

Qu'est ce qu'un service

Un service, bien que réel, est un objet difficile à définir et à représenter dans la mesure où le même terme peut désigner différents concepts selon le domaine considéré. Par définition et hors de tout contexte on peut cependant admettre que d'une manière ou d'une autre, un service apporte une fonctionnalité répondant à un besoin d'une personne réelle, même si celle-ci n'en a pas directement conscience. Ainsi, un service « offrira » généralement une fonction pratique à son bénéficiaire, celui-ci ne pouvant obtenir le même résultat par ses propres moyens, sinon de manière aussi simple et efficace (de son point de vue).

« Un service vaut ce qu'il coûte. » Victor Hugo.

La valeur ajoutée intrinsèque des services révèle un potentiel économique exceptionnel : ils répondent à autant de besoins qu'ils n'en créent de nouveaux et permettent de commercialiser des produits non matériels. L'Institut national de la statistique et des études économiques (INSEE) définit le service de la manière suivante [3] :

« Une activité de service se caractérise essentiellement par la mise à disposition d'une capacité technique ou intellectuelle. ».

Bien que relatif au secteur économique tertiaire dont il est la base, le service, de par sa qualité de production immatérielle, convient parfaitement au domaine des Technologies de l'Information et de la Communication (TIC). Ainsi, bien que de nombreux exemples de services puissent trouver résonance dans la vie de tous les jours, nous nous focaliserons sur les formes propres aux TIC, domaine exclusif de l'ensemble de ces travaux de thèse.

De nombreux exemples de services adoptant différentes formes coexistent dans les Technologies de l'Information et de la Communication, afin d'en dégager des propriétés communes (cf. 1.3) nous nous sommes intéressés à ces cas d'utilisation selon deux axes : au niveau applicatif (cf. 1.1) et dans les télécommunications (cf. 1.2).

1.1 Les services applicatifs

Par abus de langage, nous qualifierons d'« applicatif » l'ensemble des services informatiques, qu'ils soient délivrés localement par un système d'exploitation ou à distance via un réseau d'accès. Nous détaillons dans cette section ces deux cas qui abordent différemment la notion de service et précisons les concepts introduits par chacun d'eux.

1.1.1 Services locaux

Lorsque l'on s'intéresse à l'ensemble des systèmes informatiques et plus particulièrement à la raison d'être de ce que l'on appelle plus communément « ordinateurs », on constate rapidement que leur objet premier est de nous rendre des services. L'informatique étant par définition le traitement automatisé de l'information, les systèmes informatiques fournissent à leurs utilisateurs des fonctionnalités pratiques ou agréables, « *techniques et intellectuelles* » (par exemple un éditeur de texte, une base de données, un lecteur multimédia, etc).

Les caractéristiques supérieures « innées » d'un système informatique sur l'Homme en termes de vitesse de calcul et de capacité mémoire en font le serviteur par excellence pour rendre des services et alléger l'utilisateur de tâches complexes, fastidieuses ou simplement irréalisables. Ainsi, en ce sens, toute contribution d'un système informatique à l'Homme pourrait dans l'absolu être qualifiée de service.

Cependant, le terme « service » est peu employé en tant que tel dans un environnement applicatif local, on parle plus volontiers d'application, de logiciel, de programme, de tâche, de processus, . . . Je ne prétends nullement que cette riche terminologie soit incongrue ou que ces mots soient synonymes, je tiens juste à souligner que l'utilisateur face à son ordinateur pense peu en terme de « service »,

confronté physiquement à sa réalisation il le nomme plus naturellement par leur représentation concrète.

Cette particularité est d'autant plus intéressante qu'elle s'applique exclusivement aux systèmes locaux. Nous le verrons par la suite, l'utilisateur qui jouit d'une fonctionnalité sans avoir conscience de son implémentation comme c'est le cas pour les services distants ou de télécommunications est plus enclin à la nommer « service ».

Le terme exact « service » est toutefois employé dans le monde applicatif pour désigner un type précis de fonctionnalité. Introduit par le système d'exploitation Windows de Microsoft, un « *service Windows* » [4] est un programme qui possède les caractéristiques suivantes [5] :

- un cycle de vie calqué sur celui du système hôte (et donc indépendant de l'utilisateur),
- pas ou peu d'interactions avec l'utilisateur (le service est souvent rendu directement au système et contrôlé par celui-ci),
- un mode d'exécution privilégié (généralement plus de droits que l'utilisateur lambda),
- une exécution « discrète » (l'utilisateur n'a souvent pas conscience du service rendu bien qu'il en jouisse directement ou indirectement).

Cette vision de « service » est unique à MS Windows, les autres principaux systèmes d'exploitation tels que UNIX (et compatibles, *i.e.* Linux et BSD) ou encore MAC OS X d'Apple préfèrent l'utilisation du terme « démon » (*daemon* [6]) pour désigner ce type de programme. Les fonctionnalités offertes par les services Windows sont diverses tel que la gestion des réseaux, l'indexation des données ou encore la synchronisation de l'heure, cf. [7] pour plus d'information sur les composants de base.

On peut également noter que Apple Inc. introduit dans MAC OS X des « *services systèmes* ». Ces services ne constituent pas un nouveau programme mais plutôt l'interface d'une application existante par laquelle celle-ci expose les fonctions basiques qu'elle peut réaliser pour un tiers (typiquement une autre application). Ainsi les services « rendus » par les applications sont échangés via une architecture spé-

cifique qui permet le transfert de données entre programmes tel que présenté dans la Figure 1.1.

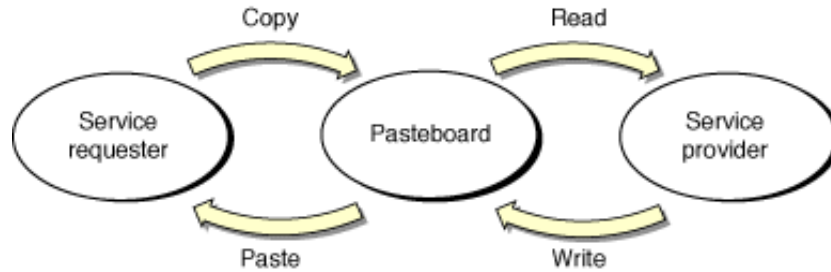


FIGURE 1.1 – Architecture de *services systèmes* dans MAC OS X.

Les services disponibles correspondant aux types de données de l'application en cours sont automatiquement listés dans un menu dédié. La Figure 1.2 présente l'exemple d'un tel menu (appelé ici « *Services* ») qui permet à un éditeur de texte (*TextEdit*) d'accéder au service fourni par un navigateur Web. L'information échangée est alors une adresse Internet (www.rabbits.com) résultant vraisemblablement au chargement du site Internet correspondant dans le navigateur Web offrant le service.

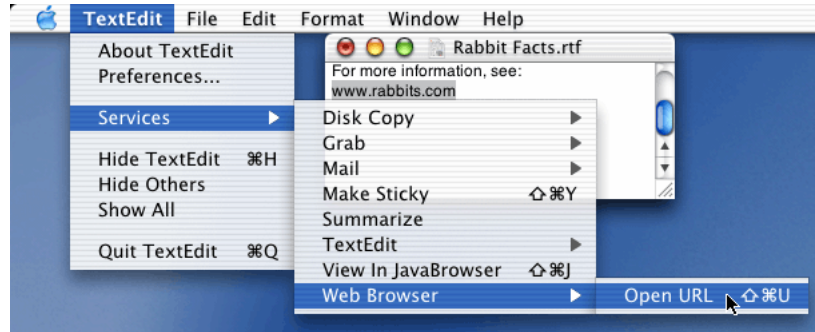


FIGURE 1.2 – Exemple de *services* MAC OS X.

Les *services systèmes* de MAC OS X et les *services Windows* présentés précédemment sont les deux seuls exemples d'introduction de la notion de service dans le domaine applicatif. De plus ces approches sont relativement différentes entre elles et ne considèrent l'utilisateur qu'indirectement : la première étant dédiée au système d'exploitation tandis que la seconde aux applications. Cependant, et nous

y reviendrons plus loin dans cette section, la majorité des fonctionnalités fournies par un environnement informatique, typiquement via des applications gérées par un système d'exploitation, sont ni plus ni moins des services pour l'utilisateur.

1.1.2 Services distants

Les services distants sont caractérisés par une architecture asymétrique « client-serveur » dissociant physiquement la consommation et l'implémentation de la fonctionnalité offerte. Le *serveur* comme son nom l'indique, implémente les mécanismes nécessaires à la fourniture du service au *client*, lequel à son tour le présente de manière « consommable » à l'utilisateur. Le service résulte ainsi de l'exécution d'au moins deux processus distincts, l'un du côté utilisateur et l'autre généralement plus complexe chez le fournisseur qui réalise les principaux traitements. Les exemples de tels services sont multiples, les plus communs étant peut-être sur le Web, rendus possible par la collaboration entre le navigateur de l'utilisateur et les serveurs Web des fournisseurs de sites sur le réseau Internet (cf. Figure 1.3).

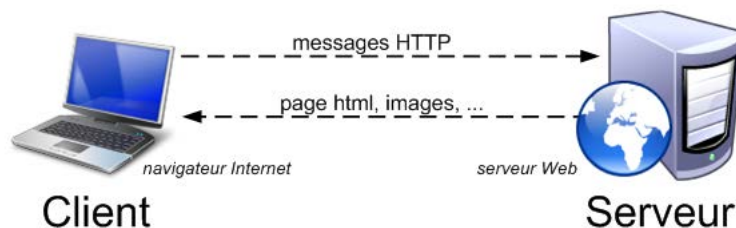


FIGURE 1.3 – Modèle client/serveur, exemple d'Internet.

À la différence d'un environnement applicatif local (cf. 1.1.1), la fourniture d'un service distant requiert la collaboration de plusieurs applications, et ce via un protocole de communication prédéfini. Cette communication permet le déclenchement et le contrôle du service, menant généralement à un transfert de données. Ces données brutes sont alors traitées et interprétées par le client puis présentées à l'utilisateur de manière à rendre la fonctionnalité désirée.

Notion de session.

Le flux d'informations échangé entre clients et serveurs est appelé *session de communication* [8] (cf. Figure 1.4). Dans un environnement distant, les services n'existent qu'au travers de ces sessions qui contrôlent et transportent les informations essentielles à la fonctionnalité offerte. De manière générale, on peut distinguer deux types de sessions présentes simultanément entre le client et le serveur lors de la fourniture d'un service.

- Les *sessions de contrôle* qui permettent aux entités de gérer le service, par exemple le protocole HTTP (*Hypertext Transport Protocol* [9]) utilisé par le navigateur pour demander le contenu d'une page Web au serveur correspondant.
- Les *sessions de données* qui transportent les informations propres au service tel que le contenu d'une page Web : code HTML, images et divers objets, transférés du serveur vers le navigateur.

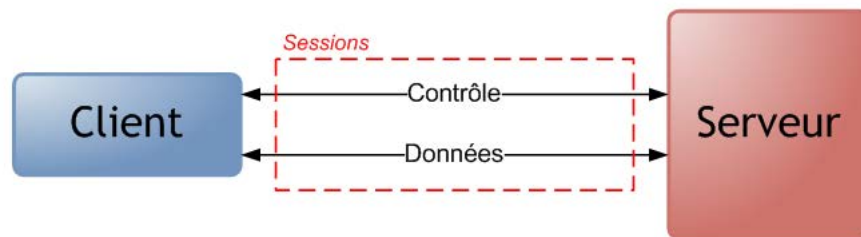


FIGURE 1.4 – Sessions d'une architecture client-serveur.

Une relation classique entre un client et un serveur implique nécessairement ces deux sessions simultanément, le format de chacune étant formellement défini afin d'assurer l'interopérabilité, soit de manière ouverte via un standard (par exemple les spécifications du W3C et de l'IETF pour HTTP [10]), soit de manière fermée via un protocole propriétaire (par exemple Adobe Flash RTMP [11]). À noter que physiquement, une session de communication peut multiplexer les informations de contrôle et les données comme c'est le cas avec HTTP ou les gérer séparément comme pour le protocole SIP (*Session Initiation Protocol* [12]). Le contrôle du service de communication étant assuré par SIP lui-même et les données multimédias acheminées via le protocole RTP (*Real-time Transport Protocol* [13]).

Modèle pair-à-pair.

D'un point de vue « services », les architectures pair-à-pair (P2P [14]) sont un cas particulier de modèle client/serveur dans la mesure où chaque entité, ou « pair » (*peer*) est à la fois client et serveur (cf. Figure 1.5). Les services sont ainsi distribués et rendus entre pairs, sans serveur centralisé. Cependant, les caractéristiques d'un service fourni via un tel modèle distribué restent les mêmes, la fonctionnalité étant créée à partir d'une (ou plusieurs) session de données gérée par une session de contrôle.

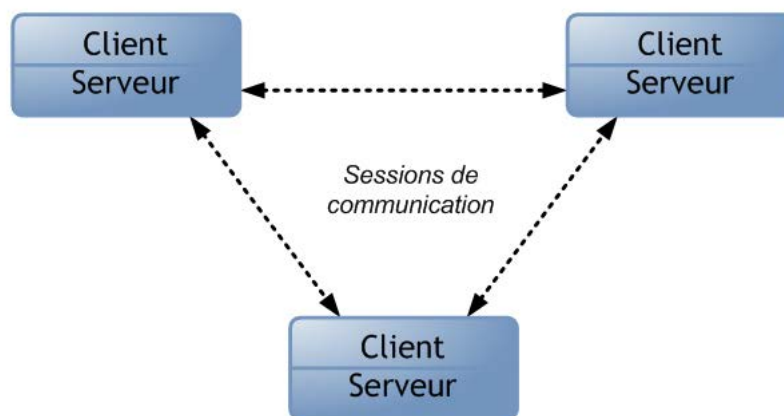


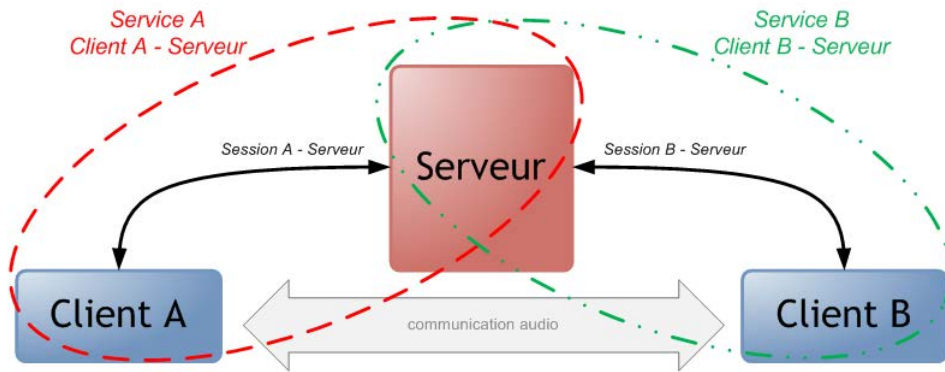
FIGURE 1.5 – Modèle pair-à-pair.

Que l'on considère une architecture client/serveur classique ou un modèle P2P, la fourniture d'un service est matérialisée par un transfert de données entre deux entités, l'une jouant le rôle de client et l'autre de serveur. Chaque session de communication (contrôle et données) correspondant à une fonctionnalité, il existe donc nécessairement une instance de service par relation client/serveur.

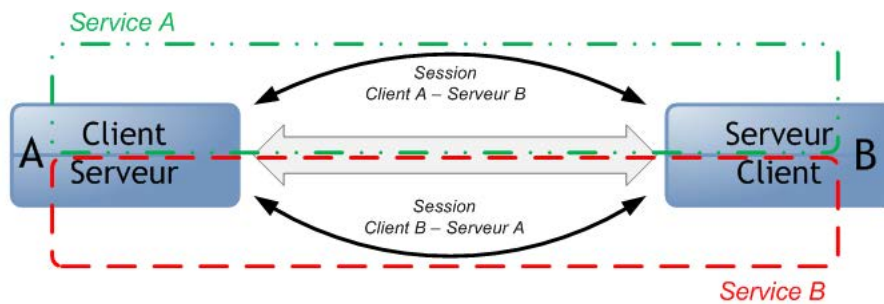
Instances de services.

Ainsi chaque utilisateur possède une instance de service qui lui est propre, et ce même si la fonctionnalité finale ne peut être rendue sans l'existence d'autres clients (typiquement un service de communication); ces derniers posséderont alors chacun leur instance de service, éventuellement fournie par le même serveur. Les Figures 1.6 illustrent les instances de services présentes lors d'une communication

audio entre deux clients dans une architecture client/serveur 1.6(a) puis selon un modèle pair-à-pair 1.6(b).



(a) Dans une architecture client/serveur.



(b) Dans un modèle pair-à-pair.

FIGURE 1.6 – Instances de services, exemple d'une communication audio.

La Figure 1.6 met en valeur la relation entre l'utilisateur et son service. Quel que soit le nombre d'entités impliquées dans le rendu d'une fonctionnalité, une instance de service est créée pour chaque utilisateur. Il appartient alors au serveur qui implémente la fonction offerte de combiner les différentes sessions afin de réaliser le traitement adéquat.

1.1.3 Services Web

Nous avons vu que le terme « service » était peu employé dans les environnements locaux (cf. 1.1.1), préférant par exemple la notion d'application. De plus,

lorsqu'une fonctionnalité est dissociée de son implémentation, l'utilisateur l'appelle plus naturellement « service » car il en perçoit directement les effets sans voir sa réalisation. C'est le cas des services distants (cf. 1.1.2) délivrés par un bien nommé « serveur » à un client, tous deux connectés via un (ou des) réseau. Cette section est consacrée aux *services Web*, un exemple de service distant particulièrement intéressant.

Interconnectant à *l'infini* les systèmes informatiques entre eux, le réseau Internet constitue la première infrastructure de services du point de vue de l'offre. Entièrement ouverte, elle favorise la création et le déploiement de tout type de services par n'importe quel fournisseur ; la richesse de l'offre ainsi proposée n'ayant d'égal que l'anarchie qui y règne. En effet, chaque fournisseur offrant une solution propriétaire sans considération de l'existant, l'architecture de services naturellement fondée sur le Web perd en efficacité et ne peut répondre à certains (nouveaux) besoins tels que (liste non exhaustive) :

- la découverte,
- la réutilisation,
- la composition,
- et l'interopérabilité des services.

Le but de ces fonctions est de traiter dynamiquement des services afin de créer une offre personnalisée plus riche, adaptée à l'utilisateur. Mais une opération automatisée sur un objet, fut-ce un service, implique un format standard et non une forme propriétaire, imprédictible, dépendante du fournisseur. C'est pour répondre à cette problématique que les *services Web* furent introduits par le World Wide Web Consortium (w3C [15]), constituant une première approche de formalisation en standardisant l'architecture de services Web.

La Figure 1.7, extraite de la spécification de l'*Architecture des Services Web* (WSA [16]) du w3C, illustre les interactions entre les différents acteurs dans la mise en œuvre d'un service. On y retrouve naturellement les deux entités essentielles : le client qui sollicite la fonctionnalité (ici *Requester Entity*) et le serveur qui la fournit (ici *Provider Entity*). Le schéma met également en avant la présence de l'Homme qui propose (fournisseur) et dispose (consommateur) du service via un système informatique, ici implémenté par des agents.

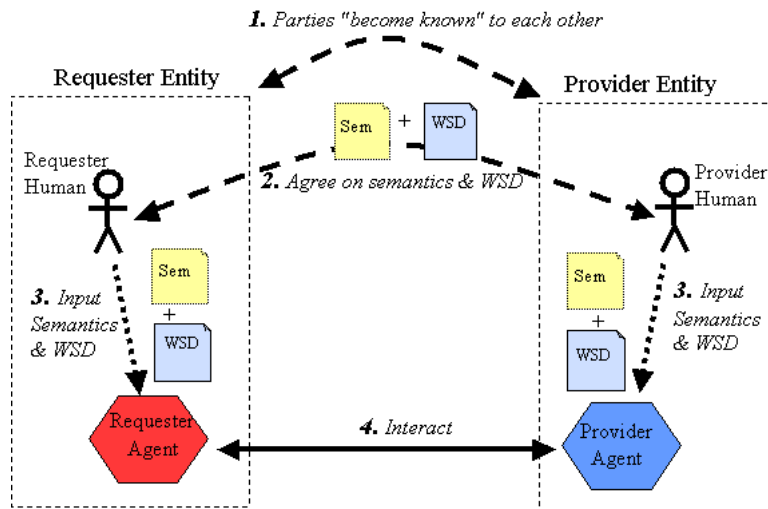


FIGURE 1.7 – Mise en œuvre d'un *service Web*.

Toujours selon la Figure 1.7, la mise en œuvre d'un service Web est décrite selon quatre étapes.

1. Phase de découverte des deux entités. C'est généralement l'utilisateur qui, selon ses besoins, recherche, sélectionne puis contacte directement un serveur. Ce dernier ayant publié préalablement les informations correspondantes à la fonctionnalité fournie, permettant aux clients potentiels de le solliciter (par exemple via un annuaire de services, un site Internet, une application, etc).
2. Le client et le serveur conviennent ensuite de la fonctionnalité à délivrer en fonction des possibilités du fournisseur et des attentes de l'utilisateur, signant une sorte de contrat numérique. Le comportement de chacune des entités y est formellement décrit à l'aide d'un langage spécifique : WSDL (*Web Services Description Language* [17]).
3. La description formelle du service Web (WSD, *Web Service Description*) est alors transmise aux agents qui implémentent les mécanismes requis pour chaque entité.
4. La fourniture proprement dite du service s'effectue alors entre le client et le serveur, matérialisée par des échanges de données entre les deux agents.

Ainsi, les services exposent des interfaces standardisées leur permettant d'être réutilisés par d'autres. On peut alors manipuler, réutiliser, composer ces *briques fonctionnelles* pour construire des services dits « riches », combinant plusieurs fonctionnalités dans une offre unique. La Figure 1.8 est un exemple de composition de services Web, elle est extraite d'un document du W3C proposant divers scénarios d'utilisation et démontrant les possibilités offertes par une telle architecture (cf. [18]).

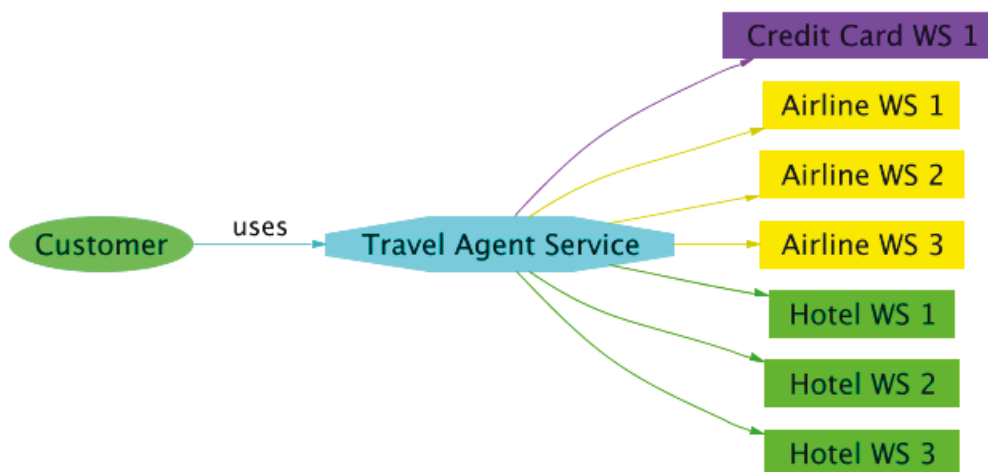


FIGURE 1.8 – Exemple de composition de services Web.

Le scénario présenté dans la Figure 1.8 est relativement basique, illustrant une utilisation assez courante de service Web. Un utilisateur souhaite réserver en ligne un séjour auprès d'une agence de voyage. Pour se conformer à la terminologie employée jusqu' alors, le *client* est donc le navigateur Internet de l'utilisateur, le *serveur* est l'application de réservation en ligne de l'agence de voyage et le *service* offert est complexe : recherche et comparaison d'hôtels et de vols à partir de critères fournis par l'utilisateur puis paiement en ligne.

La figure fait apparaître trois types d'acteurs : le client qui demande le service, le serveur qui répond à la requête et un ensemble de fournisseurs tiers qui assurent chacun une fraction de la fonctionnalité finale offerte. Le serveur sert l'utilisateur tout en jouant le rôle de client vis-à-vis de fournisseurs tiers mettant à sa disposition des briques fonctionnelles qui, une fois assemblées, constituent un service riche. Ici l'application de réservation ne possède aucune information ni contrôle sur les hôtels, les vols ou encore les mécanismes de paiement ; l'agence de voyage doit

passer par des fournisseurs tiers dont la combinaison des fonctionnalités offertes lui permet de satisfaire à la requête du client.

La spécification de ces interfaces rend possible de nombreux nouveaux usages, offrant une véritable architecture de services basée sur le Web, contrairement aux solutions locales qui restent généralement isolées, propriétaires et incompatibles (à l'exception peut-être des services systèmes de *mac os x*, cf. 1.1.1). Cette approche apporte de nombreux avantages : l'enrichissement dynamique de services par la découverte de fonctionnalités tierces, le déploiement accéléré de services complexes par la réutilisation de briques fonctionnelles existantes ou encore la personnalisation de l'offre.

Il faut cependant noter que la force de la composition basée sur la collaboration dynamique de fournisseurs de services tiers, entraîne des problèmes potentiels relatifs à l'interaction des fonctionnalités dont l'inter-fonctionnement peut générer des conflits. Cette problématique majeure appelée « interaction de services » fait l'objet de nombreux travaux de recherche, essentiellement dans le domaine des télécommunications comme nous verrons dans la section suivante (cf. 1.2.2).

1.2 Les services de télécommunications

À ses débuts dans les années 1870, le réseau téléphonique ne proposait qu'un seul et unique service : la télécommunication voix. Cependant, fondée plus d'un siècle avant son homologue informatique, l'infrastructure de télécommunications a su évoluer, proposant progressivement de nouvelles fonctionnalités : renvoi d'appel, présentation du numéro, renseignements, horloge parlante, messagerie vocale... Bien qu'exclusivement audio initialement, les services se sont progressivement enrichis avec du texte, des images, des données, permettant aux opérateurs de proposer une véritable offre multimédia : visiophonie, messagerie multimédia (MMS), vidéo à la demande (VoD), télévision, etc.

Mais l'évolution la plus remarquable du point de vue des services fut sans conteste l'interconnexion avec un autre monde : Internet (cf. 1.2.3). Avec 5 milliards de clients (abonnements et forfaits prépayés), le réseau téléphonique (fixe et mobile) est la première plateforme de services *en terme du nombre de clients*,

loin devant Internet et ses 1.6 milliards d'utilisateurs (ITU 2008 [19]). Cependant, bien que plus récente, la richesse de l'offre de services sur Internet est aujourd'hui nettement supérieure à celle proposée par l'infrastructure historique de télécommunications. Deux principaux facteurs expliquent ce constat inattendu.

Des terminaux limités.

Les appareils téléphoniques sont nettement moins performants que les ordinateurs, et ce en tous points : vitesse de calcul, capacité mémoire, interface utilisateur, connectivité, etc. C'est la conséquence des approches structurellement opposées du monde télécom (des télécommunications) et d'Internet.

Deux philosophies s'opposent. D'un côté des terminaux basiques asservis à un réseau téléphonique complexe et puissant qui assure toutes les fonctionnalités de manière centralisée (*l'intelligence au cœur*), de l'autre Internet, une simple tuyauterie qui achemine des données d'un ordinateur à un autre, ces derniers produisant et consommant les services (*l'intelligence à la périphérie* [20, 21]).

Ainsi les terminaux téléphoniques sont limités car conçus pour consommer uniquement les fonctions que le réseau dont ils dépendent est en mesure de leur offrir. Cette contrainte est double car elle limite également le réseau qui ne peut offrir de nouvelles fonctionnalités sans renouveler les appareils clients, d'où une offre de services restreinte. À l'inverse, les terminaux informatiques ne sont ni dépendants du réseau Internet (en terme de fonctionnalités), ni dédiés à un ensemble prédéfini de services, ils sont pensés pour des usages multiples et indéterminés a priori. La création de nouveaux services est alors débridée, sans considération des caractéristiques des clients, non sans quelques problèmes d'incompatibilité.

Une infrastructure fermée.

Autre différence structurelle entre les deux réseaux, le modèle Internet de part sa logique d'intelligence déportée à la périphérie du réseau, mélange producteurs et consommateurs de services. En effet, le réseau n'offrant aucun service en tant que tel, ce sont les éléments terminaux qui échangent directement, collaborent, se comportant comme producteur ou consommateur de fonctionnalités selon leurs besoins.

À l’opposé, le réseau téléphonique est géré par des opérateurs qui délivrent directement les fonctionnalités à leurs clients, l’offre est ainsi cloisonnée, fermée à des fournisseurs de services tiers (du moins sans un accord commercial avec l’opérateur). Ainsi, le modèle téléphonique cantonne l’utilisateur à son rôle de consommateur, ne pouvant devenir acteur comme avec Internet. Cette nouvelle contrainte limite intrinsèquement la génération de nouvelles fonctionnalités, le création de services étant réservée aux fournisseurs autorisés.

Si l’offre en terme de services peut paraître faible comparée au Web, l’infrastructure de télécommunications possède elle de nombreuses qualités : connectivité entre domaines et technologies d’accès, identification des clients (et authentification), mécanismes de paiement, gestion de la qualité de service, etc. L’infrastructure est l’atout majeur du monde télécom, sa robustesse (ou sa « lourdeur » selon ses détracteurs) est l’œuvre des efforts de standardisation qui assurent son interopérabilité, sa force.

1.2.1 Standardisation

L’Union Internationale des Télécommunications (ITU, créée sous le nom d’Union Internationale du Télégraphe en 1865) est le premier et principal effort de standardisation des télécommunications (infrastructure et services) jusque dans les années 1980. Alors, le développement des ordinateurs personnels a donné naissance à de nouveaux modes de communication qu’il a fallu rapidement normaliser.

L’ITU étant peu réactif, de nouveaux groupes de normalisation tels que l’Internet Engineering Task Force (IETF) ou le World Wide Web Consortium (W3C) se sont constitués afin de fournir une réponse rapide et éviter l’adoption *de facto* de solutions propriétaires. Similairement, l’évolution et l’essor des radiotélécommunications a engendré l’émergence de l’ETSI (*European Telecommunications Standards Institute*) et du 3GPP (*3rd Generation Partnership Project*). Enfin, certains groupes tels que l’OMA (*Open Mobile Alliance*) ou le *Parlay Group* se sont consacrés à la normalisation des services, assurant ainsi une certaine cohérence face à la multiplication des acteurs, inhérente à l’ouverture du modèle.

Il est intéressant de noter que les deux mondes, Web et télécom, tendant à se confondre au fil des évolutions technologiques, certains standards Internet générale-

ment plus flexibles et ouverts, sont adoptés dans les normes de télécommunications. Un exemple connu est SIP, un protocole de signalisation applicatif standardisé par l'IETF qui fut adopté par le 3GPP pour son infrastructure IMS.

Un standard de télécommunications a eu une importance toute particulière durant ces travaux de recherche : l'IP Multimedia Subsystem (IMS). Cette architecture de services très prometteuse lors de son introduction peu avant le début de la thèse propose des mécanismes de mobilité intéressants par lesquels l'étude a débuté. De plus c'est la première approche qui établit véritablement le lien entre les mondes Web et télécom. Ainsi la section 1.2.2 présentera brièvement l'IMS et quelques-uns de ses principaux concepts.

1.2.2 IP Multimedia Subsystem

L'IP Multimedia Subsystem est l'architecture de services de référence introduite par le 3GPP en 2003. Elle constitue la dernière évolution du cœur de réseau de télécommunications UMTS (*Universal Mobile Telecommunications System*, une norme de téléphonie mobile 3ème génération). Pour en savoir plus, [22] est un ouvrage particulièrement complet sur l'IMS.

La philosophie de l'IMS est de faire le lien entre services et utilisateurs de manière transparente, quelque soit la technologie d'accès employée, en d'autres termes assurer la convergence. Pour cela, le 3GPP propose de changer le modèle vertical classique (cf. Figure 1.9(a)) où chaque réseau possède sa couche de contrôle et son portefeuille de services dédiés (le modèle téléphonique fermé et monopoliste en vigueur jusqu'alors). Outre la situation protectionniste avantageuse pour l'opérateur *sur son réseau*, la verticalité est dommageable aux niveaux contrôle et service. Les mécanismes de contrôle sont dédiés à la technologie d'accès sous-jacente et donc non transposables à d'autres environnements, de même les services doivent être développés pour une couche de contrôle donnée et donc incompatibles avec d'autres infrastructures.

Face à ce manque d'efficacité manifeste, le 3GPP apporte une solution. Le modèle est désormais organisé en couches horizontales (cf. Figure 1.9(b)), services et réseaux d'accès sont articulés autour d'une couche de contrôle unique basée sur le protocole IP (*Internet Protocol*, [23]) et assurée par l'IP Multimedia Subsystem.

En dessous, l'interconnexion avec les différents réseaux d'accès est normalisé par le groupe *Telecoms & Internet converged Services & Protocols for Advanced Networks* (TISPAN de l'ETSI). Au dessus, les services sont standardisés par l'OMA (ou encore le Parlay Group et l'ETSI pour assurer la (rétro-)compatibilité).

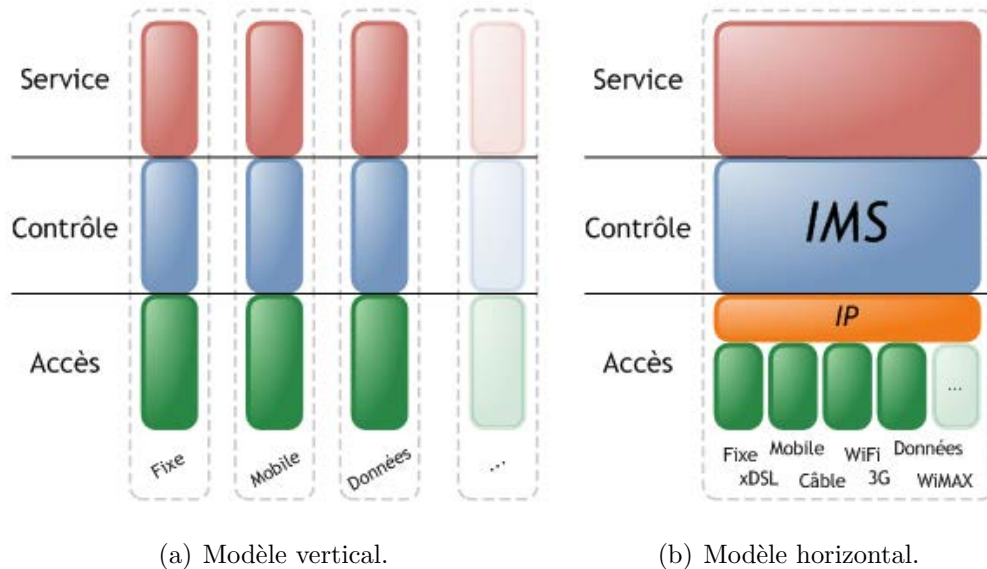


FIGURE 1.9 – Modèles d'architectures de services.

Architecture.

L'IMS est en réalité un ensemble d'infrastructures IMS possédées par divers opérateurs télécoms et inter-connectées entre elles. Par abus de langage on appelle IMS chaque domaine autonome déployé par un opérateur, un schéma simplifié de réseau 3G comportant un cœur IMS est illustré en Figure 1.10.

Un IMS, ou plus exactement un domaine IMS est composé d'un grand nombre de composants qui collaborent afin de réaliser l'ensemble des fonctions nécessaires à une infrastructure de services (télécom) : sécurité (identification, authentification, autorisation), routage (établissement de sessions, communication inter-opérateur), déclenchement de service, facturation, etc. Le tout est assuré par les *Call Session Control Functions* (CSCF), la colonne vertébrale de l'IMS constituée de trois éléments clés (cf. [24, 25]).

- **Proxy-CSCF**. Le point d'entrée d'un domaine IMS, il assure une liaison sécurisée entre le terminal et l'infrastructure.
- **Interrogating-CSCF**. Composant médiateur, il assure le routage inter-domaine et permet en outre de masquer la topologie interne de l'infrastructure sous-jacente.
- **Serving-CSCF**. Le cerveau du domaine IMS, il réalise les opérations de base (routage, authentification) mais surtout il déclenche les services de chaque utilisateur en fonction du contexte.

Enfin, ces composants de contrôle sont connectés à une base de données, le *Home Subscriber Server* (HSS) qui détient l'ensemble des informations relatives aux clients du domaine : données d'authentification, profils de services, etc.

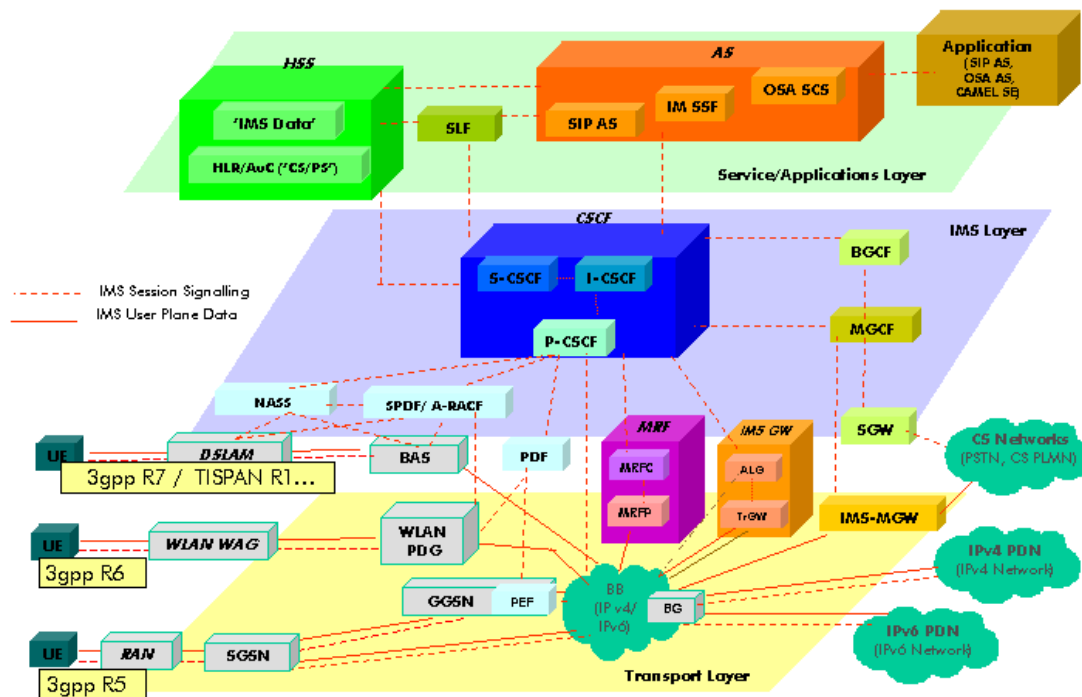


FIGURE 1.10 – Réseau 3G selon le 3GPP.

Sur la Figure 1.10 représentant l'intégralité d'un réseau 3G tel que proposé par le 3GPP, vous saurez y retrouver l'organisation structurelle horizontale vue précédemment (cf. Fig. 1.9(b)) : la couche contrôle assurée par l'IMS au milieu, la couche application/services au-dessus et la couche transport/accès au-dessous.

Au niveau IMS, les communications sont effectuées via SIP, un protocole venu du monde Internet et standardisé par l'IETF qui permet l'établissement et le contrôle de sessions multimédias. L'ensemble des services sont ainsi délivrés de bout en bout via SIP. Pour plus de détails sur ce protocole, cf. RFC 3261 [12].

Le rôle premier d'une infrastructure de services est de gérer la fourniture des services aux clients : déclenchement, composition, interactions, . . . Voyons en quoi cela consiste et de quelle manière est-ce implémenté dans l'IMS.

Déclenchement des services.

Le déclenchement (de services) est le mécanisme qui contrôle la fourniture des fonctionnalités aux utilisateurs en fonction du contexte. En effet, les services souscrits par un client doivent être fournis au bon moment conformément à ses attentes. Ainsi, pour chaque utilisateur, des règles sont établies afin de conditionner la fourniture d'un service. Par exemple, lors de l'établissement d'une communication avec un service d'anonymat, divers mécanismes entrent en jeu afin de cacher l'identité de l'appelant tel qu'il l'a souscrit auprès de son opérateur. Le déclenchement du service est réalisé de manière transparente pour le client.

Dans l'IMS, le déclenchement est assuré par le S-CSCF qui intercepte les messages SIP initiés par les terminaux clients et les redirige temporairement vers des serveurs d'applications (AS, *Application Server*) chargés de réaliser une action précise correspondant à un service tel que masquer l'identifiant de l'appelant. Ce déclenchement s'opère au sein du S-CSCF lorsque un ensemble de conditions est satisfait, ces règles appelées iFC (*initial Filter Criteria* [26]) sont stockées dans le HSS et évaluées à chaque message SIP transitant par le S-CSCF. Ainsi, en fonction de l'état d'une session de communication entre deux clients (création, modification, terminaison, etc) les services adéquats peuvent être appliqués aux utilisateurs.

Le déclenchement de services est réalisé pour chaque intervenant de la session de communication et pour chaque service souscrit et ce par le domaine dont

l'utilisateur dépend. Les messages SIP qui permettent de gérer la session de communication sont ainsi routés entre les S-CSCF et leurs AS respectifs.

Composition et interactions.

Outres les fonctions de base intégrées à la couche de contrôle (à l'IMS) tels que le routage, l'authentification ou la facturation, les autres services sont proposés au-dessus, par les AS dans la couche application. Il est à noter que certains services essentiels et appelés *enablers* sont standardisés par le 3GPP ou l'OMA tels que la *Présence* [27, 28] ou le PoC (*Push-to-talk over Cellular* [29, 30]) afin d'offrir des « briques fonctionnelles » à des services plus évolués. C'est alors en composant ces *enablers* et éventuellement d'autres services de base que des fournisseurs d'applications tiers peuvent déployer à moindre coût des services riches sur l'infrastructure IMS ; du moins, c'est l'objectif affiché.

La composition reste toutefois complexe et elle est définie que très vaguement dans l'environnement IMS [31], cependant de nombreux travaux existent sur ce sujet tels que [32, 33, 34].

Un même service pouvant être composé des fonctionnalités de plusieurs AS, le nombre de déclenchements réalisés pour chaque participant à la communication peut alors devenir très important, augmentant d'autant le risque d'interférence entre les fonctionnalités offertes, c'est ce que l'on appelle « interaction de services ».

La problématique d'interaction de services n'est pas récente, de nombreux travaux ont été réalisés sur le Réseau Téléphonique Commuté (RTC) afin d'assurer une gestion cohérente des fonctionnalités offertes (cf. [35]). Le test des services avant leur déploiement (*offline*) et l'offre de packs de services compatibles permettaient aux opérateurs de contrôler a priori les interactions. Cependant, le modèle s'est ouvert et l'opérateur ne possède plus un contrôle total sur l'offre de service, des fournisseurs tiers interviennent directement. Il devient alors impossible de résoudre a priori les interactions ; de nouvelles solutions, dynamiques (*online*), doivent être adoptées. Aucun mécanisme adéquat n'est explicitement défini dans les spécifications techniques du 3GPP, cependant de nombreux travaux existent tels que [36, 37, 38] ou encore [39] (mémoire de Master recherche du même auteur).

Il est intéressant de constater à quel point les réseaux de télécommunications ont évolué, faisant une place de plus en plus grande aux services. S'ils étaient initialement centrés sur les couches basses proposant des fonctionnalités simples tel que la voix, les améliorations techniques au niveau de l'accès (technologies sans fil, débits) ont bouleversé les modèles économiques des opérateurs. Ceux-ci misent désormais sur les services et leur infrastructure qui constituent aujourd'hui la véritable valeur ajoutée : ouverture du modèle, déploiement rapide, composition automatique, détection et résolution d'interactions, ... L'IMS en est une illustration.

1.2.3 Une fenêtre sur le Web

Au début du 21ème siècle, les évolutions technologiques aidant, notamment en matière de radiocommunications (débit accru avec l'avènement de la téléphonie de troisième génération) et d'électronique (miniaturisation des appareils informatiques), le monde télécom s'est timidement ouvert en proposant un nouveau « super-service » à ses abonnés : pas moins que le Web, la toile de services accessibles via Internet.

Avec une fenêtre sur le Web, l'infrastructure de télécommunications efface ainsi l'un de ses principaux défauts : une offre de services réduite, rigide et peu attrayante où le client reste passif. Cependant, si l'ouverture du modèle est une révolution pour les utilisateurs, elle pose de nombreux problèmes aux opérateurs télécoms qui doivent faire face à une concurrence nouvelle, introduite au sein même de leur réseau via Internet.

En effet, non seulement les opérateurs perdent leur monopole dans la fourniture des services mais les fournisseurs tiers provenant du Web sont très nombreux, performants et présents dans tous les domaines. La communication n'échappe évidemment pas à la règle et les opérateurs sont obligés dans un premier temps de restreindre techniquement les abonnés afin de conserver un minimum de contrôle et préserver leur modèle économique le temps de trouver des solutions viables.

Et c'est là tout le paradoxe né de la rencontre de ces deux mondes, les opérateurs télécoms cherchent à vendre un service qui va à l'encontre même des principes de leur infrastructure. Pour illustration, au moment de la rédaction de ce manuscrit (2008-2009) différents mécanismes sont mis en œuvre chez certains opérateurs

afin de « canaliser » les fonctionnalités Internet offertes à leurs abonnés mobiles : bridage du débit maximal des terminaux, blocage ou surtaxe de ports réseaux spécifiques, etc.

1.3 Cadre conceptuel

Via Internet, les mondes applicatif et des télécommunications tendent à se rapprocher, suffisamment du moins pour établir des propriétés communes et positionner différents concepts autour de la notion de service. Mais l'état de l'art présenté dans ce premier chapitre présente des lacunes dans la définition du concept de service, essentiellement dû au fait qu'il n'est généralement appréhendé que d'un angle restreint au domaine d'étude : économie, systèmes, Web, télécommunications, etc. Afin de permettre une compréhension constante des travaux exposés dans ce mémoire, nous proposons un cadre conceptuel qui reprend les définitions existantes et les complète afin de dessiner les relations entre ces différents concepts. La carte conceptuelle présentée en Figure 1.11 offre une vue haut-niveau d'un service et des différentes relations le liant aux notions qui l'entourent définies en section 1.3.1.

1.3.1 Autour du service

Comme nous avons vu précédemment (cf. 1.1.1), un *Service* fournit des *Fonctionnalités* à une personne réelle directement ou indirectement. En effet, conformément à l'état de l'art nous considérons qu'un service est intrinsèquement lié à un *Utilisateur*.

Le service ainsi fourni est instancié par une *Application* qui implémente et gère la fonctionnalité correspondante. Une application peut revêtir diverses formes comme nous l'avons vu dans cette première partie d'état de l'art : application locale, client/serveur, pair-à-pair, . . . elle n'est pas limitée à un seul programme ou logiciel.

Cette application est exécutée par un *Système*, lui même embarqué dans un *Terminal*. La notion de système est généralement synonyme de *système d'exploitation* (SE [40]) cependant sur certains terminaux basiques à capacités limitées, on parle plutôt de *système embarqué* (cf. [41]) où l'application se confond au sys-

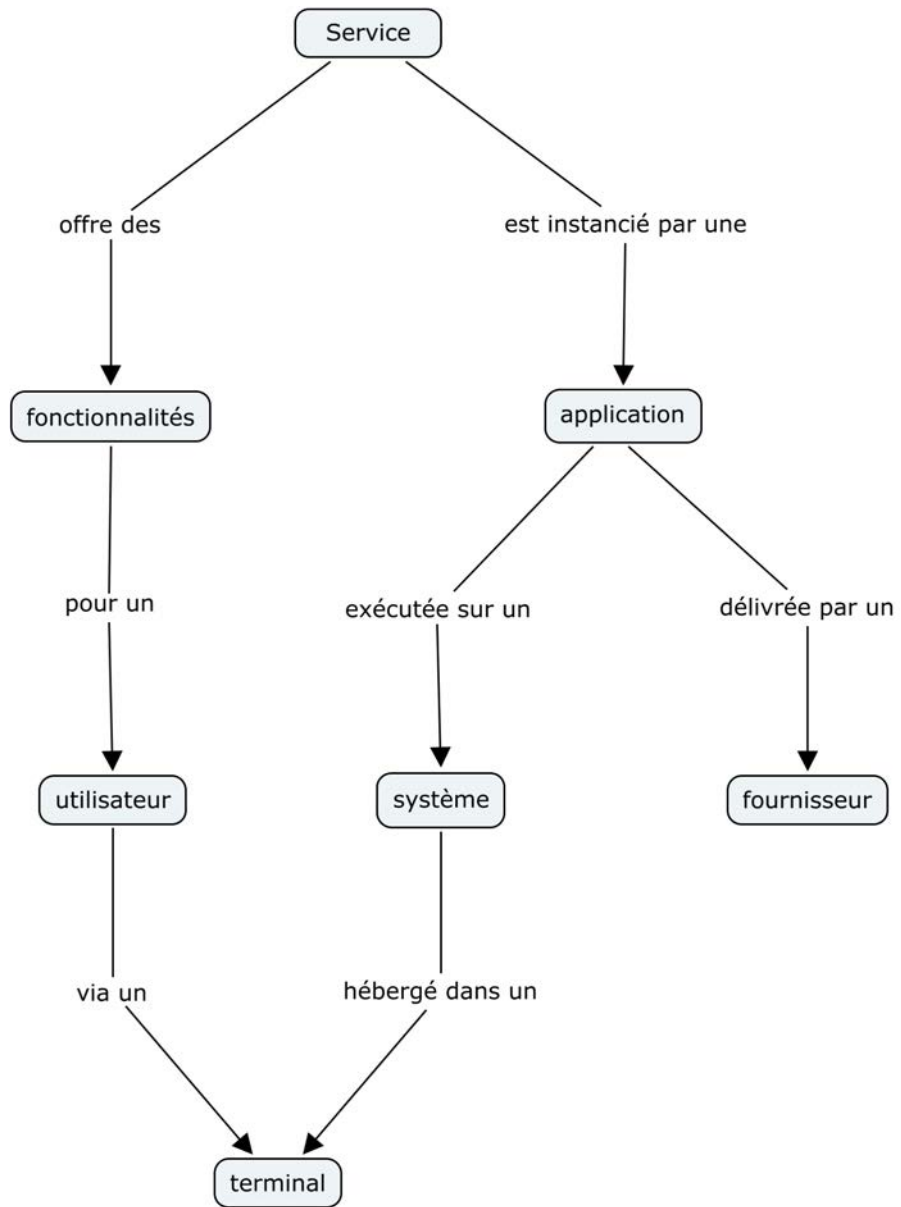


FIGURE 1.11 – Carte conceptuelle autour de la notion de *Service*.

tème dédié à un service spécifique. Les terminaux seront définis plus longuement en section 1.3.3.

Enfin, si le service est fourni à un Homme, il l'est également par un Homme. Le *Fournisseur* est celui qui offre (ou plutôt vend) le service. Dans le cadre des Technologies de l'Information et de la Communication, le fournisseur délivrera une application ou une autorisation d'accès via une licence qui offrira la fonctionnalité.

La Figure 1.11 illustre le positionnement de ces concepts et leurs relations. Les définitions proposées ici respectent et complètent l'état de l'art afin d'offrir au lecteur une meilleure compréhension des notions utilisées dans ce mémoire. Le niveau d'abstraction est volontairement très élevé afin de rester indépendant de toute implémentation, nous verrons par la suite que c'est cette abstraction même qui permet une généricité des solutions proposées. Enfin, certains de ces concepts seront développés plus loin (cf. 6.2) afin de proposer une vision plus pragmatique de la notion de service.

1.3.2 Notion de connexion

La notion de *connexion* apparaît souvent dans la littérature des Technologies de l'Information et de la Communication, cependant les concepts de modes « connecté » et « non-connecté » s'appliquent généralement à la couche réseau. Ainsi, le mode connecté correspond à la synchronisation entre l'émetteur et le récepteur d'une transmission de données s'effectuant sur une liaison réseau pré-établie qui garantit une communication fiable. L'illustration classique de ce mode est le modèle de transfert du protocole TCP (*Transmission Control Protocol* [42]). À l'inverse, le protocole UDP (*User Datagram Protocol* [43]) offre un mode non-connecté dans le sens où la liaison émetteur-récepteur n'est pas établie a priori, la transmission n'étant alors pas garantie au niveau réseau.

La notion de connexion au niveau réseau est un indicatif de fiabilité de transmission, ce qui a peu de sens au niveau applicatif. Cependant la dépendance d'un service au réseau est une propriété importante lorsque l'on considère certaines contraintes, notamment celles de mobilité tel que nous le verrons dans la deuxième partie de cet état de l'art (cf. 2). Les notions de service *connecté* et *non-connecté* sont peu présentes dans la littérature or nous pensons qu'il serait inexact de leur

faire correspondre les concepts réseaux du même nom. En effet, nous assumons que d'un point de vue « service », la notion de connexion indique une dépendance au réseau, peu importe le mode du protocole de transmission sous-jacent.

Par exemple, une application qui consomme (ou produit) un flux de données constant basé sur UDP tel qu'une communication audio ou du streaming vidéo, offrira un service connecté bien que le protocole réseau utilisé soit en mode non-connecté car l'arrêt de la transmission interromprait immédiatement la fourniture du service. Inversement, une application basée sur un protocole connecté fournit un service non-connecté si l'interruption de la connexion réseau peut être supportée, ne serait-ce que temporairement, pendant la fourniture du service ; par exemple le protocole HTTP d'un navigateur Web basé sur TCP. Les services locaux (cf. 1.1.1) seront alors nécessairement non-connectés tandis que ceux basés sur des flux de données continus seront toujours en mode connecté.

1.3.3 Les terminaux

Dernier terme qu'il reste à définir, la notion de *terminal*. Les terminaux ne sont pas directement concernés par notre étude qui porte plus sur les services et leur mobilité, cependant ils sont omniprésents dans le contexte de nos travaux car ils sont l'unique objet physique, l'interface entre l'utilisateur et le(s) service(s). Le terme « terminal » est très employé dans la littérature mais aussi très peu défini de manière générique car toujours considéré dans un contexte spécifique. Dans le cadre de ce mémoire nous resterons très haut-niveau en considérant les terminaux comme des *appareils électroniques* possédant au moins les caractéristiques suivantes :

- une interface utilisateur,
- une connectivité réseau,
- une application.

Cette définition de terminal ne tient pas pour définition générale, elle indique uniquement au lecteur comment comprendre ce terme tout au long de ce mémoire.