

Continuité des services de télécommunications dans l'IMS

3.1 Motivation

On ne peut aujourd'hui étudier les services de télécommunications sans considérer l'IP Multimedia Subsystem (IMS cf. 1.2.2). Cette architecture nouvelle génération a donc été le point de départ de notre étude sur la continuité et ce pour plusieurs raisons : une orientation « télécoms » qui correspondait à l'angle d'attaque choisi, une infrastructure complète, standardisée et prometteuse s'imposant comme la nouvelle architecture de référence et enfin la disponibilité de plateformes expérimentales IMS permettant d'évaluer nos contributions au sein même de nos laboratoires à TELECOM SudParis et Alcatel-Lucent Bell Labs France.

L'infrastructure IMS est le cœur fonctionnel de la téléphonie mobile nouvelle génération (3G), il comporte ainsi un grand nombre de mécanismes participant à la gestion de la mobilité : routage, itinérance (*roaming*), profils de service, identification, etc. De plus, l'un des objectifs majeurs étant d'assurer la convergence des réseaux, l'interopérabilité des différentes technologies d'accès est réalisée autour d'un noyau IP commun, l'IMS, sur lequel repose les services. La couche accès étant dissociée de la couche service dans ce modèle horizontal, un plus grand nombre de terminaux et donc de clients potentiels apparaissent ; les services doivent alors

être délivrés de manière cohérente entre ces appareils électroniques de plus en plus nombreux, puissants mais surtout aux capacités hétérogènes.

Avec plus de réseaux d'accès, plus de terminaux et plus de clients, le besoin en mobilité se renforce. Si les mécanismes IMS basés sur le protocole SIP (cf. 2.3.2) assurent les différents types de mobilité tels que définis par l'ITU (cf. 2.1), la mobilité de service au sens propre, qui inclut la contrainte de continuité, n'est pas assurée. Les problématiques classiques de continuité : network et terminal handover, ne sont pas prises en compte.

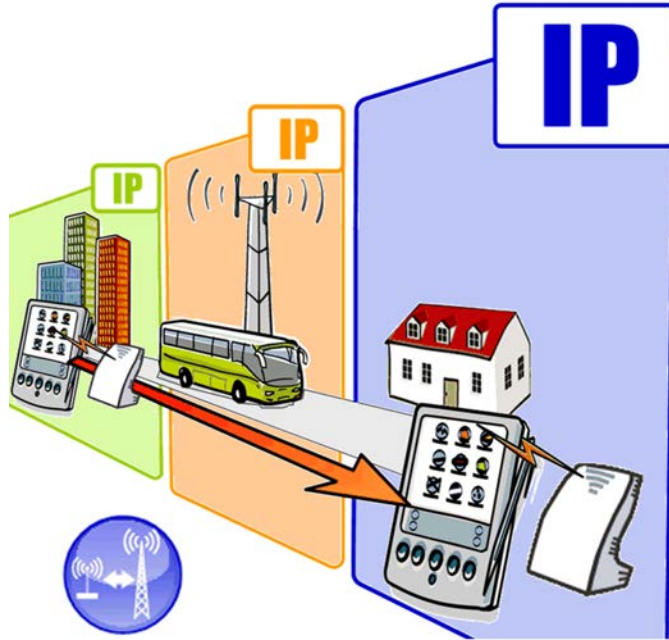
Le network handover est partiellement réalisé uniquement. Si les handovers horizontaux sont gérés de manière transparente pour l'IMS au niveau de l'accès, les handovers verticaux nécessitant un changement d'adresse ont pour incidence d'interrompre les sessions de communications. Or l'IMS, reposant sur SIP, ne délivre intrinsèquement que des services basés sur des sessions. Ce type de continuité requiert des mécanismes spécifiques qui font actuellement défaut dans l'IMS.

Le terminal handover, quant à lui, n'est simplement pas réalisable en l'état actuel de l'infrastructure. Le protocole SIP peut théoriquement effectuer un transfert de sessions entre plusieurs terminaux, mais d'un point de vue architectural l'IMS n'implémente pas les mécanismes requis pour mettre en œuvre ces solutions.

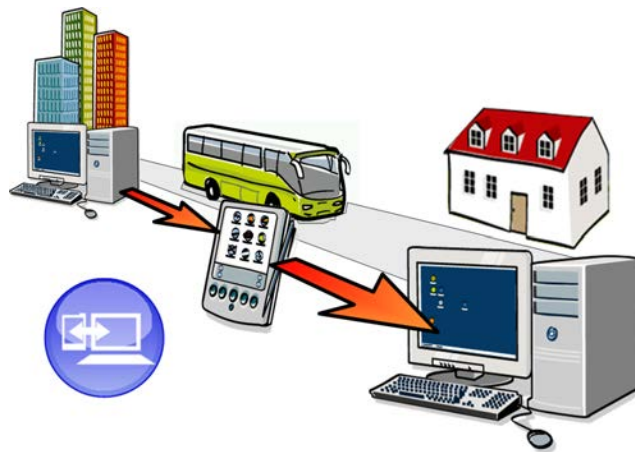
Face à un besoin croissant en mobilité et une infrastructure de services 3G telle que l'IMS qui ne peut assurer une continuité et donc une expérience utilisateur satisfaisante, nous essayons de répondre à cette problématique et ses deux principales contraintes : le network handover (mobilité de terminal) et le terminal handover (mobilité d'utilisateur). L'objectif est de permettre à un utilisateur de conserver ses services en mobilité lors de la transition entre différents milieux : au travail, à la maison, dans un lieu public... transition qui se traduit généralement par divers transferts (inter-terminaux et technologies cf. Figure 3.1). Nous étudions ces différents scénarios dans le cadre de l'IMS, donc d'un point de vue applicatif, et proposons des solutions compatibles avec les spécifications actuelles.

3.2 Problématiques et approche

L'IMS est une infrastructure fortement standardisée par de nombreux acteurs de la couche réseau à la couche applicative, assurant ainsi son interopérabilité,



(a) Network handover.



(b) Terminal handover.

FIGURE 3.1 – Scénarios de continuité.

son point fort. Afin d'offrir une solution applicable sans redéfinition de la norme, nous avons adopté une approche applicative qui essaie de résoudre les problèmes de continuité « d'en haut », via un serveur d'application (AS). Les AS sont des composants situés dans la couche service, contrôlés par l'IMS et qui offrent des fonctionnalités additionnelles (de celles de base tel que la communication audio) aux clients. Nous avons ainsi défini une application IMS supplémentaire, le *Session Continuity AS* (SCAS), dont le rôle est d'offrir la fonction de continuité de service. Cette nouvelle fonction est manipulée par l'IMS comme n'importe quel service et cible uniquement les services de télécommunications gérés nativement via SIP par l'infrastructure.

3.2.1 Network handover

Comme précisé en section 2.1.1, seul le cas du network handover *vertical* requiert un comportement applicatif spécifique pour assurer la continuité de service, le reste de cette section sera dédiée aux problématiques propres à ce cas précis dans la définition du *Session Continuity AS*.

Déclenchement.

Comment déclencher un transfert entre différents réseaux ? Les motivations d'un changement de technologie d'accès pendant la fourniture d'un service sont multiples : coût, encombrement du réseau, perte de signal, qualité de service requise, . . . De manière générale, ces motivations résultent d'une décision automatique, basée sur des règles, et effectuée au niveau du terminal ou de l'infrastructure selon des critères réseau. L'utilisateur est rarement l'initiateur d'un tel transfert, alors la question est moins de savoir comment mais quand effectuer ce transfert ?

Selon notre approche AS et avec un terminal standard, la décision ne peut s'effectuer qu'au niveau infrastructure et ce d'après les messages de signalisation SIP qui transitent par le cœur IMS. Un changement de réseau d'accès pourra alors se détecter de deux manières en fonction des capacités du terminal et de la couverture des technologies réseaux. Soit le terminal peut se connecter simultanément aux deux réseaux (on parle de *hard handover*), soit il devra se déconnecter au réseau d'origine avant d'initier la seconde connexion (on parle alors de *soft handover*) cf. [78]. On peut intuitivement supposer qu'une déconnexion préalable sera moins

optimale qu'une connexion simultanée d'un point de vue continuité (temporelle) du service.

Pour pouvoir se connecter simultanément à deux accès, deux conditions doivent être remplies au moment du transfert : le terminal doit être couvert par les deux technologies (cf. Figure 3.2) et il doit gérer les deux interfaces simultanément (au niveau réseau, propriété appelée *multihoming* [79] mais aussi au niveau radio, propriété appelée *dual mode*). Si les deux conditions sont remplies, le terminal restera connecté en continu avec l'infrastructure (via l'une ou les deux interfaces), un processus de transfert sans interruption est alors envisageable, sinon le terminal passera obligatoirement par une phase déconnectée (cf. Figure 3.3), on pourra alors seulement offrir une solution de secours ou reconnexion.

Soft handover.

Le *soft handover* est un transfert sans interruption réalisable si le terminal est connecté simultanément à l'IMS via deux interfaces différentes. Il est alors identifié doublement au niveau de l'infrastructure qui le voit comme deux entités distinctes, chacune possédant une adresse réseau différente. La problématique de network handover revient ainsi à celle de terminal handover, en transférant le service d'une adresse réseau à une autre le mécanisme de continuité réalisera en fait un changement d'interface sur le même terminal. Les détails du mécanisme en question sera présenté dans la section 3.3.3.

Il est à noter que l'infrastructure IMS doit supporter l'enregistrement multiple des clients. En effet, l'adresse SIP-IMS (appelée SIP URI) doit pouvoir être associée à de multiples adresses réseaux. Cette fonctionnalité aussi appelée *multi-registration* incluse dans la norme est parfois non supportées par les solutions IMS. Le cas échéant, le terminal devra utiliser un SIP URI différent pour chaque interface et non par terminal, le mécanisme de transfert sera lui indique.

Hard handover.

Lorsque la déconnexion est inévitable, une solution alternative de (relative) continuité consiste à rétablir le service en initiant une nouvelle connexion. En enregistrant les messages de signalisation, le SCAS peut sauvegarder les différentes caractéristiques des services de communications établis dans l'IMS. L'interruption

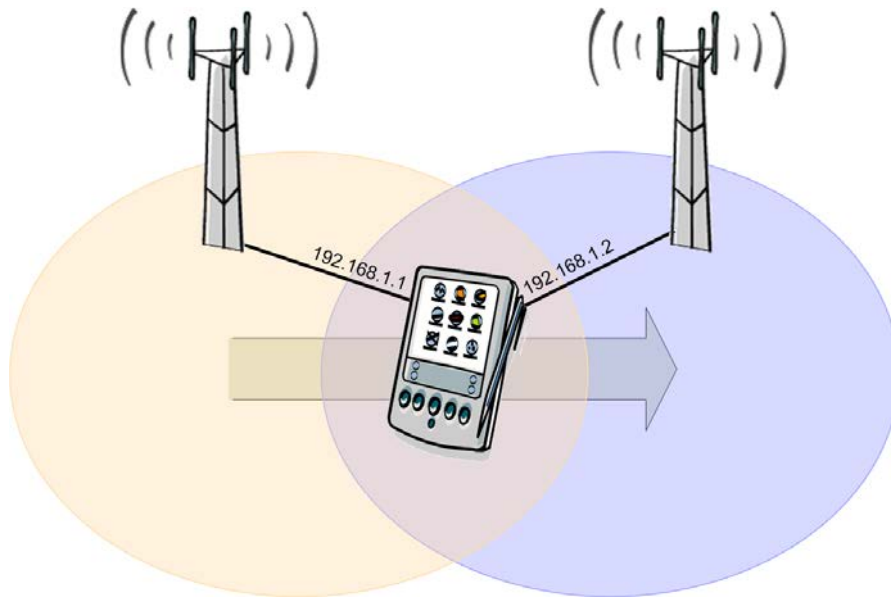


FIGURE 3.2 – *Soft handover.*

causée par le changement de réseau d'un terminal va terminer la (ou les) session correspondante de manière brutale, la laissant dans un état non-terminal. En effet, si la session n'est pas terminée « proprement » par un message de type BYE [12], le SCAS pourra déterminer qu'un incident de connectivité s'est produit. La reconnexion du terminal à l'IMS une fois le changement d'interface effectué, le SCAS possèdera suffisamment d'informations pour rétablir la communication en établissant une nouvelle session entre les interlocuteurs.

Cette solution peut être implémentée de deux manières différentes, soit via un contrôle de la session par une tierce partie (3PCC [62]) soit via le mécanisme SIP REFER (cf. [54]). Quel que soit la méthode choisie, il est nécessaire de respecter la logique de paiement de l'appel initialement établi. En effet, il doit y avoir une cohérence entre les services mis en œuvre (et donc facturés) dans l'appel initial interrompu et la communication nouvellement établie qui n'est que la continuité de ce premier. Par exemple, si *A* appelle *B* puis est interrompu, après reconnexion *A* doit être facturé (et non *B*) comme pour un unique appel et conformément à la durée réelle de connexion.

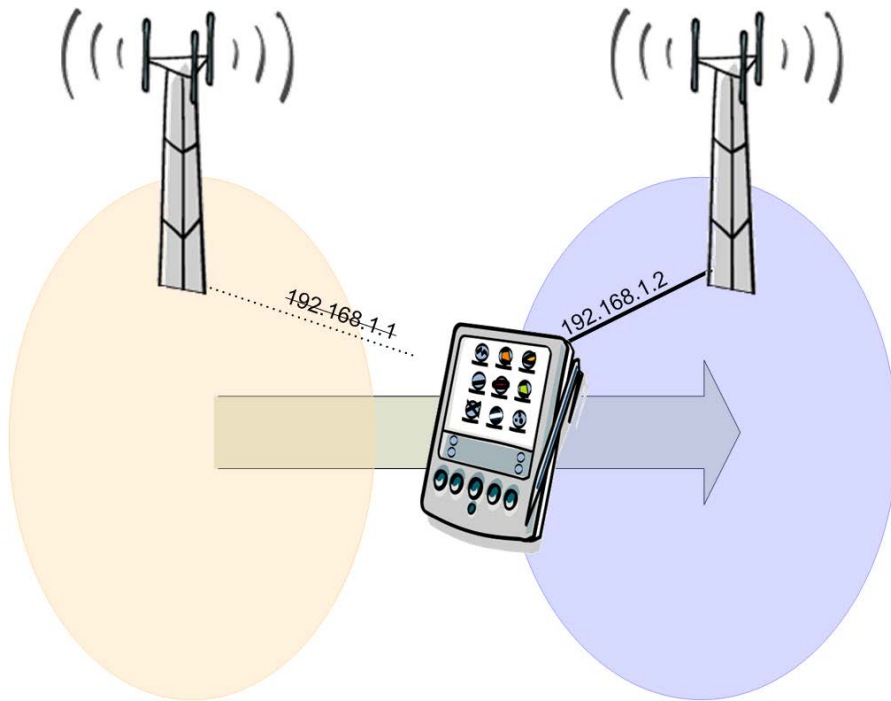


FIGURE 3.3 – *Hard handover.*

3.2.2 Terminal handover

Les mécanismes de terminal handover sont absents dans l'IMS bien que le protocole sous-jacent (SIP) comporte des fonctionnalités qui rendent cette mobilité possible. Nous proposons une solution conforme aux spécifications IMS qui repose sur ces fonctions. Tout d'abord intéressons-nous aux différentes problématiques qui interviennent avant et après le transfert proprement dit.

Déclenchement.

Contrairement au network handover, le changement de terminal est généralement déclenché manuellement. Bien que des critères réseau ou terminal puissent avoir un sens dans un terminal handover (batterie faible par exemple), changer automatiquement l'interface du service est particulièrement intrusif et potentiellement gênant pour l'utilisateur. Ainsi nous considérons le cas le plus vraisemblable, le déclenchement du transfert initié par l'utilisateur.

Une fois le transfert décidé, un grand nombre de choix se présentent à l'utilisateur : quel service transférer ? depuis quel terminal ? vers quel autre ? Un besoin fort d'identification apparaît alors, comment désigner (description et identification) un service, un terminal, un utilisateur. Ces éléments sont connus par l'IMS mais ces données destinées à un tout autre dessein ne sont pas formatées pour l'Homme... Comment choisir entre les sessions `784A5E@ims.fr?id=3263` et `D45E78@ims.fr?id=9652` ?

L'objectif est de définir une solution simple d'utilisation adaptée à un environnement télécom où les interactions entre l'utilisateur et un terminal généralement mobile sont relativement faibles. Nous proposons une approche mixte qui automatise et simplifie au maximum les mécanismes de déclenchement tout en laissant l'utilisateur maître du transfert. Les mécanismes d'identification et de désignation sont ainsi masqués. Une implémentation de la solution est détaillée en section 3.3, celle-ci permet des transferts de type « pull » et « push » (cf. 2.1.3). Dans tous les cas nous étudions le transfert entre deux terminaux du même utilisateur et non un transfert d'appel qui serait néanmoins similaire d'un point de vue technique.

3.3 Implémentation

Comme indiqué en introduction de ce chapitre, nous avons pu mener nos recherches sur deux plateformes IMS, l'IIP (INT IMS Platform) un projet open source développé au sein même du laboratoire RS2M de TELECOM SudParis et la solution Alcatel-Lucent 5400 IAS (IMS Application Server). Le *Session Continuity Application Server* que nous présentons ici a été développé et déployé dans les laboratoires d'Alcatel-Lucent, l'IAS étant plus proche de la norme. Ces travaux ont été publiés dans [80].

Le SCAS est donc un serveur d'applications qui communique avec le cœur IMS (la couche de contrôle), il a été développé en Java et est embarqué dans la solution 5400 IAS qui connecte les services fournis par les AS aux fonctions de contrôle réalisées par les CSCF (cf. 1.2.2). Le principe d'un AS est de réaliser des traitements spécifiques à partir des messages SIP qui transitent par le cœur IMS lors de l'établissement, le maintien et la terminaison de sessions multimédias (cf. 1.2.2). Le SCAS est chargé d'orchestrer les handovers afin d'assurer la continuité, pour cela

il doit déclencher et réaliser les transferts de sessions entre les terminaux IMS (cf. Figure 3.4).

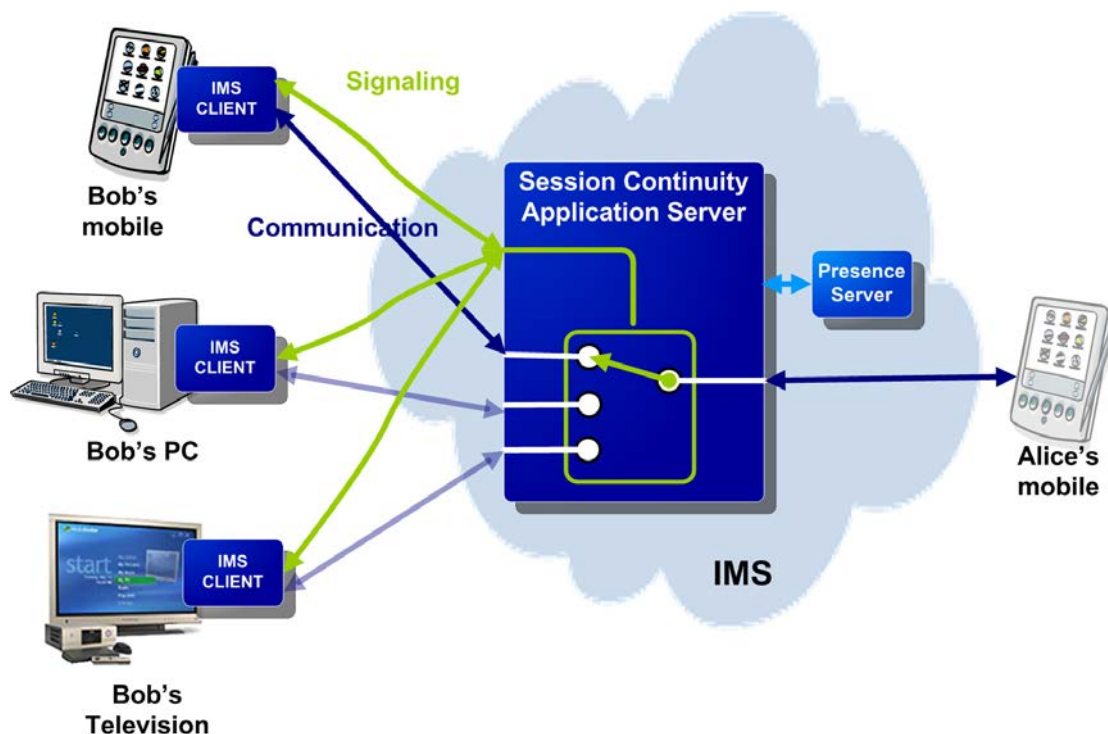


FIGURE 3.4 – Rôle du SCAS.

L'implémentation du SCAS a été réalisée uniquement dans le cadre d'un terminal handover. En effet, le mécanisme de transfert étant similaire dans l'IMS pour un network handover (*soft*) comme pour un terminal handover (cf. 3.2.1), et face à l'absence de terminaux mobiles IMS *dual-mode* supportant le *multihoming*, nous sommes focalisés sur le transfert inter-terminal.

3.3.1 Scénario

Nous avons développé deux variantes d'un même scénario qui correspondent à deux types de déclenchements, tous deux semi-automatiques requérant l'interaction de l'utilisateur pour réaliser le transfert de session en mode *push* ou *pull*.

Le scénario de base est simple : Alice est abonnée à un opérateur mobile doté d'un réseau IMS et elle a souscrit au service « *Session Continuity* » (délivré par le SCAS). Alors qu'elle est en communication avec Bob sur son ordinateur (téléphone

logiciel SIP-IMS), elle doit quitter son bureau pour aller en rendez-vous. Au lieu d'interrompre l'appel elle préfère transférer la communication sur son téléphone mobile (même abonnement IMS) pour poursuivre la discussion avec Bob pendant son déplacement. La question est comment Alice initie le transfert ?

Nous avons étudié deux modes de déclenchement : un mode *push* où Alice transfère explicitement sa session depuis le terminal origine et un mode *pull* où elle récupère la session à distance, depuis le terminal destination (cf. Figure 3.5). En mode *push*, plus manuel, c'est l'utilisateur qui déclenche le transfert via une interface spécifique sur le terminal d'origine, une confirmation peut éventuellement être requise au niveau du terminal destination. En mode *pull* le transfert est déclenché en fonction de l'état de présence des terminaux de l'utilisateur. Le SCAS récupère les informations de présence de chacun des terminaux des utilisateurs (ayant souscrit au service). En fonction de règles prédéterminées (nouveau terminal disponible, état de présence spécifique, etc) le SCAS déclenche le transfert, une confirmation peut éventuellement être requise ici aussi, le but étant de ne pas nuire à l'expérience utilisateur.

3.3.2 Architecture

Le scénario de transfert implique de nombreux acteurs :

- le terminal origine de Bob (un ordinateur doté d'un téléphone logiciel IMS),
- le terminal d'Alice (interlocutrice de Bob),
- le terminal destination de Bob (un téléphone mobile IMS),
- l'infrastructure IMS qui gère les sessions de communications,
- le serveur de présence qui indique l'état de chacun des terminaux,
- le SCAS chargé d'assurer la continuité.

Nous allons nous intéresser aux relations entre ces éléments et plus précisément aux blocs fonctionnels du SCAS durant les étapes successives d'un transfert entre terminaux.

Souscription au service de continuité.

Le SCAS fournit un service de continuité de service aux abonnés IMS qui ont explicitement souscrit à cette offre. Cette souscription est contractée à l'avance



FIGURE 3.5 – Scénario de transfert (mode *push*).

(hors connexion), entre l'opérateur télécom qui gère le domaine IMS et l'utilisateur. Au niveau de l'infrastructure, cette souscription se traduit par l'ajout de règles de déclenchement (*initial Filter Criteria* cf. 1.2.2) dans le HSS qui seront traitées par le S-CSCF, le composant IMS par lequel transitent l'ensemble des messages de signalisation. Chaque nouvel abonné est ainsi associé à un iFC.

Traçage des sessions.

Les iFCs ajoutés lors de la souscription indiquent au S-CSCF quels messages de signalisation rediriger vers le SCAS, permettant ainsi au serveur d'écouter et d'enregistrer les informations relatives à ses abonnés (terminaux, sessions de communications, etc). C'est ce que l'on appelle « traçage ».

Les messages d'enregistrement (et d'authentification) des clients IMS via la méthode REGISTER du protocole SIP sont les premiers à être interceptés. Ils permettent de connaître l'état de connexion à l'IMS d'un utilisateur, ainsi que l'adresse réseau, l'URI SIP et éventuellement le type de terminal. Les messages sont reçus et analysés par le *Session Logger* qui en extrait les informations utiles et les stocke dans une base de données locale au SCAS (*Logs* sur la Figure 3.6).



FIGURE 3.6 – SCAS durant l'enregistrement d'un client.

De plus, le processus d'enregistrement active les mécanismes SCAS relatifs à l'utilisateur en question, tel que la souscription aux informations de présence dans

le cas du mode *pull* (cf. 3.3.1) dont le déclenchement est basé sur ces informations d'état. Le composant *User Status Watcher* va alors souscrire aux informations de présence de l'utilisateur, ainsi à chaque modification opérée par le client le SCAS sera notifié, les informations seront alors stockées dans les *Logs* et pourront intervenir dans le processus de déclenchement du transfert.

Enfin, lorsque l'utilisateur créera ou sera invité à une session de communication, l'ensemble des échanges protocolaires seront analysés par le *Session Logger* qui stockera, comme pour l'enregistrement, les informations nécessaires en cas de transfert, cf. Figure 3.7. Il est à noter que le traçage des messages par le SCAS est réalisé de manière transparente pour les clients ; les échanges de requêtes SIP lors de l'enregistrement et de l'établissement des sessions de communications restent échangés vis à vis de la norme.

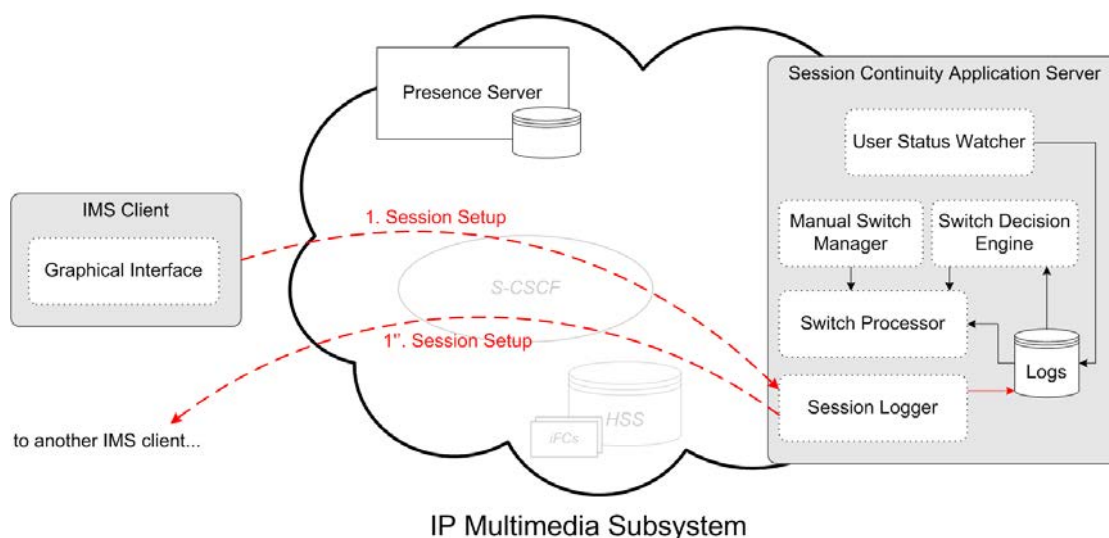


FIGURE 3.7 – SCAS durant l'établissement d'une session.

Déclenchement.

Nous avons implémenté deux types de déclenchements : un mode *push* où l'utilisateur requiert le transfert explicitement depuis le terminal d'origine et un mode *pull* où le transfert dépend de règles, l'utilisateur ne faisant que confirmer depuis le terminal destination (cf. 2.1.3). La Figure 3.8 présente l'interaction entre les différents composants dans ces deux cas. Il faudra alors considérer le client

IMS représenté comme le terminal origine pour le premier cas (mode PUSH) et comme terminal destination dans le second. Dans les deux cas, le transfert est initié depuis le terminal client, de manière plus ou moins directe et transparente pour l'utilisateur.

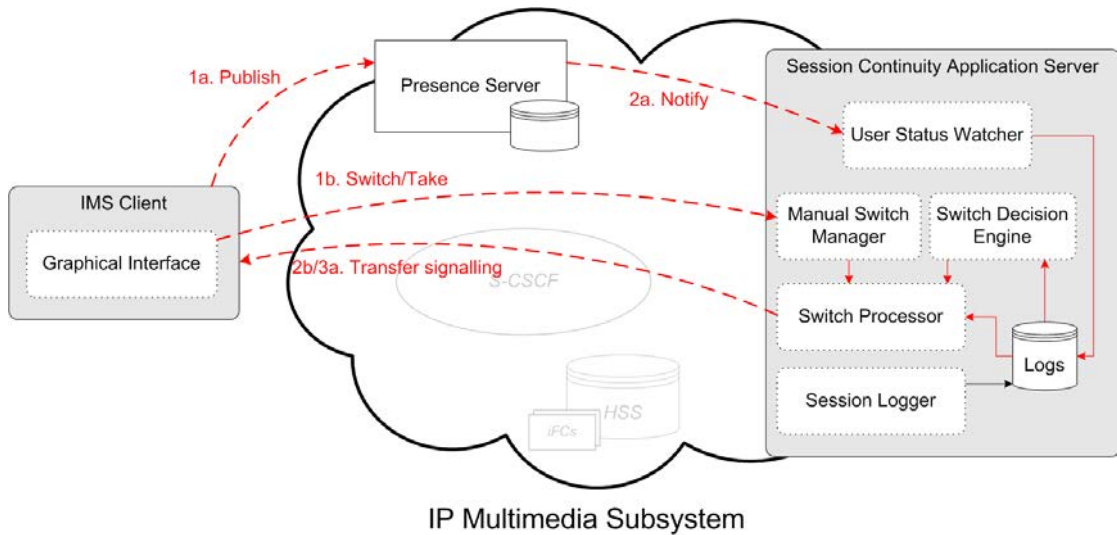


FIGURE 3.8 – SCAS durant le transfert d'une session.

En mode *push*, un ordre spécifique est envoyé au SCAS et traité par le *Manual Switch Manager* qui, en fonction de la commande *Switch* ou *Take*, va préparer et décider ou pas de déclencher le handover. La commande *Switch* marque (au niveau du SCAS) les sessions en cours sur le terminal comme « transférables », cela correspond donc à une volonté explicite du client de transférer le service de communication, cf. Figure 3.5. Ce mécanisme déclenche alors sur les autres terminaux de l'utilisateur une proposition de transfert, l'acceptation résulte de l'envoi de la commande *Take* qui identifie le terminal destination, le *Manual Switch Manager* va alors transmettre l'ordre de transfert effectif au *Switch Processor*.

Ce mode requiert la création d'un canal de communication spécifique, donc non standard, entre le client et le SCAS. Plusieurs approches sont possibles, nous avons choisi de détourner les méthodes SIP de type MESSAGE prévus pour l'envoi de messages textes courts (exception de contenu dans la signalisation SIP). Ainsi, en formatant de manière propriétaire ces messages textes, nous transmettons des informations de contrôle (ici *Switch/Take*) de manière transparente via le cœur IMS.

La force de l'approche *push* basée sur les commandes *Switch* et *Take* est sa double simplicité. Simplicité tout d'abord pour l'utilisateur qui n'a que deux interactions à réaliser en tout pour effectuer un transfert, point très important dans le cadre de services de télécommunications où le transfert mais également son déclenchement doit être simple et rapide. Cette approche est également simple d'un point de vue implémentation car celle-ci est allégée des considérations de désignation (sessions et terminaux) particulièrement complexes, les modalités du transfert sont directement déduites de l'usage de l'utilisateur : l'interface par laquelle il confirme le handover correspond au terminal destination.

Il faut cependant noter que ce mode de désignation et le mode *push* de manière générale car plus « manuel », requiert une interface spécifique, non standard, au niveau du client IMS. Nous avons travaillé avec des produits Alcatel-Lucent auxquels nous avons accès au code source, ainsi il nous a été possible de développer une interface minimaliste comportant les actions *Switch* et *Take*. Des captures d'écran correspondant au mode *push* sont présentées sur la Figure 3.5.

Le mode *pull* est légèrement plus simple mais aussi plus respectueux du standard. En effet, il n'existe pas ici de communication directe SCAS-client, bien que ce dernier soit l'initiateur du transfert. Il existe de nombreux critères de déclenchement automatique, nous avons choisi l'état de présence. Pendant une communication, tout nouveau terminal disponible (d'un point de vue présence) appartenant à l'utilisateur devient la destination du transfert. Seule une confirmation est requise au niveau de l'interface destination afin de s'assurer que le handover est désiré. Le *Switch* est ici induit par un changement d'état de présence et le *Take* correspond à la confirmation de la destination qui ne fait qu'accepter le transfert au niveau protocolaire (aucun message spécifique n'est envoyé).

Seul le message de confirmation de transfert pourrait paraître non-standard, cependant la spécification du mécanisme de transfert basé sur le message REFER ([54], cf. 3.3.3 pour plus de détails) suggère fortement que le handover soit acquitté au niveau terminal.

Enfin le *Switch Processor* va mettre œuvre les mécanismes de handover basés sur les informations précédemment collectées et stockées dans les *Logs*. Ce composant va communiquer avec les différents terminaux en SIP afin d'orchestrer le transfert, ces mécanismes sont décrits en détail dans la section 3.3.3.

Pour un mode de transfert manuel, néanmoins peu adapté aux services de communications : sélection individuelle de la session et des terminaux origine et destination, de nombreuses approches basées sur un canal de communication propriétaire client-SCAS sont envisageables. Il est toutefois intéressant de remarquer que le serveur de présence notifie les clients des états des autres terminaux et que le modèle de données, bien que standard, offre de nombreuses possibilités. Un mécanisme basé sur l'état de présence d'une session, d'un service ou encore un état spécifique indiquant la volonté ou la disponibilité pour un transfert pourrait être aisément implémenté. Nous verrons dans d'autres cas d'utilisation, notamment avec les services multimédias (cf. 5), ce genre d'approche.

3.3.3 Mécanisme de transfert

La mise en œuvre du transfert repose sur des mécanismes du protocole SIP, nous allons détailler ce processus dans cette section. Comme décrit précédemment, le SCAS trace et enregistre les messages de signalisation des utilisateurs ayant souscrit au service de continuité. Ainsi le *Switch Processor*, composant fonctionnel responsable de la négociation du transfert dispose de l'ensemble des données nécessaires.

Principe.

Au niveau SIP, un transfert de session peut être réalisé grâce à un champ spécifique nommé *Replaces* [56], une extension de l'IETF. Ce champ est utilisé dans les messages INVITE lors de l'initiation d'une nouvelle session. Lorsqu'un *Replaces* est présent dans un tel message, le récepteur de l'INVITE va alors créer la nouvelle session (s'il accepte) puis terminer la session identifiée par le contenu du champ *Replaces*. De cette manière on peut simplement substituer une session par une autre, donc un participant à une communication par un autre.

Cependant, utilisé de la sorte, seul l'un des participants à la communication peut générer un tel message car il lui faut connaître les identifiants de la session. De plus, la nouvelle session est nécessairement créée avec l'initiateur du message contenant un *Replaces*. Or nous souhaitons que le transfert soit orchestré par un tiers, le SCAS en l'occurrence. Si les informations privées de la session sont connues grâce au traçage des messages de signalisation, il réside le problème que la nouvelle

session ne doit pas être créée entre le SCAS et Bob ou Alice mais directement entre eux.

On utilise alors le message REFER [54], une autre extension du protocole SIP. Un message REFER permet d'indiquer à son destinataire une action à réaliser, typiquement initier une communication avec un autre client. De plus, des paramètres peuvent être ajoutés afin de rendre l'action réalisée par procuration plus précise encore. Dans notre cas, il suffit alors d'envoyer un message REFER indiquant au destinataire d'émettre une requête INVITE comportant un champ *Replaces* spécifique.

La Figure 3.9 illustre un exemple basique d'utilisation du message REFER permettant la substitution de la session Bob2-Alice par la session Bob1-Alice. On remarque sur ce diagramme de séquence des messages SIP que la session initiale n'est terminée qu'après l'établissement de la nouvelle, il y a donc une superposition des deux communications pendant un court laps de temps permettant un transfert plus fluide, moins perceptible par l'utilisateur. Cependant ce mécanisme suppose que le terminal d'Alice maintienne deux communications simultanément, imposant une contrainte forte à la fois logicielle (multiples sessions) et matérielle (gestion du microphone, occupation mémoire, etc).

Concrètement.

Plus concrètement, appliqué à notre SCAS, le mécanisme de transfert REFER présenté ci-dessus est géré par le *Switch Processor* grâce aux informations tracées et stockées dans les *Logs*. Les diagrammes de séquences 3.10 et 3.11 détaillent les messages échangés entre les différentes composantes du système en mode *push* (communication directe entre le client et le SCAS) et en mode *pull* via le serveur de présence.

3.4 Conclusion

Cette première approche de la continuité de service dans un cadre concret, l'IMS, a permis d'identifier des problématiques insoupçonnées jusqu'alors. En effet, bien que nous ayons restreint notre étude à un type de service particulier (télécoms), dont le protocole est clairement spécifié et qui de plus possède des mécanismes de

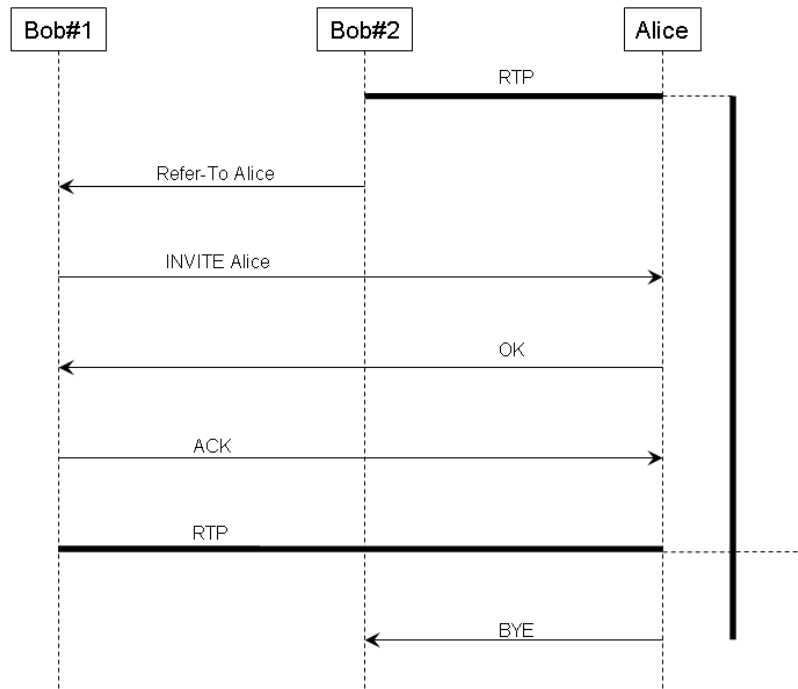


FIGURE 3.9 – Principe de transfert via le message REFER.

continuité standards, la mise en œuvre dans la vie réelle de la mobilité n'est pas encore satisfaisante d'un point de vue utilisateur.

Si l'on considère uniquement les qualités des procédés de transfert proposés, le bilan est positif conformément à l'état de l'art, cf. 2.3.2. La continuité d'une session est réalisée de manière efficace, rapide et synchronisée. L'implémentation d'un prototype a cependant mis en avant que le mécanisme de transfert seul était insuffisant, l'ensemble du processus de continuité qui commence et termine par l'utilisateur requiert de nombreuses fonctions permettant d'identifier les sessions, les terminaux, les utilisateurs, les désigner et enfin orchestrer la mobilité. Le déclenchement et l'orchestration ne consistent pas uniquement à exécuter les algorithmes de transfert mais également à contrôler les applications et la manière dont elles fournissent le service avant et après son transfert. Quelques exemples significatifs : il n'existe aujourd'hui aucune interface prévue pour déclencher un transfert depuis

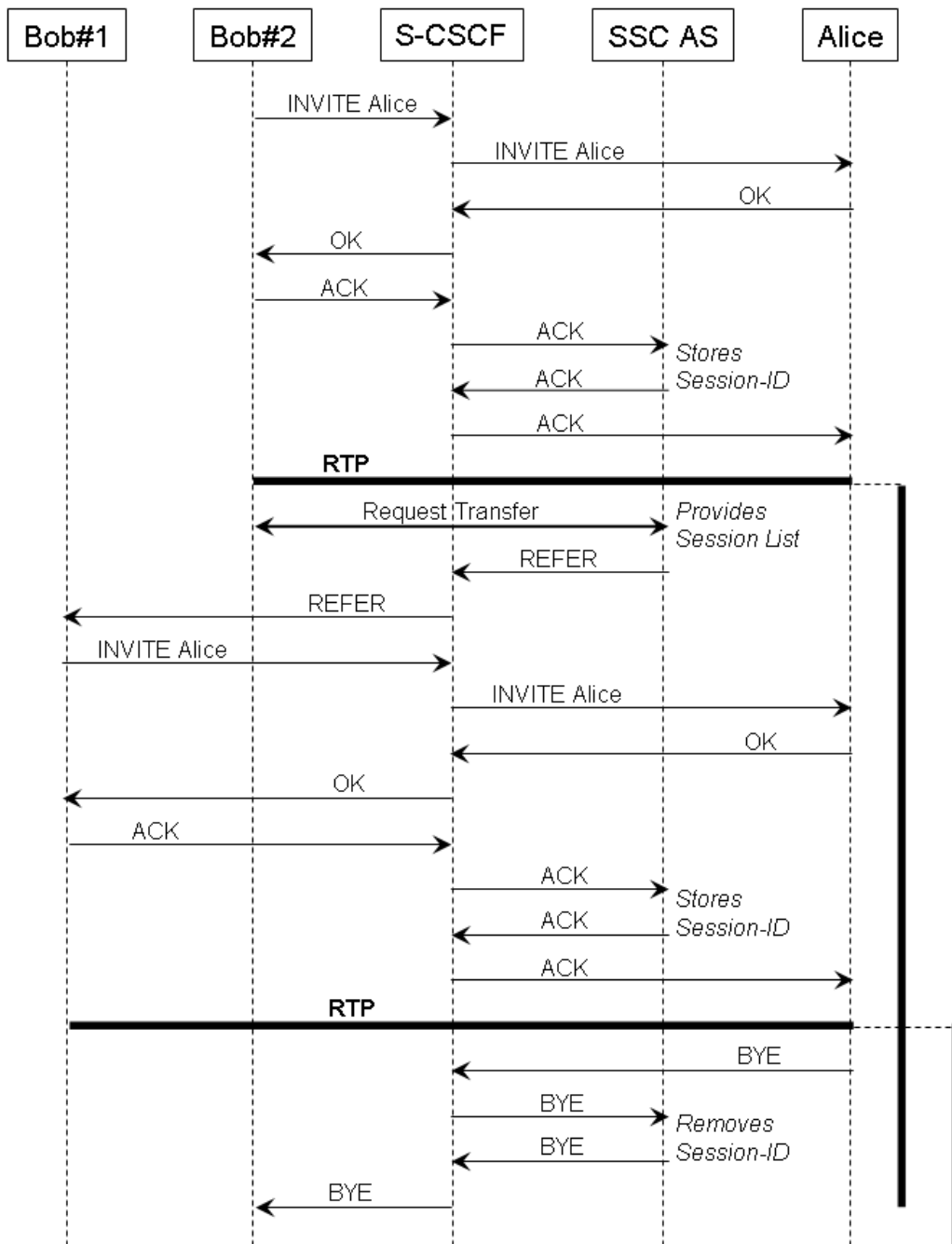


FIGURE 3.10 – Échanges SIP d'un transfert en mode *push*.

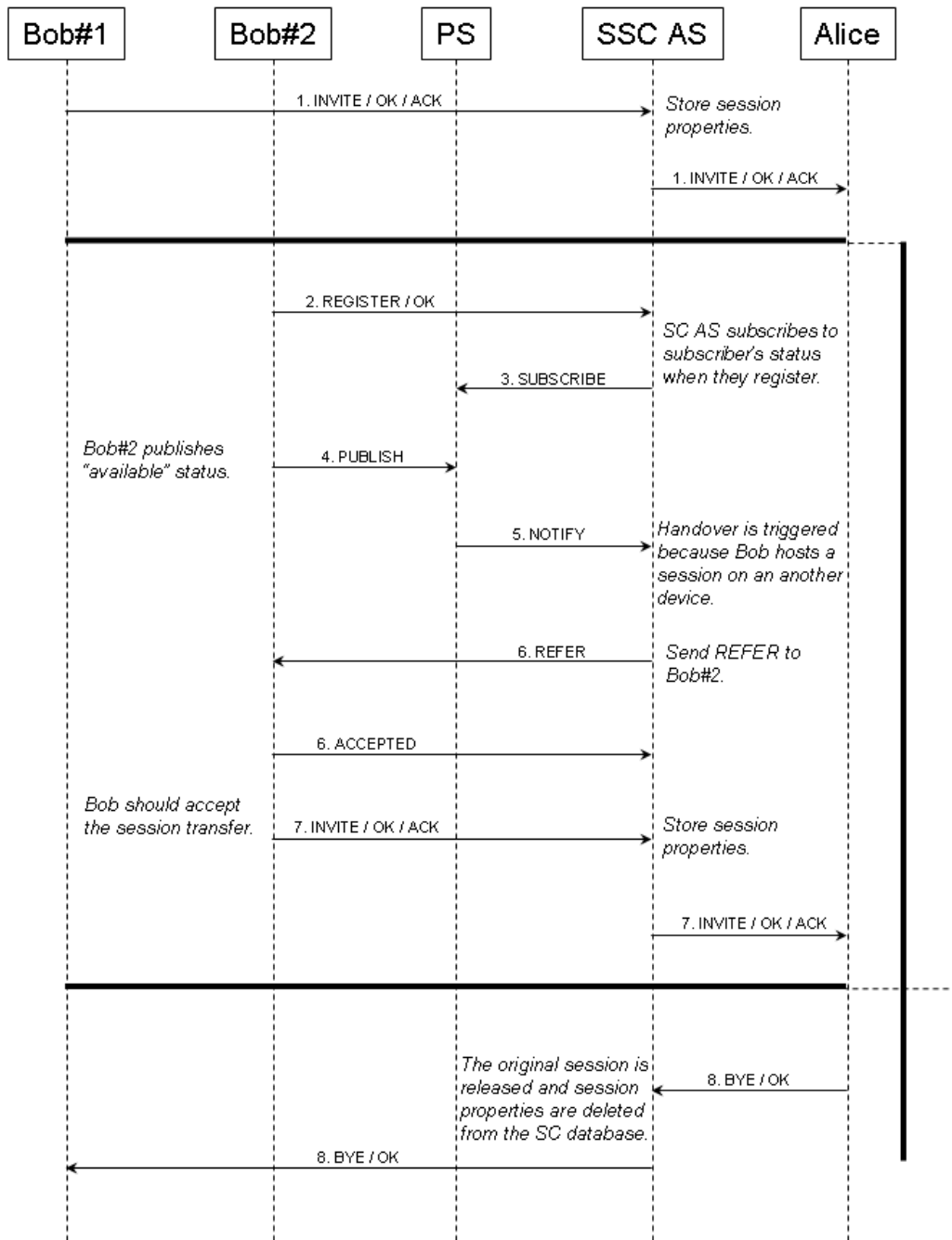


FIGURE 3.11 – Échanges SIP d'un transfert en mode *pull*.

un client IMS (nous avons dû l'implémenter), aussi un terminal ne peut accepter un transfert que si le client IMS est lancé et à l'écoute (aucun contrôle applicatif), enfin un utilisateur ne possède aucun moyen de désigner explicitement un de ses terminaux comme destinataire du transfert.

Il est relativement facile d'introduire de la mobilité dans un tel environnement homogène, l'IMS est une architecture standardisée du terminal à l'infrastructure, en passant par les services de base (*enablers*) et les protocoles. Il y a peu de surprises entre le terminal d'origine et celui de destination : capacités équivalentes, application et protocole identiques, l'adaptation du service est extrêmement faible et limitée à un rare changement d'encodage des flux de données.

Dans ce cadre rigide, la continuité temporelle est parfaite, cependant la continuité contextuelle manque cruellement à l'expérience utilisateur. Le service de télécommunication n'est pas qu'un simple flux de données, de multiples éléments sont perçus par l'utilisateur : configuration (réglage de la sonnerie, renvoi d'appel), liste de contacts, historique d'appels, avatar, etc. Si les approches basées sur une architecture de services forte telle que l'IMS permettent la continuité de certains de ces éléments et les déplaçant du terminal vers l'infrastructure (nécessitant autant de nouveaux standards, cf. présence), les caractéristiques des applications permettant d'offrir des services plus ou moins riches pour une expérience utilisateur optimale sont simplement ignorées.