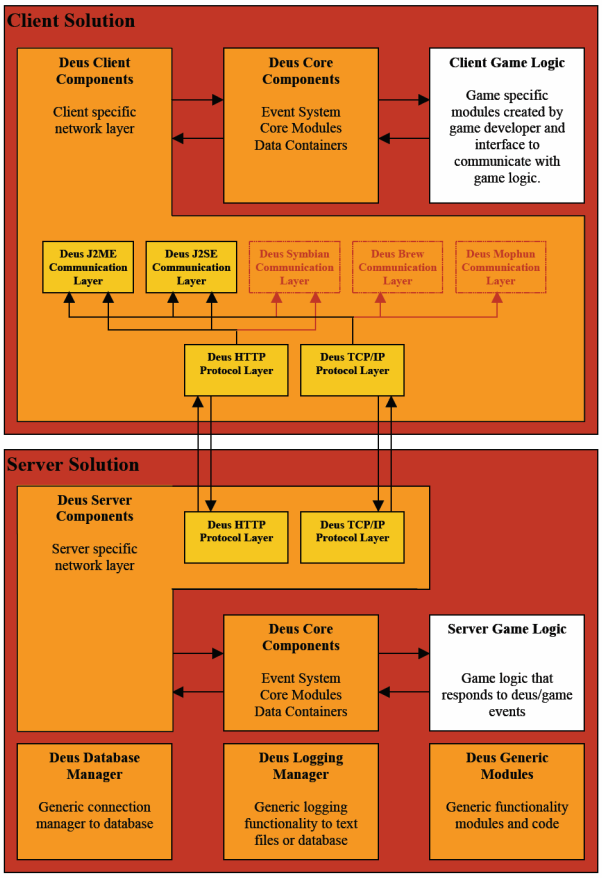
Services provider

- Interfacage avec le service paiement de l'opérateur
- Monitoring
- Administration système
- Gestion SMS/MMS, pour les envois de scores et d'informations diverses
- Mise à jour en temps réel du statut des joueurs et des sessions de jeu
- Gestion de contenu
- Gestion de tournois
- Fonctionnalités de communauté (chat, système « buddy, …)
- Gestion des utilisateurs

5.5 Deus+

Deus+ [18,19] est une plateforme de jeu multijoueur sur mobile développée par la société ExMachina, fonctionne sur tout type de terminal Java (MIDP 1.0, MIDP 2.0, Doja).

Architecture de Deus+



La plateforme Deus+ se décompose en un module client léger de 13kb et un module serveur.
Ces deux modules fonctionnent sous forme de servlets Java au sein d'un moteur de Servlet tel que Tomcat d'Apache Software par exemple. Les fichiers de configurations sont écrits en XML.
Deus+ intègre plusieurs composants :

- le **module utilisateur** : fournit des services génériques pour la gestion des utilisateurs, les services de login/logout, gestion de compte, stockage de préférences via le gestionnaire de bases de données
- **DeusLog** : logging des événements configurable via un fichier XML, produit en sortie des logs adaptés aux besoins, par exemple des logs pour la facturation
- **DeusDBManager** : gestionnaire de BD, autorisant les connections multiples à la base de données
- **LightWeight** : architecture serveur supportant plus de 2000 joueurs

Services utilisateur

- Login/Logout
- Gestion de compte
- Personnalisation
- Gestion de match
- Salle de rencontre
- Chat

Services provider

- Logging/Statistiques
- Gestion de score
- Gestion de salles d'attente pour match
- Persistance des données, gestion de caches avancées

5.6 KayaK Interactive

La plateforme proposée par la société Kayak Interactive se décompose en trois modules : KayaK Live, KayaK Studio et Kayak community [20].

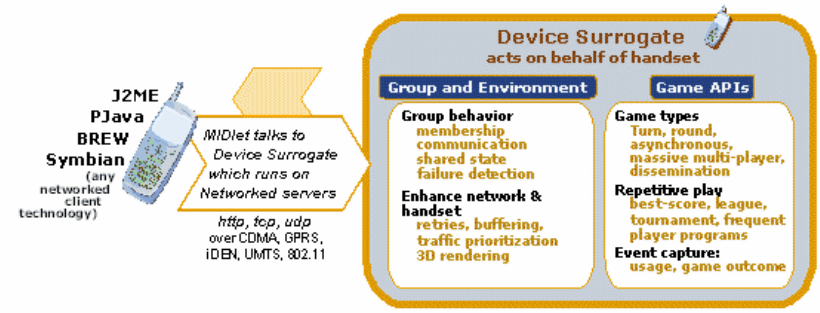
Kayak Live

Kayak Live constitue la plateforme de jeu multijoueur pour mobile basée sur Java et MIDP.
Elle offre des services avancés pour l'administration du jeu:

- Gestion de groupes de joueurs
- Gestion des intéractions entre joueurs, chat en cours de jeu par exemple
- Gestion des scores
- Hall virtuel de jeu permettant, de selectionner un jeu, de rejoindre une partie, d'inviter un ami, de répondre aux défis
- Gestion de compétitions

Elle fournit également des services systèmes :

- Persistance des données
- Monitoring
- Détection d'erreur
- Logging
- Services avancés : 3D rendering, buffering
- Gestion avancée du traffic, répartition de charge



KayaK Studio

Collection d'API pour les developpeurs contenant :

- Gestion de l'identité des utilisateurs pour l'authentification et la création de profil
- Gestion des souscriptions de l'utilisateur aux jeux
- Création de compétitions
- Outil de génération de statistiques (meilleurs scores, …)
- Monitoring de l'activité de l'utilisateur
- Reprise sur erreurs,
- Création de modèle de données persistantes
- Informations et messagerie personnalisée

KayaK Community

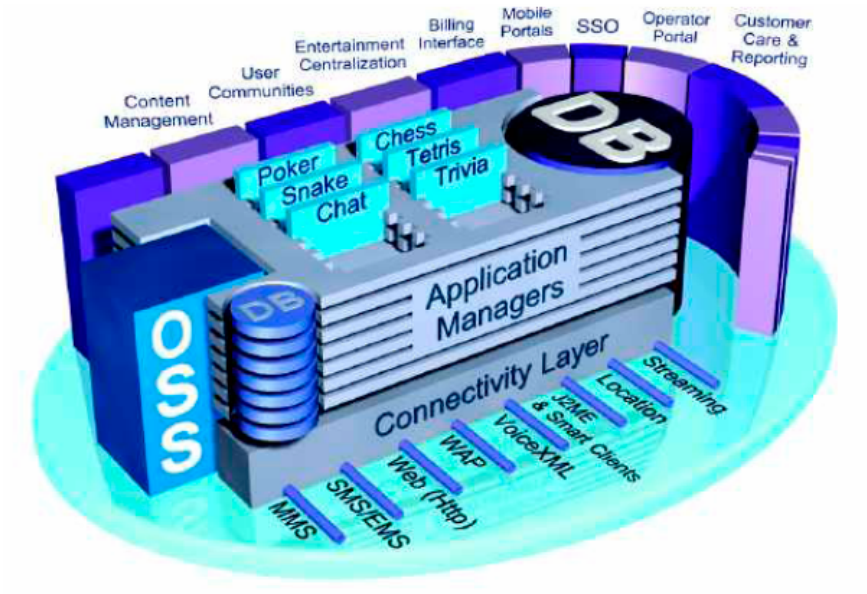
Met à disposition des outils pour la création de structures robustes pour la mise en place de communautés de joueurs :

- Création/Gestion des profils utilisateurs
- Système « buddy », réseau d'amis
- Système de blocage de joueurs indesirables
- Système permettant à l'utilisateur de rejoindre une compétition
- Gestion des compétitions, création des logiques de match (forfait, report, tours, …)
- Création de différentes catégories de jeu (gratuits, payants, betas, requièrants certains types de mobiles 3G par exemple)

5.7 Pecan Gaming

Pecan Gaming [21] est une pateforme de services avancés pour jeux sur mobile développée par Cash-U Mobile Technologies. Elle offre des fonctionnalités systèmes, de gestion de contenu, de création de communautés, de gestion de services, de gestion des utilisateurs et de gestion des interactions entre utilisateurs.
Un SDK est fournit aux developpeurs librement, Pecan SDK intégrant un environnement de développement d'application, un environnement de test et de debugging, un environnement de simulation pour tester les différents protocoles de communications et une application de certification Cash-U.

Architecture de Pecan Gaming



Cette architecture dispose d'un support avancé pour J2ME : support du jeu en temps réel, centralisation des scores, contenu dynamique, lobbies et chat rooms
Le service SSO (Single Sign-On) permet à l'utilisateur de ne se connecter qu'une seule fois et ainsi avoir accès à l'ensemble des plateformes connectées.

Services systèmes

- Tolérance aux fautes
- Persistance des données
- Gestion multi-protocoles via la couche de connectivité (J2ME, http, SMS, EMS, MMS, WAP, …)
- Logging
- Framework de Pré/Post paiement
- Statistiques pour l'analyse système et marketing
- Monitoring des utilisateurs et des applications

Gestion de contenu

- Déploiement de nouvelles applications
- Modifications de jeux
- Mise à jour de contenu

Gestion de communautés

- Salle de lobby, pour la rencontre des joueurs
- Système « buddy »
- Gestion de compétitions
- Informations et publicités

Gestion des interactions entre utilisateurs

- Chat
- Conversation vocale

Gestion des utilisateurs

- Gestion des profils et possibilité au client de mettre à jour ses informations via l'interface web

5.8 Tableau comparatif

Plateformes Services	Terraplay Move	MServer	GeX	Jive	Deus+	KayaK Inter.	Pecan Gaming
Utilisateur							
Authentification	√	√	√	√	√	√	√
Téléchargement	√	√	√	√	√	√	√
Personnalisation	√		√	√	√		√
Informations						√	√
Lobby	√	√	√	√	√	√	√
Création de parties	√		√				
Gest° de match	√		√	√	√	√	√
Gest°de compétitions	√		√	√	√	√	√
Gest° d'équipe							
Gest° de scores	√	√	√	√	√	√	√
Réseau « buddy »	√		√	√		√	√
Chat		√		√	√	√	√
Halls virtuels		√		√	√	√	√
En cours de jeu							
Gestion de session	√	√					
Communication	TCP/IP					√	√
Scores temps réel				√			
Editeur							
Configuration	√	√	√	√	√	√	
Gestion de contenu	√	√	√	√	√	√	√
Organisat° de tournois	√		√	√		√	
Gest° des joueurs		√		√		√	
Prix et récompenses							
Advertising	√					√	
Système							
Contrôle d'accès	√	√	√	√	√	√	√
Capacité du terminal	√						
Monitoring	√			√		√	√
Allocation de ressources	√						
Interfaçage Facturation	√		√	√			√
Logging	√	Log4J	√		√	√	√
Data-Mining	√		√		√		√
Persistance des données		√	√			√	
Gest° des Droits Digitaux				√			
Géo-localisation		√					
Présence / localisation							
BD	√	JDBC	√	√	√	√	√
Communication P2P		√				√	
Tolérance aux fautes							√
Gestion multi-protocoles						√	√
Routage de données avancé	√					√	

5.9 Analyse

Il existe d'autres plateformes par exemple la plateforme JAP (Java Acotel Platform) [22] d'Acotel surcouche de J2ME et MIDP ou encore la plateforme Neutron d'Exit Games [23] conforme au spécifications MGIF (voir section 6) mais ne présentent pas de différences fondamentales avec les architectures vues précédemment.

Le précédent tableau établissant un comparatif inter-plateforme nous permet de dégager une base commune de services.

Ces plateformes proposent des services utilisateurs communs orientés « compte » comme l'authentification et la personnalisation, ensuite elles offrent au joueur la possibilité de trouver des parties et des joueurs avec le « lobby », les halls virtuels et autre réseau d'amis « buddy», permettant souvent la communication inter-joueur nécéssaire à une expérience réellement interactive.
Au niveau des services en cours de jeu, on retrouve la gestion de session permettant de préserver les informations de jeu en cas de déconnexion non volontaire, ainsi que la possibilité de communiquer avec les autres joueurs.

Du côté des services pour les éditeurs, on retiendra la possibilité de configurer et gérer le contenu des jeux, la gestion des joueurs, ainsi que le service d'organisation et de gestion de compétitions.

Les services systèmes sont cruciaux dans l'établissement d'une plateforme sécurisée, performante, robuste, adaptative et lisible.
Une telle plateforme doit disposer de fonctionnalités telles que le contrôle d'accès, le routage de données avancé, l'équilibrage de charge, la résistance aux fautes, la persistance de données, la gestion multi-protocoles, le monitoring, le logging et l'analyse statistique.
Ce tableau montre certaines carences notamment en ce qui concerne la tolérance aux pannes mais aussi l'équilibrage de charge, elles peuvent être la conséquence de la volonté des éditeurs de plateformes de tenir secret le côté technique de leur plateforme afin de ne pas favoriser la concurrence.

On obtient donc après comparaison de ces plateformes, un pool de services représentant les services indispensables à la mise en place d'une plateforme complète.

Les plateformes existantes n'étant absoluement pas interopérables, ni compatibles, il convient de mettre en surbrillance la nécéssité de standardiser le domaine des plateformes de jeux multijoueur sur mobiles ceci afin de palier à cette contraignante hétérogénéité, notamment pour faciliter le travail des éditeurs de jeux.
Cette standardisation devrait disposer, comme pool de services standard, les services dégagés précédemment.
C'est dans cette logique que s'inscrit la volonté de standardisation de l'Open Mobile Alliance, présentée dans la section suivante.

6. Tentative de standardisation : OMA-MGIF

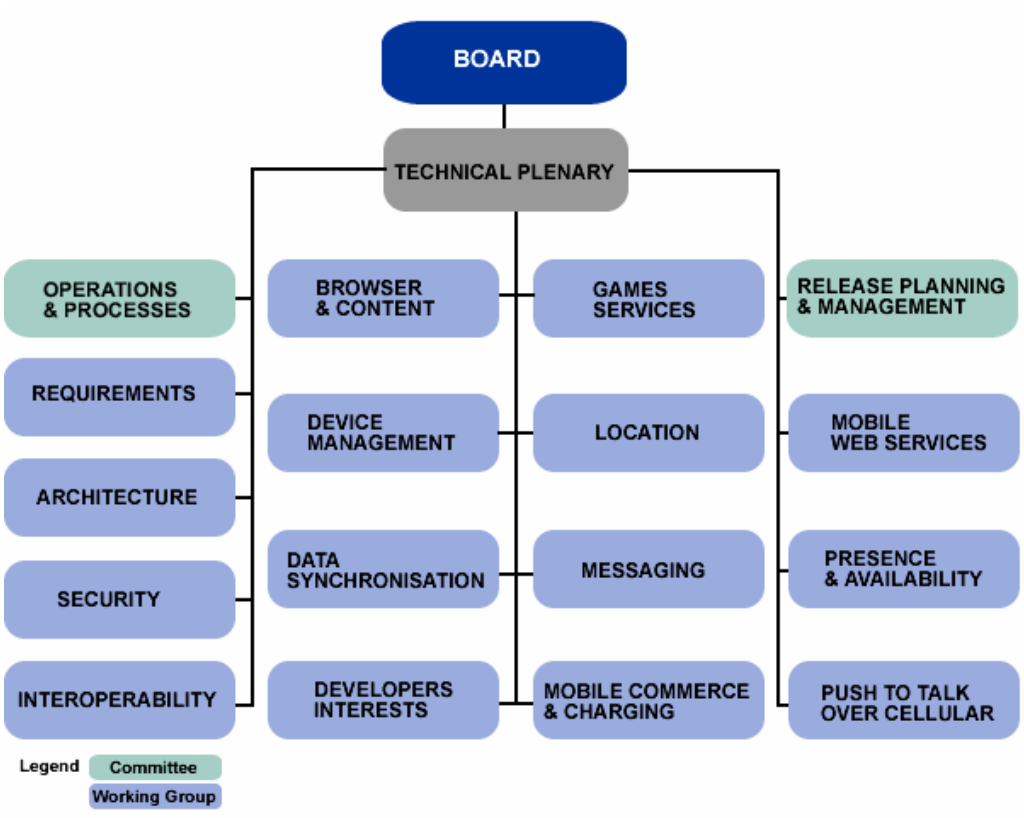
6.1 Présentation de l'OMA et du MGIF

L'OMA (Open Mobile Alliance) [24] a été formée en 2002 par un groupement de plus de 200 compagnies d'horizons divers opérateurs mobile, entreprises de fabrication de terminaux mobiles, entreprises spécialisées dans les réseaux, dans les technologies de l'information et fournisseurs de contenu.
On retrouve au sein de l'OMA : Nokia, IBM, Motorola, Siemens, Intel, Sun, …
L'objectif de cette alliance est d'établir des technologies standards pour mobile, aussi bien au niveau de l'architecture des réseaux mobiles, qu'au niveau des technologies pour la sécurité ou encore des services web sur mobile. Cette recherche de standardisation s'inscrit dans une volonté d'homogénéisation et d'intéropérabilité des technologies mobiles.
Depuis la création de l'OMA, de nombreux forums technologiques ont intégré cette organisation, notamment le forum WAP, l'initiative SyncML, Wireless Village ou encore le Mobile Wireless Internet Forum.

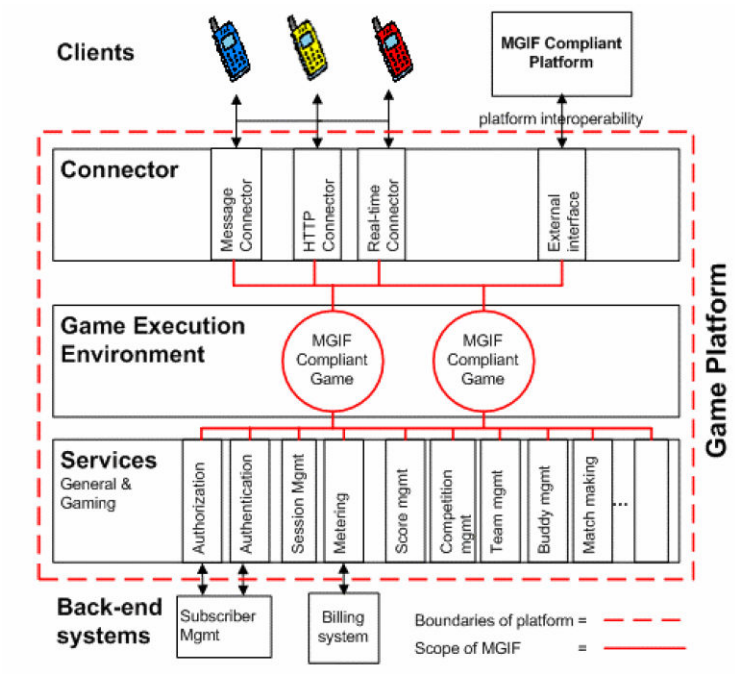
Le forum qui lui aussi à rejoint l'OMA récemment et qui nous intéresse plus particulièrement est le MGIF (Mobile Gaming Interoperability Group) [25] créé par Ericsson, Motorola, Nokia et Siemens qui a pour objectif la mise en place de services standard pour les jeux multijoueurs sur mobiles. L'intégration du MGIF au sein de l'OMA à permis la création du groupe de travail « Games services».

De nombreuses plateformes implémentent l'API MGIF 1.1, notamment la plateforme de jeu Exit Games, très active dans la normalisation des services au sein de l'OMA.
Une plateforme de ce genre se dit conforme aux spécifications MGIF, « MGIF compliant ».

Ci-dessous l'ensemble des groupes de travail au sein de l'OMA.



6.2 Architecture d'une plateforme conforme à MGIF



L'architecture d'une plateforme « MGIF compliant », a deux objectifs prioritaires l'**interopérabilité** des plateformes dans le sens ou deux plateformes MGIF ou même deux jeux peuvent communiquer et partager des ressources ou des services, et la **portabilité,** un jeu « MGIF compliant » pouvant fonctionner sur toute plateforme « MGIF compliant ».

Les services proposés par l'API MGIF sont interfaçables avec des services spécifiques à l'opérateur tels que les services de facturation ou encore les services d'authentification utilisateur.

6.3 Spécifications de la plateforme version 1.0

La version 1.0 de la platefome de jeu courante correspond à la version 1.0 de la plateforme MGIF [26]. Cette plateforme est disponible librement sous forme d'API Java et offre les services ci-dessous

Gestion de session

C'est le cœur de la plateforme établissant le lien entre l'utilisateur et l'instance du jeu côté serveur, c'est grâce à ce service que vont pouvoir être accéder les autres API et interfaces necessaires à la création de l'application. C'est cette API qui contrôle le cycle de vie de l'application.
Les entités principales constituant l'instance d'une session sont les instances des classes **ActorSession**, représentant l'utilisateur, et **ApplicationInstance,** représentant l'application.

Connectivité

Service qui permet d'établir la communication entre les clients et l'application, comment les requêtes des clients vont être acheminées et comment les réponses de l'application vont être générées et retournées aux clients**.**
Le modèle de communication est caratérisé par 4 méthodes : **client pull**, **client push**, **application pull** et **application push**.
De plus 3 modes de communications sont pris en compte : **asynchrone** (pas d'attente de l'accusé de réception) et **synchrone** (attente de l'accusé de réception avant de continuer).

A noter que dans cette version seul les clients de type messagerie ou navigateur sont adressés, les clients exécutables seront intégrés dans les futures versions de l'API.

Metering

Ce service permet de faire remonter des événements pour leur utilisation dans des statégies de facturation, ces événements peuvent être de type traffic ou spécifique au jeu.

- Evênements de type traffic : durée de la session, taille des données transférées, autorisations de souscription/pré-paiement.
- Evênements spécifiques au jeu : par exemple le changement de niveau, ce type d'évênements est crée dans l'instance de la classe ActorSession, les évênements sont paramêtrables et représentés par un entier qui est remonté.

Gestion des scores et des compétitions

Cette API fournit les mécanismes d'enregistrement et de récupération des scores. Les gestionnaires de scores et de tableaux de score sont accessibles par les classes **ScoreManager** et **ScoreTableManager**, c'est par cette dernière classe que l'on peu récupérer d'anciens scores. Cette api permet également la mise place en de nombreuses formes de compétitions.

Timers

Les applications nécessitant des délimitations de temps vont faire appel à cette API, ces Timers sont présents au sein des classes ApplicationInstance et ActorSession. Des timers spécifiques pour les fichiers de configuration ou les outils d'administration peuvent être crées via la classe **MasterApplicationInstance**.

Logging

Service permettant la création de logs pour le debugging, le repérage d'erreurs et le monitoring via l'interface **Logger**.

Bien sûr face aux nombreux services fournis par les plateformes propriétaires vues précédemment, les fonctionnalités peuvent paraître très réduites, mais c'est là commettre une erreur de taille.
La normalisation des technologies prend du temps et a pour objectif de ne retenir que les technologies pertinentes, elle confère un certain recul par rapport au fourmillement de nouvelles technologies souvent abandonnées.
L'homogénéisation des services est incontournable afin d'éviter un dédale de technologies propriétaires offrant les mêmes fonctionnalités.

6.4 Apport des futures plateformes version 2.0 et 3.0

Plateforme version 2.0

La plateforme version 2.0 est en cours de spécification, elle devrait fournir d'après quelques drafts publics les services supplémentaires suivants :

- Capacités du terminal
- Gestion « buddy », réseau d'amis
- Organisation de matchs
- Informations via messagerie
- Localisation
- Persistance des données
- Monitoring avancé

Plateforme versions 3.0

La GP3.0 (Gaming Platform) intègrerait les services suivants :

- Gestion de contenu
- Contrôle d'accès avancé
- Publicité, Annonces - « Advertising »
- Personnalisation
- Gestion des prix et récompenses
- Gestion des droits digitaux

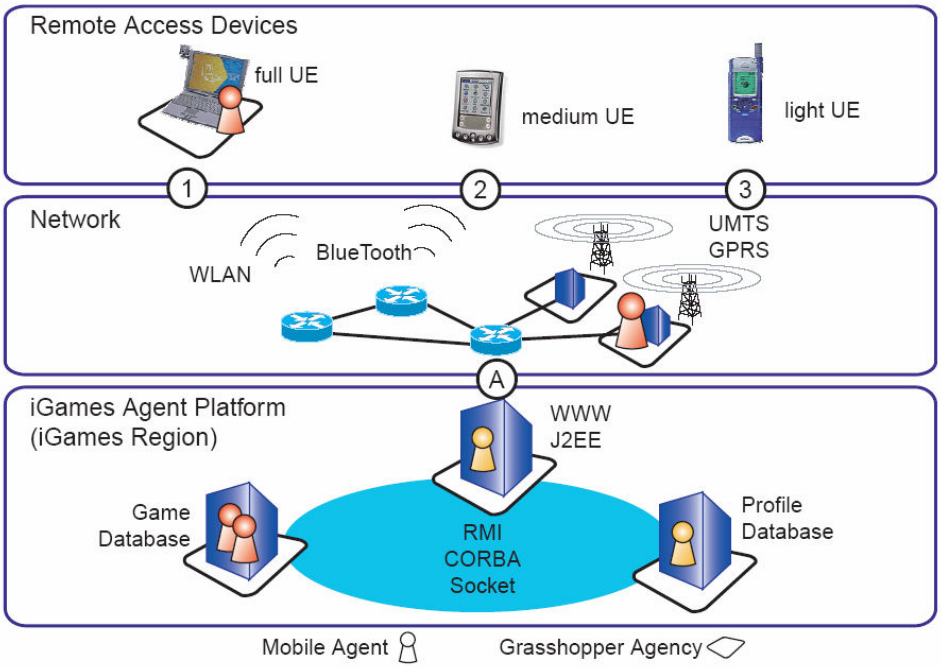
Aucunes dates n'ont été avancées en ce qui concerne la disponibilité publique de ces versions.

7. Architectures émergeantes

7.1 Approche agents mobiles pour les gaming services : iGames

Cette étude [27] réalisée par Fiorenzo Gamba et Jean-Frédéric Wagen du département de télécommunication de l'Université des sciences appliquées de Suisse Ouest s'inscrit dans le cadre de services à la demande dans des zones restreintes d'accéssibilité à ces services ou « Micro-Spots ».
Par exemple un client possédant un mobile arrive dans une zone Micro-Spot, il reçoit alors des informations envoyées par le fournisseur de service local lui indiquant la disponibilité du service iGames.Le client décide alors de se connecter à iGames, souscrit à un jeu et peut alors débuter une partie multi-joueur avec des adversaires se situant dans d'autres Micro-Spot.Le jeu est un agent mobile qui migrer sur le terminal du client en adaptant dynamiquement l'interface de l'utilisateur en fonction des profils utilisateur et terminal.

Architecture de la plateforme iGames



La plateforme de services iGames est basé sur Grasshopper, une plateforme Java de developpement d'agents mobiles développée par IKV++ compatible avec les systèmes Unix, Gnu/Linux et Windows.
Elle fournit les fonctionnalités suivantes :

- Enregistrement
- Accès
- Authentification
- Envoi de code mobile
- Gestion de joueur
- Administration système

La plateforme iGames est constituée :

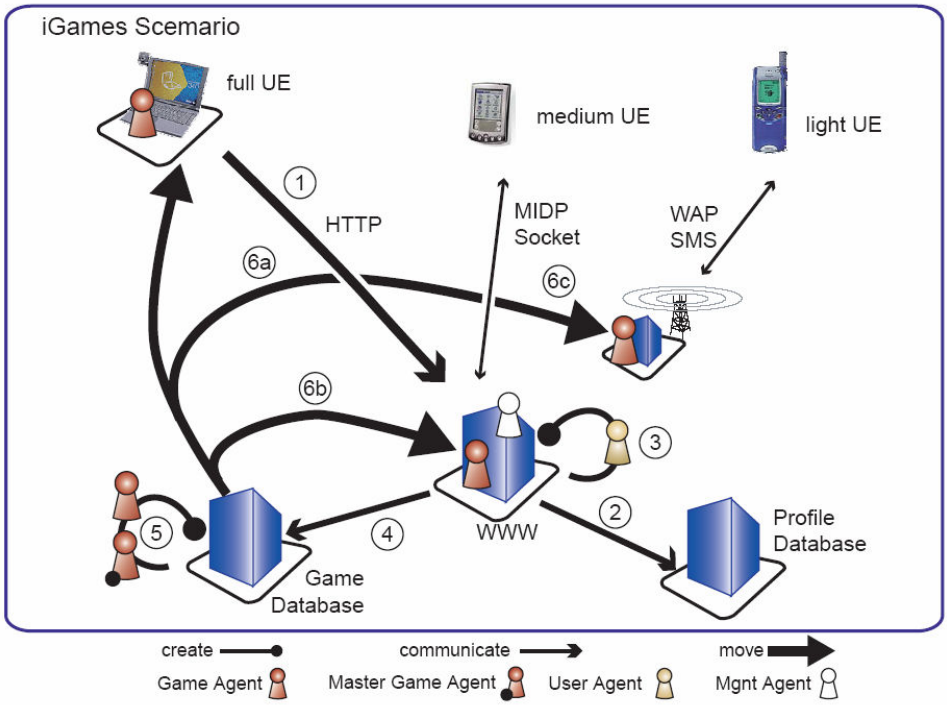
- d'une BD de profils stockant les informations sur l'utilisateur et son terminal
- d'une BD de jeux qui constitue un entrepôt de classes d'agent-jeu
- un portail d'accès Web/Wap Apache intégrant un moteur de servlet Tomcat et offrant des services de login et d'administration traditionnels

Le réseau de communication géré par iGames englobe les protocoles WLAN, bluetooth et UMTS/GPRS.

En ce qui concerne les clients, plusieurs types de terminaux sont gérés, les terminaux:

- Full UE (utilisateur/equipement), pouvant acceuillir un système complet d'agents
- Medium UE, disposant des fonctionnalités J2ME/MIDP, ne pouvant pas acceuillir de système d'agents mais capable de contrôler et de communiquer avec l'agent via un proxy sur la plateforme iGames, disposant d'une interface graphique paramétrable
- Light UE, pouvant uniquement communiquer via SMS ou WAP

Scenario de connexion à la plateforme iGames



1- Un servlet acceuille toutes les connexions offre un accès login en dialoguant avec l'agent Mgnt(management), si l'utilisateur se connecte pour la première fois un lien lui propose de s'enregistrer

2- L'agent Mgnt vérifie le login et le password de l'utilisateur et peut retrouver les profils Utilisateur/Equipement (UE)

3- Pour chaque utilisateur loggué l'agent Mgnt crée un agent dédié, le UserAgent qui est une représentation active du profil client et implémente une interface avec le client

4- Le UserAgent va pouvoir renseigner le client sur la liste des jeux disponibles

5- Le client va selectionner un jeu, le UserAgent va alors créer deux agents supplémentaires appelés MstGameAgent (master) et GameAgent dans le cas de la création d'une nouvelle partie, sinon uniquement le second. Le MstGameAgent teste le nombre de joueurs connectés et lance la partie, tandis que le GameAgent collabore avec les autres GameAgent et communique avec le MstGameAgent, et va également tenter de se rapprocher le plus possible du terminal du client en fonction de son type

6a- Le GameAgent est chargé idéalement sur le terminal Full UE mais requiert l'installation et l'enregistrement d'une agence Grasshopper

6b- Il reste sur la plateforme iGames et fonctionne comme un proxy accéssible via requête MIDP

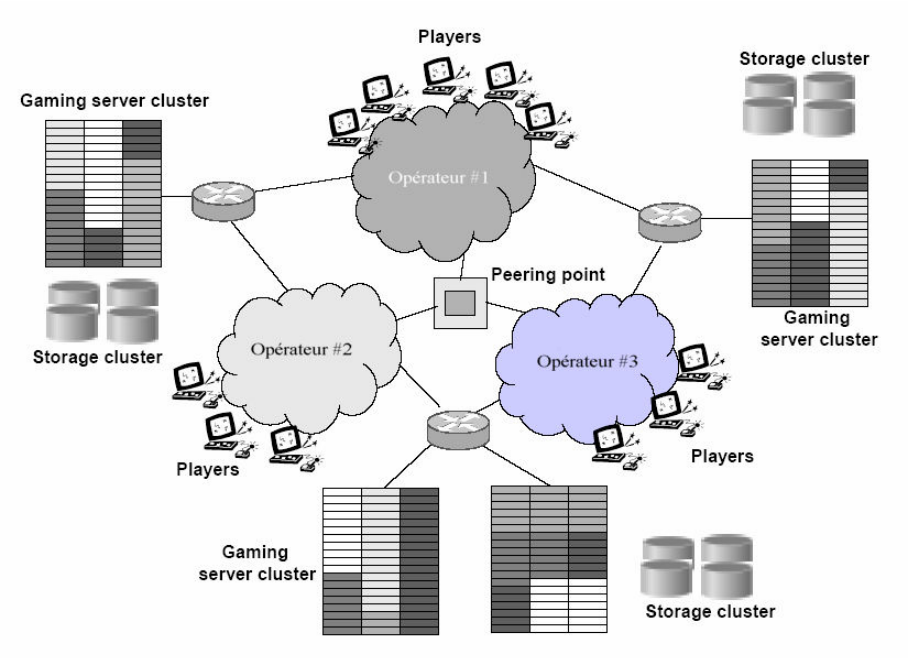
6c- Il rejoint le BSC ou le point d'accès GSM/GPRS, UMTS ou WLAN mais ceci requiert l'installation de la plateforme iGames sur le réseau de l'opérateur

Les tests effectués ont conclu à de très bonnes performances si le GameAgent est hébergé par la plateforme elle-même, cependant pour fonctionner dans de bonnes conditions dans le cas 6b Grasshopper requiert une bande passante de 1Mb/s pour le moment mais des améliorations de Grasshopper ont à venir.

Le concept d'utilisation d'agents mobiles est très intéressant dans sa capacité à proposer des services en fonction de zones prédéterminées, ce qui peut ouvrir des perspectives très intéressantes d'adaptation des services offerts à l'emplacement de l'utilisateur. Par exemple on pourrait imaginer un Micro-spot au alentour d'un cinéma diffusant actuellement un film F, ce film disposant d'une adaptation sous forme de jeu multijoueur pour mobile JF, iGames peut alors proposer à l'utilisateur de jouer au jeu JF en attendant le début de son film.

7.2 Plateformes de jeu sur GRID

Le GRID est un système distribué à grande échelle consistant en l'interconnexion de site composés de clusters puissants via des liens réseaux fiables et sécurisés à très grande vitesse, de 45 Mb/s à 2 Gb/s.
Ce type d'architecture réseau ressemble comme deux gouttes d'eau à l'architecture réseau d'un opérateur pour mobiles et de manière plus globale au réseau d'interconnexion entre réseaux opérateurs.



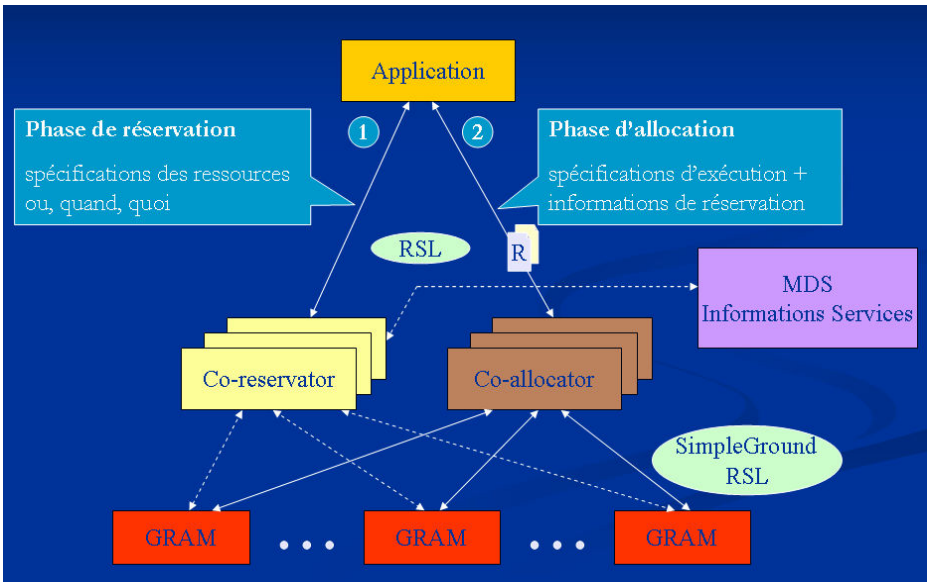
Des middleware tirant la quintescence des réseaux de type GRID existent déjà, et plus particulièrement Globus [28] de l'Alliance Globus qui offre des fonctionnalités de gestion avancée de ressources, de services d'informations sur l'état global du système (charge des différents clusters, statut, …), la gestion des données des clusters de stockage et des services de sécurité.

Des plateformes de jeux pourrait être implémentées au dessus de la couche middleware Globus afin de profiter des fonctionnalités et performances des systèmes sur GRID.

Service d'informations : MDS

Le MDS (Monitoring and Discovery Services) combine les services de monitoring du système et de découverte fournissant les informations sur les caractéristiques des clusters, type (de calcul ou de stockage), connectivité et statut (charge).

Gestion avancée de ressources : GARA



L'attribution des ressources par GARA (Globus Architecture for Reservation and Allocation) [29] se fait en deux phases, la phase de réservation des ressources (puissance de calcul requise, date de début d'execution, type de ressources) en se basant sur les informations du MDS qui va produire un ticket de réservation détenu par l'application, ensuite vient la phase d'allocation qui joint le ticket et les données à traiter et sont transmises via les allocateurs aux clusters sélectionnés.

Gestion de données : GASS

La gestion de données par GASS (Globus Access to Secondary Storage) va permettre de lire ou d'écrire de manière sécurisée des données entre deux points distants, ce qui est très intéressant de par le fait que l'on puisse transférer des éxecutables ou des données de cluster à cluster.

Sécurité : GSI

La gestion de la sécurité par GSI (Globus Security Infrastructure) permet d'établir des communications sécurisées et authentifiées basé sur la cryptographie à clé publique, GSI est particulièrement bien adapté aux stratégies de sécurité necessaires dans les réseaux opérateurs et notamment pour les services sensibles de paiement.

La société IBM a également mené des études sur ce type d'architecture [30] et aboutit à la même conclusion, les technologies de type GRID grâce à une architecture d'accueil se rapprochant beaucoup de l'infrastructure des opérateurs télécom, permettrait d'optimiser la gestion des ressources au sein des plateformes de jeux multijoueurs.

8. Conclusion

Il existe un large choix de plateformes offrant des services pour les jeux mulijoueurs sur mobile, mais ces plateformes ne sont pas interopérables, ni réellement complètes, d'autant plus que les jeux crées pour une plateforme ne sont pas portables d'une plateforme à une autre.
Même si elle accuse pour l'instant un certain retard par rapport aux solutions propriétaires, la standardisation de l'Open Mobile Alliance constitue une solution au problème de l'hétérogénéité des plateformes propriétaires.
D'autres part la recherche de mécanismes plus robustes, plus performants afin de pouvoir mettre en place des plateformes plus fiables, plus stables et surtout plus performantes, est plus que jamais d'actualité afin de satisfaire les exigences des utilisateurs qui payent ces services au prix fort.
La solution pour la création de telles plateformes peut passer par l'adoption de technologies prévues pour le GRID qui constituent une base extrêment solide, performante et surtout adaptée aux réseaux des opérateurs.
L'approche agents mobiles quand à elle permet d'entrevoir de nouveaux services, tel que des services géo-adaptés par exemple.
Il est important de noter que nous assistons actuellement à l'émergence d'un nouveau marché constituant un véritable challenge en termes technologiques, la recherche dans son intégralité se doit d'y répondre et plus particulièrement la recherche sur les systèmes distribués.

9. Bibliographie

[1] *D4.1.1 Existing and Imminent Mobile Entertainment*, MGAIN, 2003
[2] http://java.sun.com/j2me/
[3] http://java.sun.com/products/midp/
[4] http://java.sun.com/products/cdlc/
[5] J.Kikkre, *Samurai Romanesque, J2ME and the battle for mobile cyberspace*, IEEE, 2003
[6] http://www.nttdocomo/corebiz/imode/why/guideline/javamanual.html
[7] http://developper.in-fusio.com/
[8] http://www.symbian.com/technology/symbos-v8x-det.html
[9] http://www.mophun.com/developer/
[10] http://brew.qualcomm.com/
[11] http://www.microsoft.com/windowsmobile/
[12] *Terraplay Move : mobile multiplayer gaming service*, Terraplay, 2003
[13] http://www.terraplay.com
[14] *Mserver 3.0 Developer's Manual*, mForma, 2002
[15] http://devnet.mforma.com/
[16] http://www.gamefederation.com/gex_technical_summary.html
[17] http://www.pixel-tech.com/mobile.html
[18] *Deus+ SDK Whitepaper*, ExMachina, 2004
[19] http://www.exmachina.nl/
[20] http://www.kayakinteractive.com/
[21] http://www.cash-u.com/
[22] http://www.acotel.com/html/products/java2microe/index.php
[23] http://www.exit-games.com/
[24] http://www.openmobilealliance.org/
[25] *MGIF white paper v1.0*, MGIF, 2002
[26] *MGIF platform specification v1.0*, MGIF, 2002
[27] F.Gamba et JF.Wagen. *iGames: Implementation et Performance of a Gaming Service using Mobile Agents*, University of Applied Sciences of Western Switzerland, 2002
[28] http://www.globus.org/
[29] I.Foster, A.Roy et V.Sander. *A guide to GARA*, Globus Alliance, 2000
[30] D.Saha, S.Sahu et A.Shaikh. *A Service Platform for OnLine Games*, IBM, 2002