CLOUD COMPUTING

Sécurité, gouvernance du SI hybride et panorama du marché

Guillaume Plouin
Directeur technique chez Deveko

Préface de Matthieu Hug

4e édition

DUNOD

Toutes les marques citées dans cet ouvrage sont des marques déposées par leurs propriétaires respectifs.

Illustration de couverture :
Golden Gate Bridge © Michael Rosskothen - fotolia.com

5 rue Laromiguière, 75005 Paris
www.dunod.com

ISBN 978-2-10-074511-1

Préface

Lorsque j'ai démarré RunMyProcess en 2007 le terme « cloud computing » n'existait pas, « software as a service » était une expression qui apparaissait tout juste, Amazon Web Services était à peine en version beta et encore à un an d'une version de production. C'est peu après, début 2009, que j'ai rencontré Guillaume Plouin et que nos discussions, puis la première édition de son panorama du marché du cloud computing, m'ont aidé à mieux formuler et comprendre les enjeux globaux du cloud computing, que j'appréhendais plus par intuition que par analyse.

Lorsque le concept s'est répandu, on a assisté à toutes les phases classiques : déni (« ça ne sert à rien » ou « c'est uniquement pour les PME »), dénigrement (« ce n'est pas sécurisé », « c'est uniquement pour des usages simplistes »), ou tentative de banalisation (« on fait ça depuis dix ans »). Tout cela étant passé, nous sommes à présent dans une phase à laquelle notre industrie informatique n'est que trop souvent soumise : la complexification.

La complexification est l'arme ultime des acteurs en place qui perdent pied et n'arrivent pas à se transformer. Elle permet de faire croire que l'on est à la pointe grâce à un écran de fumée sémantique, en modernisant le nom des offres et des produits que l'on a déjà. Ainsi n'est-il pas infiniment plus cool de parler de « cloud privé » que de serveurs, de NAS ou de rack ? D'autant plus que ce qui n'est pas privé est forcément public, autant dire pas sécurisé ou pas sérieux, ce qui permet d'introduire un doute sur ces nouvelles offres dites « publiques » mais qui ne le sont pas...

Une autre approche consiste à donner un nom sérieux à une banalité : « cloud hybride » par exemple. Certaines applications ou charges seront dans des cloud et d'autres dans des datacenters ou sur des serveurs gérés traditionnellement, et ce pour longtemps. En gros la diversité est et restera : quelle information !

Reste que « la simplicité est la sophistication suprême » ainsi que nous l'a appris Léonard de Vinci, et qu'il est donc important de se débarrasser des scories de la complexification afin de ne s'intéresser qu'à ce qui compte. En l'occurrence le cloud computing change fondamentalement trois choses.

Tout d'abord le Web est de facto la seule plate-forme d'interconnexion à grande échelle du monde numérique mais aussi physique. Le débat des réseaux, des protocoles

et de beaucoup de normes techniques est désormais au mieux marginal : SOAP ou ReST ? La question n'a aucun intérêt, seul ReST permet de se connecter simplement au Web. Inversement cela veut dire qu'un système ou une donnée qui n'est pas relié au Web (avec la sécurité adéquate bien sûr) est mort ou en agonie : il ne produit pas assez de valeur pour son propriétaire.

Ensuite grâce à la concentration industrielle créée par le cloud computing (celui dit « public », le reste n'est ni bien ni mal, mais n'a rien à voir), le coût d'accès à la ressource informatique a été divisé grosso modo par 100 en quinze ans : calcul ou stockage, mais aussi toutes sortes d'applications ou de composants, du CRM à l'intelligence artificielle. Cela a deux conséquences : tout d'abord la simple possession d'une infrastructure informatique par une entreprise ne suffit plus à procurer un avantage stratégique car une PME peut aisément avoir des systèmes plus modernes, moins chers et plus efficaces qu'un groupe CAC 40. Ensuite sur tous les marchés un nouvel entrant peut utiliser la technologie pour s'insérer très rapidement entre les acteurs dominants et leurs clients, pour un coût faible et donc sans être visible avant qu'il ne soit trop tard : c'est l'« uberisation », néologisme assez laid, mais qui décrit néanmoins une réalité économique très concrète.

Le cloud computing, par une révolution technologique et des modèles économiques, a ainsi amené une situation paradoxale : toute l'économie est impactée en profondeur par le numérique, mais la possession de l'outil informatique ne présente plus de valeur stratégique. Où se trouve alors la valeur ? Dans la capacité à exécuter rapidement, et c'est là le troisième changement clé du cloud computing : le temps informatique s'est considérablement raccourci par rapport aux standards de l'informatique d'entreprise. Nous sommes en 2016, il y a six ans il n'y avait pas d'iPad ; six ans c'est donc une génération en informatique : aucun ERP aucun projet n'apporte assez de valeur pour nécessiter trois ans, a fortiori cinq ou six ans, de déploiement car il serait alors obsolète dès le premier jour de mise à disposition. L'obsolescence en soi n'est pas grave, mais en l'occurrence elle crée un handicap majeur dans un monde où l'informatique est le système nerveux de l'économie.

Désormais un projet informatique qui n'a pas d'impact réel en production après trois voire six mois grand maximum a été par définition mal pensé, mal organisé ou doit être arrêté. Cette accélération a un impact sur la gouvernance de l'informatique, mais elle a surtout un effet positif majeur : elle rend le coût de l'échec marginal, et donc permet la prise de risques mesurés. C'est pourquoi la prise de risque et l'échec afférent sont désormais aussi intégrés à l'éthique des start-up du numérique que le temps court et le sentiment d'urgence. À l'inverse, une entreprise dont l'informatique reste dans le temps long, où la prise de risque est exclue, devient vite peu pertinente dans son appréhension du numérique, et se trouvera donc tôt ou tard en situation d'échec grave et exclue de son marché, incapable d'y voir arriver les bouleversements ni d'en suivre le rythme.

Ainsi le cloud computing est le nom donné à un triple mouvement d'industrialisation, de démocratisation et d'accélération de l'informatique. Son potentiel créatif devient accessible à tous, bouleversant tous les secteurs économiques. C'est pourquoi il est d'une importance capitale de comprendre les tendances, d'identifier ce qui compte

réellement parmi des concepts par trop foisonnants, pour in fine gagner en rythme, en vitesse et en audace : pour ce faire la synthèse de Guillaume est depuis quelques années, et reste, une référence plus que nécessaire.

Matthieu HUG
Président, fondateur, de Fujitsu RunMyProcess

Table des matières

Préface V

Avant-propos XV

Première partie – Les concepts du cloud computing

Chapitre 1 – Contexte de l'émergence du cloud computing 3

1.1 Le cycle des interfaces informatiques 3

1.2 La montée en puissance du web 5

1.3 L'émergence de l'ASP 5

1.4 HTML5 : « le web comme une plateforme » 6

1.5 Les hébergeurs en Self Service 9

1.6 Le web 2.0 appelle le cloud computing 10

1.7 Le principe des services « device agnostic » 15

1.8 Le cloud computing : une capitalisation sur toutes les évolutions précédentes 17

Chapitre 2 – Concepts et définitions du cloud computing 19

2.1 Origine du terme cloud computing 19

2.2 Une définition plus pragmatique 21

2.3 Que signifie SaaS ? 23

2.4 Que signifie PaaS ? 25

2.5 Que signifie IaaS ? 26

2.6 Cloud computing *versus* plateformes d'entreprise 29

2.7 Le cloud, une évolution logique de l'informatique 30

Chapitre 3 – Le cloud : un nouveau modèle de consommation de services 33

3.1 L'évolution des modèles de gestion de parc applicatif 33

3.2 De la pertinence du modèle « cloud public » vis-à-vis du modèle « software interne » 42

3.3 Le cloud vu par les éditeurs 47

3.4 Cloud public*versus* cloud privé 48

3.5 Cloud public et Open Source 50

3.6 Garder les pieds sur terre... 51

Deuxième partie – Faire confiance au cloud ?

Chapitre 4 – Cloud et sécurité 55

4.1 La question de la confiance 56

4.2 Sécurité et aspects juridiques 58

4.3 Sécurité sur le plan technique 60

4.4 Gérer le risque de sécurité 69

Chapitre 5 – L'entreprise face au cloud computing 73

5.1 Par quels usages du cloud commencer ? 73

5.2 Le cloud public par secteur d'activité 74

5.3 Quel mode d'utilisation du cloud ? 76

Chapitre 6 – Bénéfices et inconvénients du point de vue des décideurs 81

6.1 Bénéfices pour l'entreprise utilisatrice 81

6.2 Inconvénients pour l'entreprise utilisatrice 86

Chapitre 7 – Bénéfices et inconvénients du point de vue des utilisateurs 89

7.1 Bénéfices pour les utilisateurs 89

7.2 Craintes des utilisateurs 95

Chapitre 8 – Bénéfices et inconvénients du point de vue des informaticiens 97

8.1 Bénéfices pour les informaticiens 97

8.2 Craintes des informaticiens 100

Chapitre 9 – Choisir un service cloud 105

9.1 L'analyse par grille de critères 105

9.2 Grille de critères cloud 106

9.3 Processus de prise de décision 109

9.4 Le cloud computing, une mutation importante pour l'entreprise 113

Troisième partie – La gouvernance du cloud

Chapitre 10 – Les premiers pas sur le cloud 119

10.1 Commencer par un pilote 119

10.2 Le premier déploiement 125

Chapitre 11 – La gestion d'un système d'information hybride 135

11.1 L'industrialisation du déploiement des applications cloud 135

11.2 L'intégration des clouds 138

11.3 La question du décisionnel dans le cloud 146

11.4 La question des référentiels dans le cloud ? 146

11.5 Comment arbitrer ses choix d'hybridation ? 148

Chapitre 12 – Généralisation du cloud et impacts organisationnels 151

12.1 Cloud et applications critiques 151

12.2 La position de la direction de la sécurité 152

12.3 Le cloud broker 152

12.4 La gouvernance d'un projet SaaS 153

12.5 La gouvernance d'un projet PaaS/IaaS 155

Quatrième partie – Les services SaaS disponibles

Chapitre 13 – Le positionnement des grands acteurs de l'IT 165

13.1 Les acteurs historiques 165

13.2 Les géants du web 170

Chapitre 14 – Les services de collaboration 173
14.1 Les concepts de la collaboration unifiée 173
14.2 Les outils de collaboration SaaS 179

Chapitre 15 – Les services FrontOffice 191
15.1 La recherche au sein d'un site web 191
15.2 Les services de cartographie 191
15.3 Les services de commerce électronique 192
15.4 Les services de mailing 194
15.5 L'analyse de trafic 195
15.6 Gestion de la relation client 196

Chapitre 16 – Les services BackOffice 199
16.1 Gestion RH 199
16.2 Analyse décisionnelle 200
16.3 Suites ERP 201

Chapitre 17 – Perspective : le modèle du cloud desktop 203
17.1 Le modèle du bureau « classique » 203
17.2 Le modèle du cloud *desktop* 204

Cinquième partie – Les plateformes de cloud computing

Chapitre 18 – Les architectures du cloud computing 211
18.1 Cloud computing et architectures multi-tiers 211
18.2 Cloud computing et architectures de services 214
18.3 Les architectures « multi-tenant » 216
18.4 Spécificités des plateformes cloud vis-à-vis des plateformes d'entreprise 218
18.5 Usage des plateformes PaaS et IaaS 222

Chapitre 19 – Les composants des plateformes PaaS et IaaS 225
19.1 Les datacenters 225
19.2 Structuration des IaaS 229

19.3 Structuration des PaaS 231
19.4 Systèmes de persistance 232
19.5 Le service d'authentification 233
19.6 Le bus d'intégration 233
19.7 La sécurisation des flux entre SI et cloud 235
19.8 L'environnement de développement 235
19.9 La gestion du cycle de vie des applications 236
19.10 Le monitoring 236
19.11 Synthèse sur l'architecture des IaaS et PaaS 237

Chapitre 20 – Les plateformes de cloud public (PaaS & IaaS) 241
20.1 Les offres à 360° (PaaS & IaaS) 241
20.2 Les offres IaaS Seul 246
20.3 Les PaaS « multi-purpose » 247

Chapitre 21 – Les plateformes de cloud privé 251
21.1 Les solutions éditeurs 251
21.2 Les solutions Open Source 255
21.3 Le cloud privé « clef en main » 256

Chapitre 22 – Les perspectives du cloud 259
22.1 Aller au bout de la logique de commodité 259
22.2 Vers des standards ? 260
22.3 Vers le cloud pervasif 261

Références bibliographiques 263

Index 265

Avant-propos

Le **cloud computing**, en français, « **informatique dans les nuages** », est en train de transformer le monde informatique. Il consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés, au même titre par exemple que les entreprises externalisent la production d'électricité vers des fournisseurs dont c'est le métier principal. Le cloud a un impact profond sur les utilisateurs et sur les stratégies informatiques des entreprises.

Les utilisateurs gagnent en autonomie et en temps de déploiement. Ils bénéficient des innovations issues du web. La composition d'applications sur le cloud leur ouvre des possibilités de collaboration inédites, entre les informations issues du web et celles issues de leurs applications métiers.

Dans les services informatiques, les études assistent à la mutation de leurs outils de développement et de leurs pratiques autour du cycle de vie des applications. Le métier des équipes d'exploitation est lui aussi profondément transformé, parfois même remis en cause.

Des plateformes comme celles de Salesforce ou Amazon ont aujourd'hui atteint un niveau de maturité satisfaisant pour les entreprises : elles offrent des services de qualité et des garanties de disponibilité supérieures à celles de beaucoup de DSI. C'est pourquoi, dès qu'elles ont surmonté leurs craintes au sujet de la confidentialité, les entreprises externalisent leurs applications non stratégiques afin de recentrer leur direction des systèmes d'information sur leur cœur de métier.

Cet ouvrage a pour objectif de présenter les concepts et les révolutions sous-jacentes au cloud computing. Nous verrons en particulier que l'on parle de **SaaS**, *Software as a Service*, pour désigner les progiciels hébergés par leurs concepteurs, et de **PaaS/IaaS**, *Platform/Infrastructure as a Service*, pour les environnements d'exécution mis à la disposition des entreprises qui souhaitent faire héberger leurs développements spécifiques.

Tous les grands cabinets d'analyse internationaux[1] s'accordent pour dire que le cloud computing va monter en puissance dans les entreprises dans les années à venir et devenir un élément à part entière du système d'information.

À qui s'adresse ce livre ?

Cet ouvrage s'adresse à tous ceux qui souhaitent comprendre les concepts et les enjeux du cloud computing. Il est bien sûr accessible aux informaticiens (chefs de projets, architectes, développeurs, équipes d'exploitation) mais aussi aux profils non techniques (métiers, maîtrises d'ouvrage, marketing, communication).

Seule la cinquième partie présente des concepts techniques qui s'adressent uniquement aux informaticiens.

Contenu

- **La première partie, « Les concepts du cloud computing »,** a pour objectif d'introduire les concepts du cloud computing, des SaaS, PaaS, et IaaS. Elle présente les différents modèles logiciels et situe le modèle cloud dans ce contexte. Elle montre la cohérence du cloud avec la tendance des entreprises à externaliser et à s'ouvrir sur Internet.
  Cette partie est accessible à tous.
- **La deuxième partie, « Faire confiance au cloud ? »** a pour objectif de présenter de manière détaillée les opportunités et les risques du cloud computing pour l'entreprise. Elle aborde successivement les points de vue de la direction, des utilisateurs, et des informaticiens, avant de dresser une synthèse en vue d'une aide à la décision.
  Un cas d'usage fictif est introduit : celui de la société INDUS, dans le secteur industriel.
  Cette partie est accessible à tous.
- **La troisième partie, « La gouvernance du cloud »,** décrit les différentes étapes que va franchir une entreprise qui souhaite aller vers le cloud computing. Elle aborde l'usage d'une première application SaaS de commodité, puis le déploiement sur les plateformes IaaS/PaaS. Elle évoque l'intégration du cloud avec le SI.
  Cette partie est accessible à tous.
- **Dans la quatrième partie, « Les services SaaS disponibles »,** on s'interroge sur les attitudes des différents acteurs informatiques vis-à-vis du cloud computing. Cette partie propose ensuite un panorama des offres SaaS, ou progiciels en ligne prêts à l'emploi. Ces offres sont classées suivant les catégories : services de collaboration, services FrontOffice, services BackOffice.
  Cette partie est accessible à tous.
- **La cinquième partie, « Les plateformes PaaS et IaaS »,** aborde les aspects techniques du cloud computing. Elle décrit les architectures sous-jacentes aux

1. Gartner, Forester, McKinsey.

plateformes cloud et elle présente leurs particularités. Elle décrit les principales plateformes disponibles pour les entreprises qui souhaitent faire héberger leurs développements spécifiques sur des plateformes cloud.
Cette partie, plus technique, est destinée aux informaticiens.

Remerciements

Ma reconnaissance va à Yahya El Mir, directeur du groupe SQLI, qui a rendu ce projet possible, à Ludovic Cinquin, directeur d'OCTO France qui en a sponsorisé les rééditions.

Je remercie en particulier Nabil Boutakhrit, Mohamed Larbi Elhafyani, Fatima Bouchnaif, Idriss Mrabti, Lamyae Jdaini, Pascal Borelli, Ludovic Piot, Marc Bojoly, Olivier Mallassi, Damien Joquet et Yannick Martel. Ils ont largement contribué au contenu de ce livre en échangeant avec moi sur le cloud.

PREMIÈRE PARTIE

Les concepts du cloud computing

Cette première partie présente l'émergence du cloud computing comme une suite logique dans l'histoire de l'informatique. Elle montre comment les évolutions successives des systèmes informatiques et l'ouverture des entreprises vers l'Internet aboutissent logiquement au cloud computing. Elle montre aussi comment le contexte actuel – volonté d'accélérer les projets, de réduire des coûts, nouveaux terminaux, etc. – constitue un cadre idéal pour le cloud computing.

Cette partie présente d'abord les grands concepts du cloud computing : élasticité, *Self Service*, paiement à la consommation. Elle aborde ensuite les concepts de SaaS, PaaS, IaaS.

Elle explique les différents modèles de consommation IT et situe le modèle du cloud computing dans ce contexte.

Elle montre la cohérence du cloud avec la tendance suivie actuellement par les entreprises à externaliser et à s'ouvrir sur Internet : l'entreprise étendue.

1

Contexte de l'émergence du cloud computing

Objectif

L'objectif de ce chapitre est de brosser le contexte qui a donné lieu à l'émergence du cloud computing.

Il présente les mouvements et les évolutions qui ont précédé l'émergence du cloud : web 2.0, RIA, ASP, terminaux mobiles.

1.1 LE CYCLE DES INTERFACES INFORMATIQUES

Depuis sa montée en puissance dans les années 1960, l'architecture informatique suit un cycle régulier de centralisation/décentralisation. Ainsi les premiers systèmes utilisés en entreprises étaient des *mainframes*, c'est-à-dire des machines dans lesquelles toute la logique de calcul et de persistance de l'information était centralisée. Les interfaces d'accès à ces systèmes étaient des terminaux passifs, à l'image du fameux Minitel, vielle fierté nationale française. Les terminaux passifs étaient composés d'un simple couple écran/clavier et constituaient des interfaces d'accès interchangeables, qui ne contenaient aucune donnée utilisateur.

Au début des années 1990, sont apparues les architectures client/serveur qui ont permis le report des traitements sur les postes de travail, les fameux ordinateurs personnels (PC ou *Personal Computer*), inventés par IBM. Ces PC ont permis la montée en puissance de Microsoft qui leur a fourni leurs logiciels embarqués : les

incontournables Windows et Office. L'idée novatrice du client/serveur était de répartir les traitements entre un serveur et un poste utilisateur devenu capable d'exécuter certains processus métier. Le rôle du serveur était dans la plupart des cas de centraliser les données et de gérer une partie des traitements, tandis que le client gérait l'autre partie des traitements et l'interface utilisateur. La communication entre ces deux « tiers » s'effectuait au travers d'une couche logicielle spécifique souvent appelée « middleware ». L'architecture client/serveur a été massivement utilisée dans la plupart des systèmes d'information, mais elle a fini par montrer ses limites. En effet, l'absence de standardisation du protocole d'échange rendait difficile la gestion des flux. De plus, la non-standardisation du frontal client a confronté les directeurs informatiques à la délicate problématique du déploiement sur les postes utilisateurs. De plus, le frontal client embarquait souvent une base de données locale, désynchronisée des bases d'entreprises.

Au milieu de ces mêmes années 1990, les architectures web ont conduit à la recentralisation de la logique de traitement sur des serveurs centraux, ramenant le PC à un simple dispositif d'affichage au travers du navigateur. Elles ont permis l'usage d'applications à l'échelle de l'Internet grâce aux standards HTTP[1] et HTML[2]. De plus, elles ont permis un accès aux applications sans passer par la douloureuse phase de déploiement logiciel sur chacun des PC du parc informatique.

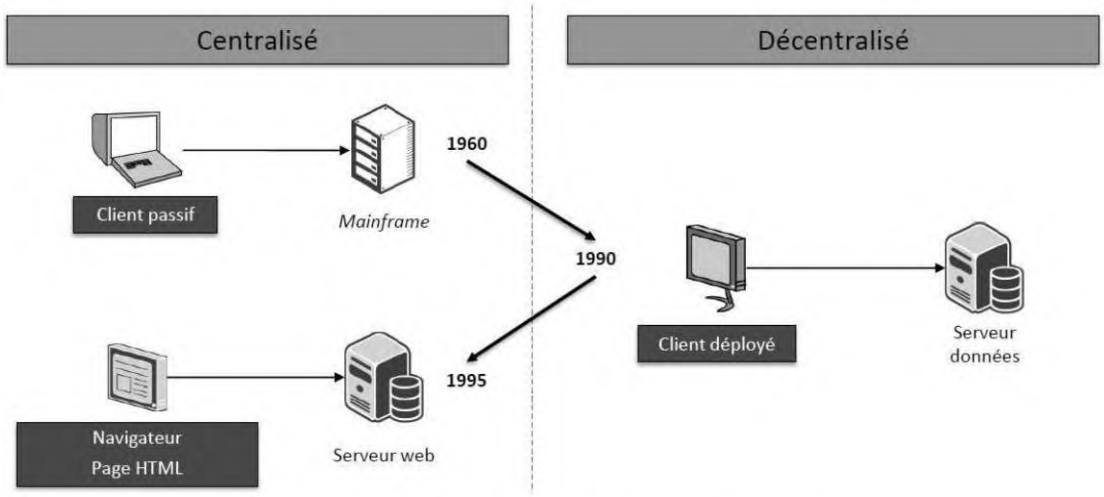

Figure 1.1 — Le cycle de centralisation/décentralisation.

1. HTTP (*HyperText Transfer Protocol*) est le protocole de communication utilisé par les sites web.
2. HTML (*HyperText Markup Language*) est le langage utilisé pour décrire les pages web. Il est notamment basé sur le principe de l'hypertexte.

1.2 LA MONTÉE EN PUISSANCE DU WEB

Les standards du web (HTTP et HTML) ont été inventés en 1990 par Tim Berners-Lee. Ce scientifique souhaitait partager des données avec ses collègues du CERN[1] : il a pour cela conçu un principe de pages présentant des fiches techniques, liées entre elles par des liens hypertextes.

L'idée initiale de Tim Berners-Lee était donc de créer une sorte d'encyclopédie en ligne, à la manière de Wikipédia. Lorsque le web est devenu une plateforme mondiale, son invention a été reprise par les entreprises qui l'ont utilisée pour diffuser des plaquettes commerciales à moindre coût : les fameux « sites vitrines ». Puis à la fin des années 1990, ces sites, au départ statiques, ont commencé à devenir transactionnels, permettant l'émergence du commerce électronique, pour devenir de **véritables applications informatiques**.

Le web a aussi introduit un changement dans l'évolution de l'informatique : en effet, des **innovations ont commencé à être testées auprès du grand public** (par exemple les moteurs de recherche), **avant d'être déclinées pour les entreprises**.

1.3 L'ÉMERGENCE DE L'ASP

C'est à cette période qu'est né le concept des **ASP**, les *Application Services Providers*. Des créateurs de start-up ont vu le parti qu'ils pouvaient tirer des architectures web : proposer aux entreprises de louer des applications métiers hébergées par leurs soins, dans leurs centres serveurs. Les ASP promettaient à leurs éditeurs des revenus réguliers grâce à un système d'abonnement. Elles promettaient aux entreprises utilisatrices de se débarrasser des problématiques d'exploitation de ces applications.

À cette époque, deux alternatives s'offraient aux applications en ASP pour construire leurs interfaces utilisateurs :

- utiliser une interface web ;
- utiliser une interface client/serveur.

1.3.1 L'ASP en interface web

À ce stade, il est important de souligner qu'accéder à une application de collaboration ou à une application métier depuis une interface HTML des années 1990 peut se révéler très frustrant : en effet, ces dernières sont limitées en termes de capacité d'interaction. Elles proposent une navigation de page en page suivant un scénario préétabli. Ce mode d'interaction est très adapté à une opération exceptionnelle comme la télédéclaration des impôts ou l'achat d'un livre sur un site de commerce électronique. En revanche, il est très limitant pour une application utilisée tous les jours, pour laquelle on souhaiterait disposer d'une bonne productivité (réactivité de

1. Le laboratoire de recherche fondamentale européen situé à Genève.

l'interface, raccourcis clavier, etc.) L'interface web élémentaire était donc inadaptée à une application ASP destinée à un usage quotidien.

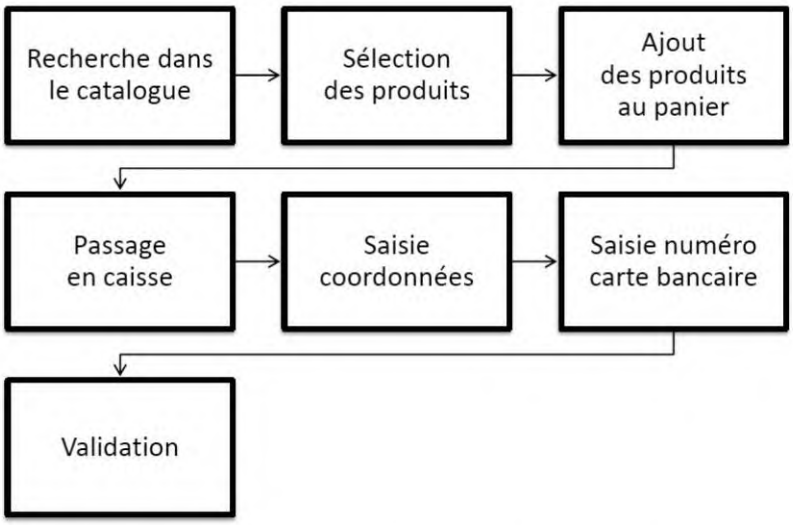

Figure 1.2 — La navigation web et la contrainte d'un scénario préétabli.

1.3.2 L'ASP en interface client natif

L'autre alternative d'interface qui s'offrait aux applications ASP était le client natif (elles proposaient souvent des applications écrites en Java). Ce type d'interface est beaucoup plus satisfaisant en termes d'ergonomie. Cependant, il nécessite un déploiement sur les postes utilisateurs, ce qui va à l'encontre de la promesse des ASP : fournir une application en mode hébergé. En effet, on retombe là dans la fameuse problématique de déploiement propre aux applications internes à l'entreprise. De plus, les middlewares utilisés par les applications client/serveur sont souvent bloqués par les firewalls d'entreprise, ce qui complexifie beaucoup leur déploiement.

Cette problématique d'interface utilisateur est la principale raison de l'échec des ASP. Nous verrons ci-après que les interfaces HTML5 ont résolu ce problème dans le cadre du cloud computing.

1.4 HTML5 : « LE WEB COMME UNE PLATEFORME »

Le concept du « **RIA**, *Rich Internet Application* » est né en 2003. Il **désignait une interface à la croisée des chemins entre les mondes client/serveur (ou client natif) et web (ou client léger)**.

Grâce à des extensions technologiques de HTML comme JavaScript, le RIA offre un supplément d'ergonomie aux pages web et permet des interfaces sophistiquées. Le RIA est basé sur un environnement d'exécution intégré au navigateur web. Lorsqu'on accède à une application RIA :

- une interface est déployée dans cet environnement ;
- l'interface échange avec des services en ligne au travers du protocole HTTP. Le RIA fonctionne alors comme une application client/serveur, le client étant l'interface RIA. Cette dernière persiste au sein du navigateur pendant toute la durée d'usage de l'application. Elle disparaît du poste utilisateur à la fermeture du navigateur.

Le RIA constitue donc une certaine forme de retour à une architecture client/serveur, mais sans la problématique de déploiement sur les postes de travail.

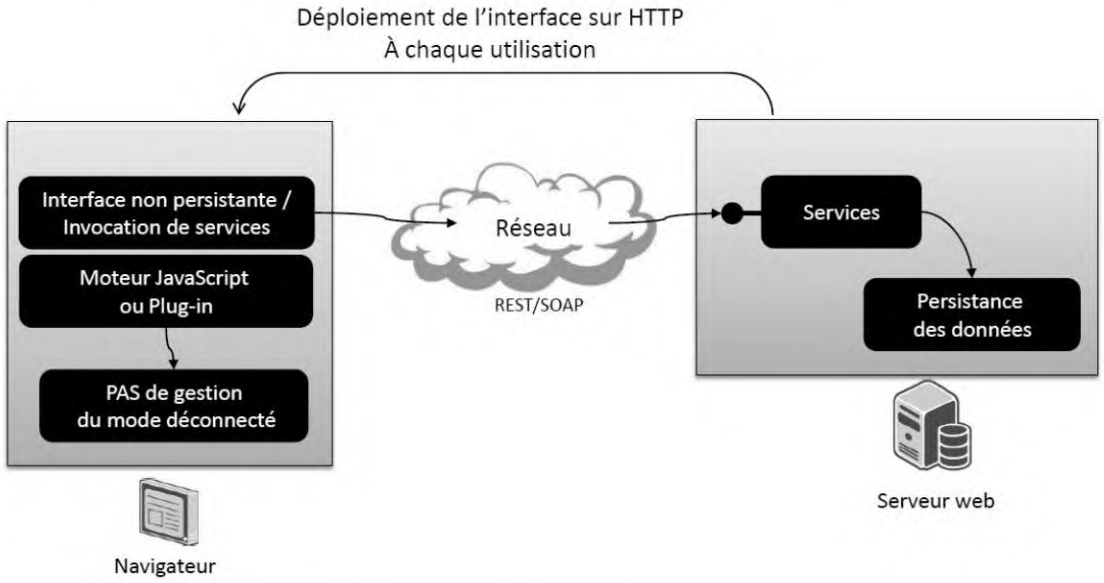

Figure 1.3 — Le fonctionnement des RIA.

Les technologies RIA disponibles aujourd'hui sont :

- HTML5, qui englobe les standards HTML, CSS et JavaScript ;
- Adobe Flash et Microsoft Silverlight, des technologies propriétaires en fin de vie.

HTML5 a l'avantage d'être entièrement basé sur des standards.

1.4.1 HTML5 : la victoire du RIA

HTML5 est une évolution majeure de HTML qui permet, entre autres, aux pages web d'échanger des données avec un service distant en tâche de fond, sans nécessiter de rechargement. Techniquement, HTML5 repose sur des appels HTTP exécutés en JavaScript.

Les nouveautés importantes de HTML5 sont :

- la création d'interfaces au pixel prêt comme en client/serveur ;
- la gestion de tous types de médias ;

- la gestion de traitements en tâche de fond ;
- la communication asynchrone avec le serveur ;
- la gestion du mode déconnecté.

Les solutions à base de HTML5 permettent de créer des pages dont l'ergonomie est identique à celle des interfaces graphiques des applications classiques – type client natif – tout en gardant la légèreté de déploiement des applications web, en étant utilisable immédiatement, sans installation, sur plus de 90 % des ordinateurs[1], et en respectant les standards.

HTML5 permet la création d'interfaces métiers ou d'interfaces grand public très dynamiques. On dispose aujourd'hui de nombreux frameworks de haut niveau pour produire des applications HTML5. On peut citer : Angular.js, Backbone.js, Ember.js.

La plateforme web basée sur HTTP et HTML devient une plateforme applicative universelle.

1.4.2 Les alternatives Flash et Silverlight

Créée en 1996, la technologie Flash a initialement été conçue pour permettre la création d'animations vectorielles au sein de pages web. Flash fonctionne avec un plug-in, une extension gratuite à installer en complément d'un navigateur web.

Silverlight a été créé par Microsoft en 2007 afin de compléter son offre de technologies d'interface et d'offrir une alternative maison à Adobe Flash.

Il faut noter que Flash et Silverlight sont fortement remis en cause par HTML5. **Steve Jobs, l'ancien patron d'Apple, avait publié une tribune en avril 2010 qui énonçait un certain nombre d'arguments en faveur de l'abandon de Flash :**

- la vidéo sur le web peut se passer de technologies propriétaires : YouTube, Daily Motion... proposent leurs contenus en HTML5 ;
- Flash est lent et instable sur les plateformes Apple malgré des années d'optimisation ;
- Flash connaît des failles de sécurité ;
- Flash consomme beaucoup de CPU et vide rapidement les batteries des appareils mobiles ;
- Flash est incompatible avec les écrans tactiles ;
- Flash est une technologie trop générique, non optimisée pour telle plateforme du marché.

On constate que le marché a entendu ce message, à tel point que Microsoft et Adobe **ont annoncé le support** de HTML5.

1. Internet Explorer 6 peut poser problème de par sa mauvaise gestion de HTML5, mais cette version est en fin de vie.

1.4.3 HTML5 au service du cloud computing

HTLM5 met fin au choix cornélien entre application web et application client/serveur. Il offre en effet une solution purement web, sans problématique de déploiement, tout en bénéficiant d'une architecture client/serveur décentralisée : une interface ergonomique, véloce, permettant une bonne productivité. HTML5 offre donc une solution très pertinente aux problématiques d'interface utilisateur des ASP. **C'est une des briques fondamentales à l'émergence des *Software as a Service*.** HTML5 intéresse aussi les plateformes d'exécution de cloud computing : ils leur offrent des interfaces sophistiquées de déploiement et de monitoring (on parlera de *Self Service* dans la suite).

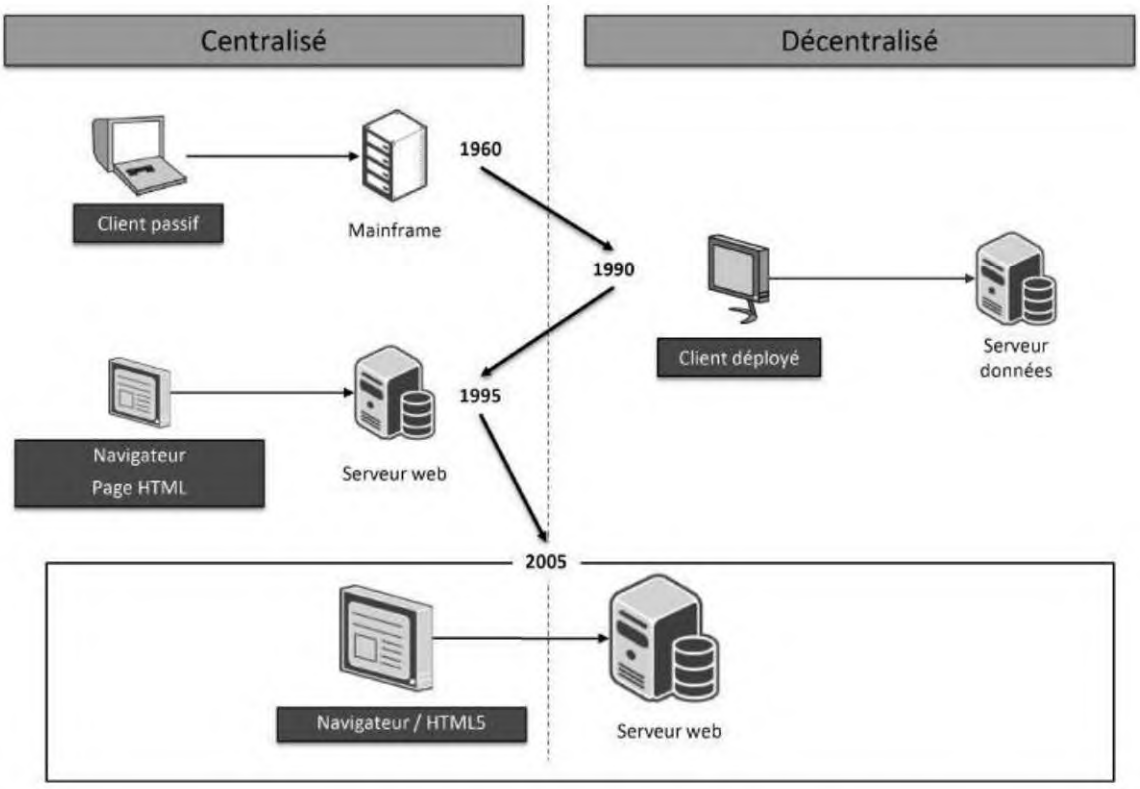

Figure 1.4 — HTML5, le meilleur des mondes web et client/serveur.

1.5 LES HÉBERGEURS EN SELF SERVICE

Le modèle traditionnel de l'hébergement repose sur des architectures techniques très customisées pour l'entreprise utilisatrice : plateforme Java ou .NET paramétrée selon un cahier des charges précis. Cette customisation nécessite des réunions d'échanges entre l'entreprise et son hébergeur ; elle entraîne des délais de mise en production de quelques semaines au minimum.

Depuis la fin des années 1990, est née une nouvelle génération d'hébergeurs qui propose une plateforme technique générique : le plus souvent LAMP[1]. On peut souscrire aux offres de ses hébergeurs depuis le web en *Self Service*. *Self Service* signifie que l'interface permet de paramétrer sa configuration, envoyer ses fichiers, et payer par Carte Bleue. Et en quelques heures, l'application est disponible.

OVH, Amen, etc. sont emblématiques de cette génération d'hébergeurs en *Self Service*. Leurs offres sont les précurseurs des environnements d'exécution du cloud. Elles permettent un service moins coûteux (parce que standardisé, banalisé et sans intervention humaine chez l'hébergeur), plus souple (parce que géré par le client) et de meilleure qualité (parce que standardisé et automatisé).

1.6 LE WEB 2.0 APPELLE LE CLOUD COMPUTING

L'idée de cette section est de montrer que les principes de web 2.0 ont naturellement débouché sur le cloud computing.

Le terme web 2.0 a été créé en 2005 par Tim O'Reilly : selon O'Reilly, le web 2.0 consiste à considérer le web comme une plateforme. Nous allons préciser ce concept dans le paragraphe suivant.

1.6.1 L'intelligence collective et les « *digital natives* »

Le web 2.0 repose avant tout sur le concept d'« intelligence collective » ou « sagesse des foules ». Ces termes un peu pompeux désignent les synergies qui peuvent avoir lieu entre des individus qui rédigent des textes sur le web afin de bâtir une somme de connaissances. Le meilleur exemple d'intelligence collective est l'encyclopédie en ligne Wikipédia, mais on peut aussi citer le système de critique de livres d'Amazon, ou la base de données musicale CDDB. La blogosphère (le monde des blogs) est aussi un des piliers de l'intelligence collective : à tel point qu'elle commence à représenter une alternative à la presse traditionnelle.

Dans l'ouvrage *Comment le web change le monde*[2], Francis Pisani préfère le terme d'« alchimie des multitudes » afin de souligner que les synergies entre les contributions ne sont pas nécessairement constructives. Nous partageons cette vision, mais nous avons cependant conservé le terme d'intelligence collective car il est consensuel et compris du plus grand nombre.

Les plus grands contributeurs à cette intelligence sont issus de la jeune génération, les fameux « ***digital natives*** » ou « **génération Y** » pour qui l'usage de l'Internet est complètement naturel. Selon les définitions, ces utilisateurs « élevés dans le numérique » sont nés après la chute du mur de Berlin ou bien sont des moins de 30 ans. Toujours est-il qu'ils introduisent une rupture dans les entreprises par rapport

1. Linux, Apache, MySQL, PHP.
2. *Comment le web change le monde*, Francis Pisani, Dominique Piotet, Pearson, 2008.

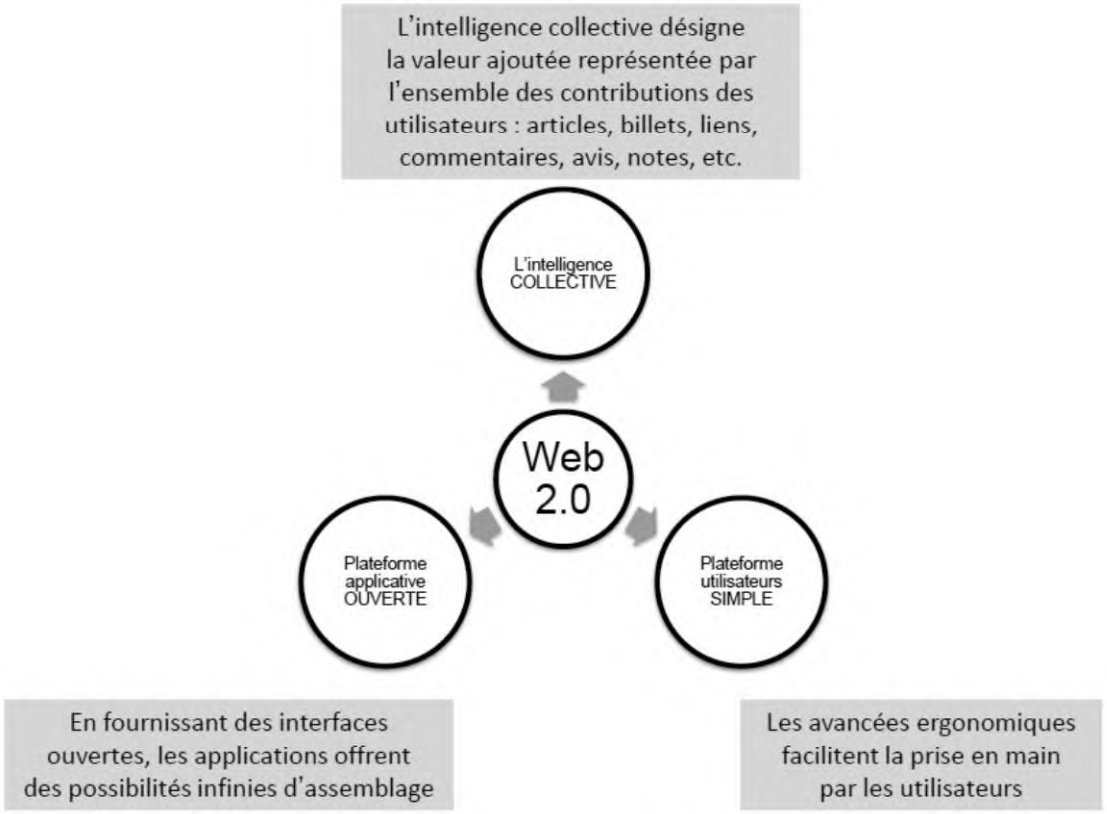

Figure 1.5 — Les concepts du web 2.0.

aux employés plus âgés, parfois appelés « ***analogists*** ». Les « *analogists* », réfractaires au numérique, ont conservé la culture du papier. On les caricature en disant qu'ils font imprimer leurs e-mails entrants par leur secrétaire et dictent leurs e-mails sortants.

Les outils associés à l'intelligence collective sont les blogs, les wikis, et plus largement les sites web qui incitent à la participation. Ils ont largement contribué à l'usage, par les « *digital natives* », d'applications web hébergées. Ces utilisateurs ont une telle habitude des espaces collaboratifs en ligne qu'ils vont naturellement pousser leur entreprise à utiliser des outils similaires, disponibles sous forme SaaS. **Ils seront donc les promoteurs du cloud, que les « *analogists* » verront d'un mauvais œil** (*cf.* chapitre 6 sur les craintes des utilisateurs).

Quelques exemples d'applications web 2.0

Blogs : Blogger, WordPress
Wikis : Wikipédia, JurisPedia, Ekopédia
Messagerie : Gmail.com, Outlook.com, Yahoo! Mail
Réseaux sociaux : Facebook, Twitter, LinkedIn
Partage de photos : Flickr, Google Photos
Partage de vidéo : YouTube, Dailymotion
Fonds cartographiques : Google Maps

1.6.2 Une plateforme utilisateurs

Le mouvement du web 2.0 s'est bâti sur des applications web plus ergonomiques, plus faciles à utiliser que les applications des générations précédentes. Cette ergonomie a été rendue possible par les technologies HTML5 introduites dans les paragraphes précédents.

La facilité de prise en main de la galaxie web 2.0 a permis une adoption massive par des utilisateurs non-informaticiens. Ainsi, ces sites ont vu leur fréquentation croître de manière très rapide, atteignant pour certains plusieurs centaines de millions d'utilisateurs.

Un des meilleurs exemples de cette amélioration ergonomique est la vidéo en ligne. Dans le passé, diffuser une vidéo sur Internet nécessitait des compétences pointues et un temps certain pour numériser, encoder et diffuser son contenu. Aujourd'hui, des plateformes comme Youtube rendent cette mise en ligne extrêmement simple.

Un autre exemple plutôt impressionnant est Google Sheets : ce service en HTML5 offre des fonctions de tableur très proches de celle de Microsoft Excel.

Ces applications orientées grand public, et souvent gratuites, ont développé une ergonomie telle que leur usage est devenu quotidien pour leurs utilisateurs, les « *digital natives* ». Et quand leurs entreprises leur proposent des applications moins ergonomiques, ils ont tendance à les délaisser pour celles qu'ils jugent meilleures. Ainsi, il est fréquent de voir de jeunes collaborateurs renvoyer leurs e-mails professionnels vers des messageries web (comme Gmail), ou de les voir stocker leurs documents sur des espaces en ligne (comme DropBox, ou Drive). Les entreprises ont bien du mal à empêcher ces pratiques parfois dommageables pour la sécurité de leurs données. On verra au chapitre 6 que les « *digital natives* » sont souvent des sponsors pour le cloud computing.

Certaines applications web 2.0 ont évolué d'un modèle grand public vers un modèle entreprise sous forme SaaS : on peut citer l'exemple de Writely, un petit traitement de texte grand public, qui a été racheté puis intégré à Google Apps, une offre SaaS aujourd'hui orientée vers les entreprises.

Ces services web 2.0 ont pris une telle importance dans le quotidien de leurs utilisateurs qu'ils sont devenus pour eux des applications critiques, à la manière des applications métiers pour les utilisateurs en entreprise. Ils ont donc dû assurer une qualité de service irréprochable, et les plateformes techniques des grands acteurs du web, comme Amazon, Yahoo! ou Google, sont devenues des modèles de performance et de robustesse.

Les applications web 2.0 ont donc créé un terreau fertile qui a fait naître les applications SaaS.

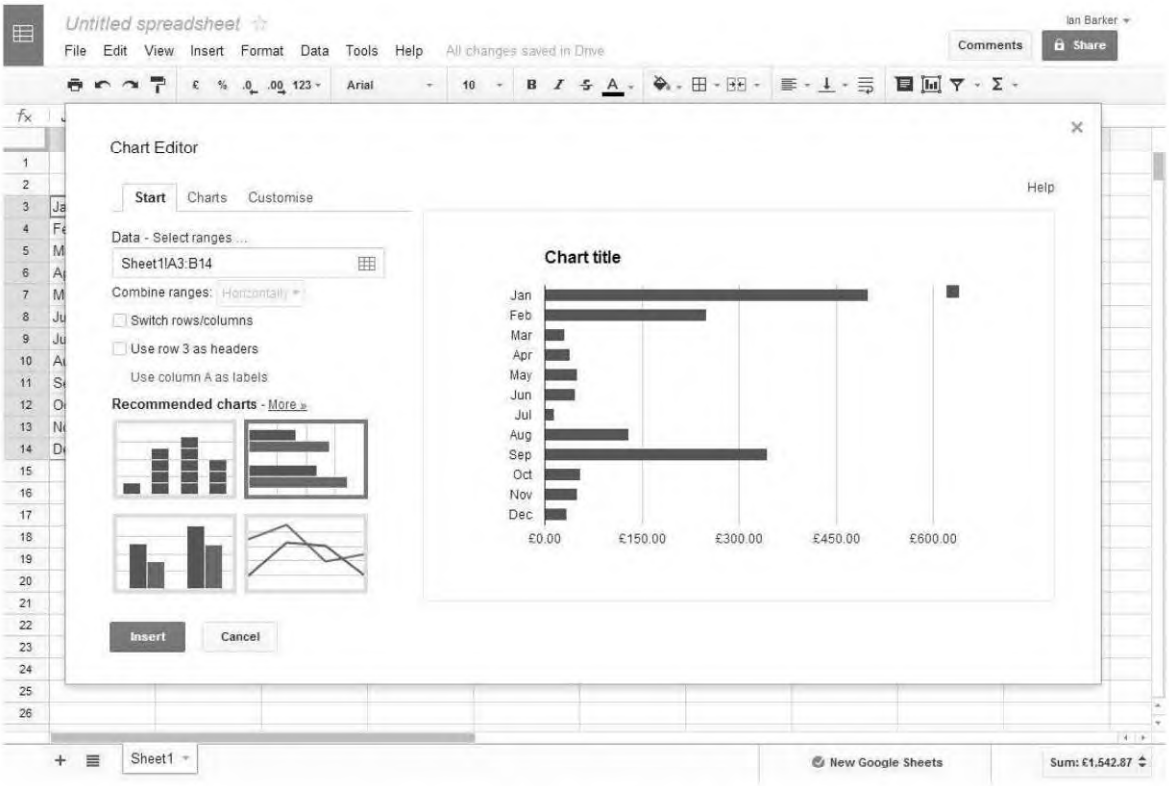

Figure 1.6 — Le tableur Google Sheets.

1.6.3 Des écosystèmes ouverts (OpenAPI)

La plupart des applications du monde web 2.0 mettent à disposition des **API ouvertes** (*Application Programming Interface*), des interfaces techniques qui permettent l'invocation de leurs services depuis d'autres applications.

Ces API sont ouvertes, publiques et utilisables par tous. Il est donc possible de créer des applications qui recourent à ces services.

Chez Amazon, la mise à disposition d'API est obligatoire...

En 2002, Jeff Bezos a donné sa vision des API dans un e-mail resté fameux :

1) All teams will henceforth expose their data and functionality through service interfaces.

2) Teams must communicate with each other through these interfaces.

3) There will be no other form of interprocess communication allowed: no direct linking, no direct reads of another team's data store, no shared-memory model, no back-doors whatsoever. The only communication allowed is via service interface calls over the network.

4) It doesn't matter what technology they use. HTTP, Corba, Pubsub, custom protocols — doesn't matter. Bezos doesn't care.

5) All service interfaces, without exception, must be designed from the ground up to be externalizable. That is to say, the team must plan and design to be able to expose the interface to developers in the outside world. No exceptions.

6) Anyone who doesn't do this will be fired.
7) Thank you; have a nice day!

Lorsque des applications sont bâties uniquement sur la base de ces API, on les appelle applications composites, ou, en anglais, ***mashups***. Elles sont construites par assemblage libre, à la manière des « legos ».

Le site housingmaps.com est un des exemples le plus connus de *mashup* : il fait appel à l'API de Craiglist, un site de petites annonces, et à l'API de Google Maps, une solution de cartographie en ligne. La résultante de cette application composite est une carte des petites annonces immobilières que l'on peut parcourir et agrandir selon ses besoins.

Ces API permettent à d'autres acteurs, entreprises ou développeurs indépendants, d'innover en les exploitant, et d'inventer de nouveaux modèles économiques. Ils peuvent ainsi construire des myriades d'applications : le site programmableweb.com référence d'ailleurs 14 000 API à ce jour (fin 2015). Cette démarche permet la constitution d'un écosystème fécond, tant pour l'entreprise qui met à disposition les API, que pour celles qui les exploitent. Cette mise à disposition permet en particulier de :

- **Créer des revenus directs, en les facturant**. Exemple : Google Maps devient payant pour plus de 1 million de transactions/an.
- **Étendre une communauté, et donc recruter des utilisateurs**. Exemple : grâce aux applications dérivées de sa plateforme, Twitter a dépassé les 300 millions d'utilisateurs et fait figure de géant du web.
- **Faire émerger de nouveaux usages pour sa plateforme et donc faire évoluer son modèle de revenu.** Exemple : en 2009, Apple a constaté que les développeurs d'applications souhaitaient vendre non seulement des applications, mais aussi des contenus pour leurs applications. Le modèle de l'AppStore a donc évolué pour intégrer cette possibilité.
- Parfois, **externaliser leur R&D**, puis racheter les start-ups les plus talentueuses. C'est ce qu'a fait Salesforce avec Financialforce.com.

Il existe aujourd'hui des start-ups qui ont bâti leur modèle économique en recourant à ces API. On peut citer TweetDeck (racheté depuis par Twitter) ou Hootsuite qui reposent quasiment entièrement sur l'API Twitter. On peut aussi citer Zynga, un éditeur de jeux reposant sur les API Facebook.

On verra dans la suite que ce principe a largement inspiré les acteurs du cloud. Les API de leurs plateformes fournissent une solution simple pour les entreprises qui souhaitent intégrer leurs applications existantes avec le cloud (*cf.* chapitre 19).

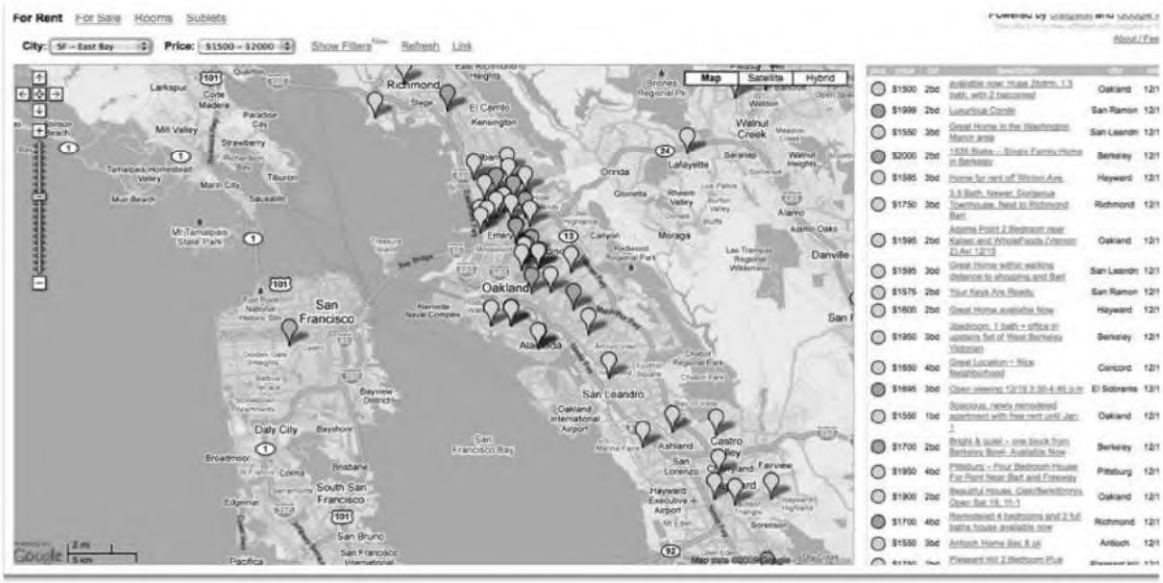

Figure 1.7 — Le principe des mashups avec l'exemple d'HousingMaps.

1.6.4 Un principe d'évolution continue

On ne peut pas terminer la présentation des principes du web 2.0 sans présenter le principe de la « **bêta perpétuelle** ».

Les applications web 2.0 évoluent suivant un cycle différent de celui des logiciels classiques avec des versions bien identifiées (*cf.* Windows XP, Windows Vista). Elles sont mises à jour en continu par leurs hébergeurs sans que les utilisateurs soient informés de l'existence d'une nouvelle version. Les nouvelles fonctionnalités apparaissent au fil de l'eau, et elles sont découvertes par hasard par les usagers. Ce mode de fonctionnement est satisfaisant pour les utilisateurs qui aiment la nouveauté et ne sont pas récalcitrants à s'adapter à des interfaces en changement perpétuel.

Le terme « bêta perpétuelle » désigne le fait que l'application n'est jamais finalisée, mais toujours en évolution : il n'y a pas de livraison de nouvelle version à proprement parler. L'enjeu de ce mode de fonctionnement est bien entendu de maintenir la stabilité de l'application tout au long de ses évolutions incrémentales.

On verra dans les chapitres suivants que ce modèle est largement repris par les acteurs du cloud.

1.7 LE PRINCIPE DES SERVICES « DEVICE AGNOSTIC »

Il y a quelques années, l'accès au monde informatique se faisait principalement par l'intermédiaire d'un ordinateur sous Windows, Mac OS ou plus rarement sous Linux.

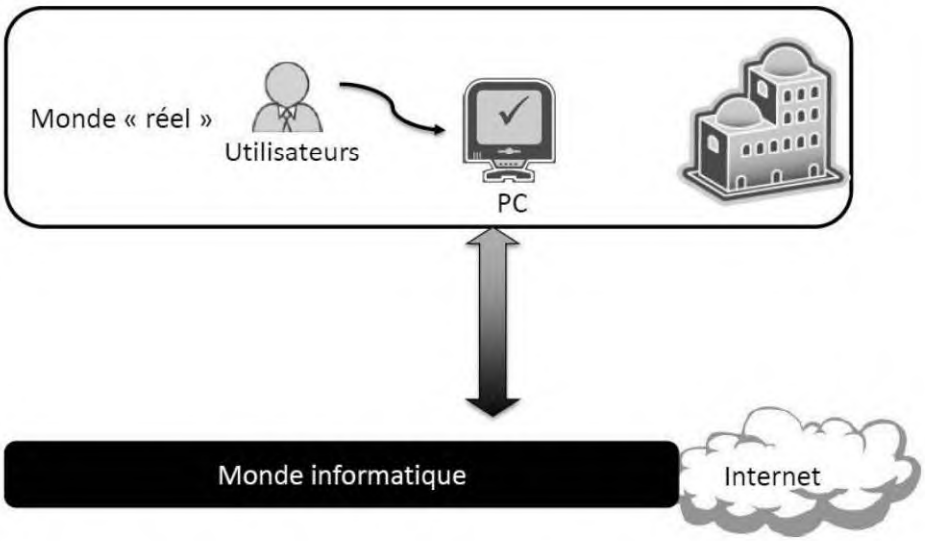

Figure 1.8 — L'accès à l'informatique aujourd'hui.

Depuis l'apparition du Palm Pilot en 1996, les appareils mobiles ont beaucoup évolué. Ils ont connu diverses appellations comme PDA (*Personal Digital Assistant*), smartphones, tablettes, etc. Ils se sont dotés de capacités de communication de plus en plus sophistiquées.

En 2015, on peut classer ces produits dans les catégories suivantes :

- **les smartphones** : ce sont des téléphones intelligents utilisant les systèmes iOS, Android, Windows Phone, etc. Leur prix et leur équipement sont très variables. Ils sont capables de se connecter à Internet. Parmi leurs périphériques, on peut trouver des appareils photo numériques, des caméras, des écrans tactiles, des accéléromètres, des systèmes de localisation GPS[1], etc. Le plus emblématique est l'iPhone d'Apple ;
- **les tablettes** : ce sont des interfaces tactiles, basées sur les mêmes systèmes que les smartphones, munies d'un écran de la taille d'un PC. Elles sont adaptées à un usage mobile[2] en station debout ou assis dans un canapé. Elles embarquent généralement des applications qui offrent une interface tactile à des services en ligne.
- **les « chromebooks »** : ces appareils proposés par Google ressemblent fortement à des ordinateurs portables. Mais ils ne disposent que d'un navigateur web pour accéder aux applications ;
- **les liseuses** : ce sont des appareils dotés d'un écran de très grande finesse, destinés à la lecture en situation nomade. Le plus connu d'entre eux est le Kindle d'Amazon.

1. Le GPS (*Global Positioning System*) est une technologie de géolocalisation basée sur des satellites en orbite autour de la Terre.

2. « Mobile » désigne un usage en train de marcher ou de rouler, tandis que « nomade » signifie que l'on est hors de son lieu de travail habituel, mais en position assise (*cf.* gare, aéroport).

Les acteurs du web 2.0 considèrent qu'il est de leur devoir de permettre un accès à leurs services depuis la plupart de ces appareils. On dit qu'ils développent des applications « *device agnostic* ». Ce caractère agnostique est assuré parfois par des applications web, parfois par des interfaces natives issues des différents AppStores pour les mobiles.

Tous ces types d'appareils mobiles savent se connecter à Internet. En revanche, ils disposent :

- **d'ergonomie très variable** : tailles d'écran, modes de saisie, robustesse des batteries, etc.
- **de socles techniques très divers** : iOS, Android, Linux, Windows, etc. ;
- **de capacités de stockage très variables** ;
- **de rapidités d'accès à Internet très variables** : EDGE[1] (200 Kb/s), 3G+ (2 Mb/s), WiFi (100 Mb/s), etc.

Par conséquent, il est très difficile de lancer des traitements gourmands en ressource sur ces appareils. C'est l'une des raisons pour lesquelles on privilégie les applications accédant à des services hébergés pour les appareils mobiles. L'interface peut être embarquée sur le mobile (*cf.* l'AppStore d'Apple), mais les données sont presque toujours hébergées.

Le mode hébergé assure aussi l'intégrité des données, mal protégées par les petits appareils, sujets à des vols, à de la casse, ou à des pannes de batteries entraînant la perte de données.

Par ailleurs, les utilisateurs d'appareils mobiles utilisent généralement en parallèle un PC classique, et ces deux appareils doivent accéder aux mêmes informations. Cet accès concurrent est grandement facilité lorsque les applications et les données sont sur le web.

La montée en puissance des appareils mobiles renforce donc la pertinence des applications hébergées, et du cloud computing.

1.8 LE CLOUD COMPUTING : UNE CAPITALISATION SUR TOUTES LES ÉVOLUTIONS PRÉCÉDENTES

Le cloud computing va bien au-delà de la synthèse des mouvements précédents : nous allons présenter ses autres bénéfices dans les chapitres suivants.

1. L'EDGE (*Enhanced Data Rates for* GSM *Evolution*) est une extension du protocole GSM, permettant des débits de transfert plus élevés.

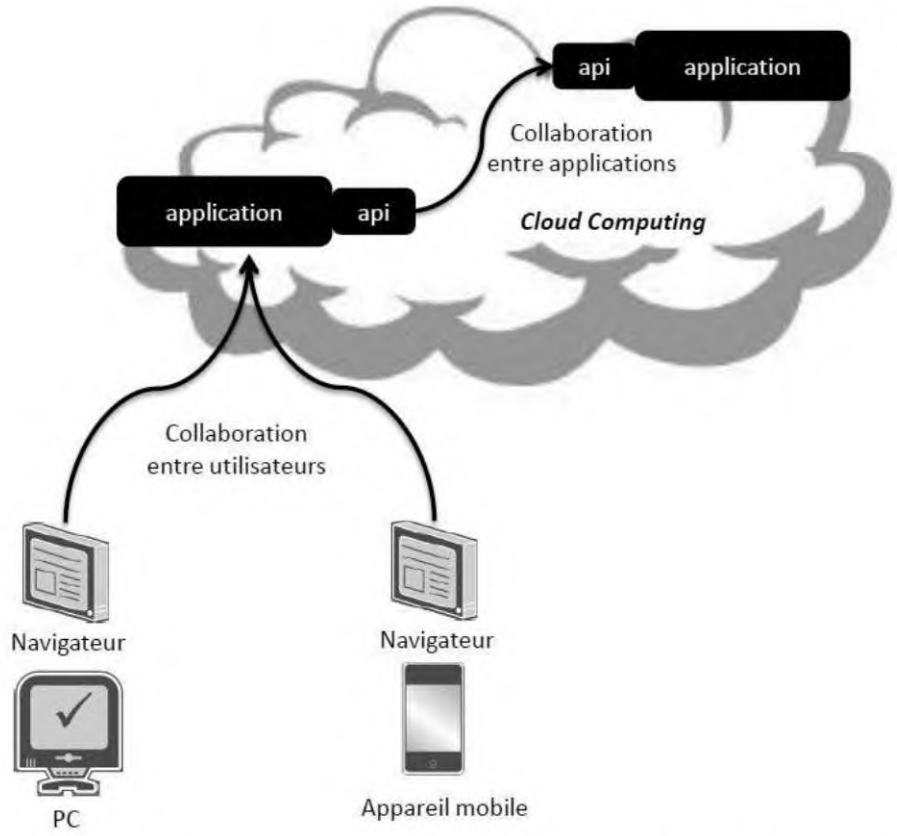

Figure 1.9 — Le cloud computing comme synthèse des évolutions précédentes.

En résumé

En proposant l'hébergement des services sur des plateformes accessibles depuis l'Internet, le cloud computing est l'aboutissement de l'ensemble des mouvements précédents :

– les applications SaaS sont dans la logique de l'évolution des sites web statiques vers les sites applicatifs transactionnels et de l'ouverture des entreprises sur Internet, et du web 2.0. Leur première déclinaison était l'ASP, qui a connu un demi-échec ;

– en parallèle, on peut considérer les hébergeurs avec portail de *Self Service* comme les ancêtres des plateformes PaaS/IaaS.

2

Concepts et définitions du cloud computing

Objectif

Nous avons mentionné à plusieurs reprises les terminologies *cloud computing*, *Software as a Service*, *Platform as a Service*, *Infrastructure as a Service* dans le chapitre précédent.

Il est temps de définir précisément ce que désignent ces concepts : c'est l'objectif de ce chapitre. Il met le modèle cloud en perspective par rapport au classique modèle *software*. Il présente aussi les bénéfices du cloud pour les éditeurs, et le concept de cloud privé.

2.1 ORIGINE DU TERME CLOUD COMPUTING

Le terme cloud computing est traduit littéralement par « informatique dans les nuages », ces nuages faisant référence à Internet et au web.

Pour bien comprendre cette terminologie, il faut rappeler qu'Internet est un réseau très complexe et difficile à appréhender car constitué de millions de connexions utilisant des technologies très disparates (fibre optique, câble, ADSL, etc.). Le réseau repose sur un maillage mondial sophistiqué, avec de nombreuses redondances qui permettent à une requête de changer de chemin si une partie du réseau est inopérante ou congestionnée. Enfin, Internet est géré par des milliers d'organisations publiques et privées différentes.

Ainsi, lorsqu'on accède à une application web comme Amazon, on n'a pas la moindre idée de son emplacement physique (peut-être en Californie, mais où précisément, difficile à dire...). On n'a pas non plus la moindre idée du chemin qu'emprunte une requête pour parvenir jusqu'à l'application (la demande a pu passer par des câbles transatlantiques ou par la Russie et le Japon...) Ainsi, le monde de l'Internet est complètement abstrait pour la plupart des utilisateurs : il n'a pas de réalité géographique tangible. L'application de cloud computing que nous utilisons peut se trouver à San Francisco, dans un satellite ou même sur la Lune : cela fait finalement peu de différence pour nous. Les nuages du cloud computing font référence à cette abstraction. Ils font aussi référence au fait que l'on représente souvent Internet sous la forme d'un nuage dans les schémas informatiques.

> Le cloud computing signifie donc que les ressources en ligne sont utilisées comme si elles étaient situées dans l'éther, dans un espace sans réalité physique. Certains acteurs du monde du cloud computing jouent d'ailleurs sur cette immatérialité : ainsi Google entretient un certain mystère autour de l'emplacement de ses centres de données ou Datacenters.

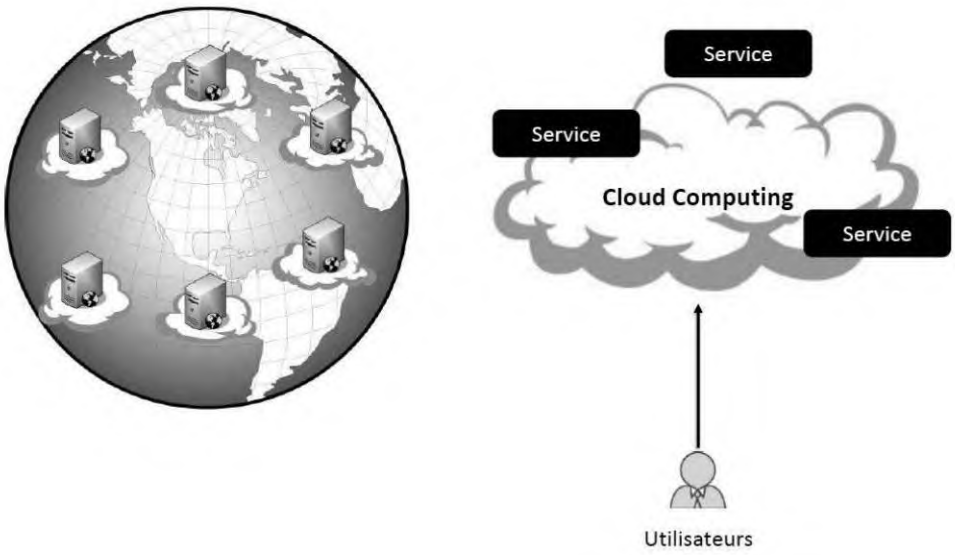

Figure 2.1 — Illustration du cloud computing.

Pour certains, la définition du cloud computing est : « ***Computing at Internet Scale*** ». On peut le traduire en français par « *usage d'une plateforme informatique à l'échelle de l'Internet* ». L'idée derrière cette expression est que, avec le cloud computing, on passe d'une salle machine à l'échelle de l'entreprise, à un réseau de milliers de serveurs distribués sur le globe. Par exemple, en utilisant la plateforme Google, on dispose de millions de machines réparties sur les cinq continents. Une plateforme de cette dimension n'a pas les mêmes propriétés, ni le même mode de fonctionnement qu'une plateforme d'entreprise : nous reviendrons sur ce point dans la partie 5.

2.2 UNE DÉFINITION PLUS PRAGMATIQUE

Sur un plan plus technique, on peut considérer que le cloud computing est une évolution des technologies de virtualisation. La virtualisation permet de donner plus d'agilité aux centres de données, grâce aux trois propriétés suivantes :

- **mutualisation des ressources :** la virtualisation permet d'affecter les ressources d'une même machine à plusieurs applications ;
- **abstraction sur la localisation :** l'application est « quelque part » sur l'une des machines constitutives de la plateforme de virtualisation. Si cette plateforme utilise des mécanismes de réplication sur des Datacenters distants, les risques de désastre (incendies, inondations) sont couverts par la distribution multisites ;
- **élasticité :** il est possible d'allouer des ressources supplémentaires à une application proche de la saturation, dans les limites physiques de la plateforme. Cette propriété est particulièrement importante : en effet, si la plateforme dispose de grandes ressources de puissance inutilisées, on peut affecter en quelques instants des capacités supplémentaires à une application. Elle permet aussi d'optimiser l'usage des ressources, en évitant le syndrome de la machine utilisée à 20 % de ses capacités (un cas classique avec une machine hébergeant un serveur HTTP).

Le cloud computing reprend ces propriétés, mais à une plus grande échelle :

- dans le cadre des plateformes de cloud computing publiques (Google, Amazon, etc.), la mutualisation de ressources se fait à l'échelle de plusieurs milliers d'entreprises. On dispose donc de bénéfices liés au facteur d'échelle ;
- l'abstraction sur la localisation est à l'échelle de plusieurs continents dans le cadre des clouds publics : la garantie sur l'intégrité des données est donc supérieure à celle d'un centre de données utilisant deux sites distants de quelques kilomètres. On retrouve ici le caractère ubiquitaire des ressources évoquées dans le paragraphe précédent ;
- avec des plateformes de plusieurs dizaines milliers de serveurs, les clouds publics proposent une réserve de puissance et donc une élasticité exceptionnelle.

Le cloud computing ajoute d'autres propriétés à celles de la virtualisation :

- **le *Pay As You Go* :** les utilisateurs paient les ressources qu'ils utilisent en fonction de leur consommation réelle et précise. Peu de DSI savent aujourd'hui mesurer précisément la consommation informatique de telle ou telle application. Les acteurs du cloud savent le faire ;
- **le *Self Service*** : l'équipe de développement peut demander l'allocation de ressources *via* un portail web. Ces ressources seront mises à sa disposition de manière automatique quelques minutes plus tard ;
- **les API ouvertes** (*cf.* chapitre 1 sur le web 2.0) : les plateformes cloud proposent des interfaces techniques accessibles à distance qui permettent de les intégrer avec le système d'information ou bien de piloter les services à distance.

Nous proposons donc la définition suivante[1] :
Cloud computing = virtualisation + *Pay As You Go* + *Self Service* + API ouvertes
Cette définition s'applique aux clouds publics (Google, Amazon, etc.) comme aux clouds privés (gérés en interne). Cet ouvrage s'intéresse avant tout aux clouds publics ; nous reviendrons néanmoins sur les *cloîuds* privés au chapitre 3.

Sur la base de cette définition, le cloud offre deux grandes familles de services :

- des services de fourniture d'application en location, appelés **SaaS**. Ces services sont généralement facturés au nombre d'utilisateurs actifs ;
- des services techniques de plateforme d'exécution en location, appelés **PaaS et IaaS**. Ces services sont facturés selon les ressources techniques consommées.

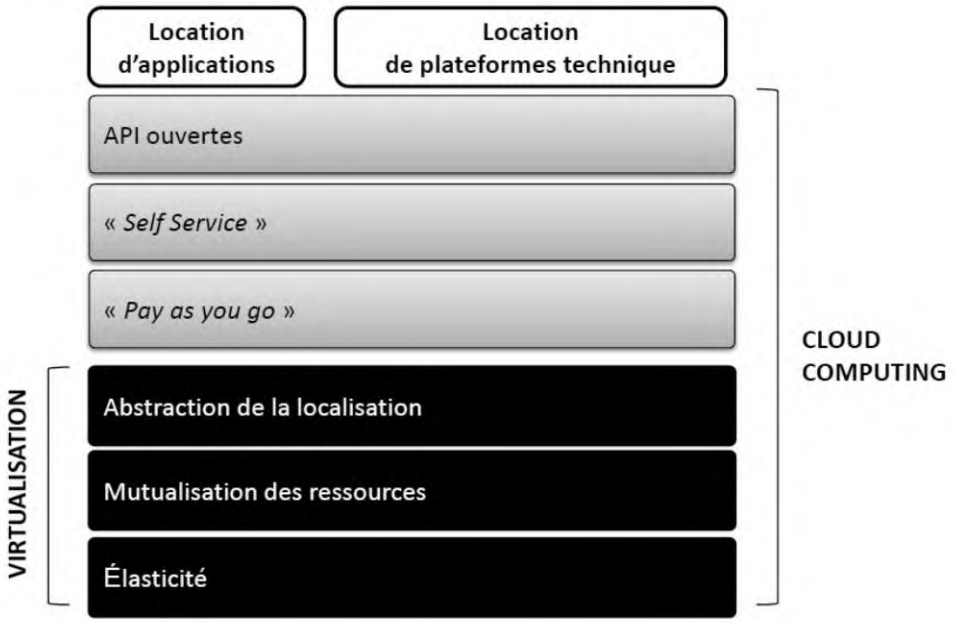

Figure 2.2 — Une définition pragmatique du cloud computing.

Le concept de cloud computing englobe les concepts de *Software as a Service* (SaaS), de *Platform as a Service* (PaaS), et d'*Infrastructure as a Service* (IaaS) que nous allons présenter dans la suite.
Le terme *as a Service* évoque bien un service, dans le sens où le fournisseur vend une fonction opérationnelle, et non des composants techniques nécessitant une compétence informatique. Pour utiliser une métaphore, il est évident pour tout le monde que le service attendu d'un restaurant est un repas et non la mise à disposition de légumes et de viande crus...

1. Le NIST américain (*National Institute of Standards and Technology*) donne une définition officielle du cloud à cette adresse : http://www.nist.gov/itl/cloud/.

Par ailleurs, le terme service évoque les architectures orientées services (SOA). Nous reviendrons dans le chapitre 18 sur les liens entre cloud et SOA.

2.3 QUE SIGNIFIE SAAS ?

SaaS signifie *Software as a Service*, c'est-à-dire un logiciel fourni sous la forme de service. Il s'agit donc de location d'application opérationnelle, clef en main, et non d'achat de logiciel informatique, à installer soi-même sur une machine. Les SaaS s'adressent donc aux utilisateurs finaux.

La différence entre SaaS et logiciel est essentielle. En effet, les SaaS proposent des logiciels opérationnels, prêts à l'emploi, sans passer par une étape d'installation, et sans aucune tâche de maintenance.

Les SaaS sont exécutés sur des plateformes conçues pour une utilisation simultanée par un grand nombre de collaborateurs qui travaillent dans de nombreuses entreprises différentes. Ces plateformes sont mises à disposition par des acteurs (comme Google ou Salesforce) que nous appellerons *opérateurs SaaS*, car leur métier est plus proche de ceux des opérateurs télécoms que de celui des éditeurs de logiciel. C'est la raison pour laquelle **dans le modèle *cloud*, on ne parle plus d'éditeur, mais d'opérateur de service.**

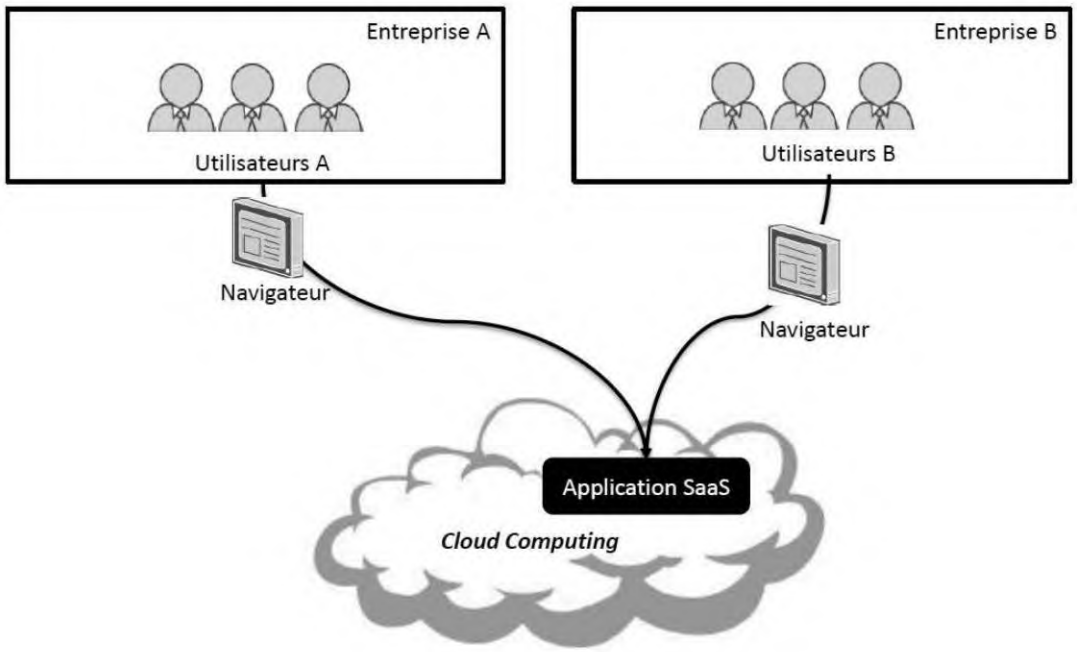

Figure 2.3 — Vision générale des SaaS.

Le canal d'accès aux applications SaaS étant le réseau Internet, on les appelle aussi « **applications en ligne** ». Dans cet ouvrage, nous utilisons indifféremment les terminologies *SaaS* et *applications en ligne*. Nous les considérons comme des synonymes.

Comme on l'a vu dans les chapitres précédents, les SaaS sont les successeurs des ASP. Ils se distinguent de ces dernières par :

- **l'usage d'interfaces HTML5** (*cf.* chapitre 1) : leurs interfaces sont exécutées dans un navigateur web ; elles n'ont aucune adhérence avec le poste de travail ; la problématique de déploiement de client natif est donc contournée ;
- **le *Pay As You Go* :** en général, les SaaS sont facturés à l'entreprise selon le nombre d'utilisateurs sur une base mensuelle ;
- **le *Self Service* :** la souscription en ligne est possible par des profils non-informaticiens, et l'activation des services est quasi immédiate. **La fourniture d'API ouvertes** : les SaaS ont fortement intégré le principe du web 2.0 ; ils fournissent des API permettant de faire appel à leurs fonctionnalités ; l'intégration entre Salesforce et Google Apps en est un exemple remarquable ; l'utilisateur retrouve ses documents au sein des écrans de CRM de Salesforce ; on verra dans le chapitre 17 que cette intégration entre SaaS peut aller jusqu'à reproduire un système d'exploitation muni de ses applications ;
- **la mise en avant de fonctions collaboratives :** les SaaS mettent l'accent sur les pratiques collaboratives héritées du web 2.0 ; comme on le verra dans la suite, on retrouve ces pratiques entre utilisateurs, mais aussi entre développeurs de SaaS ;
- des architectures spécifiques dites « **multi-tenants** » (*cf.* chapitre 18) dédiées et optimisées pour un usage en ligne : les applications SaaS sont liées à l'environnement de l'opérateur et ne peuvent pas être « déménagées » simplement sur un serveur Windows ou Linux en entreprise.

Les SaaS reprennent donc le modèle ASP dans les grandes lignes, mais ils le rendent plus viable : ils font disparaître les freins techniques qui avaient limité la portée du modèle précédent.

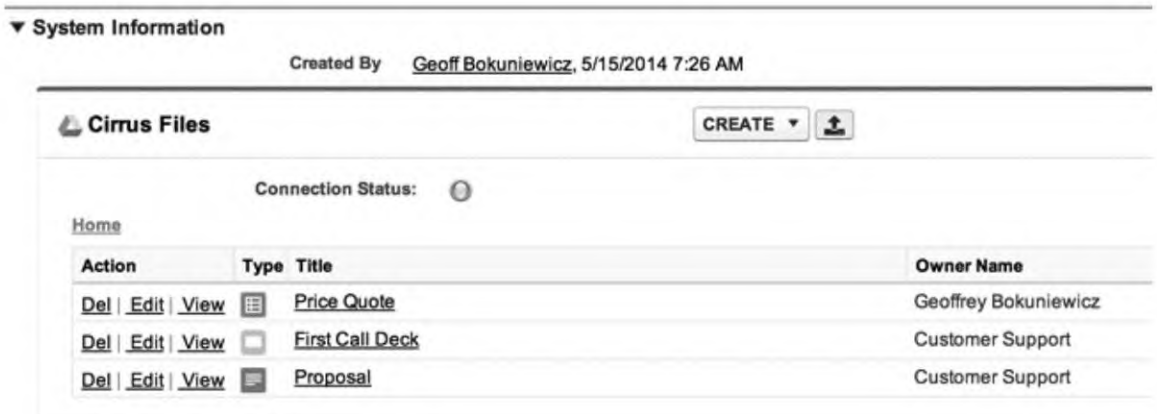

Figure 2.4 — L'intégration de Salesforce et Google Docs.

2.4 QUE SIGNIFIE PAAS ?

PaaS signifie *Platform as a Service* ou plateforme sous forme de service. Il s'agit de location de plateforme technique, permettant l'exécution de code développé en spécifique. Les PaaS s'adressent donc aux développeurs.

2.4.1 Définition

PaaS désigne une plateforme d'exécution hébergée par un opérateur et accédée depuis Internet. Cette plateforme peut être utilisée pour exécuter des sites web, des SaaS, ou tout développement spécifique issu de l'entreprise. Ces développements spécifiques doivent respecter le langage de développement et l'architecture de la plateforme PaaS. Ces contraintes sont à rapprocher de celles des plateformes LAMP en *Self Services* évoquées au chapitre 1.

Dans l'approche PaaS, l'opérateur ne fournit pas seulement un environnement d'exécution déporté, mais aussi un ensemble de services d'infrastructures. La plateforme PaaS propose ainsi :

- **un portail de *Self Service*** : pour souscrire au service, administrer et surveiller son application ;
- **un service d'exécution d'applications** : qui permet d'exécuter des applications écrites dans les langages autorisés par la plateforme, et un **service de persistance de données** qui permet de stocker des données structurées ou des fichiers ;
- **le *Pay As You Go*** : en général, les PaaS sont facturées à l'entreprise selon la consommation de CPU, réseau (bande passante, volume, etc.) et espace disque ;
- **des API ouvertes** : elles permettent l'intégration de l'application hébergée sur la PaaS avec le SI, ainsi que sa surveillance ;
- **des architectures « multi-tenants »** (*cf.* chapitre 17), dédiées à un usage en ligne : comme les SaaS, les PaaS sont liées à l'environnement de l'opérateur et ne peuvent pas être « déménagées » simplement sur un serveur Windows ou Linux en entreprise.

Nous reviendrons dans le chapitre 19 sur une description précise de l'ensemble de ces services. Dans cette partie introductive, il est important de retenir que les PaaS permettent à un développeur de mettre rapidement en ligne un développement, et ce, de manière totalement autonome, sans le support d'une équipe d'exploitation. En effet, la plateforme PaaS lui fournit une extension de son environnement de développement qui lui permet d'effectuer une mise en ligne par simple clic sur un bouton.

2.4.2 PaaS *versus* IaaS

Le PaaS est l'option qui permet la mise en ligne la plus rapide pour une petite start-up qui souhaite mettre une application web sur le marché. Le développeur se focalise sur le code et ne mobilise pas de temps pour installer matériel et logiciel.

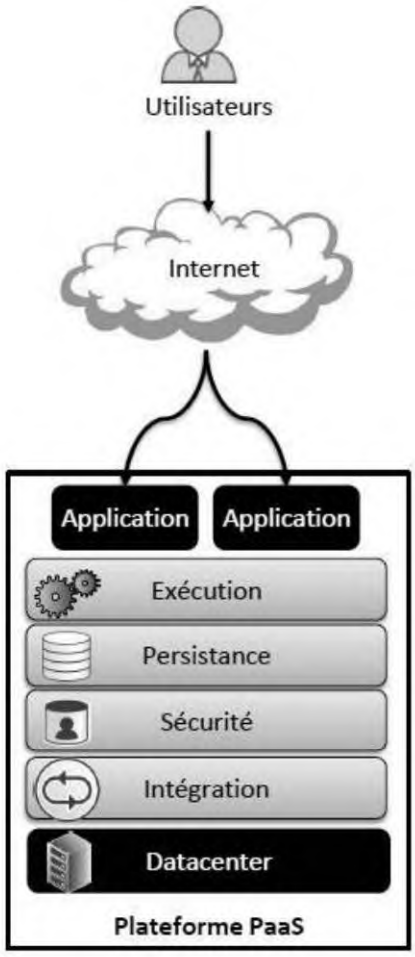

Figure 2.5 — Les plateformes PaaS.

Par ailleurs, le PaaS est facturé à la consommation processeur réelle, contrairement à l'IaaS, où l'on paie à l'« uptime », c'est-à-dire tant que le service est disponible. Prenons, par exemple, une application qui ne connaît aucun utilisateur la nuit, et qu'on laisse fonctionner au cas où : elle ne coûtera rien en PaaS, mais coûtera en IaaS.

La plateforme PaaS offre une grande souplesse : chaque service est proposé de manière unitaire, et chacun d'entre eux est accessible *via* une API, selon les principes évoqués plus haut.

2.5 QUE SIGNIFIE IAAS ?

IaaS signifie *Infrastructure as a Service* ou infrastructure sous forme de service. Il s'agit de location de plateforme technique, permettant l'exécution d'architectures applicatives complètes, comprenant base de donnés, serveur d'application, etc. Les IaaS s'adressent donc aux équipes d'exploitation.

2.5.1 Définition

IaaS désigne une plateforme d'hébergement exploitée par un opérateur et accessible depuis Internet.

Cette plateforme peut être utilisée pour exécuter des SaaS.

Elle peut aussi être mise à la disposition des entreprises qui souhaitent faire héberger toute application : progiciel acheté auprès d'un éditeur ou développement spécifique.

En effet, on a toute latitude sur l'architecture à déployer sur une plateforme IaaS : base de données, serveur d'application, annuaire de sécurité, etc. Cette liberté a une contrepartie : le déploiement sur IaaS nécessite des compétences sur l'administration de systèmes d'exploitation et de serveurs.

L'IaaS s'adresse donc à des administrateurs et non à des développeurs.

2.5.2 IaaS *versus* PaaS

L'IaaS ne permet pas un aussi bon *Time to market* que le PaaS pour déployer une nouvelle application. En revanche, il autorise des architectures atypiques pas forcément disponibles sur les plateformes PaaS du marché. Une application qui évolue et se complexifie peut donc être amenée à migrer de PaaS vers IaaS.

La plateforme IaaS fournit les services suivants :

- **un portail de *Self Service* en RIA :** pour souscrire au service, administrer et surveiller son application ;
- **un hyperviseur :** pour exécuter des machines virtuelles à l'image des solutions de virtualisation comme VMware ou HyperV. Elle fournit aussi une solution de stockage et de « Snapshot » permettant de sauvegarder l'état des machines virtuelles en cas de redémarrage ;
- **le *Pay As You Go* :** en général, les IaaS sont facturées à l'entreprise selon la consommation de CPU, réseau et espace disque ;
- **des API ouvertes :** elles permettent l'administration distante et la surveillance des applications hébergées sur l'IaaS.

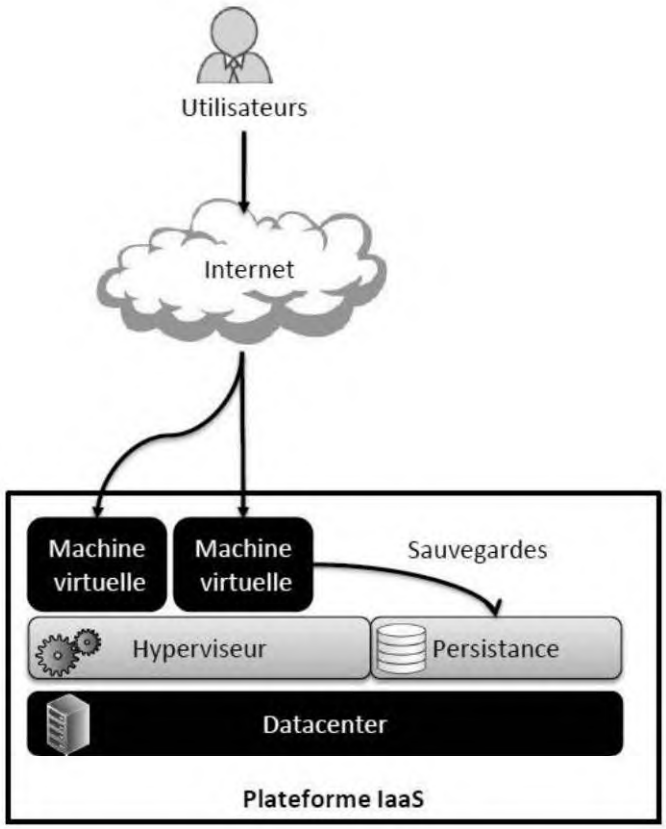

Figure 2.6 — Les infrastructures IaaS.

Amazon a été précurseur dans ce domaine avec *Amazon Web Services* (AWS). Le commerçant en ligne loue depuis quelques années une partie de ses capacités à des entreprises intéressées. AWS est ainsi devenu la première plateforme IaaS, permettant de déporter des applications dans les nuages[1]. Nous présenterons cette plateforme en détail dans le chapitre 20.

Tableau 2.1 — Services gérés par les XaaS.

Plateforme	Interne	IaaS	PaaS	SaaS
Applications	-	-	-	☑
Environnement exécution	-	-	☑	☑
Base de données	-	-	☑	☑
Système d'exploitation	-	-	☑	☑
Hyperviseur	-	☑	☑	☑
Machines	-	☑	☑	☑
Réseaux	-	☑	☑	☑

1. À noter : Amazon propose des services IaaS, mais aussi des services PaaS, ce qui peut créer une ambiguïté.

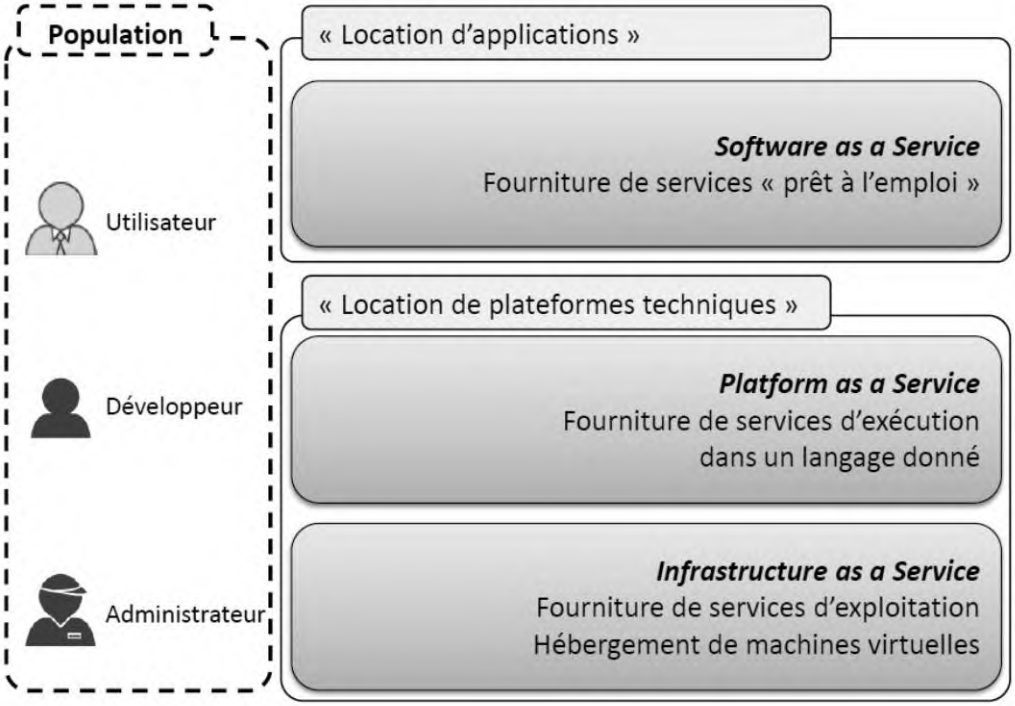

Figure 2.7 — Synthèse sur usagers des XaaS.

2.6 CLOUD COMPUTING *VERSUS* PLATEFORMES D'ENTREPRISE

Les entreprises exploitent depuis toujours deux types d'applications sur leur plateforme d'exploitation :

- **des progiciels pour des services assez standardisés,** parfois appelés services de commodité, par exemple les progiciels intégrés (SAP), la messagerie (IBM Lotus, Microsoft Exchange), etc.
- **des développements adressant des besoins spécifiques,** le plus souvent des applications métiers. Les environnements les plus utilisés dans ce contexte sont aujourd'hui JEE (*Java Enterprise Edition*) et Microsoft .NET[1].

On peut faire un parallèle entre ces plateformes d'entreprise et le cloud computing. En effet, les SaaS sont des progiciels assez standardisés, hébergés « sur le *cloud* ». Parmi les exemples les plus aboutis de SaaS à ce jour, on peut citer Salesforce, pour les progiciels intégrés, et Google Apps, pour la collaboration.

Les IaaS/PaaS sont des plateformes d'exécution hébergées « sur le cloud ». Nous présenterons dans cet ouvrage celles d'Amazon, Salesforce, Google et Microsoft.

1. Par le passé, les développements spécifiques étaient hébergés sur mainframes. Notons que les applications mainframes ne peuvent pas être migrées simplement vers le cloud.

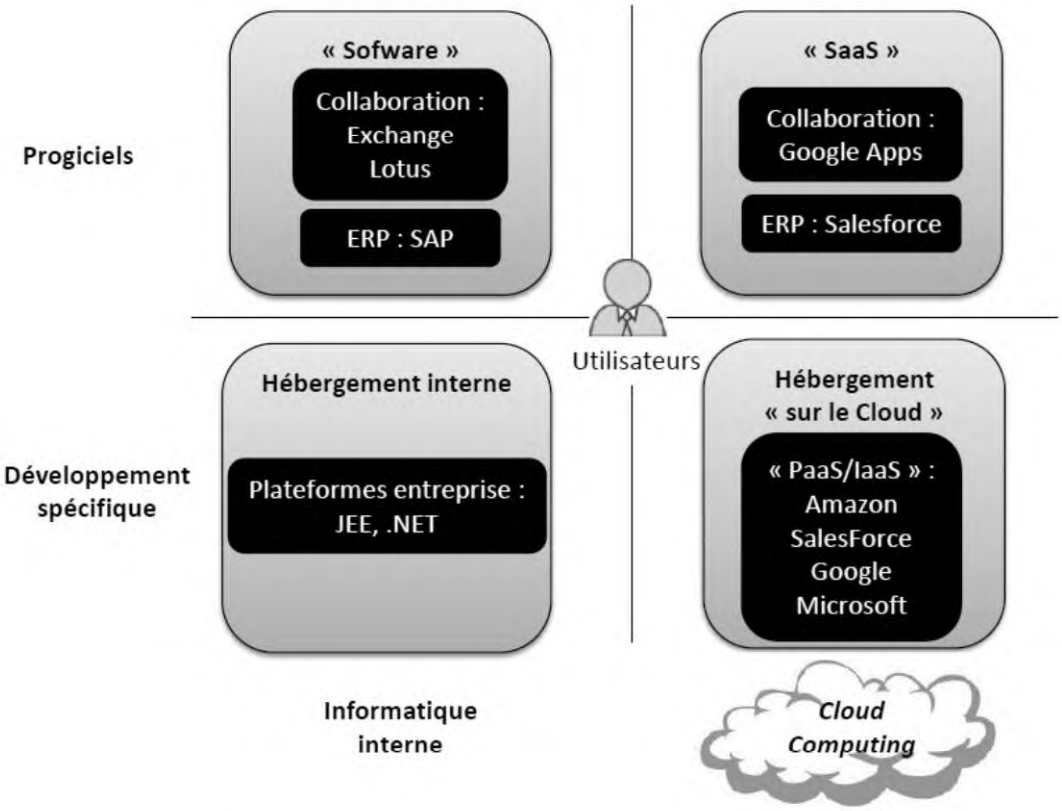

Figure 2.8 — Informatique interne *versus* cloud computing.

2.7 LE CLOUD, UNE ÉVOLUTION LOGIQUE DE L'INFORMATIQUE

2.7.1 Vers l'externalisation des ressources

Le cloud s'inscrit dans une **tendance générale de « virtualisation des ressources »**. En effet, les entreprises possèdent de moins en moins de biens matériels (locaux, flottes de véhicules, matériels informatiques, etc.) et capitalisent plutôt sur leurs données et leurs connaissances métiers.

Si l'on porte la métaphore au-delà du monde des entreprises, dans la sphère personnelle, on constate qu'il est souvent plus intéressant aujourd'hui de :

- louer une voiture à l'heure ou à la semaine, plutôt que de la posséder (*cf.* Uber, Drivy, BlaBlaCar) ;
- emprunter des livres en bibliothèque, plutôt que de les posséder (*cf.* Amazon Kindle Premium) ;
- louer des films sur des plateformes de vidéos à la demande, plutôt que de les posséder (*cf.* iTunes, Netflix, Arte VOD) ;
- etc.

Le recours au cloud s'inscrit donc dans une tendance générale de mutation des modes de consommation : de la possession vers le droit d'usage.

2.7.2 Vers l'ouverture des entreprises sur le web

Le cloud s'inscrit dans la **continuité de l'évolution des technologies du web** :

- il capitalise sur les technologies du web (HTTP/HTML), en particulier les nouvelles technologies d'interface HTML5 ;
- il capitalise sur les pratiques de paiement et de sécurité issues du commerce électronique (*Pay As You Go*) ;
- il capitalise sur l'autonomisation des utilisateurs qui font eux-mêmes leurs opérations sur les sites, comme ceux de banque en ligne (*Self Service*) ;
- il capitalise sur les pratiques ouvertes issues du web 2.0.

Le cloud est donc l'aboutissement de l'évolution du web depuis 15 ans.

Le modèle s'inscrit dans la **continuité de l'adoption du web par les entreprises.** En effet, les entreprises ont commencé par utiliser Internet à des fins de communication. Puis le réseau a intégré leur processus métiers et leur a permis d'échanger avec leurs clients et partenaires, en ouvrant de nouveaux canaux de vente et en réduisant leurs coûts. Enfin, Internet va leur permettre de déporter leur informatique, en ouvrant à nouveau des perspectives de réduction de coûts et en gagnant en performance et en robustesse.

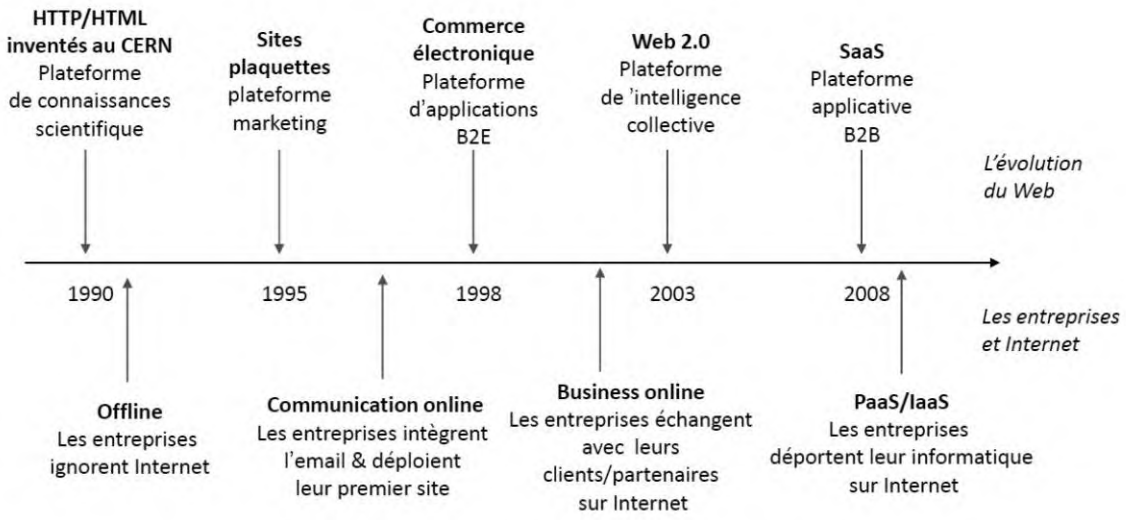

Figure 2.9 — Le cloud s'inscrit dans l'évolution du web.

En résumé

Le cloud computing désigne une informatique externalisée vers l'Internet. Il offre des possibilités de location d'applications et de plateformes techniques.

Il repose sur la virtualisation, le *Pay As You Go*, le *Self Service*, et les API ouvertes.

Il englobe les concepts de SaaS, PaaS et IaaS.

Il constitue une évolution logique de l'ouverture des entreprises vers l'Internet.

3

Le cloud : un nouveau modèle de consommation de services

Objectif

Ce chapitre présente l'historique des modèles pour la gestion des applications d'entreprises : « grand système », « software », « Open Source », « hébergeur », « *outsourcing* », « web ».

Il considère les logiciels visibles des utilisateurs, comme les logiciels techniques déployés sur des serveurs.

Il introduit le modèle cloud, en le comparant aux modèles précédents et il présente les bénéfices du *Pay As You Go* et du *Self Service* pour les entreprises.

3.1 L'ÉVOLUTION DES MODÈLES DE GESTION DE PARC APPLICATIF

3.1.1 Le modèle « grand système » des années 1970

Aux débuts de l'informatique, lorsqu'IBM était le seul acteur de la commercialisation de serveurs, les logiciels n'avaient pas beaucoup de valeur. En effet, IBM vendait ses machines et considérait la partie logicielle comme de la cosmétique, un peu comme

une couche de peinture. Le matériel était alors le principal sujet de l'informatique : il était très encombrant, coûteux, et son accès était réservé à quelques spécialistes.

Le logiciel étant déconsidéré, il n'avait pas de modèle économique. C'est seulement au début des années 1980 à l'époque de MS-DOS qu'a commencé à se développer le principe de vente des licences logicielles.

3.1.2 Le modèle traditionnel « software interne »

Le modèle logiciel dominant que nous connaissons aujourd'hui est apparu dans les années 1980 avec Microsoft MS-DOS, Office, puis Windows. Nous l'intitulerons modèle « software » dans cet ouvrage. Nous ajoutons « interne » dans l'intitulé, car nous considérons dans ce paragraphe que le logiciel est exploité en interne.

Ce modèle est basé sur la commercialisation par un **éditeur** de licences, dites **perpétuelles**, que l'utilisateur achète à un prix fixe et qui lui permet d'utiliser son logiciel sans limite dans le temps.

Le modèle « software » comprend souvent une option de **support** qui, en échange d'un abonnement mensuel ou annuel, donne accès aux équipes de l'éditeur afin de résoudre des problèmes d'installation et de maintenance du logiciel, et de mieux comprendre son fonctionnement. Il propose aussi souvent une souscription annuelle qui permet de bénéficier des **mises à jour du logiciel**.

> **Le cas de Microsoft**
>
> Microsoft propose depuis quelques années le paiement d'une licence perpétuelle accompagné d'une « software assurance », une assurance logicielle qui couvre ses clients contre les problématiques d'obsolescence en leur garantissant de disposer de la dernière version du logiciel. Pour bénéficier de la nouvelle version, il faudra migrer le logiciel, ce qui est une tâche assez fastidieuse.

Dans le modèle « software », les prestations de type installation, paramétrage, maintenance ne sont pas assurées par l'éditeur. L'entreprise utilisatrice doit gérer ces tâches elle-même ou les déléguer à un prestataire de service. L'hébergement du logiciel est généralement assuré dans les locaux de l'entreprise utilisatrice.

Enfin, le déploiement du logiciel auprès des utilisateurs cibles passe par des workflows de validation et des manipulations manuelles : son délai de mise à disposition est généralement de l'ordre de quelques semaines.

On constate, dans la pratique, que l'exploitation d'un logiciel « software » multiplie environ son prix par quatre. Ainsi le coût total de possession du logiciel (TCO[1] en anglais) est très supérieur à son coût d'acquisition.

1. TCO : *Total Cost of Ownership*.

Les principales problématiques auxquelles doit faire face l'entreprise utilisatrice sont :

- déployer le logiciel sur ses infrastructures ;
- garantir la disponibilité[1] du logiciel vis-à-vis des utilisateurs (on parle de disponibilité 24 heures sur 24, 7 jours sur 7, etc.) en déployant et en exploitant une architecture adaptée ;
- assurer la sécurité du logiciel (sa capacité à protéger les données vis-à-vis d'un accès illicite, ou d'une altération accidentelle ou malveillante) en déployant et en exploitant une architecture adaptée ;
- gérer les montées de version du logiciel (le passage de la version 1.0 à la version 1.2, par exemple) en assurant la migration des données, la continuité du service, et la formation des utilisateurs.

Ces problématiques sont extrêmement coûteuses en termes de ressources humaines et nécessitent de bons niveaux d'expertise sur le logiciel utilisé.

Tableau 3.1 — Le modèle « software » en synthèse.

Licence	Support	Mises à jour	Exploitation	Hébergement	*Time to market*
Perpétuelle : paiement en une seule fois à l'éditeur	Paiement annuel à l'éditeur	Paiement annuel à l'éditeur	Interne	Interne	Se compte en semaines

3.1.3 Le modèle « Open Source interne »

Le modèle Open Source s'est développé à la fin des années 1990. Open Source signifie que le code du logiciel est ouvert, que sa relecture par des tiers est possible ; il ne signifie pas nécessairement que le logiciel est gratuit. Nous ajoutons « interne » dans l'intitulé, car nous considérons dans ce paragraphe que le logiciel est exploité en interne.

Les logiciels Open Source les plus utilisés sont ceux qui constituent les socles applicatifs, comme les systèmes de gestion de base de données (MySQL, PostGreSQL), les serveurs HTTP (Apache), les serveurs d'application (Tomcat, Jboss, JonAS, Glassfish, etc.), les systèmes d'exploitation (Linux). L'Open Source aborde peu à peu la partie applicative, visible de l'utilisateur final, avec des logiciels de gestion de site web (exemple : eZpublish), gestion documentaire (exemple : Alfresco), de portail (exemple : Liferay), de CRM (exemple : SugarCRM), etc.

Il existe des sous-modèles dans l'Open Source : le logiciel peut être gratuit ou non, accompagné d'un support payant ou non. Les licences d'utilisation (GPL[2], BSD[3], etc.)

1. Voir *Performance des architectures IT*, G. Grojean et al, Dunod, 2011.
2. GNU : *General Public License*.
3. *Berkeley Software Distribution*.

sont plus ou moins contraignantes pour l'entreprise utilisatrice. Néanmoins, les modèles dominants sont :

- un logiciel gratuit, dont les mises à jour sont gratuites, dont le support peut être assuré par les équipes de l'entreprise utilisatrice ou acheté à des tiers comme les SSLL[1] : c'est le modèle d'Apache, OW2, Eclipse, etc.
- un logiciel gratuit, dont les mises à jour sont gratuites, avec un support payant : c'est le modèle de RedHat, Alfresco, etc. Ces sociétés garantissent la stabilité de versions destinées aux entreprises.

Le cas de Red Hat

Red Hat commercialise notamment des distributions de Linux, un système d'exploitation par ailleurs totalement gratuit. Red Hat vend un support annuel qui permet aux entreprises habituées au modèle « software » de pouvoir se retourner vers un fournisseur en cas de problème. Red Hat est une des plus belles réussites financières du monde Open Source.

Un des avantages du modèle Open Source est que l'on peut tester le logiciel pour se faire une idée précise de ses fonctionnalités et de sa performance technique, sans frais de licence[2]. Une expérimentation ou un test réel donnent toujours beaucoup plus d'information que la lecture d'une documentation éditeur, laquelle n'est pas toujours objective. Cette possibilité est particulièrement intéressante pour les PME qui disposent de compétences techniques, mais n'ont pas la possibilité de se faire prêter les logiciels de type « software » par leurs éditeurs.

Dans le modèle Open Source, les prestations de service de type installation, paramétrage, maintenance ne sont assurées par aucun éditeur. L'entreprise utilisatrice doit s'en charger elle-même ou déléguer ces tâches à un prestataire de service. L'hébergement du logiciel est généralement assuré dans les locaux de l'entreprise utilisatrice.

Le déploiement du logiciel auprès des utilisateurs cibles passe par des workflows de validation et des manipulations manuelles : son délai de mise à disposition est généralement de l'ordre de quelques semaines.

Le coût des logiciels Open Source est essentiellement lié à leur exploitation. Leur TCO se calcule donc par rapport au prix des machines et au coût salarial des équipes d'exploitation. Le TCO n'est pas nécessairement plus bas que celui du modèle « software » pour les grandes entreprises qui ont des équipes performantes dans le modèle « software », et devront changer leurs habitudes.

1. Sociétés de services en logiciels libres.
2. C'est aussi le cas avec certains éditeurs de « software interne ».

Tableau 3.2 — Le modèle « Open Source interne » en synthèse.

Licence	Support	Mises à jour	Exploitation	Hébergement	*Time to market*
Gratuit	Interne ou paiement annuel à l'éditeur	Gratuit	Interne	Interne	Se compte en semaines

En revanche, les start-ups high-tech s'orientent souvent vers l'Open Source, car elles ont la possibilité de s'organiser pour optimiser leur TCO en embauchant des administrateurs spécialisés dans l'Open Source. Ainsi, avec la baisse du coût des machines, les logiciels Open Source permettent à ces entreprises de se lancer sans investissement initial conséquent. Elles gagnent ainsi en indépendance vis-à-vis des capital-risqueurs.

3.1.4 Le modèle « hébergeur »

Une première approche vers l'externalisation peut être d'exploiter ses applications chez un hébergeur, tout en en gérant l'exploitation.

Cette approche comporte deux bénéfices :

- l'entreprise déporte sur un tiers les problématiques matérielles (achat, installation, branchement, etc.) et se focalise sur les aspects logiciels. Ainsi le périmètre de compétences dont il faut disposer en interne se réduit ;
- l'hébergeur achète le matériel en gros et obtient des économies d'échelle sur les prix, dont il peut faire profiter ses clients.

Néanmoins, l'entreprise utilisatrice doit continuer à gérer le parc applicatif avec les coûts afférents. Et le délai de mise à disposition des applications auprès des utilisateurs reste de l'ordre de quelques semaines.

Tableau 3.3 — Le modèle « Hébergeur » en synthèse.

Licence	Support	Mises à jour	Exploitation	Hébergement	*Time to market*
Gratuit ou payant selon modèle logiciel	Interne ou paiement annuel à l'éditeur	Gratuit ou payant selon modèle logiciel	Interne	Fournisseur	Se compte en semaines

3.1.5 Le modèle « outsourcing »

Le modèle « *outsourcing* », ou « externalisation » en français, consiste à confier à un prestataire de service les tâches d'exploitation des applications. IBM Global Services ou HP Enterprise Services sont des prestataires classiques d'« *outsourcing* ».

Dans ce modèle :

- le logiciel est de type « software », Open Source, ou développement spécifique ;
- l'hébergement du logiciel est assuré dans les locaux du prestataire de service.

L'avantage du modèle « *outsourcing* » par rapport au modèle interne est que les tâches d'exploitation et donc la compétence sont externalisées. Elles sont fortement rationalisées par le prestataire. En effet, le prestataire de service exploite de nombreux logiciels pour ses différents clients : il va donc naturellement mettre en place des procédures et des outils pour faciliter et accélérer ces tâches, ce qui lui permettra de réduire ses coûts. Ces coûts seront d'autant plus réduits que l'on fera appel à une main-d'œuvre en offshore où les salaires sont plus bas.

Du fait des chaînes d'intermédiaires, le délai de mise à disposition des applications auprès des utilisateurs reste de l'ordre de quelques semaines.

Le cas d'IBM

IBM dispose dans son offre de matériel, de logiciel et de services. La société a donc une très bonne maîtrise de tous les métiers de l'informatique. Elle est capable de concevoir, déployer et exploiter des applications très performantes basées sur son portefeuille de produits. Le métier de fournisseur de service d'IBM est le plus rentable aujourd'hui : il intègre l'« *outsourcing* ».

On retrouvera ce principe avec le modèle cloud : la concentration facilite la réduction des coûts.

Tableau 3.4 — Le modèle « *outsourcing* » en synthèse.

Licence	Support	Mises à jour	Exploitation	Hébergement	*Time to market*
Gratuit ou payant selon modèle logiciel	Interne ou paiement annuel à l'éditeur	Gratuit ou payant selon modèle logiciel	Fournisseur	Fournisseur	Se compte en semaines

3.1.6 Le modèle « web »

Le modèle web est principalement destiné au grand public. Il est présenté ici en introduction au modèle cloud.

Le modèle web propose un service disponible uniquement sur les serveurs de son éditeur, spécialisé sur une fonction donnée (exemple : la recherche, la vente de livres). Dans ce modèle, le service évolue progressivement sans consultation, ni avertissement à ses utilisateurs : on reparlera plus loin du principe de la bêta perpétuelle. Par ailleurs, les utilisateurs souscrivent au service de manière autonome, selon le principe du ***Self Service***.

Le modèle web connaît plusieurs modèles économiques :

- **le commerce électronique :** le site vend des produits physiques ou numériques (exemples : Amazon, iTunes Store) ;
- **le modèle publicitaire :** une utilisation gratuite rentabilisée par la vente d'espaces publicitaires (exemple : moteurs de recherche) ;
- **le « Freemium »** (deux versions : gratuite (free) et payante (premium)) : l'offre gratuite incite les utilisateurs à tester l'application, à s'y habituer, puis les pousse vers la version payante qui offre des fonctions supplémentaires (exemples : Evernote, Trello) ;
- **la place de marché :** l'application met en contact des acheteurs et vendeurs et prélève une commission (exemple : eBay) ;
- **le don/bénévolat :** le site publie un logiciel gratuit et propose aux utilisateurs de faire un don (exemple : Mozilla).

On verra que certains clouds reprennent en partie ces modèles.

Le modèle web a permis de concevoir et d'expérimenter des architectures à très haute disponibilité. En effet, certains sites reçoivent plusieurs millions de visites par jour. Le web a ainsi fait progresser les architectures permettant de hautes performances : ces technologies ont été réutilisées dans les entreprises, mais aussi dans les clouds.

Tableau 3.5 — Le modèle « web » en synthèse.

Licence	Support	Mises à jour	Exploitation	Hébergement	*Time to market*
Gratuit, sur abonnement, à l'usage,etc.	Gratuit	Gratuit	Fournisseur	Fournisseur	Se compte en minutes (*Self Service*)

3.1.7 Le modèle « cloud public »

Avec le modèle cloud, le service est intégré et hébergé par son fournisseur. Il emprunte beaucoup aux pratiques du web 2.0 et au modèle web présenté ci-dessus, et constitue la déclinaison de ces modèles dans le cadre de l'entreprise.

Le service est standardisé pour faciliter sa maintenance par l'opérateur : une seule version du service s'exécute pour toutes les entreprises utilisatrices. Il s'appuie sur une architecture « multi-tenant » qui ressemble plus à celle des sites web, qu'à celle d'un logiciel de type « software » (*cf.* chapitre 18). Cette architecture, complètement spécialisée pour le logiciel, facilite les tâches liées à l'exploitation :

- gestion de la haute disponibilité ;
- gestion des montées de version ;
- gestion des sauvegardes ;
- etc.

Son objectif est d'automatiser au maximum ces tâches jusqu'alors confiées à des opérateurs humains afin de réduire les coûts d'exploitation. Elle permet aussi une

plus forte maintenabilité du service, géré comme une instance unique et globale. Par conséquent, la mise à disposition du service auprès des utilisateurs est très rapide.

Les services cloud sont totalement liés à leur plateforme : il est difficile de les proposer sous une forme « software » internalisable par l'entreprise.

Le cas de Google

Le modèle économique de Google est à la croisée des chemins entre le modèle web et le modèle cloud. Le géant de la recherche dispose sur ses Datacenters de trois environnements distincts pour ses applications :

– **google.com :** c'est l'environnement utilisé par les employés de Google. Il intègre tous les services collaboratifs (messagerie, calendrier, portail, partage de fichiers, bureautique en ligne, etc.) ;

– **gmail.com :** c'est l'environnement utilisé par le grand public. Il intègre les services collaboratifs cités ci-dessus et d'autres (réseau social, recherche dans l'historique de recherche, etc.) ;

– **Google Apps :** c'est l'environnement utilisé par les entreprises clientes de Google. Il intègre les services collaboratifs cités ci-dessus, en utilisant le nom de domaine de l'entreprise à la place de gmail.com.

Lors du développement de nouvelles fonctions, Google suit les étapes suivantes :

– **alpha-test** des fonctions par les collaborateurs Google sur le SaaS google.com ;

– **bêta-test** par le grand public sur le site web gmail.com ;

– **mise en production** pour les entreprises sur le SaaS Google Apps.

Gmail et Google Apps sont donc des mondes séparés : web et SaaS. Mais ils reposent chez Google sur les mêmes plateformes.

Tableau 3.6 — Le modèle cloud public en synthèse.

Licence	Support	Mises à jour	Exploitation	Hébergement	*Time to market*
Pay As You Go	Compris	Compris	Fournisseur	Fournisseur	Se compte en minutes (*Self Service*)

3.1.8 En synthèse

Le tableau présenté ici tente de faire une première synthèse sur les spécificités des différents modèles logiciels. Nous reviendrons plus en détail dans la suite de ce chapitre sur une comparaison détaillée entre les modèles « software interne » et cloud.

Tableau 3.7 — Synthèse comparative sur les modèles logiciels.

Modèle	Licence	Support	Mise à jour	Exploitation	Hébergement	*Time to market*
Software interne	Perpétuelle : paiement en une seule fois à l'éditeur	Paiement annuel à l'éditeur	Paiement annuel à l'éditeur	Interne	Interne	Se compte en semaines
Open Source interne	Souvent gratuit	Interne ou paiement annuel à l'éditeur	Souvent gratuit	Interne	Interne	Se compte en semaines
Hébergeur	Gratuit ou payant selon modèle logiciel	Interne ou paiement annuel à l'éditeur	Gratuit ou payant selon modèle logiciel	Interne	Fournisseur	Se compte en semaines
Outsourcing	Gratuit ou payant selon modèle logiciel	Interne ou paiement annuel à l'éditeur	Gratuit ou payant selon modèle logiciel	Fournisseur	Fournisseur	Se compte en semaines
Web	Gratuit ou sur abonnement	Gratuit	Gratuit	Fournisseur	Fournisseur	*Self Service* (se compte en minutes)
Cloud public	*Pay As You Go*	Compris	Compris	Fournisseur	Fournisseur	*Self Service* (se compte en minutes)

3.2 DE LA PERTINENCE DU MODÈLE « CLOUD PUBLIC » VIS-À-VIS DU MODÈLE « SOFTWARE INTERNE »

Nous avons passé en revue, dans les paragraphes précédents, les différentes approches de la gestion de parc applicatif. Nous allons maintenant nous arrêter plus longuement sur la comparaison entre le modèle dominant, le modèle « software interne », et le modèle « cloud public ».

3.2.1 Le prix du logiciel

L'avantage principal du modèle « software interne » est la notion de licence perpétuelle : on paye une fois pour toutes et on est propriétaire de son logiciel. Cependant, comme on l'a vu précédemment, ce modèle cache des coûts récurrents : le support annuel et le renouvellement des licences tous les deux ou trois ans.

Il est clair que le terme « perpétuel » est totalement déplacé dans le monde informatique où les cycles de renouvellement du matériel et du logiciel sont extrêmement courts. L'obsolescence rapide est d'ailleurs une des principales sources de coût de la filière. Les montées en version de certains logiciels sont parfois une course en avant dont l'objectif est de vendre plus de licences, plutôt que d'apporter de véritables nouvelles fonctions aux utilisateurs. Certains éditeurs n'hésitent pas à proposer des mises à jour cosmétiques pour maintenir leur carnet de commandes.

L'exemple de Microsoft Office

Office est un logiciel formidable de par sa puissance et sa productivité. Il est d'ailleurs totalement incontournable dans le monde des entreprises aujourd'hui. Office a connu de grandes périodes d'innovation : entre sa naissance et sa version 2000, des fonctionnalités très pertinentes sont apparues (par exemple la correction de grammaire et d'orthographe).

Cependant, les versions 2003, 2007, 2010 et 2013 ont connu un ralentissement du rythme des innovations car le monde de la bureautique est arrivé peu à peu à maturité. Ainsi les dernières versions du logiciel misent sur l'évolution de l'interface plutôt que sur les fonctionnalités. Et le bénéfice d'une montée de version pour une entreprise est de plus en plus discutable. Ce, d'autant plus, que toute montée de version a un coût.

Le modèle cloud fonctionne quant à lui, à la manière de celui des opérateurs télécom : un opérateur de téléphonie mobile facture mensuellement un forfait et fait payer les SMS à l'usage ; de la même manière, un opérateur ADSL facture un abonnement mensuel et fait payer à l'usage les communications vers les mobiles et les locations de films à la demande. On peut d'ailleurs noter que les opérateurs télécom font leur marge principalement sur les services facturés à l'usage : c'est leur déclinaison du « *Pay As You Go* ».

Les opérateurs cloud suivront probablement ce modèle en offrant un abonnement avec une large gamme d'options et des services facturés à l'usage.

Dans le modèle cloud, les mises à jour sont incluses dans le service : il n'est pas nécessaire de racheter un droit d'usage tous les trois ans ; cela peut être une source de réduction des coûts[1]. C'est en tout cas, une source de lissage des coûts.

Cette différence avec le modèle « software interne » est fondamentale. Les opérateurs cloud n'ont pas besoin d'alourdir inutilement leurs applications pour continuer à percevoir des revenus.

Par ailleurs, le modèle cloud est basé sur la mutualisation des ressources, qu'elles soient matérielles ou logicielles. Ce principe de mutualisation est dans la droite ligne de la mondialisation de l'économie : en effet, les industriels ont aujourd'hui tendance à se spécialiser pour être compétitifs. Par exemple, les constructeurs automobiles sont aujourd'hui devenus, par souci d'optimisation des coûts, des assembleurs de composants (siège, moteur, etc.) produits par des tiers. De la même manière, **les clouds arrivent à faire baisser leurs coûts grâce à la mutualisation de leur plateforme entre de nombreuses entreprises.**

Enfin, avec les clouds, le paiement s'effectue seulement suivant le nombre d'utilisateurs, sans coût fixe forfaitaire. A contrario, il existe de nombreuses grandes entreprises qui achètent des licences en masse sans véritablement les utiliser. **Encore un facteur de coût qui disparaît avec le modèle cloud.**

La métaphore de l'électricité, proposée par Nicolas Carr

Nicolas Carr est influent dans le monde des technologies de l'information. Il a écrit plusieurs ouvrages qui ont fait beaucoup de bruit dans la profession[2]. Son ouvrage intitulé *The Big Switch*[3] se penche sur le monde du cloud.

Nicolas Carr explique que de nombreuses entreprises américaines produisaient elles-mêmes leur électricité au début du XX^{e} siècle. Puis elles ont pris conscience du fait que cette production leur revenait cher et qu'il était économiquement plus intéressant de faire appel à un opérateur spécialisé dans la production d'énergie. Ainsi, ces entreprises se sont peu à peu mises à acheter leur électricité et à se recentrer sur leur cœur de métier (production d'automobiles, de biens de consommation, etc.)

Selon Nicolas Carr, la même histoire va se reproduire prochainement dans le monde informatique. Les entreprises vont constater que la gestion courante de leurs logiciels de messagerie, de ressources humaines, ou la gestion de la relation client, etc. leur coûteraient moins cher en l'externalisant chez des opérateurs spécialisés. Ainsi, elles vont externaliser ces services et se recentrer sur leur informatique cœur de métier.

La métaphore est parlante et efficace. Si nous ne partageons pas tous les éléments du discours de Nicolas Carr, nous pensons aussi que la montée en puissance du cloud est inévitable pour des raisons de rationalisation. Nous verrons dans le chapitre 5 les autres arguments pour basculer vers le cloud.

1. Le calcul de retour sur investissement doit être fait au cas par cas.
2. En particulier : *Does IT Matter* ? Information Technology and the Corrosion of Competitive Advantage, N. Carr Harward Business School Press, 2004.
3. *The Big Switch, Rewriting the World, from Edison to Google*, N. Carr, Norton, 2008.

3.2.2 Le prix de la maintenance du parc utilisateur

Les évolutions régulières des logiciels « software » sont à l'origine de coûts importants liés aux changements de configuration des serveurs et des postes utilisateurs. Le déploiement de mises à jour sur un parc de postes de travail passe par la télédistribution au travers d'un logiciel de type « Windows Update » ou par l'installation poste à poste. Dans le premier cas, le déploiement et la maintenance de la télédistribution ont un coût important en termes de licence (le logiciel de télédistribution n'est pas gratuit) et de main-d'œuvre. Dans le second cas, l'installation a un coût exorbitant en termes de main-d'œuvre pour un parc excédant la centaine de postes.

Ces problématiques n'existent pas dans le modèle cloud, où aucun déploiement n'est nécessaire sur le parc utilisateurs.

3.2.3 Le prix de la maintenance du parc serveur

L'exploitation des logiciels serveurs par l'entreprise nécessite des équipes de production aux compétences coûteuses, et qui sortent souvent du métier de l'entreprise. Cette assertion se vérifie pour les progiciels comme pour les développements spécifiques.

Les mises à jour de logiciels sur les serveurs comportent des risques importants en termes de sécurité : une mise à jour peut être à l'origine de corruption ou de perte de données ; elle peut aussi provoquer des pertes de performance ou des régressions fonctionnelles au niveau des applications. Pour bien gérer ces risques, il est important de disposer d'une plateforme de test sur laquelle on va éprouver, éventuellement plusieurs fois, le protocole de mise à jour afin de le maîtriser avant de le mettre en œuvre sur la plateforme de production. La plateforme de test servira aussi à éprouver la nouvelle version du logiciel et à valider sa performance et son absence de régression fonctionnelle. Ces bonnes pratiques de mise à jour des applications serveurs ont un coût important en termes de machines et de main-d'œuvre. **Ces problématiques n'existent pas dans le modèle cloud, où la maintenance est prise en charge par le fournisseur de service.**

3.2.4 Le coût du parc de postes de travail

L'obsolescence rapide du matériel est une des principales sources de coûts dans le domaine informatique. On peut la gérer en utilisant un parc loué plutôt qu'acheté : c'est la stratégie qu'utilisent de nombreuses entreprises.

Certaines entreprises se penchent aujourd'hui sur le remplacement des postes de travail classiques sous Microsoft Windows/Office par des alternatives moins coûteuses. Deux pistes intéressantes vont dans le sens de cette démarche (*cf.* chapitre 1) :

- **Les clients légers :** il s'agit d'une nouvelle génération de terminaux passifs qui constituent des interfaces vers des environnements de travail déportés sur des serveurs ; ils utilisent donc des protocoles de déport d'écran[1] ; ils sont durcis,

1. Comme Citrix ou Microsoft RDP, *Remote Desktop Protocol*.

c'est-à-dire qu'ils sont monoblocs et ne comportent aucune pièce mécanique. Parmi les avantages de ces appareils, on peut citer le moindre coût, la moindre consommation en énergie (environ le dixième de celle d'un PC normal), les moindres nuisances sonores et une pérennité de travail de cinq à sept ans. En revanche, ces appareils ne peuvent pas fonctionner en cas de perte de contact avec le serveur.

- **Les chromebooks :** ce sont des ordinateurs portables à moindre coût, souvent équipés de Linux et dépourvus de disques durs. Ils ne peuvent exécuter les clients lourds développés pour Windows et sont donc plutôt destinés à exécuter des applications HTML5. Le système ChromeOS proposé par Google début 2011 a été conçu pour porter cette vision.

Le modèle cloud autorise le déploiement massif de tels postes de travail à moindre coût. En effet, les applications SaaS ont des interfaces HTML5 qui fonctionnent avec ces deux types d'appareils. Par ailleurs, elles nécessitent peu de puissance sur le poste de travail et permettent d'envisager l'usage de clients légers sans avoir à déployer d'énormes fermes de serveurs pour gérer les déports d'exécution. **Le cloud permet donc de réduire les coûts matériels liés aux postes de travail.**

3.2.5 Le prix du parc de serveurs

Même si le prix du matériel informatique a beaucoup chuté ces dernières années, le coût d'un parc serveur reste assez élevé. Un hébergement de qualité nécessite en effet :

- au moins deux centres de données ;
- des « salles blanches » équipées de climatisation et de planchers techniques ;
- des armoires de serveurs intégrant des « racks », des systèmes mutualisés de clavier et d'écran, des dispositifs de redémarrage à distance ;
- un système de badges de sécurité pour réglementer l'accès aux serveurs hébergés ;
- un lien redondant pour l'accès à Internet ;
- une alimentation en électricité redondante ;
- un groupe électrogène et/ou des batteries de secours en cas de coupure d'électricité ;
- un système d'injection de gaz inerte (souvent de l'azote) pour étouffer les flammes en cas d'incendie ;
- etc.

La nécessité d'avoir au moins deux centres de données s'explique par le besoin de prévenir les désastres (incendies, tremblements de terre, etc.). Pour optimiser cette sécurité, il est conseillé que ces centres soient séparés par une distance de quelques dizaines de kilomètres. Et par conséquent, il faut deux équipes d'exploitation, car une seule équipe ne pourrait pas intervenir assez rapidement sur les deux sites. **Ces arguments plaident pour le déport vers des spécialistes de l'exploitation, comme les opérateurs de clouds.**

Le coût des « salles blanches » est lié à la taille et à l'emplacement du Datacenter. Des acteurs comme Google font le choix de créer de grands centres dans des zones isolées où le prix du mètre carré est dérisoire. Cette approche leur permet de réduire leurs coûts. **Cet argument plaide pour le déport vers des spécialistes de l'exploitation, comme les opérateurs de clouds.**

Il faut noter que le coût électrique d'un centre de données est très conséquent. Ainsi, le matériel informatique représente aujourd'hui 1,2 % de la consommation électrique des États-Unis. On estime même que la consommation de l'informatique égalera celle des hommes en 2035[1]. Les acteurs du cloud computing utilisent des machines peu gourmandes, ce qui leur permet de réduire ce coût. **Cet argument plaide pour le déport vers des spécialistes de l'exploitation, comme les opérateurs de clouds.**

Par ailleurs, beaucoup d'entreprises constatent aujourd'hui que leurs serveurs sont sous-employés et qu'elles ont des charges d'achat, de maintenance et de consommation électrique inutiles. Une étude d'IBM a montré que de nombreux serveurs sont utilisés à 20 % de leur charge, ce qui signifie que 80 % de leur facture part en pure perte. Les entreprises se tournent aujourd'hui vers les technologies de virtualisation pour réduire ces pertes. Mais ces technologies sont complexes à maîtriser. **Là encore, le déport des serveurs vers un opérateur de cloud est plus efficace.**

On a parfois tendance, dans le modèle « software interne », à chiffrer le coût matériel d'une plateforme serveur en se basant sur la production. Il est essentiel de bien noter qu'une plateforme « software interne » comprend des machines de production, mais aussi de développement, test, et recette. Le coût en termes de serveurs d'une plateforme « software » est ainsi souvent le triple du coût de la plateforme de production. **Ces problématiques n'existent pas dans le modèle cloud, où le matériel est géré par le fournisseur de service.**

3.2.6 En synthèse

On a vu dans ce paragraphe que le modèle cloud est très intéressant à deux titres :

- **il permet généralement[2] la réduction du coût total de possession (TCO)** : la consommation par abonnement est souvent moins coûteuse que la somme de l'achat de licences dites « perpétuelles », de l'achat de support, de l'achat de mises à jour et des frais d'exploitation. Par ailleurs, les clouds permettent des modèles de commercialisation variés. On a évoqué précédemment le « freemium », l'abonnement mensuel/annuel, le paiement à l'usage... Le calcul de ce TCO doit néanmoins être fait au cas par cas par les entreprises utilisatrices ;
- **il permet de déporter vers des sociétés spécialisées les problématiques** d'exploitation. Ainsi, les entreprises peuvent se recentrer sur leur cœur de métier

1. Source : Gerhard Fettweis de l'université de Dresde. La consommation des hommes intègre l'éclairage, le chauffage, les transports, etc.

2. Mais pas toujours : l'étude doit être menée au cas par cas.

plutôt que de disperser leurs compétences, en embauchant des spécialistes de l'exploitation informatique.

3.3 LE CLOUD VU PAR LES ÉDITEURS

La plupart des éditeurs se posent la question de changer de modèle économique : aller du modèle « software interne » vers le modèle cloud. Ce paragraphe présente les avantages de ce dernier modèle pour les éditeurs.

3.3.1 La « perversion des versions »

On a vu précédemment que les entreprises utilisatrices devaient, suivant le modèle « software interne », effectuer des montées de version régulièrement, et que ces montées de version avaient un coût important.

Le problème se pose aussi pour les éditeurs. En effet, assurer le support, les mises à jour de sécurité et la maintenance évolutive sur plusieurs versions d'un même logiciel est un terrible casse-tête, et un casse-tête coûteux.

Prenons l'exemple de Microsoft : l'éditeur de Redmond a dû, à un moment donné, gérer les évolutions de Windows XP, Vista, 7 et 8. On imagine les quatre équipes de développeurs mobilisées pour un seul et même logiciel : une terrible dispersion d'énergie et un cauchemar pour un éditeur moins fortuné que Microsoft.

Avec le modèle cloud, l'opérateur ne propose qu'une seule version de son logiciel à un instant *t*.

Par ailleurs, la gestion de versions signifie une roadmap avec des jalons de publication des « *releases* ». Ces jalons sont généralement espacés dans le modèle « software interne ». Ainsi, si un bug s'est glissé dans une « release », il risque de perdurer plusieurs mois.

A contrario, les opérateurs SaaS font des mises en production incessantes et au fil de l'eau, suivant le principe de la « bêta perpétuelle ». Le fait de maîtriser la plateforme de production leur donne un très grand degré de liberté dans l'activation des nouvelles fonctionnalités : ils peuvent les activer pour une population restreinte de bêta testeur, les généraliser, ou tout simplement les désactiver si elles s'avèrent non pertinentes ou boguées. Le « A/B testing » est au cœur de ces pratiques : il consiste à activer une fonction pour une part des utilisateurs, la population A, et une autre pour la population B. La comparaison des comportements des deux populations donnera des indicateurs sur la pertinence de cette fonction. Ce type de test peut être utilisé pour des choix élémentaires : la forme, la couleur, la position d'un bouton...

Le SaaS permet donc un grand confort aux éditeurs dans leur gestion de version.

3.3.2 La « *Customer driven roadmap* »

La « *Customer driven roadmap* » est un élément complémentaire et vertueux de la « bêta perpétuelle ». Comme les opérateurs SaaS maîtrisent la plateforme de production, ils peuvent faire des statistiques sur l'usage de leur logiciel. Cela leur permet de mesurer l'accueil qui est fait aux nouvelles fonctionnalités offertes. À ce titre, les opérateurs SaaS suivent énormément de métriques : on dit d'ailleurs qu'ils ont le « culte de la mesure ».

Une bonne illustration de la « *Customer driven roadmap* » est fournie par les Labs de Gmail : ce sont de petites fonctionnalités unitaires que les utilisateurs peuvent décider d'activer ou non. En fonction de leur adoption, Google décide ou non de les intégrer à la version standard de son service.

De manière plus classique, le fait de maîtriser la plateforme de production permet de lancer des sondages auprès de certains segments de population afin d'obtenir un feedback utilisateurs.

3.3.3 Un nouveau modèle économique

Les éditeurs sont souvent séduits par les arguments évoqués ci-dessus. Il reste que le modèle cloud constitue un nouveau modèle de revenu auquel il faut d'adapter :

- les rentrées d'argent sont diluées dans le temps, ce qui a un impact sur les investissements ponctuels en R&D ;
- les références manquent sur le bon prix à facturer en abonnement. Par exemple, le prix de Google Apps est bas par rapport au nombre de services offerts. Ce prix n'est pas forcément la bonne référence pour de petits éditeurs ;
- il existe un risque de cannibalisme entre le modèle cloud et le modèle « software » pour les éditeurs qui souhaitent maintenir les deux offres. Ainsi Microsoft a beaucoup tardé à sortir « Office Web Apps » qui constitue un concurrent pour Office, son offre historique ;
- enfin, l'éditeur qui fait le choix du SaaS devra arbitrer entre créer sa propre infrastructure de Datacenters ou bien se reposer sur un cloud public existant. La seconde option paraît plus prudente pour les petits acteurs.

3.4 CLOUD PUBLIC *VERSUS* CLOUD PRIVÉ

Comme on l'a évoqué précédemment, on entend par cloud public un cloud opéré par un acteur tiers, comme Google ou Amazon, et accessible depuis Internet. Un cloud privé est une plateforme élastique gérée en interne.

Rappelons que, selon notre définition, une plateforme interne n'est un cloud privé que si elle respecte les propriétés suivantes (*cf.* chapitre 2) :

- élasticité ;
- abstraction sur la localisation ;

- mutualisation ;
- *Pay As You Go* ;
- *Self Service* ;
- API ouvertes.

Employer le terme cloud pour désigner toute plateforme de virtualisation interne est selon nous un abus de langage : on parle parfois de « cloud *washing* » pour qualifier l'usage du mot cloud tous azimuts. Certains éditeurs utilisent abondamment le « cloud *washing* » pour faire la promotion de leurs outils, même s'ils ne les ont pas fait évoluer pour intégrer les propriétés ci-dessus.

Même en respectant ces propriétés, nous gardons de fortes réserves sur le cloud privé :

- Il n'est possible de tenir la promesse d'élasticité qu'avec un « effet d'échelle », c'est-à-dire un parc de serveurs important, de l'ordre de centaines, voire de milliers de machines. Ainsi, un cloud privé fera réellement sens chez un grand compte, beaucoup moins dans le cadre d'une PME.
- La promesse de ***Self Service*** et de ***Pay As You Go*** en interne ne peut être tenue qu'avec un degré avancé d'automatisation dans le data center. Cette automatisation revient cher pour un cloud privé de dimension moyenne. Cependant, des solutions commencent à émerger : nous les évoquerons dans le chapitre 19.
- Le cloud privé nécessite un **important investissement initial** (CAPEX[1]) pour construire un centre de données. Il nécessite des immobilisations, sous la forme de réserve de machines, pour assurer l'élasticité. A contrario, les clouds publics permettent beaucoup d'agilité en se limitant à des OPEX[2]. Notons qu'il est possible de lever cette réserve en déportant le cloud privé chez un hébergeur qui prendra en charge l'investissement initial : on parlera alors de « cloud privé hébergé ».
- Les clouds publics assurent une haute garantie sur l'intégrité des données au travers de deux propriétés : environnement multi-site géographiquement distribué, et certifications (ISO 27001, SAS 70 type II). Ces propriétés sont difficiles à assurer avec un cloud privé de taille moyenne.
- Les clouds publics proposent des prix attractifs, rendus possibles avec la mutualisation des ressources à grande échelle, entre plusieurs entreprises.

Malgré tout, certaines entreprises ne peuvent pas envisager le cloud public pour des raisons réglementaires, par exemple dans la banque (nous reviendrons sur ces problématiques au chapitre 5). Elles doivent donc faire le choix du cloud privé. Plusieurs scénarios sont alors possibles en fonction de leurs contraintes de sécurité (figure 3.1) :

1. CAPEX (*Capital Expenditure*) : c'est-à-dire dépenses d'investissement de capital.
2. OPEX (*Operating Expenditure*) : c'est-à-dire dépenses d'exploitation.

- un cloud strictement réservé à l'entreprise dans les murs de l'entreprise ;
- un cloud strictement réservé à l'entreprise dans les murs d'un hébergeur (cloud privé hébergé) ;
- un cloud mutualisé avec d'autres acteurs du même secteur, dignes de confiance, dans le cadre d'un groupement d'intérêts communs (GIE) ;
- une solution hybride entre cloud public et privé, les traitements étant poussés vers le cloud public, tandis que les données restent au sein du cloud privé ;
- un cloud public situé sur le territoire national, qui peut être éligible dans certains contextes réglementaires.

Nous pensons que le cloud strictement réservé à l'entreprise est une option vouée à disparaître dans le futur.

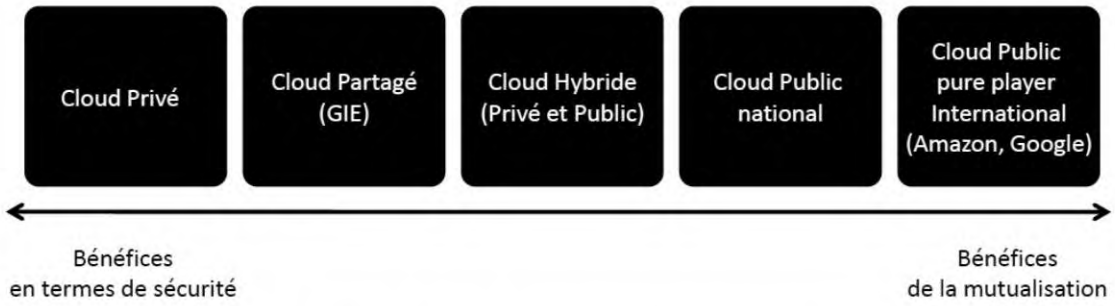

Figure 3.1 — Typologies de cloud et bénéfices.

3.5 CLOUD PUBLIC ET OPEN SOURCE

3.5.1 Clouds et socles Open Source

La fourniture de service cloud passe par la création de Datacenters de plusieurs milliers de serveurs. Dans ce cadre, l'usage de logiciels Open Source est source d'économies très substantielles. Si l'on prend l'exemple de Google, cet acteur est réputé posséder plus de 1 million de machines serveurs sous Linux. Si ces machines fonctionnaient sous un Unix propriétaire, la facture à payer serait très conséquente...

Le monde du cloud public s'appuie donc fréquemment sur des socles d'exécution Open Source (système Linux, base de données MySQL, serveur HTTP Apache, langages PHP/Python/Ruby, etc.) afin de réduire ses coûts d'infrastructures. Les acteurs ont par conséquent acquis de fortes compétences Open Source, ainsi qu'une grande inclinaison pour ce modèle, qui les a amenés naturellement à ouvrir le code de leurs applications.

3.5.2 Ouverture du code des services cloud

On a vu précédemment que le cloud reprenait le principe de partage de connaissance issu du web 2.0. En particulier, dans le domaine technique, les clouds exposent des API ouvertes afin de permettre une intégration simple de leurs fonctionnalités par des applications tierces.

Certains acteurs cloud vont plus loin dans l'ouverture, en mettant à disposition tout ou partie de leur code source, adoptant ainsi le modèle Open Source. C'est le cas de Google, mais c'est loin d'être le cas de tous les acteurs. Certains observateurs vont jusqu'à parler d'un modèle « Open cloud » pour ces typologies d'outils disponibles en mode hébergés et téléchargeables.

Une licence Open Source spécifique a d'ailleurs émergé pour gérer les logiciels libres distribués en mode cloud : la licence AGPL (*Affero General Public License*).

Les clouds peuvent donc avoir une forte appétence avec le modèle Open Source.

3.5.3 Cloud public et Open Source : un modèle pour les entreprises ?

Les entreprises qui souhaitent réduire au maximum leurs coûts d'achat de licence « software » ont aujourd'hui l'opportunité de mêler les deux modèles cloud et Open Source :

- **le modèle cloud** pourra être utilisé pour l'informatique de commodité, suffisamment générique pour être externalisée ;
- **le modèle Open Source** pourra être utilisé comme socle à l'informatique métier, vouée à rester au sein de l'entreprise.

Cette approche commence à intéresser de nombreuses entreprises qui y voient une opportunité intéressante de réduction de leur TCO.

3.6 GARDER LES PIEDS SUR TERRE...

Avant de refermer cette partie introductive, nous souhaitons modérer l'enthousiasme excessif pour le cloud computing. En effet, le cloud est un *buzz word* du moment et certains y voient une solution générique à tous les problèmes, du fait de sa promesse de scalabilité phénoménale.

Il faut bien garder en tête les aspects suivants :

- le cloud public n'est pas éligible dans certains secteurs d'activité ;
- le cloud public peut se révéler plus cher que l'exploitation en interne dans certains cas : il convient donc de faire un calcul de TCO ;
- le cloud a aussi un effet « boîte noire » (on ne connaît pas les architectures de Google ou Amazon) qui peut se révéler satisfaisant pour ceux qui ne souhaitent pas s'encombrer de problématiques techniques. Il existe cependant des cas où il est nécessaire de comprendre et d'optimiser son architecture, même sur le cloud ;

- le cloud privé est un vain mot si l'on ne réorganise pas les équipes de la DSI pour assurer le *Pay As You Go* et le *Self Service*. Autant ne pas se voiler la face et parler de plateforme de virtualisation.

En résumé

Ce chapitre a démontré les avantages du modèle cloud public vis-à-vis du modèle « software interne » traditionnel : nombreux axes de réduction des coûts, *Pay As You Go*, *Self Service*.

Il a introduit les bénéfices du cloud pour les éditeurs.

Il a présenté les différences entre cloud public et cloud privé.

Enfin, le modèle cloud public connaît beaucoup de synergies avec le modèle open source.

DEUXIÈME PARTIE

Faire confiance au cloud ?

L'objectif de cette partie est de présenter de manière détaillée les opportunités et les risques du cloud computing pour l'entreprise.

Un cas d'usage est introduit : celui de la société INDUS, dans le secteur industriel.

Après une présentation des problématiques de sécurité, une rapide introduction sur les usages du cloud computing, les chapitres de cette partie abordent successivement les points de vue de la direction, des utilisateurs, puis des informaticiens, avant de dresser une synthèse en vue d'une aide à la décision.

Cette partie met l'accent sur l'usage des SaaS et IaaS, car nous constatons que, jusqu'à présent, les entreprises étaient méfiantes vis-à-vis des PaaS, du fait de leurs architectures très spécifiques.

Pour mentionner quelques chiffres, 59 % des grands comptes utilisent aujourd'hui au moins un service cloud, contre 44 % pour les entreprises de moins de 100 salariés, plus méfiantes[1]. Le principal frein à l'adoption du cloud est la question de la sécurité des données, chez 64 % de décideurs IT et 45 % de décideurs métiers.

1. Selon une étude de juillet 2015 par NetMediaEurope, réalisée auprès de 428 décideurs issus de France, d'Allemagne, du Royaume-Uni, d'Italie et d'Espagne.

4

Cloud et sécurité

Objectif

L'objectif de ce chapitre est d'apporter un éclairage factuel sur les problématiques de sécurité liées au cloud, afin de sortir des peurs irrationnelles. Il aborde les aspects juridiques, réglementaires et techniques de la sécurité. Il donne aussi des pistes pour l'adaptation de la politique de sécurité au cloud.

Un cas d'usage : la société INDUS

Pour illustrer les différents éléments de cette partie, nous proposons d'introduire un cas d'usage : l'expérience d'une entreprise fictive, basée sur des retours d'expérience réels. Cette entreprise appartient au monde de l'industrie européenne. Nous l'appellerons INDUS. Elle commercialise des composants électroniques destinés à une intégration dans des produits électroménagers. INDUS compte 10 000 employés.

Monsieur Vincent, le directeur général d'INDUS est un homme pragmatique. Sa préoccupation principale est de maintenir sa société face à la concurrence des pays émergents qui pratiquent des tarifs plus bas. Sa stratégie consiste à produire des composants innovants, de haute qualité. Il investit beaucoup dans la recherche et développement. Pour Monsieur Vincent, l'informatique est un outil nécessaire qui rend ses collaborateurs plus performants. Mais il ne veut pas faire de l'informatique pour l'informatique. L'innovation qui l'intéresse est avant tout celle qui concerne son métier. Il s'intéresse un peu à la mouvance cloud.

Monsieur François est directeur informatique d'INDUS. Il a travaillé dur pour mettre en place des solutions de CAO[1], afin de concevoir rapidement les nouveaux produits ;

1. Conception assistée par ordinateur.

il a aussi déployé des outils pour gérer les échanges avec les fournisseurs et les clients : flux d'informations B2B, intégration avec des places de marché, etc. Il se méfie des belles plaquettes des éditeurs de logiciels et préfère le développement spécifique par ses équipes. Monsieur François dispose d'une petite équipe qui gère les postes de travail et les outils de travail collaboratifs (messagerie, calendriers partagés, etc.) de ses collaborateurs. Ces outils ne sont pas les plus intéressants pour lui : il préfère travailler sur l'outillage CAO. De fait, l'équipe « poste de travail » dispose de peu de moyens et elle est mal payée. Son turn-over est important, et les outils collaboratifs sont souvent indisponibles.

Monsieur Paul est directeur commercial d'INDUS. Il rêve depuis longtemps d'une application de CRM pour pouvoir mieux encadrer le travail de ses commerciaux. Ces derniers sont en permanence sur le terrain. Ils sont très autonomes et difficiles à suivre. Monsieur Paul s'est vu répondre plusieurs fois par Monsieur François que les logiciels de CRM sont chers et inefficaces. Il aimerait bien trouver une solution pour pouvoir se passer de l'aval de ce dernier.

Enfin, Monsieur Jules est directeur marketing d'INDUS. Son équipe fourmille d'idées de services innovants et souhaiterait pouvoir rapidement faire des maquettes applicatives pour pouvoir les tester auprès des utilisateurs. Mais il faut souvent 6 mois à la DSI pour faire ces maquettes. Monsieur Jules trouve que cette dernière manque d'agilité.

4.1 LA QUESTION DE LA CONFIANCE

Lorsqu'on présente le modèle cloud à un néophyte, sa première réaction tourne invariablement autour de la confidentialité des données. En effet, le monde de l'entreprise est **universellement et intimement persuadé que l'on doit conserver ses données informatiques dans ses locaux pour assurer leur sécurité.**

Cette intime conviction est un peu étrange quand on sait que les entreprises utilisent depuis des années les services de prestataires d'hébergement pour leurs applications web et leurs extranets clients ; des données parfois critiques concernant l'entreprise et ses clients transitent par ces tiers. La confiance accordée aux prestataires d'hébergement tient à leur réputation sur le marché et à leurs engagements contractuels. En effet, ils s'engagent juridiquement au travers de « règles de confidentialité » à ne pas divulguer les données de leurs clients. Les opérateurs cloud prennent les mêmes engagements, mais attirent moins la confiance que les prestataires d'hébergement. Selon nous, cette méfiance est essentiellement due aux origines du cloud : ces acteurs sont issus du monde du web, un monde connoté grand public ; tandis que les prestataires d'hébergement sont issus du monde des télécoms, un monde connoté entreprise.

La différence de traitement entre documents papier et données numériques

Dans le cadre de la conservation des documents papiers, il est fréquent de recourir à un archivage externe. Les documents contractuels, réglementaires, administratifs sont en effet souvent conservés dans des banques, chez des notaires, des experts

comptables, etc. En revanche, nous sommes naturellement plus méfiants lorsqu'il s'agit de données numériques.

On peut citer de nombreux exemples de cette méfiance :

– nous hésitons à communiquer nos données d'identité (âge, profession, etc.) sur Internet, alors que nous le faisons très facilement auprès d'un banquier ou d'un assureur sur simple demande orale ;

– les entreprises ne transmettent jamais d'informations numériques sur les salaires de leurs collaborateurs ; en revanche, elles recourent souvent à des prestataires externes pour imprimer et acheminer leurs feuilles de paye ;

– nous hésitons à payer par Carte Bleue sur Internet, alors que nous le faisons facilement dans un restaurant, où le risque est parfois plus grand (*cf.* pratique de la copie de Carte Bleue en arrière-boutique chez les escrocs) ;

– pour prendre un dernier exemple un peu provocateur : nous avons une totale confiance dans le restaurant de notre entreprise, tandis que nous nous méfions du stockage de documents chez Google. Et pourtant, ces deux prestataires sont encadrés par une législation stricte. Dans le premier cas, nous risquons notre vie, dans le second, un peu d'indiscrétion...

Il y a une grande part de « théorie du complot »[1] et de « peur de l'inconnu » dans cette méfiance vis-à-vis de l'informatique. L'informatique est, pour beaucoup de gens, mystérieuse et effrayante. La presse informatique participe d'ailleurs à cette paranoïa en relatant sans cesse l'apparition de nouveaux virus. Pourtant bien peu d'entreprises ont souffert de virus ces cinq dernières années.

Le seul argument qui justifie la méfiance vis-à-vis de l'informatique est la possibilité d'analyse à grande échelle qu'elle permet et que ne permet pas la lecture de millions de documents papier.

Pour nous, la confiance que l'on accorde ou pas à ses partenaires ne doit pas être conditionnée par leur mode de stockage : papier ou numérique pour les notaires, numérique pour les hébergeurs et les opérateurs cloud. La confiance doit s'établir sur des garanties contractuelles.

Pour aborder cette question de la confiance, on peut recourir à la **métaphore de la banque** : notre compte chèque est dématérialisé, nous ne savons pas où sont nos données bancaires, et nous faisons confiance à notre banque pour les gérer selon l'état de l'art. C'est comparable avec le cloud. De même que nous ne stockons pas nos billets de banque sous notre matelas par peur des voleurs, nous pouvons considérer qu'il vaut mieux conserver nos données dans les Datacenters industriels des opérateurs cloud (*cf.* chapitre 18). Pour nuancer ce discours, il faut noter que la différence entre banque et cloud se situe au niveau de la maturité : les acteurs du cloud sont plus récents, et on sait que la confiance s'installe dans le temps.

1. Les chercheurs en sciences sociales qualifient de « théorie du complot » les fantasmes de manipulation des masses par un groupe de pouvoir secret, minoritaire et élitiste. Il y a eu des théories du complot autour de l'assassinat de Marilyn Monroe, des attentats du 11 septembre, etc.

4.2 SÉCURITÉ ET ASPECTS JURIDIQUES

4.2.1 Les problématiques juridiques

Les problématiques réglementaires limitent les possibilités de recours au cloud. Dans certains secteurs d'activité, comme la banque, les entreprises doivent respecter des contraintes légales très fortes.

Il est nécessaire de considérer les réglementations sur :

- les données sectorielles (*cf.* lois Sarbanes-Oxley, suite du scandale Enron) ;
- les données personnelles (encadrées par la CNIL[1]).

Le cloud propose souvent une abstraction sur la localisation des données : cette abstraction vient compliquer la donne. En effet, il peut être difficile de déterminer le droit applicable : celui du fournisseur (souvent américain), du client ou du pays d'hébergement des données.

Par exemple, dans le cas de Google, on est amené à contractualiser avec une société de droit américain, dont le bureau européen est en Irlande, sans savoir où seront stockées les données. Dans le cas d'Amazon, c'est un peu plus simple car on peut choisir de faire héberger ses données en Europe, en l'occurrence en Irlande ou en Allemagne.

La localisation des données en Europe simplifie le problème juridique car les lois sur la protection des données métiers et personnelles sont cohérentes sur l'ensemble du continent. Il est possible de faire héberger des données personnelles aux États-Unis grâce à un accord transatlantique : le **Safe Harbour**. Malheureusement, tous les acteurs du *cloud* ne le proposent pas.

Il convient donc de faire clarifier ses engagements au fournisseur, de négocier le contrat pour qu'il respecte le droit du client et non le droit américain. Il faut avoir en tête qu'en cas de litige, il peut être difficile de faire respecter une décision de justice dans un pays tiers. Nous conseillons vivement de faire travailler le service juridique de l'entreprise sur ce contrat avant de signer quoi que ce soit. La négociation sera un peu complexe mais possible pour un grand compte. Elle sera quasi impossible pour une PME qui se verra imposer un contrat standard. Néanmoins, en France, les PME pourront s'appuyer sur le droit du consommateur pour annuler des clauses abusives.

Rappelons cependant que les éditeurs du « modèle software » (Microsoft, Oracle, etc.) ont des pratiques similaires : ils tentent aussi d'imposer des contrats standards de droit américain. Mais la problématique de localisation des données ne se pose pas avec eux.

1. Commission nationale de l'informatique et des libertés.

4.2.2 Sécurité et espionnage

Le **Patriot Act**, une loi votée à la suite des attentats du 11 septembre 2001, permet à l'administration américaine de demander l'ouverture de ses bases de données à toute société ayant son siège au États-Unis. Cette loi complique la confiance vis-à-vis des opérateurs cloud américains. Et ce, d'autant plus que les révélations d'Edward Snowden en 2013 ont révélé l'ampleur de l'espionnage pratiqué par la NSA[1].

En effet, ces révélations ont levé le voile sur la collecte de données effectuée par la NSA chez certains acteurs du cloud, comme Google et Microsoft.

Il existe probablement des actions similaires dans les administrations européennes, peut-être à moins grande échelle, en tout cas moins médiatisées.

4.2.3 Les certifications

Pour rassurer leurs clients sur leurs bonnes pratiques de sécurité, les opérateurs cloud ont souvent recours à des certifications. Les certifications les plus courantes sont :

- **ISO 27001 :** on a évoqué cette norme internationale plus haut. Elle représente la norme de référence de l'ISO[2]en matière de sécurité ;
- **SSAE[3] 16 type II et ISAE[4] 3402 Type II (successeurs de SAS[5] 70) :** destinée aux acteurs de l'outsourcing, cette certification garantit la conformité en matière de protection des données, la mise en œuvre de contrôles d'accès efficace dans les Datacenters, la mise en œuvre de politique de sauvegardes efficace. Elle est validée par un audit tous les six mois ;
- **FISMA[6] :** cette certification permet l'hébergement de services pour le compte des administrations américaines.

Ces certifications sont validées par des sociétés d'audit, comme Ernst & Young, qui effectuent un contrôle des pratiques de sécurité 1 à 2 fois par an.

Les grands acteurs comme Salesforce, Google, Microsoft, Amazon ont passé ces certifications.

Malgré tout, il n'est pas trivial de comprendre ce que recouvrent ces certifications. Il faut donc bien analyser les garanties offertes.

1. *National Security Agency*, aux États-Unis.
2. *International Organization for Standardization*.
3. *Statement on Standards for Attestation Engagements*.
4. *International Standards for Assurance Engagements*.
5. *Statement on Auditing Standards*.
6. *Federal Information Security Management Act*.

4.2.4 Un besoin de maturation

Les contrats proposés par les acteurs du cloud doivent clairement mûrir pour mieux répondre aux exigences des entreprises non américaines. Pour mieux servir leurs clients, ils doivent prendre en compte les législations locales, et non se ramener au droit américain.

Par ailleurs, il y a un vrai sujet sur les législations locales dans le monde globalisé du cloud et de l'Internet. Il est nécessaire que les législations s'homogénéisent.

En effet, les acteurs de l'Internet profitent abondamment des différences législatives pour tenter d'imposer des contrats à leur avantage, et parfois aussi pour faire de l'optimisation fiscale (c'est-à-dire ne pas payer leurs impôts en France).

Une clarification des droits et des actions de la NSA est aussi nécessaire pour crédibiliser les acteurs du cloud. Un meilleur encadrement de cette agence est en cours de discussion au Congrès américain. De plus, les acteurs du cloud ont renforcé le chiffrement des données échangées ou stockées afin de contrer la NSA. Ils rendent publiques les demandes d'informations de l'agence. Un bras de fer s'organise entre eux et le gouvernement américain.

Les problématiques juridiques et réglementaires constituent à notre avis le principal frein à l'adoption du cloud par les entreprises aujourd'hui.

4.3 SÉCURITÉ SUR LE PLAN TECHNIQUE

4.3.1 Analyse de risques

Lorsqu'on évalue la sécurité au sens informatique, on mène généralement une analyse au travers du prisme :

- authentification des utilisateurs ;
- confidentialité des données : stockées ou en transit sur le réseau ;
- intégrité des données : stockées ou en transit sur le réseau ;
- disponibilité ;
- traçabilité.

Ce paragraphe propose d'analyser les risques du cloud selon le prisme.

4.3.2 Cloud et authentification

L'authentification est une procédure qui consiste, pour un système informatique, à vérifier l'identité d'un accédant avant de lui autoriser l'accès au système.

Politique d'authentification par défaut

Les plateformes SaaS, IaaS et PaaS proposent, dans la grande majorité des cas, une classique authentification par identifiant et mot de passe.

Elles proposent rarement des « politiques de mot de passe » contraignantes, n'obligent donc pas à modifier les mots de passe régulièrement, pour ne pas compliquer la vie des utilisateurs. **Néanmoins, des politiques de mot de passe peuvent être activées chez les acteurs matures (Google, Amazon, Microsoft, Salesforce, etc.).**

À propos des politiques de mot de passe

Une « politique de mot de passe » sert à limiter les risques de vol de mot de passe et d'usurpation d'identité. Les pratiques courantes d'une telle politique sont les suivantes :
– changement du mot de passe tous les 3 mois ;
– interdiction de réutiliser un mot de passe précédemment utilisé ;
– mot de passe devant contenir au minimum huit caractères, et intégrant des lettres et des chiffres ;
– verrouillage du compte après cinq tentatives d'authentification avec un mot de passe erroné.

Ces mauvaises pratiques de politique de mot de passe sont issues du monde du web grand public. Elles se révèlent particulièrement dangereuses pour les comptes d'administrateurs. En effet, si une personne mal intentionnée arrive depuis le web à percer un mot de passe administrateur, elle peut à sa guise supprimer tous les comptes utilisateurs (SaaS) ou même une application (PaaS).

Dans le cas d'une messagerie externalisée comme Google Apps, le pirate qui a capturé le mot de passe administrateur peut tout simplement détruire les boîtes de messagerie de toute l'entreprise. Ce risque est généralement mieux maîtrisé pour une application interne à l'entreprise.

Authentification renforcée

De plus en plus d'acteurs du cloud proposent une authentification renforcée, reposant sur un mot de passe à durée de vie courte, généralement 30 secondes, en complément du mot de passe classique.

Ces systèmes d'authentification sont proches des « token » SecurID[1] de la société RSA. Mais les acteurs du cloud privilégient des systèmes logiciels à des objets matériels pour des raisons de facilité de distribution.

Ainsi Google propose Authenticator, une application facilement téléchargeable depuis l'AppStore Apple ou le Google Play Store. Microsoft, DropBox et Amazon reposent sur cette application pour proposer, eux aussi, une authentification renforcée. Des acteurs français comme OVH et Gandi proposent aussi d'utiliser cette solution.

1. Ces porte-clefs génèrent des mots de passe à usage unique et limités dans le temps. Il est nécessaire de les transporter sur soi pour connaître son mot de passe.

Selon nous, l'authentification renforcée constitue aujourd'hui l'état de l'art pour un administrateur de solution cloud.

Fédération d'identité

La plupart des opérateurs SaaS et PaaS proposent une option de délégation de l'authentification auprès de l'annuaire de l'entreprise, appelée fédération d'identité (voir chapitre 11). Avec cette option, l'utilisateur est redirigé vers le système d'information d'entreprise lors de la phase d'authentification. Lorsque celle-ci est terminée, il retourne en toute transparence vers la plateforme *cloud*. Cette opération utilise le principe du *Single Sign On* (SSO).

Google, Microsoft et Salesforce proposent en particulier un tel service. Cette option permet à l'entreprise de maîtriser sa politique de sécurité.

4.3.3 Cloud et confidentialité

La confidentialité porte sur la certitude qu'une donnée n'a pas été lue par une personne non habilitée ou malintentionnée. Elle doit être traitée pour des données persistantes et lors d'échanges avec des tiers.

La confidentialité des données hébergées chez l'opérateur cloud

Le risque d'espionnage par les collaborateurs de l'opérateur cloud est en principe couvert :

par le contrat entre l'entreprise utilisatrice et l'opérateur ;

par les certifications obtenues par l'opérateur (voir ci-dessus).

Ces garanties peuvent être considérées comme suffisantes ou non selon les cas. Par exemple, une entreprise qui dispose de secrets industriels hésitera probablement à les placer dans le cloud.

Notons qu'afin de se protéger de l'espionnage par la NSA, plusieurs acteurs cloud chiffrent maintenant leurs données.

La fin de ce chapitre abordera le rôle de la classification des données pour mitiger ce risque.

La confidentialité des données transitant sur le réseau

Les opérateurs cloud utilisent presque toujours le protocole SSL[1] pour assurer la confidentialité des données échangées. Son niveau de sécurité est suffisant. Certains opérateurs cloud, comme Amazon ou Microsoft, vont plus loin et proposent un chiffrement IPSEC[2], qui offre un plus haut niveau de sécurité.

1. *Secure Socket Layer* : c'est le protocole le plus utilisé pour chiffrer les échanges sur le web.
2. IPSec permet le chiffrement au niveau TCP/IP : le protocole travaille plus bas que SSL dans les couches réseaux. Il est donc plus sécurisé.

Notons qu'afin de se protéger de l'espionnage par la NSA, plusieurs acteurs cloud chiffrent maintenant les échanges de données au sein de leur propre réseau interne.

Un avantage au cloud ?

Dans le cas d'une entreprise qui gère ses serveurs sur son réseau informatique, il est fréquent d'avoir à gérer plusieurs scénarios d'accès :

- des utilisateurs qui accèdent aux applications depuis le siège ;
- des utilisateurs qui accèdent aux applications depuis un site secondaire ;
- des utilisateurs qui accèdent aux applications en situation de nomadisme.

Les responsables du réseau sont alors amenés à gérer plusieurs scénarios de sécurité, incluant authentification et chiffrement, en fonction de ces populations :

- accès direct pour les utilisateurs du siège ;
- VPN (*Virtual Private Network*) pour les utilisateurs du site distant ;
- Reverse Proxy pour les nomades.

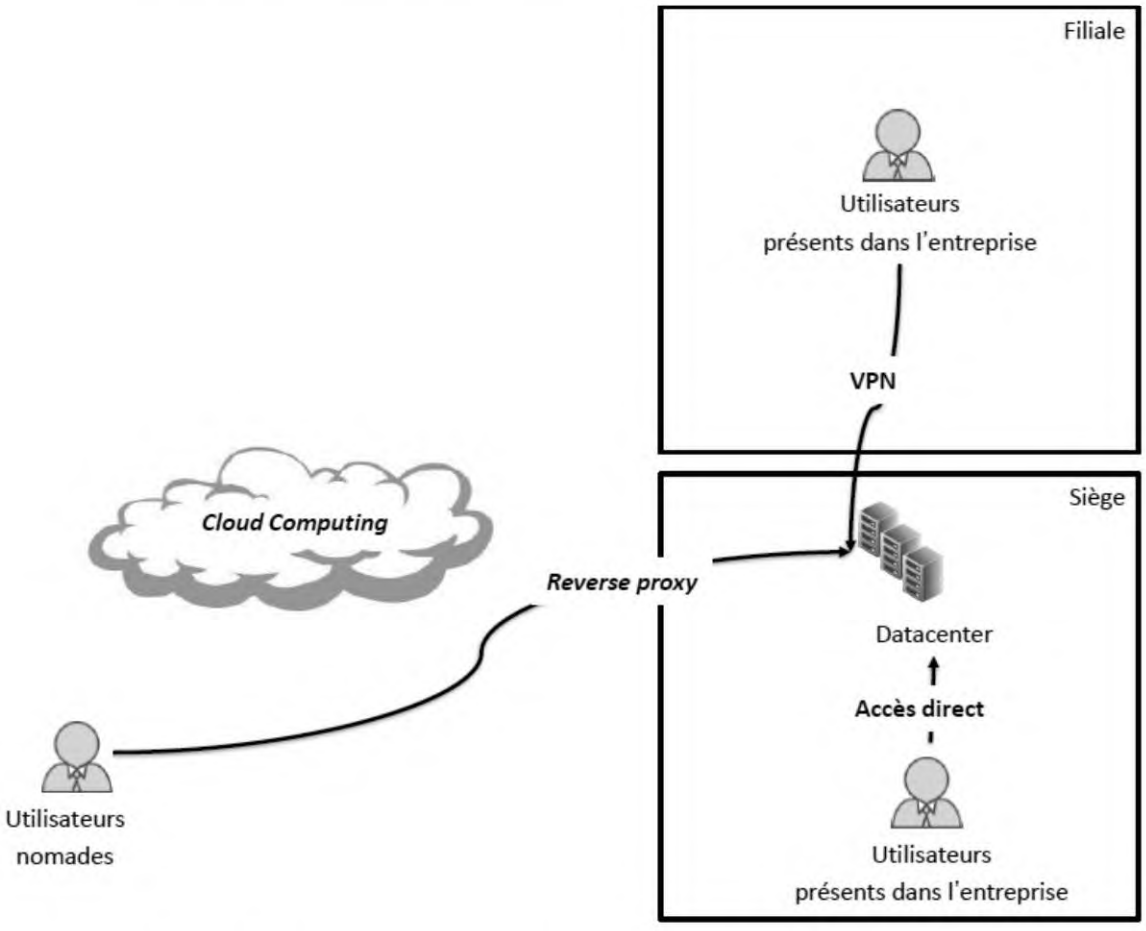

Figure 4.1 — Variétés des accès en entreprise et problématiques de sécurité.

Le but de ce paragraphe n'est pas de détailler ces mesures de sécurité, mais de souligner qu'elles sont complexes à mettre en œuvre, comme à maintenir. Cette complexité entraîne des coûts, mais parfois aussi des failles de sécurité : il peut en effet arriver qu'une des mesures de sécurité soit moins forte que les autres, ou bien qu'une erreur se soit glissée dans la complexité du paramétrage.

Avec les clouds, il existe un unique scénario de sécurité. En effet, que l'accès soit effectué depuis le siège, depuis une filiale ou depuis Internet, le système de sécurité est le même. Ainsi, les mesures de sécurité sont plus simples à déployer et à administrer.

Les grandes entreprises commencent aujourd'hui à adopter un nouveau modèle de sécurité qui considère que tous les utilisateurs représentent un danger pour les serveurs à cause de leur infection potentielle par des virus ou des chevaux de Troie. Il recommande donc de traiter les utilisateurs du siège et les nomades avec le même niveau de méfiance, et d'isoler les centres serveurs des réseaux d'utilisateurs. **Le modèle cloud s'intègre parfaitement avec ces pratiques.**

Par ailleurs, le modèle cloud permet aux populations nomades d'accéder au parc applicatif sans complexité additionnelle, ce qui est généralement une source de satisfaction pour elles.

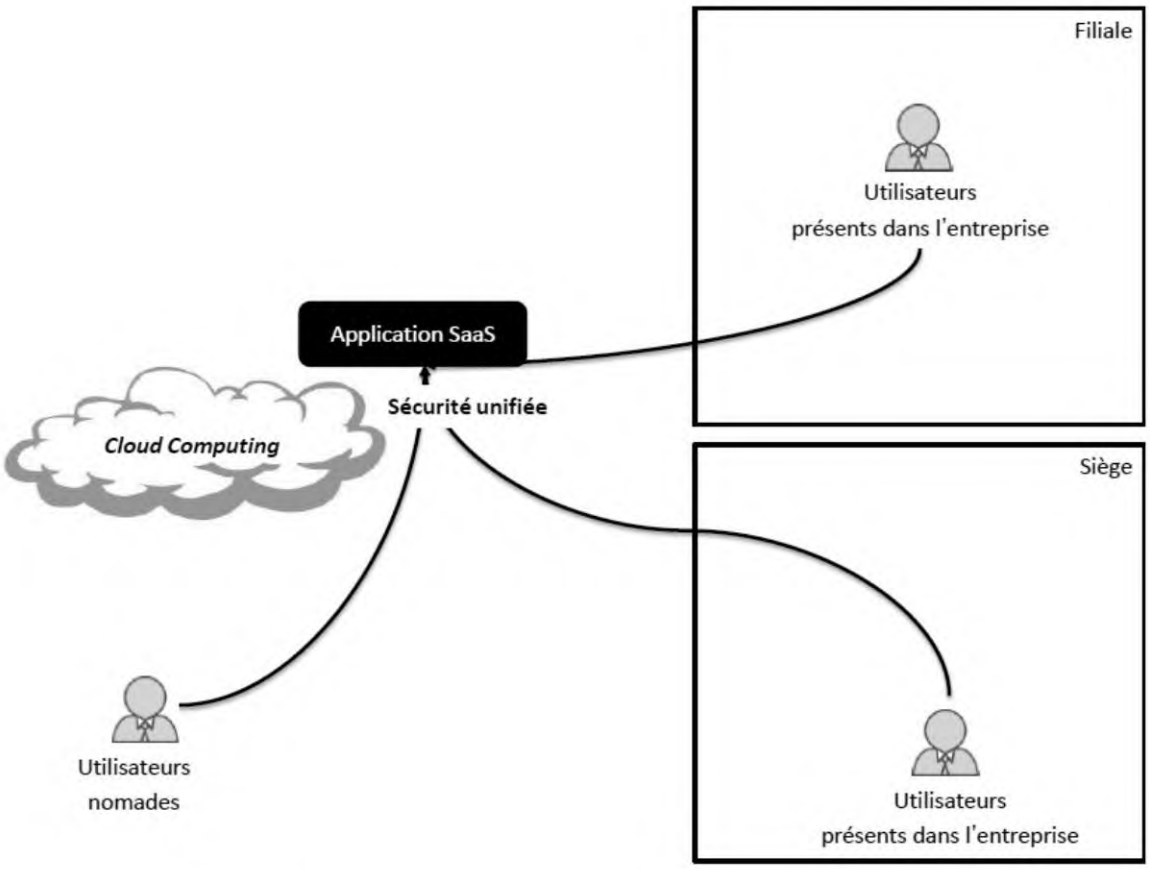

Figure 4.2 — Cloud et simplification des problématiques de sécurité.

Le cas d'INDUS

Monsieur Paul, directeur informatique d'INDUS, considère que la sécurité est mieux maîtrisée lorsqu'elle est internalisée. Ses collaborateurs ont mis en place de multiples passerelles de sécurité correspondant à plusieurs scénarios d'usage des applications.

Cependant, à force de multiplier les systèmes « maison », ils ont rendu le niveau de sécurité global difficile à mesurer. D'ailleurs, un audit commandé par le responsable de la sûreté de l'entreprise a montré des failles dans le système.

Il est ainsi apparu qu'une sécurité pensée de manière unifiée et centralisée par des experts était plus souhaitable que les pratiques hétérogènes de la DSI. Certains opérateurs cloud remplissent ces critères, selon le responsable de la sûreté.

4.3.4 Cloud et intégrité

L'intégrité porte sur la certitude qu'une donnée n'a pas été altérée de manière accidentelle ou malintentionnée. Elle doit être considérée pour des données persistantes et lors d'échanges avec des tiers.

L'intégrité des données hébergées chez l'opérateur cloud

Les opérateurs cloud disposent généralement de plusieurs Datacenters distants ce qui leur permet de dupliquer les données à grande échelle, de mettre en place des plans de reprise d'activité de grande qualité (*cf.* chapitre 18).

Google, par exemple, dispose de 13 Datacenters.

Figure 4.3 — Les Datacenters de Google dans le monde selon google.com/about/datacenters.

Ces Datacenters garantissent l'intégrité des données grâce à des processus que bien peu d'entreprises peuvent se permettre. En particulier, les PME bénéficient

grâce aux clouds d'un niveau de sécurité et d'un temps de remise en production impossibles à déployer avec une simple sauvegarde sur bande. Les grandes entreprises sont moins sensibles à cet argument : elles préfèrent gérer elles-mêmes leur plan de reprise d'activité.

Les politiques de sauvegardes et de réplication des opérateurs cloud ne sont pas toutes identiques. Il convient donc de se renseigner sur leur stratégie pour valider leurs garanties d'intégrité. C'est au RSSI de faire cette étude.

L'intégrité des données transitant sur le réseau

La problématique de l'intégrité lors des échanges est similaire à celle de la confidentialité : elle est garantie par SSL.

Un avantage au cloud ?

Il est assez fréquent que les populations nomades perdent leur PC portable ou le cassent. Les scénarios classiques sont les suivants :

- le collaborateur renverse du café sur son clavier, ou son enfant fait tomber la machine par terre : dans les deux cas, l'ordinateur est hors d'usage ;
- le collaborateur se fait voler son PC portable dans sa voiture, un train, un aéroport, etc.

Dans ces scénarios hélas courants, l'entreprise fait face à deux risques : celui d'un accès par des tiers non habilités à des données confidentielles et celui de la perte de données critiques.

Avec les clouds, les données ne sont pas stockées sur le poste de travail mais sur la plateforme de l'opérateur. Les deux risques ci-dessus sont donc écartés. **Le poste de travail change de statut : il n'est plus un espace de stockage mal maîtrisé par la DSI, mais une simple interface interchangeable.**

4.3.5 Cloud et disponibilité

La disponibilité d'une application désigne le ratio de temps pendant lequel elle est en état de fonctionner correctement sur une période de temps donnée. La disponibilité s'exprime en pourcentage.
Par exemple, une disponibilité de 99 % correspond à 4 jours d'inactivité par an, et une disponibilité de 99,9 % correspond à 9 heures d'inactivité par an.

La qualité de service offerte par les opérateurs *cloud* est souvent supérieure à celles que peuvent s'offrir les entreprises. Ainsi, leurs plateformes industrielles (*cf.* chapitre 18) garantissent généralement une disponibilité de 99,9 %, et parfois même de 99,99 %.

Il est néanmoins recommandé de contrôler cette disponibilité régulièrement *via* des outils de métrologie.

Par ailleurs, il faut bien être conscient que cette disponibilité est assurée en sortie du Datacenter de l'opérateur cloud, et non au niveau du poste utilisateur. Les aléas de la latence d'Internet peuvent avoir un impact sur la disponibilité perçue par les utilisateurs, en particulier dans le cas d'un Datacenter cloud situé hors de l'Europe.

La disponibilité de bout en bout peut être maîtrisée en réseau local, dans les murs de l'entreprise. Elle peut difficilement l'être dans le cadre du cloud. Certains opérateurs cloud disposent d'un réseau de serveurs de cache (CDN, *Content Delivery Network*) : il leur permet de mieux contrôler cette disponibilité de bout en bout.

La métaphore de l'avion

Certains acteurs du cloud (Microsoft et Amazon, par exemple) ont subi des pannes. Pour relativiser ces problèmes, on peut faire appel à la métaphore de l'avion. En effet, les catastrophes aériennes sont hyper-médiatisées, mais l'avion reste le moyen transport le plus sûr. De la même manière, la presse pointe le moindre défaut des acteurs du cloud, mais ne dit rien sur les accidents informatiques qui ont lieu dans les entreprises moins en vue. Lorsqu'on fait un bilan sur les cinq dernières années, il apparaît que le cloud est plutôt très fiable.

Il n'est pas simple pour une entreprise d'assurer un tel niveau de disponibilité sans coûts exorbitants.

Les utilisateurs sont les premiers bénéficiaires de cette qualité de service : en effet, les applications déployées sur le cloud ont des temps de réponse très rapides et des périodes d'indisponibilité très rares. Ce n'est pas toujours le cas des applications internes lorsqu'elles ne sont pas jugées stratégiques. Dans le cadre de l'informatique de commodité, la DSI souhaite parfois limiter ses coûts, et ce, aux dépens de la qualité de service.

Le cas d'INDUS

La messagerie d'INDUS est basée sur une infrastructure serveur qui date de 10 ans. Monsieur Paul, le directeur informatique, a un budget réduit et il préfère le consacrer à l'évolution de l'informatique métier.

De fait, la messagerie d'entreprise donne des signes de fatigue. Lorsque certains collaborateurs partent trop longtemps en vacances, leur boîte mail s'engorge, et toute la plateforme subit des ralentissements.

Les utilisateurs ont coutume de se plaindre de leur messagerie à la pause-café : ils la comparent souvent à Gmail ou Yahoo! Mail qui leur paraissent plus robustes. Certains d'entre eux transfèrent tous leurs messages professionnels vers Gmail : ils se demandent pourquoi la société ne migre tout simplement pas la messagerie vers la plateforme de Google.

4.3.6 Cloud et traçabilité

La traçabilité porte sur la génération de traces (souvent des « logs ») permettant de suivre le comportement des applications en production. Ces traces permettent de comprendre les comportements des utilisateurs et ceux d'éventuels pirates. Elles permettent aussi d'optimiser le fonctionnement des applications.

Les plateformes SaaS et PaaS offrent aujourd'hui des possibilités réduites en termes de traçabilité des actions utilisateurs : des API pour collecter les traces et les traiter au sein de l'entreprise. Certaines offrent parfois des tableaux de bords pour analyser ces données depuis leurs interfaces. Les opérateurs cloud SaaS et PaaS considèrent qu'ils fournissent un service intégré, qui débarrasse les DSI des problématiques de monitoring des applications : la gestion de la traçabilité est donc du ressort de leurs équipes internes. C'est pourquoi l'outillage fourni est limité.

Dans le cas de l'IaaS, c'est à l'équipe d'exploitation de déployer et de gérer ses outils de monitoring sur la plateforme.

4.3.7 Récapitulatif des réponses du cloud aux problématiques de sécurité

Le tableau 4.1 montre que l'argument sécuritaire pour contester le modèle cloud n'est pas valable sur le plan technique. D'autant plus que les opérateurs cloud peuvent consacrer beaucoup plus de moyens à fiabiliser et à sécuriser leurs plateformes qu'une entreprise utilisatrice. En définitive, la problématique essentielle tourne autour du juridique et de la confiance vis-à-vis de l'opérateur cloud sur la confidentialité de données.

Tableau 4.1 — Réponses des SaaS aux problématiques de sécurité.

Problématiques de sécurité	Réponses des SaaS
Authentification	Possibilité de gérer une politique de mot de passe, une authentification renforcée et de déléguer l'authentification chez certains opérateurs.
Confidentialité	Pas de vrai risque technique. Décision à prendre selon la politique de sécurité et les problèmes juridiques (NSA).
Intégrité	Pas de vrai risque technique. Garantie d'intégrité souvent supérieure à celle du SI.
Traçabilité	Peu de choses disponibles avec SaaS/PaaS. Idem SI pour IaaS.
Disponibilité	Pas de vrai risque technique. Garantie de SLA souvent supérieure à celle du SI.

4.4 GÉRER LE RISQUE DE SÉCURITÉ

4.4.1 Gestion du risque pour une PME

Les dirigeants de PME sont souvent ouverts à l'usage du cloud, car ils sont intéressés par les garanties sur l'intégrité des données offertes par ce modèle. Disposer de telles garanties en interne est en dehors de leurs moyens.

N'ayant pas un poids suffisant pour négocier, ils devront accepter les contrats sur étagères.

En ce qui concerne le risque sur la confidentialité, ils considérèrent généralement que leurs données seront noyées dans la masse de celles hébergées chez l'opérateur SaaS, et qu'elles ne revêtent pas un caractère assez intéressant pour donner lieu à des tentatives de piratage. En effet, qui s'intéresse aux e-mails, aux documents d'une PME lambda ? (On parle ici d'une PME qui n'aurait pas de secret industriel.)

L'usage du cloud pour une PME est donc un compromis entre des bénéfices d'intégrité et des risques de confidentialité.

4.4.2 Gestion du risque pour une grande entreprise

Prenons maintenant le cas d'une grande entreprise. Une telle entreprise peut considérer plusieurs typologies de risques :

- **risque de vol de secret industriel :** ce risque existe dans les secteurs très concurrentiels, par exemple, ceux où le dépôt de brevet est critique pour s'assurer de nouveaux marchés ;
- **risque de vol de données confidentielles sur ses clients :** ce risque existe dans des secteurs comme celui de la banque, où la protection des données est critique ;
- **risque de vol de données de fonctionnement interne :** ce risque porte essentiellement sur l'image d'une entreprise connue. Si le public apprenait qu'on lui a volé des données, son image serait salie.

La plupart des grandes entreprises gèrent ces risques au travers d'une « politique de sécurité »[1] mise en œuvre par le « responsable de la sûreté de l'entreprise ». Parfois, l'entreprise choisit de segmenter sa politique de sécurité en nommant un responsable de la sécurité informatique, le RSSI (Responsable de la sécurité du système d'information), et un responsable de la sécurité physique, en charge de la gestion des accès aux locaux. Nous avons choisi, dans cet ouvrage, de considérer un acteur unique : le « responsable de la sûreté de l'entreprise ».

La politique de sécurité est une pratique incontournable à laquelle les grandes entreprises consacrent généralement une équipe permanente. Cette équipe s'appuie sur des standards, comme la norme ISO 27001, pour produire une documentation sur les règles à respecter au sein de l'entreprise.

1. Voir Plouin G., Soyer J., Trioullier M.-É., *Sécurité des architectures web*, Dunod, 2004.

Parmi les documents issus de la politique de sécurité, on doit trouver une « **classification des données** » qui établit le degré de confidentialité des données manipulées dans l'entreprise. Cette classification permet de mener des « **analyses de risque** » sur les données lorsqu'on envisage leur externalisation, par exemple chez un opérateur cloud. Ces analyses de risque intégreront bien entendu les possibilités d'écoutes par la NSA. Le tableau 4.2 présente un exemple d'une telle « analyse de risque ».

La « classification des données » et l'« analyse de risque » sont des outils d'aide à la décision indispensables à tout arbitrage sur la sécurité des données.

Les grandes entreprises qui réfléchissent sur l'usage du cloud ne doivent pas prendre une décision hâtive basée uniquement sur la méfiance de leurs équipes, comme c'est parfois le cas. **Elles doivent utiliser les outils fournis par la « politique de sécurité ».**

Tableau 4.2 — Exemple d'analyse de risque pour l'externalisation de données chez un opérateur SaaS.

Type de donnée	Secret industriel	Données confidentielles clients	Données de fonctionnement interne
Classification	Stratégique	Critique	Confidentiel
Risque si externalisation	Très important	Important	Modéré
Externalisation	Impossible	Souvent impossible	Possible

À propos du risque d'intrusion

Quelques intrusions réussies ont défrayé la chronique ces derniers mois : par exemple les attaques de Sony Music et Ashley Madison. Ces deux attaques ont eu des répercussions désastreuses pour les entreprises concernées. Dans les deux cas, les pirates se sont attaqués à des SI autonomes. Les clouds Google, Microsoft, Amazon n'ont pas subi ce genre d'intrusion. Peut-être sont-ils mieux armés contre les hackers ? C'est impossible à établir...

4.4.3 Les tâches du RSSI vis-à-vis du cloud

Si l'entreprise souhaite utiliser des plateformes cloud, le RSSI va devoir intégrer de nouveaux éléments dans la politique de sécurité :

- une mention sur la possibilité d'externaliser vers le cloud au sein de la classification des données ;
- des règles sur les niveaux d'authentification que doivent offrir les plateformes cloud ;
- des règles sur les niveaux de sécurisation des flux SI/cloud ;
- des règles sur la réplication et la sauvegarde des données ;
- des règles sur le chiffrement des données persistant sur les plateformes cloud ;
- des règles sur la gestion des traces et logs ;

- des règles sur les SLA.

Le RSSI sera amené à homologuer certains opérateurs cloud. Il utilisera pour cela les certifications SAS 70 Type II et ISO 27001.

Le cloud constitue donc une nouvelle charge de travail assez conséquente.

En résumé

Ce chapitre a présenté brièvement les différents aspects de la sécurité du cloud computing :

– aspects juridiques, et réglementaires ;

– analyse de risques ;

– rôle du RSSI dans le contrôle de l'usage du cloud.

5

L'entreprise face au cloud computing

Objectif

L'objectif de ce chapitre est de situer les usages possibles du cloud computing pour les entreprises : il s'agit d'aborder les bénéfices et les risques suivant les secteurs d'activité et la taille des entreprises.

Le chapitre présente aussi quelques cas d'usage du cloud.

5.1 PAR QUELS USAGES DU CLOUD COMMENCER ?

Pour une entreprise, l'externalisation d'application commencera forcément par des applications peu stratégiques. En effet, les applications métiers stratégiques représentent le savoir-faire de l'entreprise, elles sont critiques pour sa survie, elles manipulent des informations particulièrement confidentielles et sont difficiles à externaliser. Par ailleurs, elles sont souvent développées en spécifique et il peut être difficile de les faire s'exécuter sur une plateforme tierce qui ne saurait pas parfaitement reproduire le socle d'exécution de l'entreprise. Enfin, les applications métiers échangent souvent des flux d'informations avec des partenaires : leur externalisation nécessiterait donc de déplacer ou de recréer ces flux.

A contrario, il existe une autre catégorie d'applications que nous appellerons ici « informatique de commodité ». Cette terminologie recouvre les applications génériques, dont le périmètre est bien maîtrisé et relativement standardisé, et que l'on

retrouve dans toutes les entreprises, indépendamment de leur métier. On peut citer par exemple :

- **les applications collaboratives** : la messagerie, les outils d'échange synchrone (visioconférence, messagerie instantanée, partage d'écran, etc.), les calendriers partagés, les espaces de partage documentaire, etc.
- **les applications de gestion des ressources humaines** : édition de feuilles de paie, suivi d'activité, demande de congés, etc.
- **les applications de gestion de la relation client** : suivi des affaires, organigramme des sociétés clientes, calendrier de relance, etc.
- **les applications de gestion financière** : suivi du chiffre d'affaires, des centres de coûts, des immobilisations, etc.

Les applications de commodité permettent à une entreprise de faire ses armes sur le modèle cloud, d'autant plus que ces applications échangent moins de flux d'informations que les applications métiers.

Il existe aussi des applications métiers assez génériques dans un secteur d'activité donné : elles pourront être consommées en mode SaaS, dans un second temps.

Enfin, les entreprises pourront, dans un troisième temps, déporter leurs applications stratégiques vers les PaaS/IaaS si leurs contraintes le leur permettent.

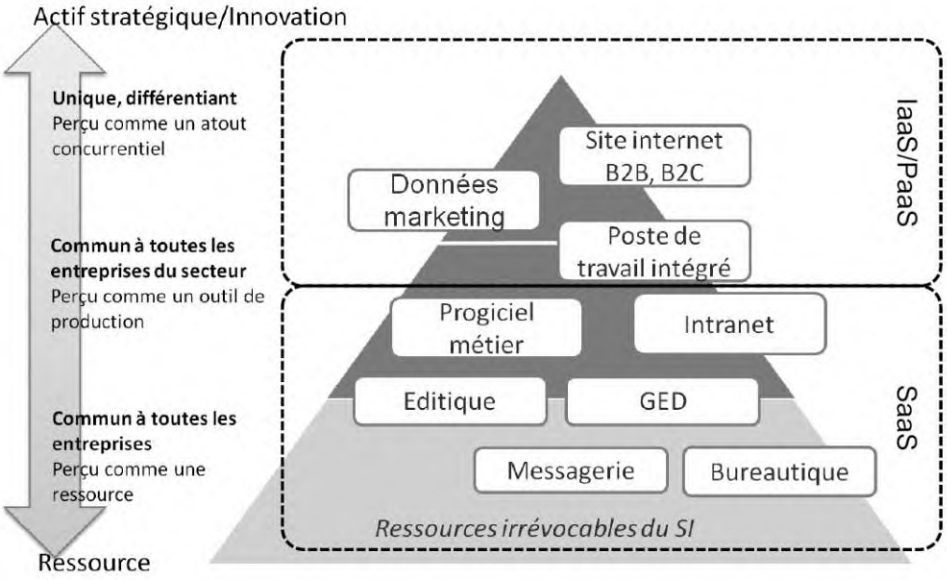

Figure 5.1 — Usage des XaaS par les entreprises.

5.2 LE CLOUD PUBLIC PAR SECTEUR D'ACTIVITÉ

Comme on l'a évoqué précédemment, certains secteurs d'activité sont plus réservés que d'autres vis-à-vis du cloud. Par exemple, la banque est un secteur à fortes contraintes réglementaires, ce qui rend difficile le recours au cloud public. De plus, les banques

ont, de manière historique, développé de fortes compétences informatiques car l'IT est indispensable à leur cœur de métier : la gestion de transactions financières. De fait, passer au cloud signifierait un changement culturel important et poserait le problème du reclassement du personnel existant.

Le cas du secteur des télécommunications est particulier : en effet, les grands opérateurs se posent parfois en concurrents des clouds publics, souhaitant bâtir leur propre offre. Et le cloud est d'autant plus pertinent pour eux que leurs services informatiques pourraient être les premiers clients de leurs infrastructures publiques.

Dans d'autres secteurs, comme l'industrie, l'informatique est plus vue comme une commodité, dont il faut réduire les coûts au maximum. Le recours au cloud sera donc plus aisé.

Enfin, dans le secteur de l'Internet et du web, le recours au cloud est naturel, dans la continuité du web 2.0.

Nous avons ainsi tenté de synthétiser notre vision de l'adoption du cloud par secteur d'activité dans la figure 5.2. Cette vision est valable fin 2015.

Éligibilité du cloud

Internet / médias / tourisme	Industrie / énergie	Télécoms	Assurance / banque / finance	Services publics
Recours au cloud naturel.	Recours au cloud possible. Informatique = commodité	Recours au cloud envisageable. Parfois opérateur cloud.	Recours au cloud ponctuel, pour raisons réglementaires et culture informatique.	Recours au cloud souhaité pour raisons financières, mais difficile pour raisons réglementaires.

Figure 5.2 — Éligibilité du *cloud* par secteur d'activité.

Il est aussi pertinent de faire un distinguo selon la taille de l'entreprise utilisatrice. Ainsi, les PME et les start-up sont naturellement avantagées par le cloud computing qui leur permet d'éviter un investissement dans des infrastructures informatiques au démarrage de leur activité. Sur le plan financier, ces petites structures bénéficient beaucoup plus qu'un grand compte de l'effet d'échelle lié à la mutualisation de ressources entre plusieurs entreprises. Et les PME ont souvent une vision différente des grands comptes sur la confidentialité des données : elles considèrent souvent que leurs données sont « noyées dans la masse ». Nous reviendrons sur ces aspects dans les chapitres 6 et 8.

Paradoxalement, les petites entreprises ont un recours limité au cloud par manque d'information. Elles ont peu de ressource pour faire de la veille. Elles sont donc mal informées sur l'état de l'art du SaaS. De plus, un certain nombre de petits patrons sont

rebutés par la lecture en anglais, et il est vrai que le cloud parle encore beaucoup en anglais et en dollars.

5.3 QUEL MODE D'UTILISATION DU CLOUD ?

Une entreprise peut tirer parti des applications en ligne suivant plusieurs approches, correspondant à divers degrés d'investissement dans le modèle cloud.

L'adoption des XaaS en France ?

Selon une étude de Markess International sur les entreprises françaises en juillet 2015 :
– 80 % des entreprises utilisent le SaaS,
– 28 % utilisent l'IaaS,
– 11 % utilisent le PaaS.

5.3.1 Approche 1 : usage d'une application SaaS

Dans cette approche, l'entreprise se lance dans le modèle cloud et externalise une partie de son informatique de commodité, par exemple son logiciel de gestion de la relation client.

Dans ce scénario, des données confidentielles de l'entreprise sont stockées chez un tiers, l'opérateur SaaS. Il sera donc nécessaire de s'assurer de la non-divulgation de ces données. Par ailleurs, si l'application se révèle indisponible, l'impact peut être assez significatif : l'équipe commerciale peut en effet se retrouver au chômage technique.

Enfin, si le fournisseur du service venait à disparaître, il serait possible de le remplacer par un autre. Bien sûr, ce changement ne serait pas trivial car il faudrait reprendre toutes les données pour les réimporter chez le nouveau fournisseur, mais il serait surmontable. Contractuellement, il est indispensable de prévoir des clauses de réversibilité pour préserver l'accès à la donnée même en cas de défaut du fournisseur.

Tableau 5.1 — Risques liés à l'usage d'une application SaaS « de commodité ».

Risque sur la confidentialité des données	Risque si indisponibilité plateforme cloud	Risque en cas de disparition de l'opérateur cloud
Existant, à traiter	Critique	Minime

Un panorama des SaaS non-métiers disponibles aujourd'hui sera proposé aux chapitres 14, 15 et 16.

Avec le recours au SaaS, la DSI devra gérer la création/suppression des comptes utilisateurs sur la plateforme cloud, c'est-à-dire le *provisioning* des comptes (*cf.* chapitre 9).

Figure 5.3 — Impacts organisationnels du SaaS pour la DSI.

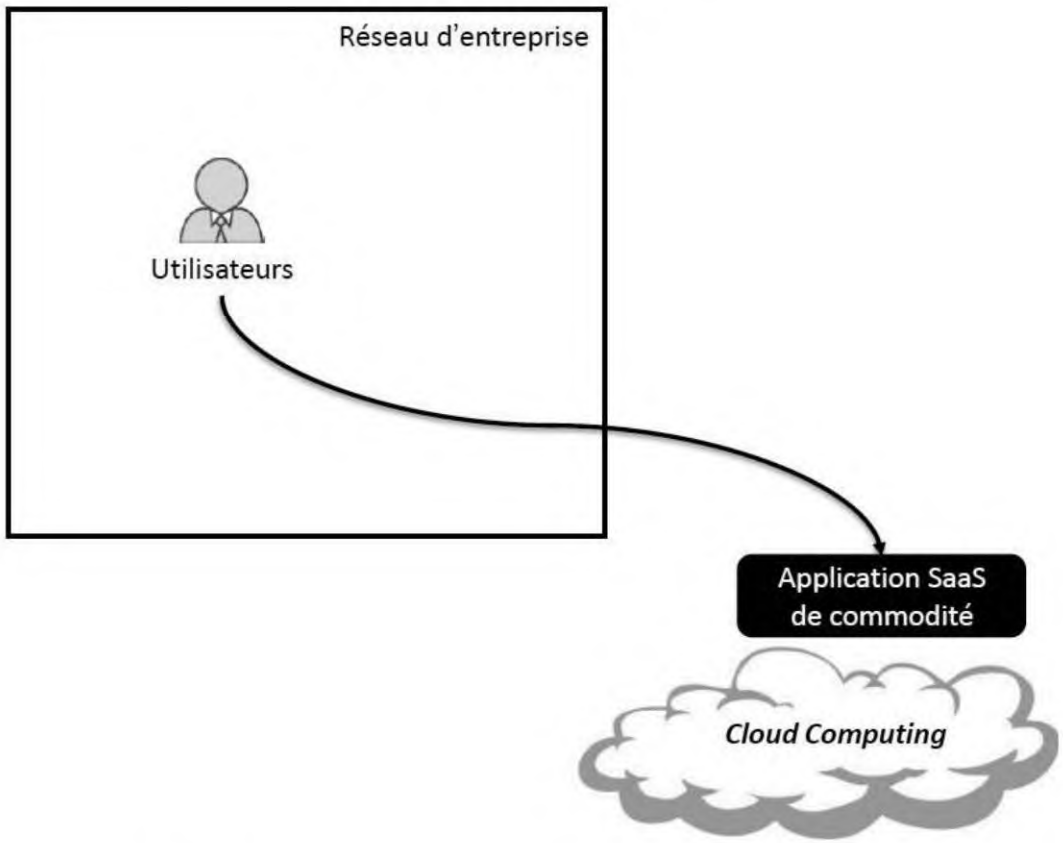

Figure 5.4 — Usage d'une application SaaS « de commodité ».

5.3.2 Approche 2 : déploiement d'applications sur IaaS

Dans cette approche, l'entreprise utilise une plateforme IaaS, pour exécuter une application métier sans assurer l'exploitation du Datacenter. Elle gère la couche logicielle de son application et délègue la partie matérielle. L'entreprise maîtrise son architecture sur une plateforme IaaS. La disparition de l'opérateur est modérément critique : il est possible de redéployer l'architecture chez un autre acteur. Il va sans dire qu'une indisponibilité de la plateforme ou un vol de données serait très dommageable.

Il existe d'autres cas d'usages intéressant des plateformes IaaS. Elles peuvent être utilisées comme environnement de développement, test et recette, sans pour autant assurer la production d'une application. Elles peuvent aussi servir à tester de nouveaux concepts en hébergeant des maquettes applicatives à durée de vie courte.

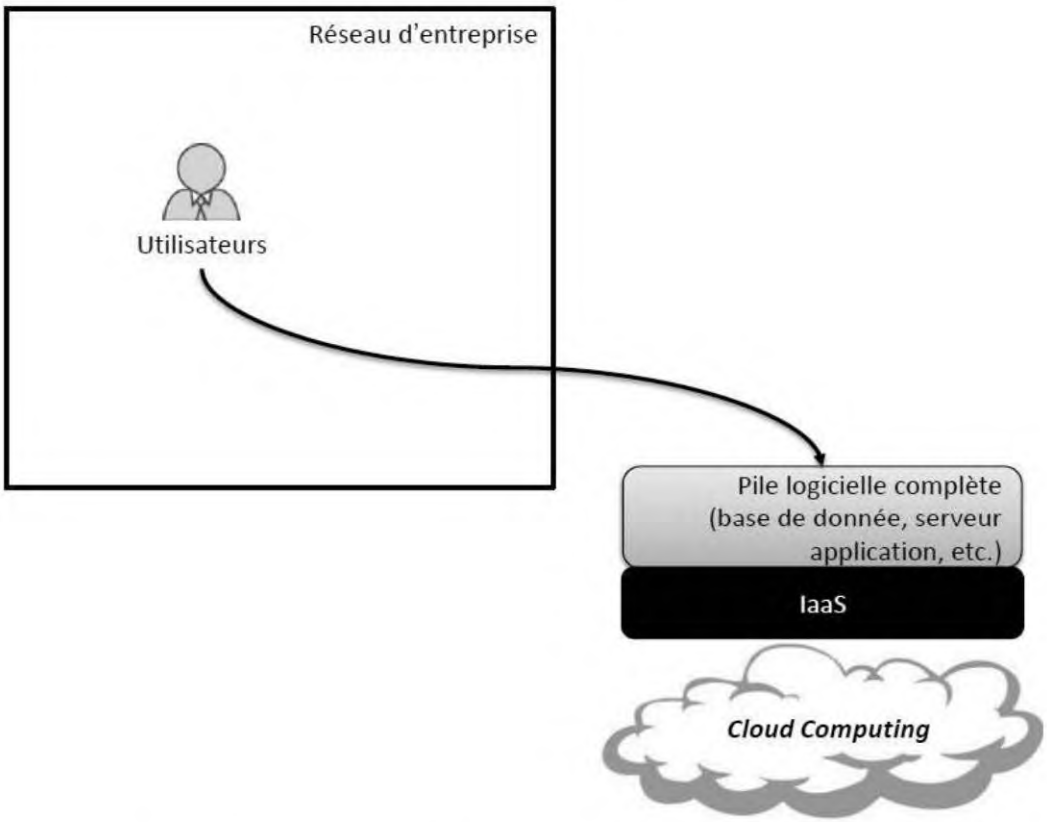

Figure 5.5 — Usage d'une plateforme IaaS pour l'hébergement d'une application métier.

Tableau 5.2 — Risques liés à l'usage d'une plateforme IaaS pour le développement d'une application métier.

Risque sur la confidentialité des données	Risque si indisponibilité plateforme cloud	Risque en cas de disparition de l'opérateur cloud
Existant, à traiter	Critique	Modérément critique

Un panorama des plateformes IaaS disponibles aujourd'hui sera proposé au chapitre 20.

L'IaaS change le mode de travail de l'équipe d'exploitation : en effet, celle-ci doit travailler sur une plateforme déportée à laquelle elle n'a pas d'accès physique.

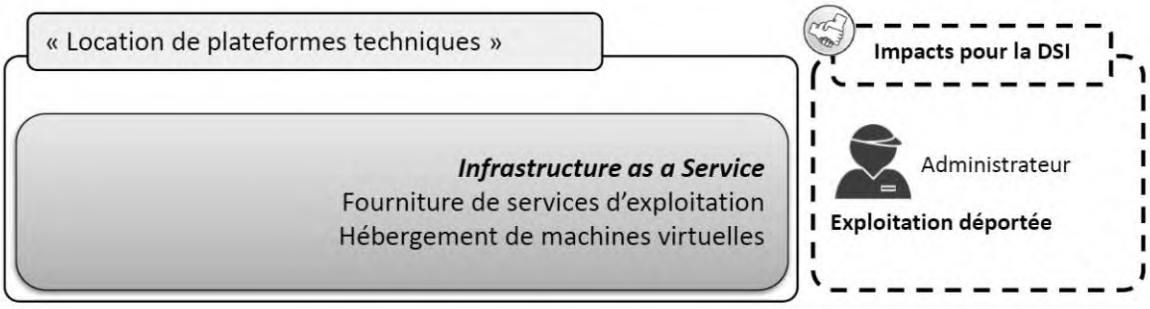

Figure 5.6 — Impacts organisationnels de l'IaaS pour la DSI.

5.3.3 Approche 3 : développement d'applications sur PaaS

Dans cette dernière approche, l'entreprise utilise une plateforme PaaS, pour développer une application métier sans en assurer la production.

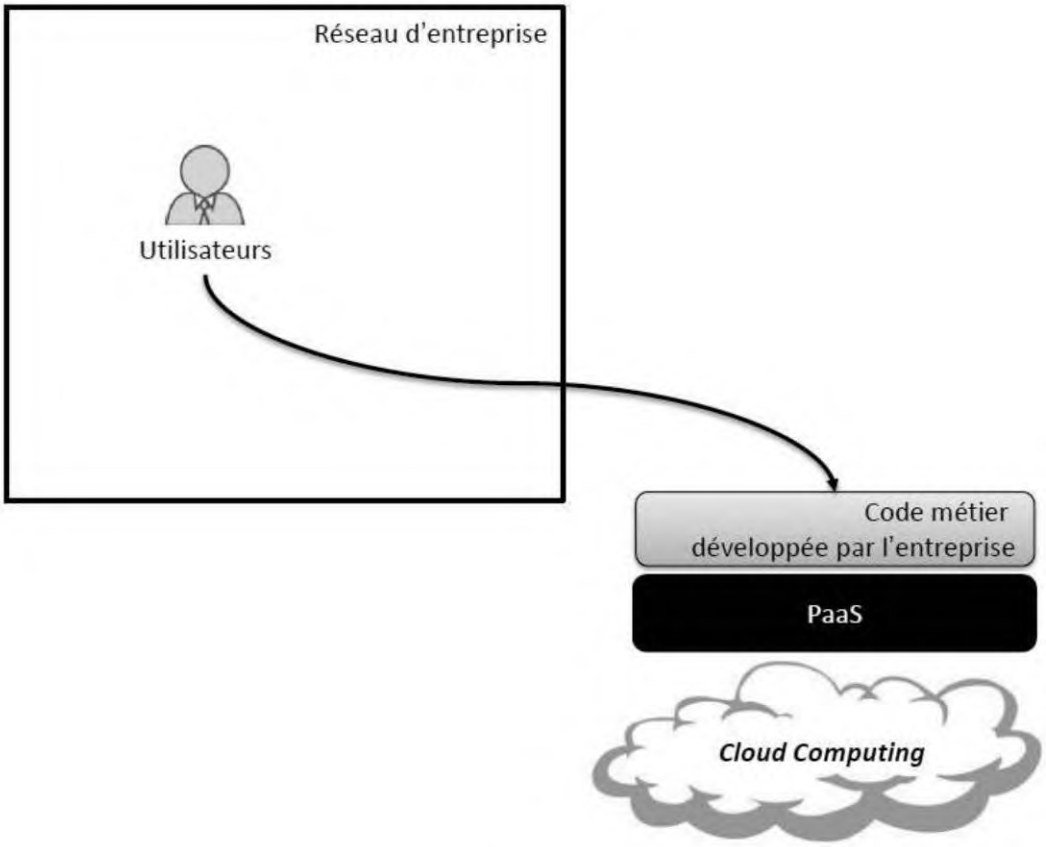

Figure 5.7 — Usage d'une plateforme PaaS pour le développement d'une application métier.

Comme on le verra dans le chapitre 20, les plateformes PaaS ne sont pas encore standardisées et une application développée sur une plateforme PaaS A pourra difficilement être migrée vers une plateforme PaaS B. La disparition de l'opérateur de la plateforme PaaS A serait donc critique pour l'entreprise utilisatrice, qui devrait

réécrire son application. Il va sans dire qu'une indisponibilité de la plateforme ou un vol de données serait très dommageable.

Tableau 5.3 — Risques liés à l'usage d'une plateforme PaaS pour le développement d'une application métier.

Risque sur la confidentialité des données	Risque si indisponibilité plateforme cloud	Risque en cas de disparition de l'opérateur cloud
Existant, à traiter	Critique	Critique si PaaS

Un panorama des plateformes PaaS disponibles aujourd'hui sera proposé au chapitre 20.

Avec le recours au PaaS, les développeurs codent leur application en respectant les contraintes architecturales de la plateforme. Puis, ils testent et déploient leurs applications en direct, sans passer par l'équipe d'exploitation.

Figure 5.8 — Impacts organisationnels du PaaS pour la DSI.

En résumé

Ce chapitre a présenté brièvement les trois cas d'usages du cloud computing :

– le recours à une application SaaS ;

– le recours à l'hébergement sur une plateforme IaaS ;

– le recours au développement sur une plateforme PaaS.

Les offres disponibles pour ces trois cas d'usages seront présentées dans les quatrième et cinquième parties.

6

Bénéfices et inconvénients du point de vue des décideurs

Objectif

Comme son nom l'indique, l'objectif de ce chapitre est d'introduire le point de vue des décideurs vis-à-vis du cloud computing.

On abordera ici des aspects qui touchent à la stratégie et aux coûts, aux problématiques juridiques et à la relation au fournisseur.

6.1 BÉNÉFICES POUR L'ENTREPRISE UTILISATRICE

6.1.1 La réduction des coûts liés aux infrastructures

Le premier bénéfice du cloud computing perçu par les entreprises est la réduction des coûts. Notons que cette réduction n'est pas avérée dans tous les scénarios d'usage du cloud.

La **réduction des coûts liés aux équipes d'exploitation** est possible grâce à la mutualisation ; avec le cloud computing, une équipe unique gère les applications de plusieurs centaines d'entreprises. Les coûts salariaux sont donc moindres que ceux

correspondant à une centaine d'équipes dans laquelle les ressources sont potentiellement sous-employées. Dans une équipe mutualisée, il est aussi possible de spécialiser les collaborateurs et donc de les employer au mieux de leurs compétences. Par ailleurs, la gestion d'infrastructures importantes incite à la mise en place d'outils spécialisés. Ainsi, les équipes d'exploitation des clouds déploient des outils de surveillance des infrastructures, de reprise sur incident en cas de défaillance de serveur, ou de déploiement qui sont complètement adaptés à leurs besoins ; ce qui n'est pas le cas des équipes d'exploitations plus petites qui doivent utiliser des outils génériques et coûteux du marché.

Par ailleurs, pour assurer un haut niveau de sécurité sur ses applications informatiques, en particulier en cas de sinistre grave, comme un incendie ou une inondation, il est nécessaire de disposer d'un « **plan de reprise d'activité** ». Ce plan consiste généralement à mettre en œuvre un système de basculement vers un autre centre serveur sur un lieu distant. La pratique du « plan de reprise d'activité » nécessite de disposer de plusieurs Datacenters distants, et donc de plusieurs équipes d'exploitation. Là encore, ce coût est plus facile à lisser pour les opérateurs cloud grâce à la mutualisation.

La **réduction des coûts d'exploitation** est aussi favorisée par l'automatisation. On verra, dans la cinquième partie, que les acteurs du cloud computing déploient des systèmes automatisés à un degré avancé. **Cet outillage est d'ailleurs le cœur de métiers des opérateurs cloud, leur « secret industriel ».** Ainsi, chez Google, l'abonnement à l'offre de collaboration ne nécessite aucune intervention humaine. De la même manière, en cas de défaillance d'une machine, la reprise de traitements par un autre serveur est automatisée. Un luxe que peu d'entreprises peuvent se permettre à l'heure actuelle.

Les opérateurs cloud déploient d'importants centres serveurs, souvent plus importants que ceux des entreprises. De fait, ils **achètent des machines en gros** et peuvent négocier les prix à la baisse. **Certains, comme Google, vont jusqu'à concevoir et réaliser en interne leurs serveurs, routeurs, et Datacenters.** De plus, ils ont les moyens d'optimiser l'usage des ressources de leur parc serveur, tandis que les entreprises exploitent souvent des serveurs utilisés à 50 % de leurs capacités, ce qui induit des surcoûts. Par ailleurs, les plateformes de test, recette et préproduction cloud représentent un investissement négligeable sur l'ensemble de leurs parcs de machines : a contrario, pour de nombreuses entreprises, ces plateformes multiplient par deux la facture matérielle.

Les opérateurs cloud **achètent leur énergie en gros** et peuvent donc négocier le prix de l'électricité. Pour citer encore l'exemple de Google, il semble que le leader des moteurs de recherche ait placé certains de ses Datacenters près de centrales électriques pour réduire sa facture d'énergie. Le déport des problématiques d'achat d'énergie est d'autant plus intéressant que les coûts énergétiques des Datacenters sont en pleine explosion (figure 6.1).

Enfin, comme on l'a vu dans le chapitre 3, les opérateurs cloud utilisent fréquemment des logiciels serveurs Open Source pour minimiser leurs frais d'achats en logiciel :

une pratique rare dans les entreprises qui préfèrent souvent utiliser des logiciels simples à prendre en main comme ceux de Microsoft.

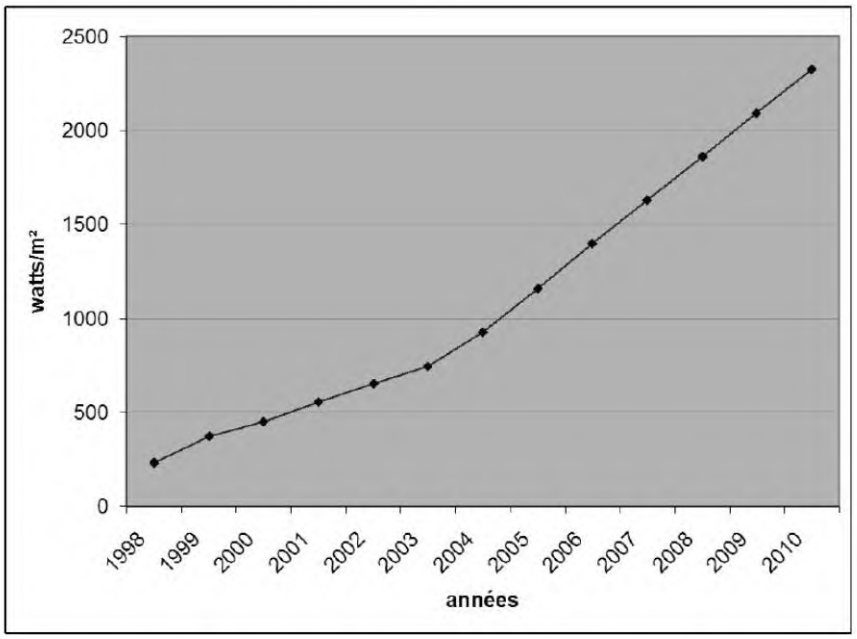

Figure 6.1 — Croissance des coûts énergétiques des Datacenters.

L'ensemble des facteurs évoqués ci-dessus concourt au moindre coût de possession d'une application cloud par rapport à une application internalisée.

6.1.2 Le « green IT »

Certaines entreprises sont sensibles à l'impact environnemental de l'informatique et ont lancé une démarche « green IT », ou informatique verte, pour limiter cet impact.

Une grande partie de cet impact est liée à la consommation énergétique des postes de travail. Or, on a vu, dans le chapitre 1, que les interfaces HTML5 utilisées par le cloud permettaient de déployer des postes de travail allégés, moins gourmands en ressources.

Une autre grande partie de l'impact environnemental de l'informatique est liée au gaspillage énergétique des Datacenters. Les acteurs du cloud computing sont assez avancés dans la réduction de ces impacts : ils ont des dispositifs de ventilation mieux optimisés et ils commencent à utiliser des énergies renouvelables dans leurs Datacenters.

Le recours au cloud computing peut donc s'inscrire dans une démarche « green IT » de développement durable.

Cependant, avec leur promesse de grande capacité de stockage, certaines offres cloud incitent à l'absence de sobriété et au gaspillage de ressources. Est-il vraiment utile de conserver des e-mails durant 15 ans, sachant la consommation électrique que cela occasionne ? Cette question remet en cause le caractère écologique du cloud.

6.1.3 La réduction des coûts d'usage

Les éléments présentés dans le paragraphe précédent montrent que les opérateurs cloud parviennent à réduire leurs coûts et donc à proposer des tarifs qui permettent un TCO inférieur à celui d'une solution internalisée.

De plus, comme on l'a vu au chapitre 3, les services cloud sont proposés en *Pay As You Go*, paiement à la consommation. L'abonnement revient presque toujours moins cher que l'internalisation (*cf.* encadré). De plus, le paiement d'un abonnement est très intéressant pour les entreprises qui souhaitent lisser leurs coûts dans le temps.

Le mode cloud permet aussi de disposer en permanence d'outils à la pointe, sans avoir à se soucier du reclassement des matériels obsolètes.

De nombreuses entreprises se tournent vers la location ou le leasing pour leurs locaux, les flottes de véhicules, leurs parcs de PC, etc. En suivant cette logique, il est naturel d'aller vers le modèle cloud.

6.1.4 Une meilleure sécurité

Comme on l'a vu au chapitre 4, le cloud offre de bonnes garanties sur l'intégrité des données.

6.1.5 Un recentrage sur le métier

On aborde généralement le cloud computing par l'« informatique de commodité », et non par l'informatique métier (*cf.* chapitre 5).

Dans ces conditions, le modèle cloud est intéressant pour l'entreprise car elle lui permet de recentrer ses efforts sur son cœur de métier. En effet, toutes les ressources et compétences de la DSI se consacrent alors à améliorer l'agilité du système d'information pour répondre au mieux et au plus vite aux maîtrises d'ouvrage sur des besoins réellement spécifiques. La démarche SOA[1] pourra fournir une méthodologie structurante pour arriver à cet objectif. Les gains pour l'entreprise seront :

- une plus grande optimisation des processus métiers, et donc des gains de productivité ;
- des nouvelles fonctionnalités métiers déployées plus vite, éventuellement avant la concurrence, ce qui permettra de gagner des parts de marché ;
- un système d'information métier mieux structuré et donc plus pérenne ;
- un environnement « bac à sable » dédié à l'innovation et à la création rapide de maquettes pour les faire tester par les utilisateurs.

Ce recentrage aura aussi un impact positif sur les collaborateurs de la DSI : ils travailleront sur des problématiques plus intéressantes que, par exemple, du dépannage

1. Voir Fournier-Morel X., Grojean P., Plouin G., Rognon C. SOA. *Le guide de l'architecte d'un SI agile*, 3e édition, Dunod, 2011.

de serveur de messagerie. Ils seront plus proches des interlocuteurs métiers de leur entreprise et auront l'impression d'être plus valorisés en participant à sa croissance. Le turnover de la DSI sera donc réduit, et ses collaborateurs auront une meilleure connaissance métier qui leur permettra d'évoluer vers un poste moins technique si tel est leur souhait. Le modèle cloud contribue à ces évolutions positives.

Le cas d'INDUS

Les équipes informatiques sont associées, dans l'esprit des collaborateurs d'INDUS, à des problèmes de plantage de Windows et de messagerie qui ne marche pas.

Monsieur François, directeur informatique, est exaspéré par cet état de fait : les collaborateurs ignorent totalement l'énorme travail de la DSI sur les applications métiers.

Il est parfois tenté par l'infogérance de la bureautique, mais il hésite car il ne veut pas mettre une partie de son équipe au chômage technique.

6.1.6 La fin du syndrome de l'« administrateur héroïque »

Il y a, dans certaines entreprises, des équipes d'exploitation qui, faute de formation, de temps et de moyens, effectuent des paramétrages de type « boîte noire ». Nous entendons par là des configurations de logiciels serveurs customisées, non conformes à l'état de l'art et maîtrisées par une seule personne, parfois deux. Ces pratiques existent plus souvent dans le cadre de l'informatique de commodité que dans celui de l'informatique métier, plus stratégique donc mieux documentée et mieux maîtrisée. Il est clair que ces pratiques rendent l'entreprise dépendante d'un « administrateur héroïque », seul à pouvoir intervenir sur les serveurs. Les congés de cet administrateur sont souvent sources de blocages et il est généralement amené à faire du support téléphonique pendant cette période. Par ailleurs, un accident ou une démission touchant cette personne sont très ennuyeux pour l'entreprise.

Ces configurations risquées sont parfois aussi liées à une volonté de contrôle sur les infrastructures, pour se rendre indispensable afin de mieux négocier une augmentation.

Ce syndrome touche plutôt les petites entreprises que les grandes. Dans tous les cas, le syndrome de l'« administrateur héroïque » constitue un risque pour l'entreprise qui disparaît avec les clouds. En effet, ces derniers fournissent une console d'administration simple sous la forme d'une interface web. La console est standardisée et facile à appréhender : elle peut être prise en main par un informaticien non spécialisé. Il n'est, par exemple, pas nécessaire qu'il connaisse les arcanes de l'optimisation système. La compréhension des grands concepts et des connaissances standards est suffisante. De ce fait, il est assez aisé de faire un transfert de compétence vers un autre informaticien.

6.2 INCONVÉNIENTS POUR L'ENTREPRISE UTILISATRICE

6.2.1 Les problématiques d'achat

On l'a vu précédemment, les acteurs du cloud sont souvent issus du web. De fait, ils utilisent les solutions d'achats des sites de commerce électronique : paiement par Carte Bleue, PayPal, etc. Si ce modèle de paiement convient aux PME, il ne correspond pas aux habitudes des services achats des grands comptes. Ces services ont rarement une Carte Bleue à disposition pour les achats en ligne, et n'ont pas l'habitude de s'engager sans connaître un montant à l'avance (*Pay As You Go*).

Des acteurs comme Google, Amazon sont capables de travailler avec des factures en relation avec des services achats, mais ce n'est pas la démarche qu'ils privilégient. Ils préfèrent que tout passe par le web (le fameux Self Service). Ils privilégient l'approche « ***Customer as a commodity*** », c'est-à-dire un client standard, sans spécificité.

Ainsi Google dispose d'une dizaine de commerciaux en France pour Google Apps, et Amazon a ouvert son premier bureau en France en 2012. Pour toute entreprise qui n'est pas un grand compte, il est difficile d'avoir un interlocuteur humain : on les pousse à passer par le réseau de partenaires, un réseau qui n'a pas de latitude de négociations sur les contrats et les tarifs. Les choses sont cependant en train d'évoluer du fait de la demande du marché : il est probable que les acteurs du cloud s'équipent de forces de vente comparables à celles des éditeurs.

Le cloud computing peut donc représenter une rupture pour les services achats. C'est moins vrai avec certains acteurs comme Salesforce ou Microsoft qui proposent une relation « classique » à leurs clients entreprise. Néanmoins, dans tous les cas, le cloud propose un modèle de facturation par abonnement au mois, qui n'est pas toujours habituel pour les services achats.

Enfin, les contrats standards des acteurs du cloud prévoient des pénalités peu contraignantes en cas de rupture de service. Ainsi, si Google Apps connaît une panne de 24 heures, Google vous offre 24 heures de service en plus. Cet engagement sera vu comme insuffisant par beaucoup d'entreprises, même si les ruptures de services de Google Apps ont été rarissimes au cours des trois dernières années.

6.2.2 Les risques commerciaux

Le fait que les données soient hébergées chez l'opérateur cloud introduit un risque commercial : que se passe-t-il si l'opérateur dépose le bilan ? S'il change ses conditions ou tarifs ? Si la promesse de réversibilité des données n'est pas tenue ? Si l'État américain décide de faire cesser l'activité de l'opérateur cloud (*cf.* les anecdotes de Wikileaks et de Megaupload) ?

Il n'est pas simple de gérer ces situations de crise avec un simple formulaire sur le web... Il est important de bien mesurer ces incertitudes et d'en tenir compte au moment du choix de son opérateur cloud.

6.2.3 Un rejet de la part des clients

De nombreuses entreprises détiennent, au sein de leur système d'information, des données qui concernent leurs clients. Il peut s'agir d'e-mails, de documents, de données de suivi de la relation client, etc. Lorsque ces entreprises se penchent sur le modèle cloud pour des données concernant leurs clients, elles sont tenues de prendre en compte l'avis de ces derniers.

Par exemple, une société de services qui souhaite s'appuyer sur un espace collaboratif SaaS comme DropBox (*cf.* partie 4) devra au préalable demander l'avis de son client, afin de s'assurer que les données à partager ne sont pas classifiées comme critiques par ce dernier. En effet, l'externalisation des données concernées pourrait aller à l'encontre de la politique de sécurité de ce client.

Il est donc essentiel d'un point de vue déontologique d'informer ses clients et de leur faire valider le recours au modèle cloud.

Le client de l'entreprise qui souhaite recourir au modèle cloud peut avoir plusieurs types de réaction :

- une réaction de rejet liée à une méconnaissance du modèle (*cf.* la « théorie du complot »). Dans ce cas, on pourra essayer d'argumenter pour le convaincre. La réduction des coûts liée au modèle cloud pourra peser dans la balance ;
- une réaction de rejet justifiée par la politique de sécurité du client. Dans ce cas, il faudra lui proposer une alternative ;
- une réaction positive liée à une image innovante du modèle cloud. Ce cas est heureusement assez fréquent et devrait le devenir de plus en plus.

En résumé

Ce chapitre a présenté les principaux bénéfices du modèle cloud du point de vue de la direction des entreprises :

– réduction des coûts par industrialisation/rationalisation ;

– déport des problématiques d'exploitation et recentrage sur l'informatique métier ;

– garantie d'intégrité et de disponibilité des Datacenters des opérateurs cloud ;

– fin du syndrome de l'« administrateur héroïque » ;

Il a aussi présenté les principaux risques pour l'entreprise :

– les problématiques juridiques et commerciales ;

– un rejet de la part des clients.

Pour nous, la principale problématique reste la confiance que l'on peut accorder à l'opérateur cloud sur la confidentialité de données.

7

Bénéfices et inconvénients du point de vue des utilisateurs

Objectif

L'objectif de ce chapitre est d'introduire le point de vue des utilisateurs vis-à-vis des bénéfices et des risques du cloud computing.

On abordera ici des aspects qui touchent au *Time to market*, à l'ergonomie, à l'accessibilité des applications pour les utilisateurs, ainsi qu'aux bénéfices en termes de collaboration.

7.1 BÉNÉFICES POUR LES UTILISATEURS

7.1.1 Le Time to market

L'usage du modèle cloud permet aux utilisateurs de bénéficier de nouvelles applications sans passer par les étapes d'un cycle projet imposé par la DSI et souvent long. En effet, de nombreuses DSI imposent un « cycle projet en V » avec des étapes incontournables comme :

- spécifications fonctionnelles ;
- conception de l'architecture ;

- spécifications détaillées ;
- implémentation ;
- tests techniques ;
- tests fonctionnels ;
- recette.

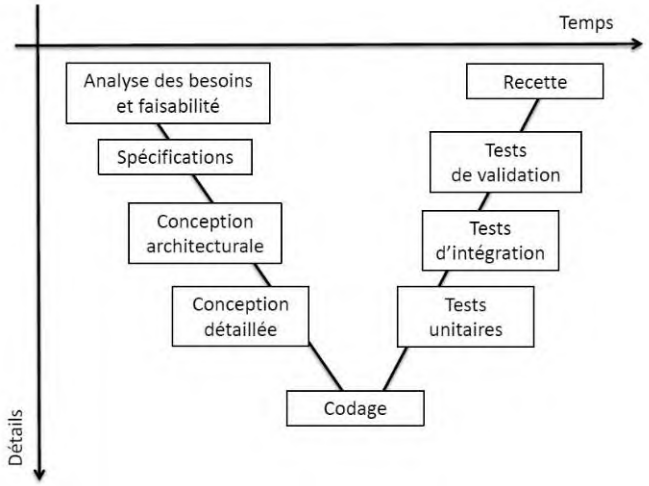

Figure 7.1 — Le cycle en V (source : Wikipédia).

Le cycle en V peut être pertinent pour la construction d'applications métiers complexes qui nécessitent une grande pérennité et une grande maintenabilité. Cependant, il est souvent vécu par les métiers comme un fardeau pour des projets plus simples, des applications web non pérennes ou comme des applications de commodité. Ainsi, les utilisateurs sont parfois contents de pouvoir contourner la DSI lorsqu'ils ont besoin d'agilité pour de petits projets. Pour ce faire, ils peuvent utiliser des sociétés de services ou des applications sur le cloud.

Les applications SaaS sont particulièrement pertinentes pour répondre à un besoin d'agilité. En effet, elles sont installées et exploitées par leurs fournisseurs et leur intégration se limite à du paramétrage au travers d'une console web, simple à prendre en main. Le cycle projet d'une application SaaS se réduit, la plupart du temps, à des spécifications légères et à un paramétrage rapide. Les choses se compliquent néanmoins lorsqu'une intégration forte avec le SI est nécessaire : nous en parlerons dans le chapitre 11.

Le recours à une plateforme PaaS ou IaaS permet d'accélérer le déploiement des applications développées au sein de l'entreprise. En effet, il est fréquent que les services d'exploitation des DSI demandent plusieurs mois pour mettre à disposition de nouvelles machines et les configurer. Avec le cloud, les plateformes sont disponibles en quelques minutes et les équipes de développement peuvent mettre en production sans attendre les quelques mois imposés par la DSI. Là encore, le cloud permet un gain important en TTM, pour la partie plateforme.

De manière générale, la promesse de TTM du cloud est très séduisante pour les équipes métiers. Et elles sont parfois tentées de **contourner la DSI** pour gagner

du temps sur la mise en ligne de leurs nouveaux outils. C'est un des impacts organisationnels structurants du cloud. Nous y reviendrons dans le chapitre 8.

Le cas d'INDUS

Monsieur Paul, directeur commercial d'INDUS a le projet de déployer une solution de gestion de la relation client (CRM). Il en a parlé à Monsieur François, le directeur informatique. Et ce dernier est revenu vers lui avec une proposition de logiciel de CRM maison « aux petits oignons ». Le projet de Monsieur François prévoit 1 an de développement et un coût de 700 000 euros. Monsieur Paul est un peu agacé car la CRM est un service très banal pour une force de vente et il espérait disposer de cette application dans 2/3 mois.

Il a par hasard discuté avec un collaborateur de la R&D qui lui a parlé de la solution SaaS Salesforce. Salesforce, lui a-t-on dit, est proposé pour un coût mensuel raisonnable, et son paramétrage est assez rapide. Une fois les spécifications fonctionnelles terminées, l'implémentation est l'affaire de quelques semaines.

Il va sans dire que Monsieur Paul est très intéressé.

7.1.2 Ergonomie et évolutivité des applications SaaS

Comme on l'a vu au chapitre 3, le cloud permet une « *Customer driven roadmap* » dans le cadre des applications SaaS. Cela signifie que la collecte des retours utilisateurs est au cœur des roadmaps de ces applications.

Ces bonnes pratiques permettent l'émergence d'interfaces qui sont des modèles d'ergonomie. Ces interfaces sont généralement simples, efficaces et très satisfaisantes pour les utilisateurs.

Elles permettent aussi la « bêta perpétuelle », c'est-à-dire une évolution au fil de l'eau des fonctionnalités applicatives, plutôt que le principe des nouvelles versions tous les un à trois ans.

Par ailleurs, les SaaS proposent des fonctionnalités souvent plus simples que celles des applications d'entreprise issues de développements spécifiques ou d'intégration de logiciel de type « software ». Cette simplicité permet une meilleure maîtrise des fonctions de l'application et améliore par conséquent la productivité.

L'exemple de la bureautique en ligne

Des offres comme Google Apps ou Zoho (*cf.* chapitre 14) fournissent des services bureautiques en ligne comparables à ceux de Microsoft Office. Cependant, ces services sont simplifiés : par exemple, les traitements de textes intégrés à ces offres proposent des fonctions d'édition, mais pas de tables d'index, de tables d'illustrations. On dit souvent que ces offres proposent 20 % des fonctionnalités d'Office, mais que ces 20 % couvrent les besoins de 80 % des utilisateurs.

Elles évoluent à un rythme mensuel avec de nouvelles fonctionnalités issues des tests utilisateurs.

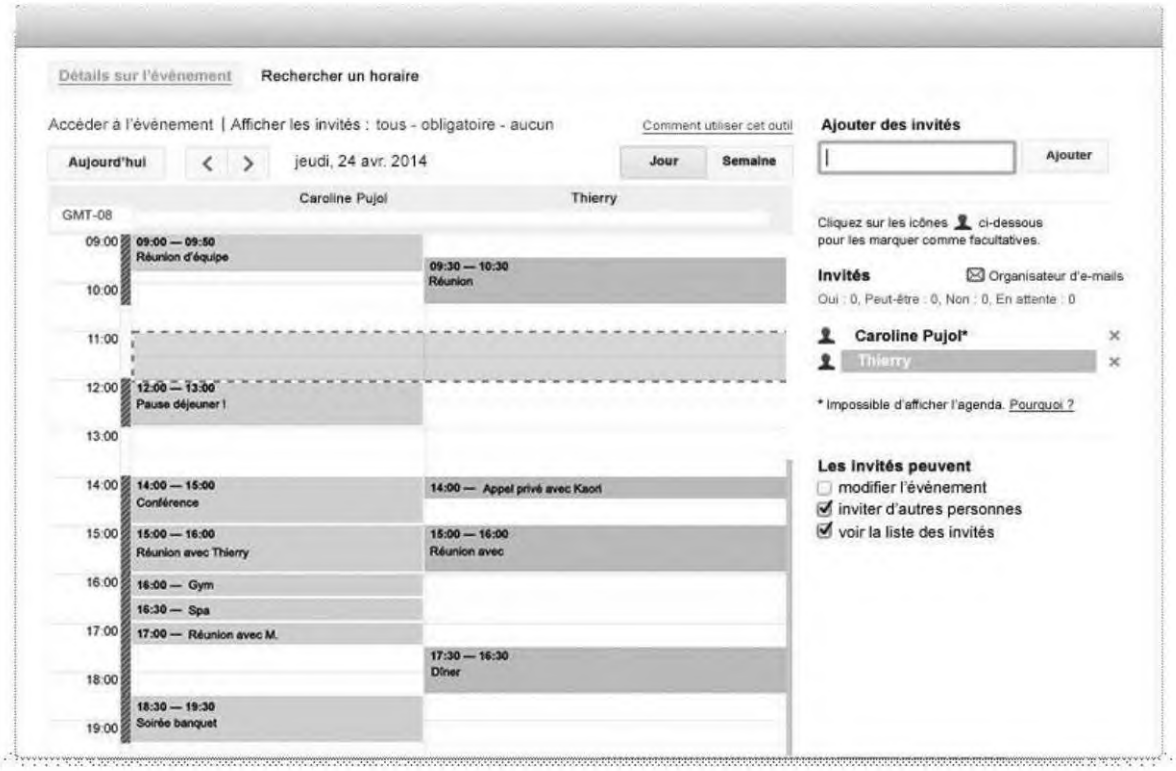

Figure 7.2 — Ergonomie de Google Agenda.

7.1.3 Les API ouvertes

Les applications SaaS opérées sur le cloud mettent l'accent sur la collaboration tant au niveau des utilisateurs qu'au niveau des couches logicielles (*cf.* les fameuses API du web 2.0 évoquées dans le chapitre 1).

Ces API permettent de bénéficier d'écosystèmes capables de collaborer entre eux par la création simplifiée de *mashups*. Dans certains cas, il est même possible de laisser les utilisateurs créer ces *mashups* sans taper une ligne de code.

On peut citer quelques exemples :

- l'intégration de la messagerie instantanée Hangout dans la bureautique Google ;
- l'intégration de la messagerie instantanée Hangout au sein de l'écosystème Salesforce ;
- la possibilité d'accéder à la bureautique en ligne docs.com de Microsoft *via* son compte Facebook ;
- la possibilité d'accéder à la CRM Zoho *via* son compte Google ;
- etc.

Les API fournies par les SaaS permettent aussi aux utilisateurs de les faire collaborer. Ils peuvent ainsi intégrer plusieurs interfaces entre elles en fonction de leurs besoins. Par exemple, il est possible d'utiliser une interface unifiée pour accéder aux fonctionnalités de Google Apps et de Salesforce. Il est aussi possible d'utiliser la messagerie instantanée dans le tableur de Google.

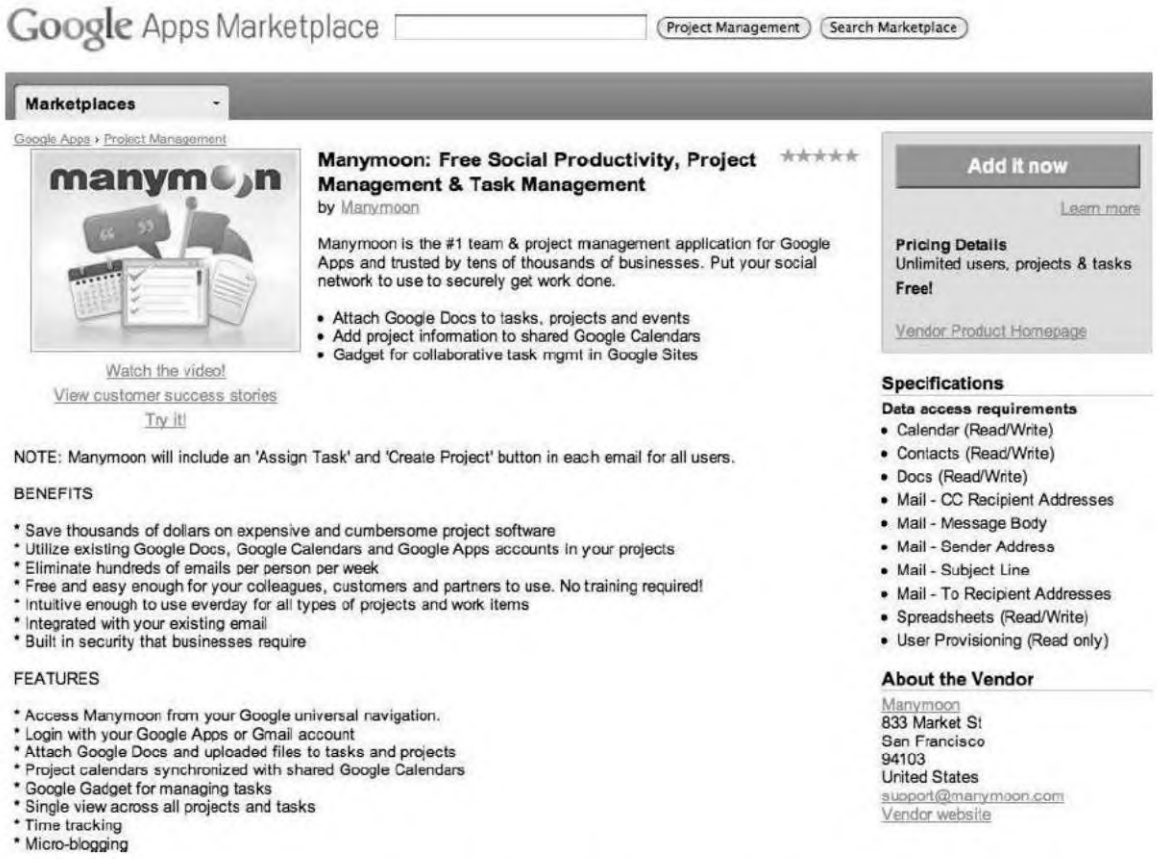

Figure 7.3 — L'ajout de services à Google Apps depuis la Marketplace.

À propos de Google Apps Marketplace

Google Apps Marketplace est un écosystème d'applications capables de s'intégrer en profondeur avec Google Apps. Cette place de marché se présente sous la forme d'un catalogue, à partir duquel on peut souscrire à des services gratuits ou payants. Ces services seront ensuite directement intégrés dans l'interface de Google Apps. Dans la démarche de souscription, on demande à l'utilisateur s'il autorise le service tiers à lire des documents, à écrire dans son calendrier, etc. Tout se passe *via* des formulaires. Les possibilités d'enrichissement de l'environnement Google Apps *via* ses API sont immenses.

7.1.4 Accessibilité des applications

On a vu dans le chapitre 4 que le cloud offre un très haut niveau de disponibilité et unifie les modes d'accès aux applications. Ainsi, les utilisateurs accèdent à leurs applications au travers d'un simple navigateur web, qu'ils soient au bureau, en situation de nomadisme, en télétravail chez eux, dans un cybercafé, etc., cet accès pouvant se faire depuis un ordinateur ou depuis un terminal mobile. En effet, les outils cloud sont presque toujours multi-écrans (disponibles sur Smartphone, tablette, PC).

Cette plus grande accessibilité permet aux utilisateurs en déplacement de consulter leurs e-mails sans repasser au bureau, de travailler depuis un aéroport, etc.

Elle offre un grand confort aux utilisateurs et plus de productivité à l'entreprise : des déplacements inutiles sont évités, des temps morts sont utilisés, etc.

De plus, le cloud assure aux utilisateurs un environnement de travail totalement identique quel que soit leur lieu d'accès. Il permet ainsi un retour sur investissement rapide grâce à une productivité améliorée et à une réduction de coûts de déplacement.

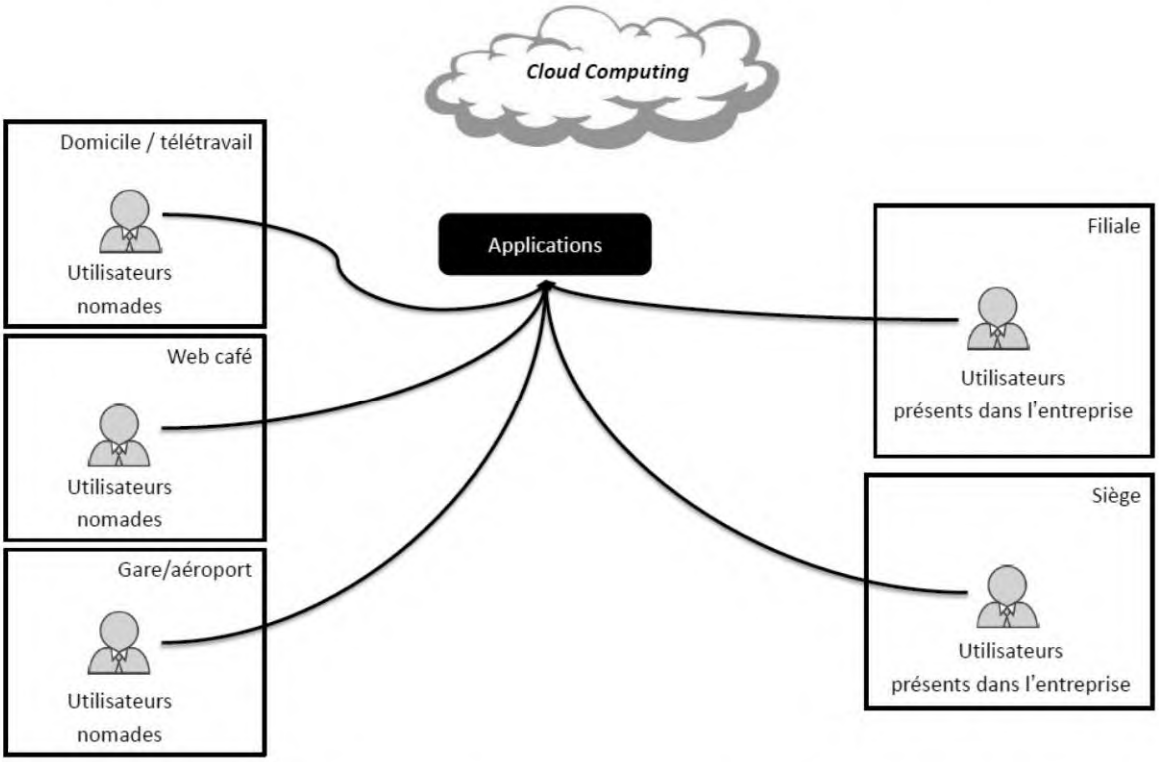

Figure 7.4 — Accessibilité des applications déployées sur le cloud.

7.1.5 Une migration plus rapide des postes de travail

Lorsqu'un poste de travail arrive en fin de vie, la DSI fournit à l'utilisateur une machine de remplacement et assure de manière plus ou moins complète la migration de l'environnement utilisateur vers le nouveau poste. Dans la pratique, les collaborateurs perdent généralement plus d'une journée de travail à transférer leurs données, à reparamétrer leurs programmes et leur environnement afin de retrouver leur niveau de productivité optimal.

Avec le cloud, ces problématiques de migration disparaissent totalement : en effet, l'environnement des utilisateurs est stocké sur la plateforme de l'opérateur. Ainsi, un changement de poste de travail est une opération totalement anodine.

Il existe même des postes de travail explicitement destinés au cloud : les chromebooks (*cf.* chapitre 1).

7.2 CRAINTES DES UTILISATEURS

7.2.1 La confidentialité des données

Les craintes des utilisateurs portent invariablement sur la confidentialité des données comme nous l'avons évoqué précédemment. Mais ils ne sont pas à même de mesurer réellement ce risque, cette tâche étant du ressort du « **responsable de la sûreté de l'entreprise** » comme on l'a vu au chapitre 4.

Ces craintes sont faciles à lever dès lors que la direction de l'entreprise a étudié la question et validé l'usage d'applications en mode cloud. Il s'agit alors de donner les bonnes explications aux utilisateurs et d'accompagner le changement.

7.2.2 Le mode déconnecté

Les interfaces web des applications cloud gèrent mal le mode déconnecté sur les ordinateurs. Si HTML5 le prend en charge depuis peu, il est rarement implémenté dans la pratique. Google est l'exception qui confirme la règle.

On a vu que de nombreux acteurs du cloud proposent des applications accessibles sur ordinateur et sur mobile : le mobile est souvent la solution pour accéder à ses données en situation de nomadisme. Il permet de continuer à travailler, même si son interface est moins confortable que celle d'un ordinateur.

7.2.3 Dépossession du poste de travail

Certains utilisateurs attachent beaucoup d'importance à la puissance de leur poste de travail, dont ils annoncent fièrement le nombre de mégaoctets de mémoire et de disque dur. Leur PC est parfois, à l'image de leur voiture, un symbole de réussite. Ils adorent le customiser avec des applications qu'ils ont dénichées sur Internet.

On a vu que les applications SaaS utilisent des interfaces HTML5 et poussent les utilisateurs à sauvegarder leurs données « dans les nuages ». Dans ce contexte, le poste de travail tend à devenir une coquille vide de données, une machine interchangeable. Il n'y a plus d'intérêt à le customiser. De plus, l'absence de données en local sur le PC peut donner à l'utilisateur une impression de dépossession de ses données.

Cette impression de dépossession touchera les *analogists* et non les *digital natives*. Un accompagnement au changement sera nécessaire pour leur faire accepter ce nouveau mode de travail. Il faudra prendre le temps de leur expliquer les bénéfices du cloud et de répondre à leurs interrogations.

Pour certaines typologies d'utilisateurs comme les directeurs ou les managers, il pourra être nécessaire de faire des exceptions, en leur permettant de disposer d'une copie locale de leurs données.

En résumé

Ce chapitre a présenté les principaux bénéfices du modèle cloud du point de vue des utilisateurs :

– *time to market* ;

– ergonomie et évolutivité ;

– *mashups* ;

– accessibilité ;

– qualité de service et disponibilité.

Les risques généralement évoqués par les utilisateurs sont la sécurité et le mode déconnecté. Le premier n'est pas de leur ressort, et le second est à considérer.

8

Bénéfices et inconvénients du point de vue des informaticiens

Objectif

L'objectif de ce chapitre est d'introduire le point de vue des informaticiens vis-à-vis des bénéfices et des risques du cloud computing.

On abordera ici les points de vue de la direction des études, de la production, de la cellule d'architecture transverse, du RSSI.

8.1 BÉNÉFICES POUR LES INFORMATICIENS

8.1.1 Plus d'agilité pour les études

Nous proposons dans ce paragraphe une petite liste de cas d'usage du cloud pour les directions des études.

- Le cloud peut être utilisé pour **héberger une UDD (Usine de développement)** dans le cadre des études. L'UDD est un ensemble de composants qui permet de centraliser les sources, compiler le code, tester le build, générer les documentations, lancer des tests de non-régression, déployer l'application en

environnement de test, etc. Les données concernées par l'externalisation chez un tiers sont ici le code source, dont la confidentialité est plus ou moins citrique selon les entreprises. L'avantage de recourir à une UDD sur le cloud est de gagner du temps en termes de mise à disposition des machines si l'on utilise une plateforme IaaS (par exemple, des machines virtuelles Hudson sont au catalogue d'Amazon Web Services). Si l'on décide de recourir à une UDD PaaS, on dispose d'un environnement opérationnel en seulement quelques minutes, alors que le temps de déploiement classique d'une UDD se compte en semaines.

- Le cloud peut être utilisé pour héberger les environnements de développement/test/recette dans le cadre des études. La disponibilité de ces plateformes en interne peut être retardée par les temps de commande et installation inhérents à la direction de la production. Les déployer sur le cloud permet de gagner un temps précieux, et ce, d'autant plus qu'elles sont généralement non pérennes et qu'elles manipulent rarement des données critiques.
- Le cloud peut être utilisé pour héberger un « bac à sable d'incubation » dans le cadre d'une démarche innovation. En effet, on peut être amené à créer des maquettes dans le cadre de Proof Of Concept, afin de tester des fonctionnalités innovantes auprès de populations réduites d'utilisateurs. Dans ce cadre, il est intéressant de disposer d'environnements rapides à déployer, puis rapides décommissionner. En effet, les applications expérimentales sont souvent abandonnées après une période de test infructueux. On parle en anglais de « *fail fast* », c'est-à-dire d'abandon rapide si l'expérimentation est un échec. Le cloud est particulièrement intéressant pour les plateformes provisoires, car il permet de souscrire à un environnement pendant quelques semaines, selon le principe du *Pay As You Go*. Il évite ainsi un investissement inutile dans des machines dont l'usage n'est pas pérenne.

8.1.2 Plus d'agilité pour la production

Le cloud permet aux équipes de production de bénéficier d'une plateforme d'hébergement en quelques minutes. Deux grands scénarios s'offrent à elles :

- **utiliser une plateforme IaaS :** dans ce cas, l'infrastructure (machines virtuelles, stockage, réseau) est disponible en quelques minutes, mais les tâches de déploiement, tests de charge, optimisation, monitoring, etc. demeurent. Ainsi, le cloud fait gagner quelques précieuses semaines, mais il n'autorise pas une mise en ligne immédiate ;
- **utiliser une plateforme PaaS :** dans ce cas, non seulement l'infrastructure est disponible en quelques minutes, mais en outre les tâches de déploiement, optimisation, monitoring sont directement prises en charge par la plateforme. Il est donc réellement possible de faire une mise en ligne immédiate. De fait, avec le PaaS, le travail des équipes de production est réduit au strict minimum : suivre les métriques de l'application dans la console.

Le cloud permet donc un gain de temps important pour les mises en production dans les deux scénarios IaaS et PaaS. Il rend aussi possible d'autres scénarios spécifiques grâce à ses propriétés d'élasticité :

- **scénario de débordement :** il est possible d'utiliser le cloud de manière ponctuelle lorsque la charge d'une application connaît un pic saisonnier (exemple : un site de commerce électronique au moment des fêtes de Noël). Dans ce cas, un réplica de l'application est déployé sur une plateforme cloud, mais n'est activé que lorsque cela est nécessaire. Et, grâce au principe du *Pay As You Go*, l'entreprise ne paie rien tant que l'application est inactivée ;
- **scénario de déport des traitements à forte consommation de ressources :** le cloud peut être utilisé pour absorber des calculs gourmands en ressources (exemples : gros calculs statistiques, facturation mensuelle, migration de plan comptable) ;
- **scénario d'ajustement :** le cloud peut être utilisé au démarrage d'un service dont on ne sait pas anticiper la charge utilisateur : son élasticité lui permet de s'adapter à une charge imprévisible (illustration : lorsque le service GeoPortail a été annoncé par le président de la République en 2006, l'annonce a provoqué une très forte charge qui a rendu le service indisponible pendant les premières semaines). Le cloud permet ainsi de gérer le démarrage d'un service : lorsque ce dernier sera en vitesse de croisière, il pourra être réintégré au SI.

8.1.3 Le recentrage sur l'informatique métier

Comme on l'a vu au chapitre 4, le bénéfice des SaaS pour les DSI est un recentrage sur l'informatique métier au détriment de l'informatique de commodité.

La DSI va pouvoir se débarrasser de nombreuses tâches ingrates d'exploitation et de mises à jour sur les serveurs et sur le parc utilisateur. Le bénéfice sera d'autant plus grand que le support (*helpdesk*) dans le cadre de l'informatique de commodité est source de nombreuses frictions avec les utilisateurs et offre peu de satisfactions pour les équipes d'exploitation.

8.1.4 Plus de temps pour penser le SI

Débarrassée d'un certain nombre de tâches d'exploitation, la DSI va pouvoir se consacrer à deux projets essentiels pour son entreprise :

- **les études,** c'est-à-dire concevoir de nouvelles applications métiers au plus près des attentes des utilisateurs. En effet, une DSI moderne doit se positionner comme un prestataire de service vis-à-vis des métiers : être à l'écoute de leurs nouveaux besoins, faciliter la mise en œuvre de nouvelles applications correspondant à la stratégie de l'entreprise et, enfin, assurer une qualité de service irréprochable. La disparition des tâches d'exploitation concernant l'informatique de commodité sera un accélérateur pour la mise en œuvre de cette approche ;

- **l'urbanisation de son système d'information,** c'est-à-dire de travailler à la rationalisation du système d'information, pour le pérenniser, améliorer sa maintenabilité et son agilité. Le travail d'urbanisation est complexe et difficile à mettre en œuvre tout en maintenant en parallèle les tâches de production. De fait, peu d'entreprises arrivent à le mener à bien et à fournir à leurs métiers le niveau d'agilité qu'ils attendent. La réussite des projets d'urbanisation est néanmoins une source importante de gains de productivité et de compétitivité. Il est donc important d'y consacrer le maximum de temps et de ressources.

Dans le cadre de sa mission d'urbanisation, la DSI va devoir **élargir sa réflexion à la bonne cohabitation** des applications internes à l'entreprise avec celles déportées chez des opérateurs cloud : l'architecture hybride.

8.2 CRAINTES DES INFORMATICIENS

8.2.1 La perte de pouvoir et de ressources

La première réaction négative des équipes d'exploitation vis-à-vis du modèle cloud est une réaction très humaine de protection. En effet, ce nouveau modèle est perçu comme la menace de suppression d'une partie de leurs attributions.

Le cas d'INDUS

Lorsque Monsieur Vincent, directeur général d'INDUS, a parlé pour la première fois à son DSI, Monsieur François, de l'offre de messagerie d'entreprise hébergée de Google, celui-ci a eu une réaction de repli.

Il a rétorqué à son patron qu'INDUS gère quotidiennement plus de 20 000 e-mails et qu'il n'est pas sûr que Google puisse en faire autant[1]. L'argument est un peu court : les infrastructures de Google gèrent des millions d'e-mails par jour. Cela montre bien que Monsieur Vincent était sur la défensive, à cours d'arguments véritables.

Pour la DSI, la perte d'une partie de son périmètre peut être synonyme d'une réduction de budget et d'une réduction des effectifs. Cette perspective est très anxiogène pour les équipes d'exploitation qui craignent pour leurs emplois.

Le spectre de l'offshore

Lorsque le concept de développement offshore est apparu il y a quelques années, les équipes informatiques occidentales ont vu leur emploi mis en danger. Les développeurs se sont sentis menacés par un remplacement par des ressources moins coûteuses en Inde. L'expérience a cependant montré que le marché de l'emploi informatique en Europe ne s'est pas complètement écroulé avec l'offshore.

1. Cette anecdote a été vécue par l'auteur.

Le modèle cloud représente une menace similaire mais, cette fois, pour les équipes d'exploitation. Il y a un véritable risque pour ces équipes de se retrouver au chômage.

Enfin, comme on l'a vu au chapitre 6, les offres SaaS métier permettent aux utilisateurs de se passer des services de la DSI. Ils peuvent contourner cette dernière pour gagner en agilité. Cette perspective est frustrante pour les études. Cependant, cette pratique de contournement était déjà là pour les projets web, confiés à une *web agency*.

8.2.2 La sécurité

Un des principaux arguments de la DSI pour refuser le cloud est la sécurité. La sécurité entre souvent dans les attributions des équipes d'exploitation et elles connaissent cette problématique plus en profondeur que les utilisateurs (*cf.* le « mythe du complot » pour ces derniers).

Les équipes d'exploitation sont capables de décomposer les problématiques de sécurité suivant les critères classiques : authentification, confidentialité, intégrité, traçabilité, disponibilité (*cf.* chapitre 4).

8.2.3 La « réversibilité » des applications

On appelle réversibilité la capacité à quitter une solution informatique pour une autre. Cette réversibilité implique que l'on puisse récupérer ses données et éventuellement ses composants métiers chez son prestataire d'hébergement pour les migrer chez un autre acteur ou éventuellement les rapatrier en interne.

Les clouds fournissent de nombreux assistants d'import pour permettre la migration des données vers leurs plateformes. Par contre, ils ne fournissent pas d'outils pour récupérer ces données.

Prenons l'exemple de Flickr, un service de stockage et de partage de photos ; cet opérateur dispose d'une offre premium payante destinée aux professionnels. Il fournit gratuitement un logiciel d'envoi de photos vers sa plateforme : le « Flickr uploadr », mais il ne fournit pas de logiciel de « Flickr downloadr » pour les récupérer en masse. Il est donc nécessaire de les récupérer une à une, ce qui est très fastidieux.

Heureusement, on a vu que les plateformes cloud fournissent des API permettant de se connecter à leurs plateformes. Il est donc possible pour la DSI de développer des composants sur la base de ces API pour autoriser la réversibilité. Ces composants sont souvent fournis par des éditeurs tiers (c'est le cas du Flickr downloader).

Si les API fournies par les plateformes cloud permettent la réversibilité en théorie, il est prudent de le vérifier au cas par cas. Plusieurs stratégies peuvent être envisagées :

- **protocole de test des API** : la DSI construit une maquette pour vérifier la réversibilité ;

- **demande d'engagement auprès de l'opérateur** : l'opérateur fournit la maquette pour prouver la réversibilité ;
- **demande de solution de réversibilité clés en main** : l'opérateur fournit une solution de réversibilité opérationnelle et industrialisée.
- **La réversibilité est une vraie question :** l'accepter montre une vraie stratégie d'ouverture de la part d'un opérateur cloud et assure qu'il reste vertueux dans les relations avec ces clients, à l'inverse d'une stratégie d'enfermement, comme cela se pratique chez certains vendeurs de logiciel...

8.2.4 L'intégrabilité des applications

La capacité d'une application cloud à s'intégrer au SI est un vrai sujet. Sans capacité d'intégration, elle est vouée à rester un silo, ce qui rappelle les travers de l'informatique antérieure aux années 2000, et aux démarches d'urbanisation. Heureusement, les API fournies par les plateformes cloud permettent généralement cette capacité d'intégration. On y reviendra au chapitre 11.

8.2.5 La dépendance au réseau

Faire appel au cloud suppose que la connexion de l'entreprise au réseau Internet est de qualité irréprochable. En effet, si, par exemple, le logiciel de CRM est externalisé et que le lien à Internet est rompu, l'équipe commerciale se retrouve plus ou moins au chômage technique.

De fait, les équipes de la DSI avancent parfois les arguments suivants : « *Le modèle cloud constitue un risque grave de perte d'accès aux applications. Pour sécuriser le réseau, il va falloir redonder toutes les infrastructures et ça va coûter cher. La facture télécom va supprimer le bénéfice du cloud en termes de réduction des coûts.* »

La problématique de dépendance au réseau du modèle cloud est très réelle. Il convient cependant de la relativiser. En effet, sans le modèle cloud, une rupture de réseau est déjà très critique pour de nombreuses entreprises. Elle implique :

- une rupture dans l'envoi et la réception de messages avec les clients ou les partenaires ;
- une rupture des flux de données avec les partenaires ;
- une rupture de l'accès au site web ou/et à l'extranet pour les clients ;
- la perte de l'accès au web pour les collaborateurs.

Il est donc indispensable pour les entreprises aujourd'hui de se doter d'un lien réseau de haute qualité avec la redondance adéquate pour remédier à des pannes. Quasiment aucune entreprise ne peut travailler sans accès à Internet, avec ou sans cloud.

8.2.6 L'augmentation du trafic réseau

Un autre argument souvent avancé par les informaticiens pour contester le modèle cloud est l'augmentation du trafic réseau. En effet, des échanges qui avaient lieu sur le réseau interne de l'entreprise vont passer par le lien Internet avec le modèle cloud.

Par exemple, pour envoyer un message à un collègue *via* Google Apps, on le fait transiter deux fois par Internet : une fois pour le transmettre à la plateforme Google, et une fois pour le collecter à partir de la plateforme Google.

La problématique d'augmentation du trafic réseau est à considérer. Il convient néanmoins de la modérer. En effet, les SaaS proposent des interfaces web et les pages web ont un poids très modeste.

Précisons les choses sur un angle un peu plus technique dans le cas de la messagerie :

- lorsqu'on utilise une messagerie interne, les messages transitent le plus souvent suivant les protocoles SMTP[1] et POP[2] ; ces protocoles font transiter tout le contenu du message ainsi que les pièces jointes sur le réseau ;
- lorsqu'on utilise une messagerie SaaS, les messages sont lus au travers d'une page web et éventuellement *via* le protocole IMAP[3] ; dans ces deux cas, on accède d'abord aux en-têtes des messages ; puis seulement si c'est nécessaire au contenu du message ; et enfin, dans un troisième temps, on peut choisir de télécharger les pièces jointes. Dans bien des cas, on supprime les messages avant d'avoir téléchargé leur contenu.

Le trafic réseau généré par les applications SaaS est optimisé pour un transit par Internet. Cette optimisation est indispensable pour que les utilisateurs situés loin des serveurs aient des temps de réponse rapide. Le trafic généré par les SaaS est dans bien des cas très inférieur à celui des applications internes de l'entreprise.

L'argument de l'augmentation du trafic réseau est donc à relativiser.

En résumé

Ce chapitre a présenté les principaux bénéfices du modèle cloud du point de vue des informaticiens :

– plus d'agilité pour les études ;

– plus d'agilité pour la production ;

– se recentrer sur l'informatique métier.

Les risques généralement évoqués par les informaticiens portent sur :

– la sécurité ;

– la réversibilité ;

1. *Simple Mail Transfer Protocol.*
2. *Post Office Protocol.*
3. *Internet Message Access Protocol.*

– l'intégrabilité ;
– la dépendance au réseau ;
– l'augmentation du trafic réseau.

On a vu que la réversibilité et l'intégrabilité étaient de vrais sujets, les autres risques sont à relativiser.

Il reste que le cloud représente une rupture organisationnelle importante pour la DSI, à ne pas sous-estimer.

9

Choisir un service cloud

Objectif

Nous pensons que l'usage de services cloud est aujourd'hui incontournable. En effet, même les entreprises les plus méfiantes, grandes banques et assurances, utilisent à minima un de ces services.

L'objectif de ce chapitre est donc de proposer une méthodologie d'aide au choix, afin de bien prendre une décision raisonnée. Nous souhaitons en particulier aider les métiers qui vont vers le cloud sans l'aide de la DSI, afin qu'ils ne passent pas à côté d'un problème technique.

9.1 L'ANALYSE PAR GRILLE DE CRITÈRES

Nous avons balayé dans les chapitres précédents les risques et les bénéfices du modèle cloud pour les divers acteurs de l'entreprise. Nous proposons maintenant une **méthodologie d'aide au choix** sous la forme d'une analyse par grille de critères.

Cette démarche passe par les étapes classiques :

- **définition des critères de décision** sur la base de l'existant, des besoins utilisateurs, de la politique de sécurité de l'entreprise ;
- **pondération de la grille de critères :** il s'agit d'attribuer à chaque critère un poids sur une échelle généralement de 1 à 4 (importance critique, haute, moyenne, ou basse). Ainsi, on pourra quantifier de manière précise les critères les plus importants du point de vue de l'entreprise (par exemple la réduction des coûts ou la sécurité) ;

- **analyse de la solution :** il est ensuite temps de confronter la solution à la grille. Cette étape peut s'effectuer au travers de sa documentation (disponible sur Internet), ou bien en rencontrant l'opérateur cloud pour lui soumettre les questions de la grille. On donnera alors une note pour chacun des critères (par exemple 1 point si telle exigence est partiellement remplie, 2 points si une autre exigence est complètement remplie) ;
- **agrégation des résultats :** cette étape consiste à analyser les notes et les pondérations afin d'en sortir des indicateurs. Ces indicateurs, graphiques si possible, seront les outils d'aide à la décision.

La figure 9.1 propose un exemple d'extrait de grille de critères et un exemple de graphe d'aide à la décision.

N°	Intitulé du critère	Coefficient de pondération	Note	Commentaires
1.	Pérennité de l'opérateur SaaS	Haut	1	
2.	Engagement de confidentialité	Critique	2	
3.	Haute disponibilité de la solution	Critique	0	
4.	Nb clients, principales références	Critique	2	
5.	Coût de l'abonnement	Moyen	2	
6.	Sécurité des accès	Critique	1	

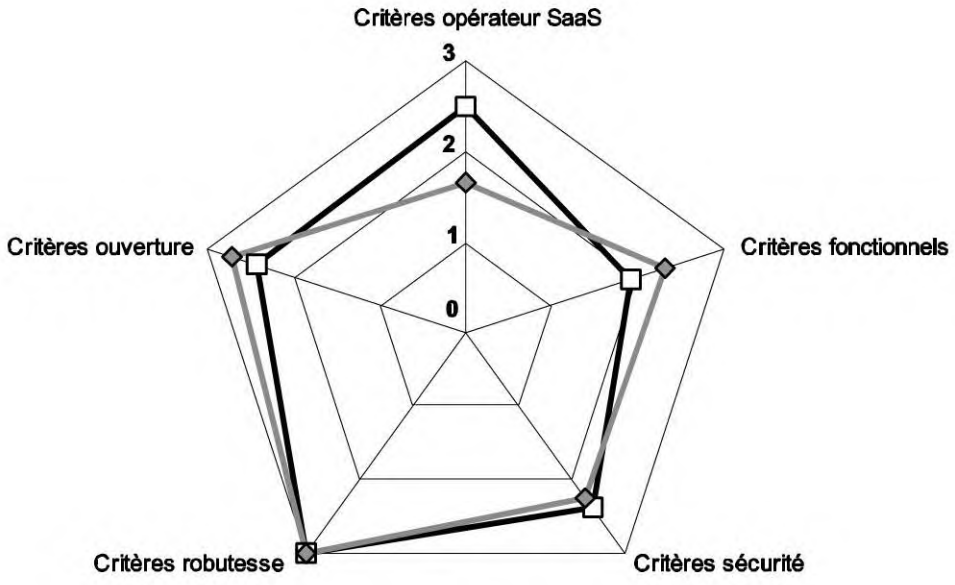

Figure 9.1 — Outils d'analyse par grille de critères.

9.2 GRILLE DE CRITÈRES CLOUD

Le tableau 9.1 présente quelques critères d'aide à la décision. **Ces critères devront être beaucoup plus détaillés selon le contexte et les exigences de l'entreprise utilisatrice.**

Tableau 9.1 — Grille de critères cloud.

Famille	Criticité	Critère	Commentaire
Métier	Critique	Adhérence à opérateur	Possibilité de trouver un autre service cloud équivalent ?
Métier	Très critique	Intégrabilité	Découverte après coup d'une difficulté d'intégration avec le SI
Métier	Moyen	Croyance excessive dans la baisse des coûts	Faire une vraie étude de ROI
Métier	Très critique	Dépôt de bilan de l'opérateur	Essayer de trouver un opérateur cloud équivalent, capable de prendre le relais ?
Métier	Important	Application non customisable et mises à jour forcées	Problème existant avec logiciels classiques, mais contrôle du planning de migration dans ce cas
Métier	Important	Travail impossible en mode déconnecté	Voir si c'est un point bloquant pour nomades
Métier	Important	Service Level Agreement	Contrôler la SLA avec DSI
Métier	Moyen	Pas de support téléphonique	Compensé par forums en ligne ?
Métier	Très critique	Vol de secret industriel ou de compétences	À évaluer avec la sécurité
Métier	Moyen	Se passer de la DSI ?	Risque sur intégration et réversibilité, à voir avec DSI
Juridique	Très critique	Localisation des données inconnue (*cf.* Google)	À exclure ?
Juridique	Très critique	Données stockées hors EU	Limiter à données peu critiques ?
Juridique	Très critique	Données stockées en EU	Cloud en Europe pour données critiques ?
Juridique	Très critique	Données personnelles stockées à l'étranger	Exiger Safe Harbor ?
Juridique	Très critique	Données stockées à l'étranger : quelle loi s'applique ?	Négocier application lois françaises ?
Achats	Important	OPEX plutôt que CAPEX	Les achats doivent s'adapter ?
Achats	Important	Coûts difficiles à anticiper	Mettre en place un seuil à ne pas dépasser ?
Achats	Important	Politique « à prendre ou à laisser »	Peut-on accepter un paiement par CB, un contrat sur étagère, non négociable ?
Sécurité	Très critique	Non-respect de la politique de sécurité de l'entreprise	Adapter la politique ? Tolérer des exceptions provisoires ? Construire une politique de sécurité propre au cloud ?

Tableau 9.1 — (suite)

Sécurité	Critique	Évaluation des opérateurs cloud	Homologuer les opérateurs ? Diffuser une liste des opérateurs homologués ?
Sécurité	Critique	Données éligibles au cloud	Réintroduire une classification ?
Sécurité	Critique	Certifications à exiger	ISO27001 & SAS 70 II nécessaires ?
Sécurité	Moyen	Tests d'intrusion	En faire ?
Sécurité	Moyen	Scan d'applications web	En faire ?
Sécurité	Critique	Politique d'accès	Rendre l'usage de la fédération d'identité d'entreprise obligatoire ?
Sécurité	Critique	Données persistantes	Introduire chiffrement pour les données confidentielles ?
Sécurité	Critique	Protection des flux échangés sur Internet	Chiffrement SSL à minima ? IPSec pour données critiques ?
Sécurité	Critique	Accès au cloud	Passer par VPN et authentification renforcée pour données critiques ?
Sécurité	Moyen	Cloud personnel	Laisser les employés utiliser des clouds de type Evernote ou autre à titre personnel ?
DSI	Très critique	Réversibilité	Réversibilité des données et modèles de données ? Garantie d'effacement « intégral » ?
DSI	Très critique	Solution d'intégration avec le cloud	Exclure les opérateurs cloud ne proposant pas de technologies standards ? Usage d'un middleware fourni par la DSI permettant les échanges bidirectionnels ?
DSI	Très critique	Gestion des identités dans le cloud	Rendre l'usage de la fédération d'identité d'entreprise obligatoire ?
DSI	Important	Surveillance des SLA	Utiliser un fournisseur tiers pour mesurer les SLA réelles
DSI	Important	Logs	Exiger au moins des logs d'accès
DSI	Important	Latence réseau	Exiger au moins des serveurs de cache en Europe. S'assurer de la possibilité d'augmenter facilement le lien Internet
DSI	Moyen	Support utilisateurs	Utiliser les processus et outils de ticketing habituels. Exiger support de niveau 3 par opérateur cloud

9.3 PROCESSUS DE PRISE DE DÉCISION

Nous vous proposons dans ce paragraphe un cheminement en vue d'une prise de décision. Cette méthodologie s'adresse avant tout aux entreprises de taille conséquente. Pour une PME de 20 personnes, le processus de prise de décision pourra être plus rapide, et ce d'autant plus s'il n'y a pas d'existant à remplacer.

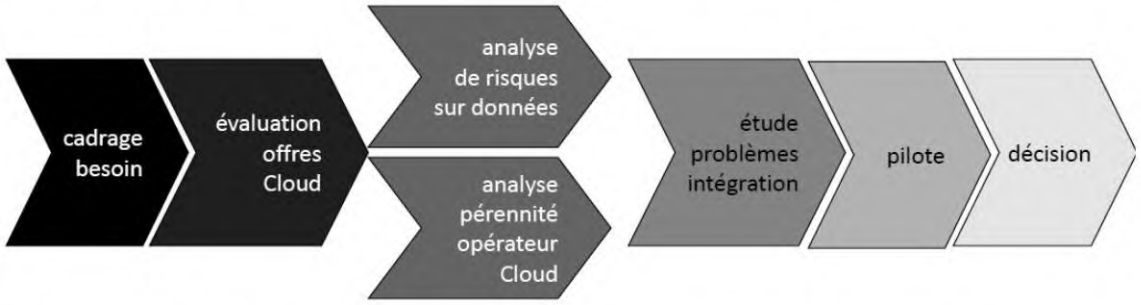

Figure 9.2 — Processus de décision pour l'adoption du modèle cloud.

9.3.1 Qui prend la décision ?

Dans certaines entreprises, des équipes métiers contractualisent avec un opérateur cloud sans consulter la DSI. Cette décision peut être légitime pour plusieurs raisons :

- les équipes métiers gèrent leur budget ;
- elles ont une bonne connaissance de leurs besoins et sont le plus à même de trouver une solution SaaS qui y répond ;
- elles recherchent une agilité que la DSI ne leur propose pas, dans certaines situations.

Néanmoins, cette démarche peut être source de problèmes lorsque ces équipes métiers n'ont pas conscience des enjeux d'intégration, de réversibilité, du mode déconnecté, de la mise en ligne de nouvelles fonctionnalités au fil de l'eau, etc.

Il nous semble qu'un compromis doit être trouvé entre déléguer entièrement le choix du service cloud à la DSI et la contourner. Nous y reviendrons dans la partie 3.

9.3.2 Cadrer le besoin

Il est clair que le passage au modèle cloud ne doit pas être déclenché par un souhait de la direction informatique d'utiliser des outils de nouvelle génération, mais par un vrai besoin des utilisateurs. Il convient donc de bien clarifier les bénéfices que l'on attend du cloud. Ces bénéfices peuvent être un meilleur *Time to market*, une réduction des coûts d'exploitation, une meilleure disponibilité des applications, etc.

Le cloud peut être une opportunité pour amener des changements au sein de l'entreprise. Le management arbitrera entre deux stratégies :

- **stratégie « plus de la même chose » :** le cloud est destiné à offrir les mêmes fonctionnalités aux utilisateurs, selon un modèle économique différent, et en externalisant les problématiques d'exploitation. Un exemple de cette stratégie est l'externalisation de la messagerie ;
- **stratégie de transformation :** le cloud est envisagé dans le cadre d'une refonte de certains processus métiers ou collaboratifs dans l'entreprise, ou bien pour faire évoluer la DSI.

Parmi les scénarios de transformation, on peut envisager :

- des outils collaboratifs SaaS pour faciliter une collaboration ouverte avec des clients et partenaires : on reviendra sur ces offres dans le chapitre 13 ;
- des outils métiers sur le cloud pour donner plus d'agilité, d'ergonomie, d'évolutivité aux utilisateurs ;
- des plateformes cloud pour donner plus d'agilité aux études et à la production ;
- des plateformes cloud comme « bac à sable innovation » ;
- etc.

9.3.3 Évaluer les solutions cloud disponibles

Le marché des solutions cloud est en pleine expansion et les solutions éligibles pour répondre aux besoins des entreprises vont devenir rapidement très nombreuses.

Si le choix est aujourd'hui limité à quelques solutions de références (nous avons déjà beaucoup cité Google, Salesforce et Amazon), il deviendra rapidement nécessaire d'évaluer un nombre conséquent de solutions avant de déterminer laquelle peut correspondre à ses besoins. On utilisera alors les démarches de choix classiques correspondant aux logiciels « software ».

Le cas d'INDUS

Les collaborateurs d'INDUS n'ont pas un usage avancé des logiciels de collaboration. Ils utilisent depuis des années des services de base : messagerie électronique et calendriers partagés. Ils sont prêts à changer de solution s'ils retrouvent ces mêmes fonctionnalités, surtout si elle amène une qualité de service supérieure à celle qu'ils vivent au quotidien. Quelques « *digital natives* » parmi eux aimeraient pouvoir **partager leurs calendriers avec des personnes externes à l'entreprise et utiliser une messagerie instantanée ouverte**. Une solution de collaboration en mode SaaS, comme Google Apps, répond donc bien à leurs besoins.

On a vu, dans les paragraphes précédents, que la direction commerciale souhaitait **s'équiper d'une solution de gestion de la relation client de manière rapide**. L'évaluation des fonctions de Salesforce a montré que cette solution répond aux exigences de l'équipe commerciale. Le développement spécifique, proposé par le directeur informatique, a été écarté car le planning proposé était trop long.

Enfin, la direction marketing souhaite s'équiper d'une **plateforme d'incubation** pour tester des maquettes de nouveaux produits auprès de ses utilisateurs. La plateforme Amazon Web Services offre la souplesse et l'agilité indispensable à ce projet. Le

principe du *Pay As You Go* permet un coût raisonnable pour des maquettes à durée de vie de l'ordre du mois.

9.3.4 Mener une analyse de risques

Nous avons abondamment abordé la problématique de confidentialité dans les précédents chapitres. Il est clair que cette problématique est centrale dans la décision d'aller ou non vers le cloud.

Rappelons que la problématique doit être envisagée non pas sur la base d'impressions et de rumeurs, mais sur la base d'une **solide analyse de risques**. Cette analyse doit intégrer une **classification des données** candidates à l'externalisation : données stratégiques, critiques, confidentielles, données de fonctionnement. Elle doit envisager la sécurité selon le prisme : **authentification, confidentialité, intégrité, disponibilité, traçabilité**.

Rappelons enfin que seul le **responsable de la sûreté de l'entreprise** peut statuer sur cette problématique : les informaticiens et les utilisateurs sont souvent incompétents sur le sujet car ils n'envisagent pas la problématique dans son ensemble.

Le cas d'INDUS

Les messages électroniques et calendrier sont considérés par le responsable de la sûreté d'INDUS, Monsieur Jacques, comme des données de fonctionnement, non critiques. Le déport de ces fonctions vers un opérateur SaaS a donc été validé. Monsieur Jacques a cependant émis la directive suivante : les contrats échangés avec les clients stratégiques d'INDUS seront chiffrés à l'aide un logiciel Open Source courant. Monsieur Jacques est très satisfait car la migration vers les SaaS lui a permis de faire passer une mesure qu'il appelait de ses vœux depuis longtemps. En effet, jusqu'alors INDUS avait toujours échangé ses contrats *via* des courriers électroniques non chiffrés.

Concernant l'utilisation du service Salesforce, Monsieur Jacques a considéré qu'il s'agissait aussi de données de fonctionnement, tant que les contrats n'étaient pas intégrés à la solution.

Enfin, la plateforme d'incubation n'a pas vocation à stocker des données métiers, mais plutôt à tester de nouvelles interfaces homme machine. Si une expérimentation nécessite de telles données, elles seront interrogées à la volée à partir d'un Web Service exposé depuis le SI INDUS.

9.3.5 Étudier la pérennité de l'opérateur cloud

En parallèle de l'analyse de risque, il convient d'évaluer la pérennité de l'opérateur cloud. En effet, la démarche de migration vers une plateforme cloud est une démarche lourde, dont la réversibilité est non triviale. Il est donc indispensable de s'assurer de la stabilité de son partenaire.

Le cas d'INDUS

INDUS est totalement rassuré par la santé financière de Google, un acteur qui est devenu en quelques années un des géants de l'informatique. Une autre solution SaaS de collaboration avait été envisagée : elle a été écartée car elle était proposée par une start-up qui n'avait pas atteint son équilibre financier.

L'examen des résultats financiers de Salesforce sur les trois dernières années a aussi rassuré INDUS.

Enfin, Amazon est leader sur divers segments : vente en ligne, livres électroniques, cloud computing. Sa pérennité est donc considérée comme suffisante.

9.3.6 Étudier l'intégrabilité et la réversibilité

L'intégration de flux d'informations entre l'opérateur cloud et l'entreprise peut revêtir différents niveaux de complexité :

- dans un scénario simple et fréquent, il se limite à la création manuelle de comptes utilisateurs chez l'opérateur ;
- dans un scénario un peu moins simple, on peut souhaiter maintenir une synchronisation entre l'annuaire d'entreprise et la base de comptes hébergés chez l'opérateur ;
- dans un scénario nettement plus complexe, on peut être amené à échanger des flux d'informations en continu avec l'opérateur cloud.

Le troisième type de scénario est peu fréquent dans un contexte où le cloud est utilisé pour de l'informatique de commodité. Il est cependant indispensable de bien anticiper ces contretemps car ils peuvent se révéler très lourds à gérer par la suite.

Une étude de premier niveau des problématiques d'intégration est donc un préalable au choix d'aller vers une solution cloud.

Rappelons que le niveau zéro de l'intégration est la possibilité de récupérer ses données depuis l'opérateur cloud, la fameuse réversibilité. Nous déconseillons le recours à un opérateur cloud qui n'offrirait pas cette garantie.

Le cas d'INDUS

Après une petite étude, INDUS a décidé de synchroniser son annuaire réseau actuel avec les comptes de messagerie hébergés chez Google. Les API fournies par l'opérateur ont permis de monter une maquette fonctionnelle rapidement, rassurant les équipes de la direction informatique.

Concernant Salesforce, compte tenu de la taille réduite de l'équipe commerciale (50 collaborateurs), il a été décidé de procéder à des imports/exports manuels.

Enfin, le bus d'intégration *Simple Queue Service* fourni par Amazon pourra être utilisé pour appeler les Web Services du SI en cas de besoin de données d'entreprise sur la plateforme d'incubation.

9.3.7 Faire un pilote

Dans la mesure où le passage au modèle cloud implique des **changements dans les modes de travail** des collaborateurs, il apparaît nécessaire de passer par une phase de pilote. Fort heureusement, le *Pay As You Go* se prête particulièrement bien à des expérimentations sur une population réduite. En effet, l'expérimentation n'implique aucun déploiement : tout se passe chez l'opérateur cloud ; et il est simple d'acheter un droit d'usage de quelques mois pour une population pilote. Cette phase de pilote permettra de confirmer ou d'infirmer la décision de passer au modèle cloud.

9.4 LE CLOUD COMPUTING, UNE MUTATION IMPORTANTE POUR L'ENTREPRISE

Pour terminer ce chapitre sur le choix d'un opérateur cloud, nous avons souhaité récapituler les impacts du cloud computing sur les entités de l'entreprise.

- Pour la **direction de l'entreprise**, le cloud computing permet de recentrer la DSI sur le métier en externalisant la « plomberie informatique ». Il peut permettre une réduction des coûts, selon les cas. Le cloud introduit une problématique juridique autour de la localisation des données. Il nécessite l'acceptation de l'externalisation des données par les employés, les clients et les partenaires.
- Pour les **utilisateurs**, le cloud computing permet de gagner en *Time to market*, en ergonomie (*Customer driven roadmap*), en évolutivité (bêta perpétuelle), en disponibilité et en accessibilité. Le cloud introduit une problématique de mode déconnecté pour les utilisateurs.
- Pour **les achats**, le cloud computing permet de basculer les coûts et OPEX et de bénéficier du *Pay As You Go*. Il introduit de nouvelles difficultés : interlocuteurs difficiles à joindre, contrats standards à l'avantage du fournisseur, garanties limitées en termes de qualités de service ; possibilité limitée de négociation ou d'adaptation de clauses sur un service banalisé.
- Le **responsable de la sûreté** voit sa politique de sécurité remise en question par l'externalisation vers le cloud. Il est tenté de résister, mais en prenant le risque d'être **contourné par les utilisateurs** qui contractualiseront directement avec des opérateurs SaaS sans lui demander son avis. Il a donc plutôt intérêt à aménager la politique de sécurité et à participer activement à l'homologation des opérateurs cloud en examinant de près leurs garanties et leurs certifications. Il pourra aussi mener des tests d'intrusions sur les plateformes. Le cloud est pour lui une opportunité de faire évoluer son métier et d'acquérir de nouvelles compétences.
- La **cellule architecture de la DSI** voit ses standards remis en cause : de nouveaux silos de données et annuaires utilisateurs vont éclore sur les plateformes cloud, et la maintenance de la synchronisation des données va se complexifier. De nouvelles problématiques de latence réseau vont apparaître, et des solutions d'intégration simplistes vont devoir être utilisées. La cellule est tentée de résister,

mais en prenant le risque d'être **contourné par les utilisateurs** qui créeront de nouveaux référentiels de données sur les plateformes SaaS sans lui demander son avis. Elle va donc plutôt développer un centre de compétence cloud, et proposer des solutions sur étagère pour l'intégration avec les référentiels situés sur les plateformes cloud. Elle pourra aussi s'impliquer dans les tests de réversibilité.

- La **direction des études** gagne en agilité grâce aux plateformes PaaS et IaaS en *Self Service*. Dans le cadre du PaaS, elle devra intégrer de nouvelles contraintes dans ses pratiques de développement (*cf.* chapitre 19). Le cloud ne constitue pas une remise en question pour elle.
- La **direction de la production** gagne en agilité grâce aux plateformes IaaS. En revanche, son spectre d'intervention se réduit, pour se recentrer sur les logiciels métiers. Avec les plateformes PaaS et SaaS, le spectre se réduit encore plus car l'administration est grandement simplifiée. La réduction des tâches techniques peut entraîner une réduction des effectifs de la production. Cette dernière se recentrera sur le pilotage de sous-traitants et sur le support aux utilisateurs.

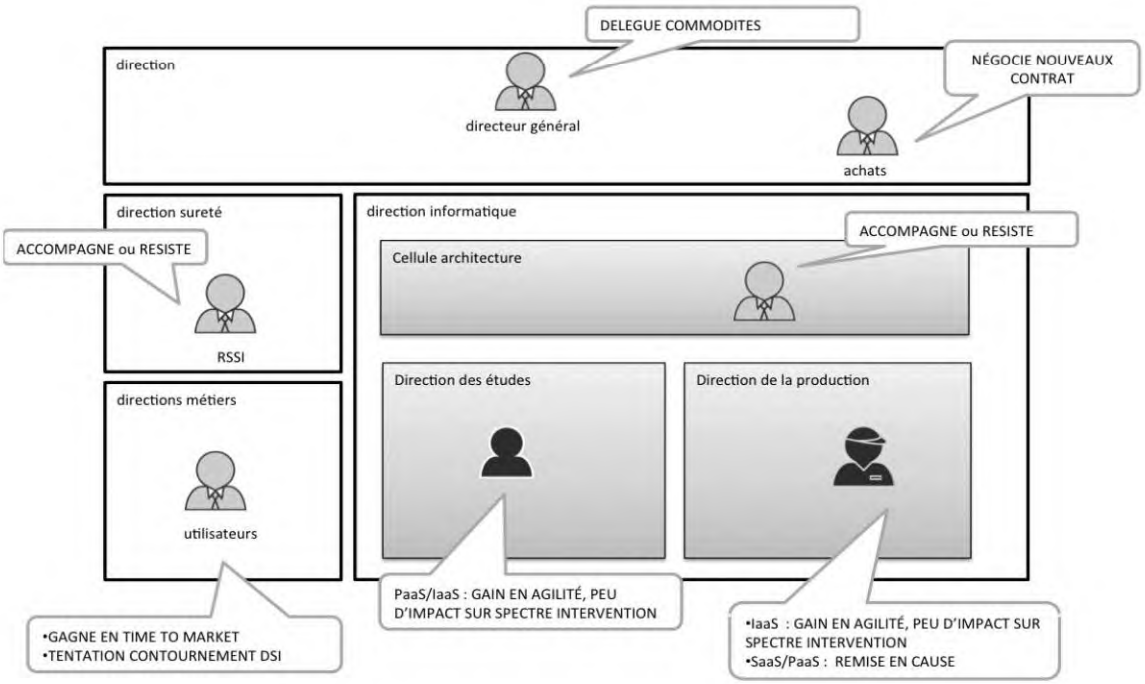

Figure 9.3 — Impact du cloud sur les populations de l'entreprise.

Tableau 9.2 — Impacts du cloud sur les populations de l'entreprise.

Entités	Bénéfices du cloud	Impacts du cloud	Changements nécessaires
Entreprise	Recentrage sur le métier (Parfois) réduction des coûts	Problèmes juridiques Risque de rejet par client/partenaire	Montée en compétence du service juridique
Utilisateurs	*Time to market* Ergonomie et évolutivité Qualité de service Accessibilité		
Achats	OPEX plutôt que CAPEX *Pay As You Go*	Difficulté à obtenir un interlocuteur Garanties limitées	Montée en compétence du service achats
Responsable de la sûreté		Remise en cause de la politique de sécurité	Aménagement politique de sécurité Homologation opérateurs cloud
Cellule d'architecture	Recentrage sur urbanisation	Nouvelles problématiques d'architecture	Création d'un centre compétence cloud Homologation d'opérateurs cloud Solutions d'intégration sur étagères
Études	Agilité	Contraintes de développement sur PaaS	
Production	Agilité	Réduction du spectre d'intervention	Recentrages sur le support utilisateurs

En résumé

On a vu dans ce chapitre une méthodologie d'aide à la décision autour du cloud. Avant de prendre cette décision, il nous apparaît important de :

– clarifier le besoin ;
– évaluer les offres cloud disponibles ;
– faire une analyse de risques sur les données de l'entreprise ;
– faire une analyse de la pérennité de l'opérateur cloud ;
– étudier les problématiques d'intégration et de réversibilité ;
– faire un pilote.

On a aussi évoqué les impacts du cloud sur les populations de l'entreprise : achats, responsable sûreté, cellule d'architecture transverse, études, production.

TROISIÈME PARTIE

La gouvernance du cloud

Cette partie présente les différentes étapes que peut ou doit franchir une entreprise qui souhaite aller vers le cloud computing.

Nous aborderons d'abord l'usage d'une première application SaaS de commodité, suite à une phase de pilote.

Puis nous décrirons l'intégration des applications cloud avec le SI, l'organisation à mettre en place, le rôle du *cloud broker*.

Enfin, nous évoquerons un SI déporté vers le cloud computing.

10

Les premiers pas sur le cloud

Objectif

Ce chapitre présente une démarche pour expérimenter le modèle cloud de manière progressive, en passant par une étape de pilote.

Nous utiliserons le cas d'usage d'INDUS, une société qui a décidé de migrer ses fonctions collaboratives vers le modèle cloud, de se doter d'une solution de gestion de la force de vente en SaaS et de monter une plateforme d'incubation sur IaaS.

10.1 COMMENCER PAR UN PILOTE

10.1.1 Les applications et plateformes de commodité

Comme on l'a vu au chapitre 4, l'adoption du modèle cloud commence généralement par des fonctions d'informatique de commodité. Rappelons que l'informatique de commodité, par opposition à l'informatique métier, désigne les fonctions nécessaires à toutes les entreprises : collaboration entre les employés, gestion des ressources humaines, gestion de la relation client, gestion de la paye, gestion financière, etc. Ces fonctions sont assez génériques pour être facilement utilisées sous forme d'abonnement auprès d'un opérateur SaaS.

En revanche, même si ces fonctions sont dites de commodité, elles peuvent être stratégiques pour l'entreprise : par exemple, la messagerie électronique est aujourd'hui une fonction vitale pour de nombreuses entreprises.

Ces fonctions pourront donc être externalisées suivant le modèle cloud, mais cette démarche d'externalisation doit être prudente et rigoureuse. **Elle doit suivre un processus de prise de décision raisonné, et la démarche de déploiement doit être effectuée de manière très encadrée.**

On peut parler de plateforme de commodité pour désigner une infrastructure qui héberge des outils accessoires, non critiques pour le fonctionnement de l'entreprise. On pense par exemple à des outils de développement ou des outils d'incubation. Une interruption de ces services n'est pas critique pour le business de l'entreprise, et ces outils ne manipulent pas de donnée critique (à moins que l'on considère que le code source soit une donnée critique : c'est un autre débat).

On peut donc expérimenter l'hébergement de ces outils sur des plateformes cloud. Dans la pratique, on commence le plus souvent par de l'IaaS car on peut y déplacer des applications sans les réécrire. A contrario, avec le PaaS, il est nécessaire de les adapter à l'architecture de la plateforme (*cf.* chapitre 19). On constate d'ailleurs qu'aujourd'hui (fin 2015) les entreprises utilisent plus volontiers l'IaaS que le PaaS, sans doute pour cette raison.

10.1.2 Le recours à une application SaaS occasionnelle

Par le passé, un certain nombre d'entreprises ont utilisé le modèle ASP (*Application Service Provider*), ancêtre du SaaS, pour des applications faisant l'objet d'un usage ponctuel. On peut citer dans ce domaine :

- les plateformes de gestion de paye (par exemple, la solution ADP GSI) ;
- les plateformes d'envoi d'e-mails en masse à des fins marketings (par exemple, les solutions Sarbacane, MailChimp, SendinBlue) ;
- les plateformes de dématérialisation des marchés publics pour les collectivités locales (par exemple, la solution Omnikles) ;
- les plateformes d'édition de documents (par exemple, la solution New Works) ;
- etc.

Avec un service de ce type, l'entreprise peut « se faire la main » sur le recours à une solution SaaS, et expérimenter les problématiques de sécurité des données, de disponibilité du service, de qualité du support aux utilisateurs et de confiance dans la relation avec l'opérateur.

Ce type de service n'est pas structurant puisqu'il s'agit de fonctions assez périphériques. Il constitue donc seulement un avant-goût du modèle cloud, sans en démontrer les réels bénéfices : déport des problématiques d'exploitation, élasticité, etc.

Pour tester véritablement le modèle cloud, il faudra faire le choix d'une application de commodité à usage quotidien. C'est seulement dans ce contexte que l'on pourra bénéficier complètement des avantages du modèle.

Le cas d'INDUS

INDUS utilise aussi des services de mailing marketing auprès de ses distributeurs.

À la suite d'une réflexion sur le modèle cloud et d'un processus de prise de décision rigoureux, INDUS a décidé de lancer en parallèle deux projets SaaS : la migration de ses outils collaboratifs vers Google Apps et l'utilisation de Salesforce pour la gestion de sa relation client.

INDUS souhaite par ailleurs monter une plateforme d'incubation sur Amazon Web Services.

10.1.3 Le pilote SaaS

On a vu dans le chapitre 8 que la rupture créée par le modèle cloud rendait nécessaire une phase de pilote. Ce pilote est simple à mettre en œuvre car le déploiement est entièrement fait chez l'opérateur cloud, auquel il suffit d'acheter un droit d'usage pour quelques mois. Le pilote n'impacte pas le système d'information de l'entreprise : il implique uniquement les utilisateurs et détermine leur appréhension des fonctionnalités proposées en SaaS.

Il est pertinent de mener une expérimentation avec une population réduite, de 10 à 200 personnes selon la taille de l'entreprise, pendant une durée de 1 à 6 mois, selon la complexité des fonctions à tester.

Cette phase pilote peut être basée sur le volontariat si l'on souhaite commencer le test avec les utilisateurs les plus enthousiastes, les « *early adopters* », souvent issus de populations de « *digital natives* ». On peut aussi choisir un modèle un peu contraignant, si l'on souhaite que l'expérimentation reflète les difficultés d'adoption qu'on aura avec l'ensemble de la population. Dans ce cas, il faudra convaincre les expérimentateurs de la pertinence du pilote. Cette seconde option permet de choisir des populations représentatives de l'entreprise (utilisateurs métiers, commerciaux, informaticiens, administratifs, etc.). En effet, la première option peut déboucher sur une population constituée uniquement d'informaticiens.

Les étapes de l'expérimentation sont généralement les suivantes :

- écriture d'un manuel de prise en main ;
- ouverture d'un forum de discussion, pour permettre les échanges entre expérimentateurs ;
- basculement des expérimentateurs vers la solution SaaS ;
- support par un expert de la solution et au travers du forum ;
- recours à des outils d'analyse de fréquentation pour mesurer leur adoption de la solution SaaS ;
- recours à des sondages pour mesurer la satisfaction des utilisateurs, ainsi que leur perception des bénéfices vis-à-vis de la solution précédente.

Le cas d'INDUS

INDUS a choisi de mener deux expérimentations :

– le pilote sur la migration des outils de collaboration vers le modèle cloud impliquera 200 utilisateurs pendant 6 mois. Les utilisateurs ont été choisis parmi les populations jeunes dans chaque corps de métier. La problématique d'accompagnement au changement des utilisateurs moins enthousiastes sera traitée après le pilote ;

– le pilote sur le déploiement d'une solution CRM SaaS impliquera 10 commerciaux volontaires pendant 3 mois.

Le pilotage de l'expérimentation

La phase expérimentale peut être encadrée par diverses typologies d'acteurs : DSI, département R&D, direction métier, cabinet de conseil.

On peut imaginer que la DSI prenne en main le pilote, mais l'équipe en charge du pilote peut être influencée par ses collègues de la production qui auront à cœur de démontrer l'inefficacité du modèle cloud, par suite des craintes qu'ils nourrissent sur la perte de leur périmètre de responsabilité. De plus, de par leurs responsabilités sur la stabilité et la continuité de service des systèmes, les équipes d'exploitation sont souvent allergiques au changement. Enfin, il arrive que les informaticiens privilégient la technique sur la relation avec les utilisateurs, ce qui est fort dommage dans une expérimentation centrée sur les utilisateurs.

Certaines entreprises disposent d'un département R&D ou département innovation, tourné vers la veille et les nouveaux usages. Ce département pourra mener à bien l'expérimentation en utilisant des méthodes agiles et en cultivant la proximité avec les expérimentateurs. Le risque, avec ces équipes enthousiasmées par les nouveaux usages, est de passer à côté de problèmes très spécifiques, qui peuvent se révéler bloquants (par exemple, la solution SaaS n'autorise pas une fonction qui paraît anecdotique mais qui est très utilisée dans l'entreprise).

Enfin, on peut faire appel à un cabinet de conseil qui apportera deux avantages intéressants : une indépendance vis-à-vis des problématiques politiques internes à l'entreprise, et une expérience de la démarche dans d'autres entreprises.

L'accompagnement aux expérimentateurs

La phase de pilote constitue un galop d'essai dans l'accompagnement au changement. Elle permet de tester la compréhension des utilisateurs et leur adhésion vis-à-vis de la solution SaaS. Il est donc essentiel de bien les accompagner.

Il est souhaitable de créer, en vue de cette phase, un document de prise en main de l'application SaaS, synthétique, pragmatique et opérationnel. Le recours à la documentation en ligne de l'opérateur SaaS peut, en effet, faire perdre du temps aux expérimentateurs. Le document de prise en main pourra être partagé au travers d'une solution SaaS (*cf.* chapitre 15 sur les solutions de stockage SaaS), afin d'élargir le périmètre d'expérimentation du modèle.

Il est aussi important de cultiver les échanges avec les utilisateurs afin de répondre en temps réel à leurs interrogations. L'expérimentation doit être effectuée en conditions de travail réelles pour être probante, ce qui générera une surcharge de travail pour les utilisateurs. Il faut donc les débloquer rapidement en cas de problème pour s'assurer qu'ils continuent à utiliser l'application SaaS.

La mise en place d'un forum de discussion est très indiquée pour faciliter les échanges et permettre à tous les expérimentateurs de bénéficier des retours d'expérience des utilisateurs les plus avancés.

Le forum de discussion pourra s'appuyer sur une solution SaaS (*cf.* chapitre 14 sur les solutions de collaboration SaaS), afin d'élargir le périmètre d'expérimentation du modèle.

Le pilote SaaS pour une nouvelle application

Lorsque l'expérimentation porte sur une nouvelle application, l'impact sur le SI est totalement nul et il est possible de mener le pilote sans participation active de la direction informatique.

Le pilote SaaS en remplacement d'une application

En revanche, lorsque l'expérimentation porte sur le remplacement d'une application, il est nécessaire de maintenir les deux systèmes opérationnels en parallèle pendant la durée du pilote. Il est en effet essentiel de maintenir la réversibilité : les expérimentateurs doivent pouvoir revenir à l'ancien système si l'expérimentation se passait mal.

Par ailleurs, si l'application implique la collaboration d'utilisateurs intégrés au pilote avec des utilisateurs travaillant sur l'ancien système, il est nécessaire de bâtir des passerelles entre les deux systèmes pendant la durée de l'expérimentation, **c'est-à-dire de faire de l'intégration entre SI et cloud.**

Prenons l'exemple de calendriers partagés : pour pouvoir travailler dans des conditions normales, il faut que tous les collaborateurs puissent partager leurs calendriers qu'ils fassent ou non partie du pilote. Cela implique une synchronisation des calendriers entre les deux systèmes parallèles.

Il est essentiel de noter que les problématiques évoquées ne sont pas propres aux projets SaaS : on les rencontrera dans tous les projets de migration d'un système A vers un système B.

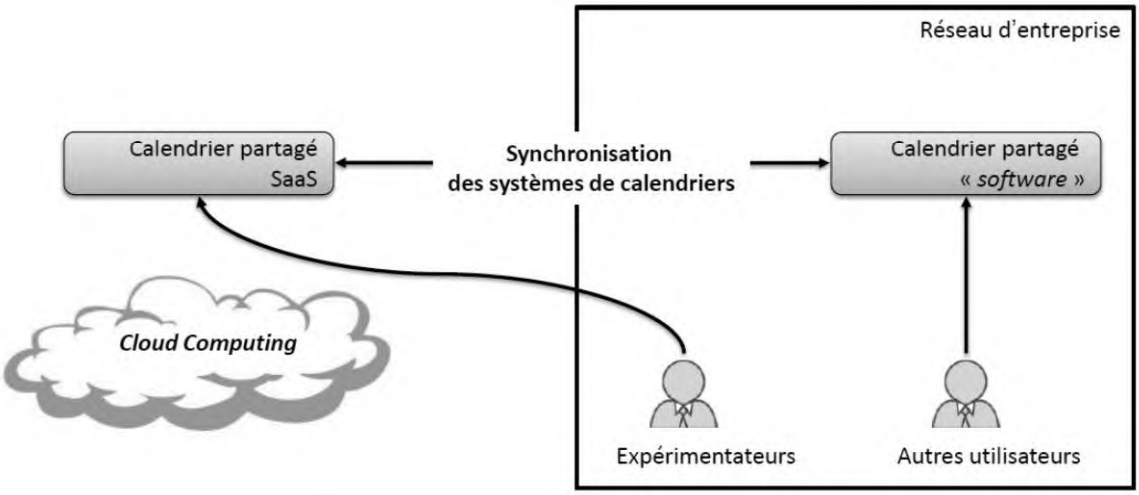

Figure 10.1 — La coexistence de deux systèmes pendant le pilote.

La mesure de la réussite de l'expérimentation

Pour mesurer la réussite du pilote, il est indispensable de définir, avant son démarrage, les indicateurs et les seuils indiquant une réussite. En effet, un travail sur les indicateurs en cours d'expérimentation serait biaisé par les premiers retours utilisateurs.

Ces indicateurs devront intégrer deux types de mesure :

- **une mesure quantitative** qui portera sur des résultats tangibles, collectés à partir de capteurs, comme le suivi des accès utilisateurs à l'application SaaS, le suivi de la quantité d'informations manipulés par les utilisateurs au sein de l'application SaaS. La collaboration de l'opérateur SaaS sera nécessaire pour collecter ces indicateurs : il devra fournir une console de suivi ou une API permettant de collecter des données brutes ;
- **une mesure qualitative** qui portera sur la satisfaction des utilisateurs. Elle sera effectuée au travers de sondages de satisfaction. On recommande de procéder à une série de sondages tout au long de l'expérimentation pour mesurer la progression de l'adhésion des utilisateurs à l'application SaaS. Pour effectuer ces sondages, il est possible de recourir à un outil SaaS, afin d'élargir le périmètre d'expérimentation du modèle.

C'est la confrontation des mesures aux seuils définis auparavant qui permettra de mesurer si l'expérimentation est un succès ou non.

10.1.4 Le pilote IaaS

De même que pour le SaaS, le pilote est simple à mettre en œuvre car le déploiement est entièrement fait chez l'opérateur cloud, auquel il suffit d'acheter un droit d'usage pour quelques mois. Le pilote n'impacte pas le système d'information de l'entreprise, ni les utilisateurs : c'est un pilote technique qui ne concerne que les collaborateurs de la DSI. Son expérimentation est donc grandement simplifiée.

Le pilotage de l'expérimentation

Le pilote peut être pris en charge par la cellule architecture de la DSI dans le cadre de la création **d'un centre de compétence cloud**. Il pourra aussi être géré par la cellule innovation, si la DSI en possède une.

L'expérimentation porte par exemple sur les environnements de développement, ou un « bac à sable innovation ». Elle consiste à prendre en main les machines virtuelles de la plateforme pour y installer les couches logicielles nécessaires. Cette prise en main est relativement simple dans la mesure où beaucoup de DSI utilisent déjà des technologies de virtualisation aujourd'hui.

L'expérimentation

Le pilote permet d'affiner :

- **la compréhension du modèle de coût :** coût effectif, pertinence d'éteindre les machines virtuelles en cas d'inactivité, etc. Ce point est important, car certaines plateformes, comme Amazon Web Services, ont un modèle tarifaire très complexe ;
- **la connaissance de la plateforme :** rapidité de prise en main, mode de déploiement, temps de déploiement d'une nouvelle application, système de sauvegarde, sécurité du pilotage à distance, système de persistance des données en cas d'arrêt des machines virtuelles, fonctionnalités spécifiques propres à la plateforme (*cf.* chapitre 19), etc.

La connaissance des spécificités du cloud : problématiques de latence réseau, de disponibilité réelle, etc.

La mesure de la réussite de l'expérimentation

De la même manière que pour le pilote SaaS, il est pertinent de se donner des indicateurs de réussite pour valider le pilote avant l'expérimentation, par exemple :

- la plateforme IaaS revient effectivement moins chère que l'hébergement en interne ;
- le *Self Service* permet effectivement la mise en ligne d'une application en moins d'une heure ;
- la migration d'une application du SI vers le cloud prend moins de 2J ;
- la disponibilité est de 99,9 % sur l'IaaS ;
- les temps de réponse sont inférieurs à 200 ms sur l'IaaS ;
- etc.

10.2 LE PREMIER DÉPLOIEMENT

Une fois la phase de pilote terminée et la décision d'aller vers un service cloud entérinée, il est temps de préparer la conduite du changement.

On a vu dans les paragraphes précédents que cette phase pilote pouvait être menée par le département innovation de l'entreprise, par une société de conseil, etc. Cette première phase terminée, l'application va devoir être reprise en main par l'équipe d'exploitation : on parle de phase d'« **industrialisation** » car on quitte le mode expérimental pour rejoindre un mode production.

10.2.1 La mise en production sur le cloud

Une fois les étapes de négociation et de contractualisation avec l'opérateur cloud terminées, la direction informatique reçoit un identifiant/mot de passe pour l'accès à la console d'administration de la plateforme cloud.

Elle doit alors prendre en main cette console et se familiariser avec les fonctions proposées :

- **gestion des comptes et des droits pour les utilisateurs :** nous allons détailler cet aspect ci-après ;
- **paramétrage des options proposées :** choix des fonctionnalités à activer dans le cas du SaaS ; choix des machines virtuelles dans un catalogue dans le cas de l'IaaS ; choix des services techniques à utiliser dans le cadre du PaaS ;
- **customisation des domaines et DNS[1] des applications :** il s'agit d'utiliser les noms de domaines de l'entreprise pour donner une cohérence au parc applicatif ; par exemple, on accédera à la messagerie depuis l'adresse http://mail.monentreprise.com ;
- **API pour l'échange de données avec le SI d'entreprise :** nous allons détailler cet aspect ci-après ;
- **parfois, API pour la collecte des traces des actions utilisateurs :** ce besoin de collecte de traces pour assurer le monitoring a été évoqué ci-dessus.

Les consoles cloud sont généralement simples à prendre en main. Elles proposent un paramétrage standard très compréhensible, et parfois un paramétrage avancé. Pour illustrer cette simplicité, on peut dire que les plus simples ressemblent aux espaces clients des opérateurs ADSL ou des banques en ligne ; les plus complexes ressemblent à celles des hébergeurs, comme OVH.

Gestion des comptes utilisateurs

Les plateformes SaaS et PaaS fournissent généralement un annuaire d'utilisateurs qu'il faudra gérer suivant les scénarios suivants. Dans le cas de l'IaaS, on crée seulement quelques comptes administrateurs, et les paragraphes suivants ne s'appliquent pas (en effet l'annuaire des utilisateurs fera partie de la pile logicielle installée par l'administrateur).

L'intégration des comptes utilisateurs dans la plateforme SaaS/PaaS suit généralement l'un des deux scénarios suivants : gestion manuelle ou gestion automatique.

1. *Domain Name Service.*

Dans le cadre de la création des comptes de manière manuelle, le gestionnaire crée les comptes avec des identifiants identiques à ceux utilisés au sein de l'entreprise. Les mots de passe sont laissés à l'appréciation des utilisateurs : ces derniers pourront utiliser leur mot de passe d'entreprise, sans aucune obligation. Si leur mot de passe d'entreprise change tous les trois ou six mois, selon la politique de mot de passe en vigueur, ils devront répercuter manuellement ce changement sur leur mot de passe SaaS/PaaS. Le gestionnaire de la plateforme cloud devra penser à supprimer les comptes des collaborateurs démissionnaires. Cette gestion manuelle est recommandée pour une population réduite (moins de 50 comptes), elle peut se révéler très lourde pour une population conséquente (plus de 1 000 comptes).

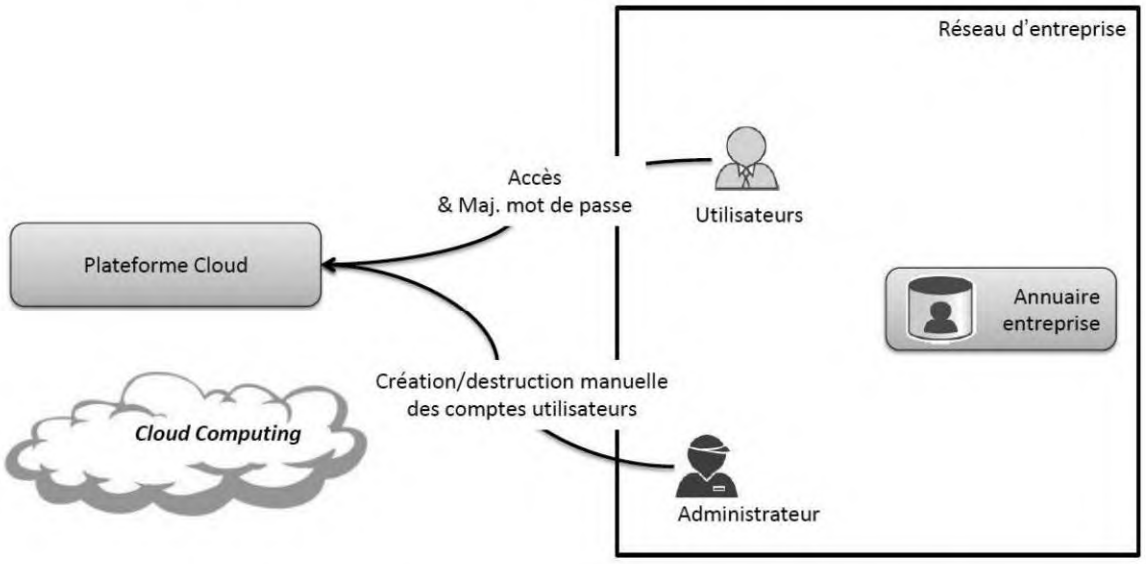

Figure 10.2 — Création manuelle des comptes SaaS/PaaS.

Dans le cadre de la création des comptes de manière automatisée, les comptes sont créés depuis le système d'information d'entreprise. Dans ce cas, la DSI va utiliser les API mises à disposition par l'opérateur cloud pour la gestion des comptes. Trois processus seront déployés :

- création automatique d'un compte depuis l'annuaire d'entreprise, à l'arrivée d'un nouveau collaborateur ;
- modification automatique du mot de passe utilisateur depuis l'annuaire d'entreprise ;
- suppression automatique de compte depuis l'annuaire d'entreprise, au départ d'un collaborateur.

Bien entendu, les mots de passe seront envoyés vers l'opérateur cloud au travers d'un protocole sécurisé, comme SSL.

Ces processus de gestion de compte sont généralement appelés « *provisioning* ». Il convient d'apporter le plus grand soin à ces processus car ils sont déterminants pour la sécurité de l'application cloud.

Nous verrons dans le chapitre 11 que cette intégration peut se révéler plus complexe dans le cas de plusieurs clouds, et d'une industrialisation de la gestion des comptes.

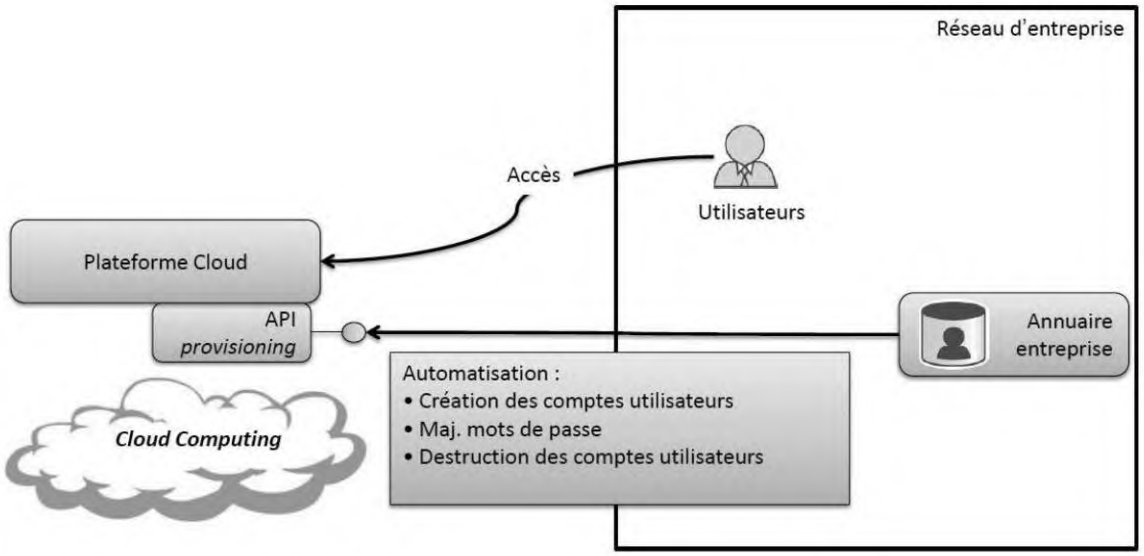

Figure 10.3 — Création automatisée des comptes SaaS/PaaS.

Dans le cadre de la fédération d'identité, les comptes sont stockés uniquement dans le système d'information d'entreprise. L'opérateur cloud délègue l'authentification au service d'identité du SI, l'*Identity Provider* (IDP).

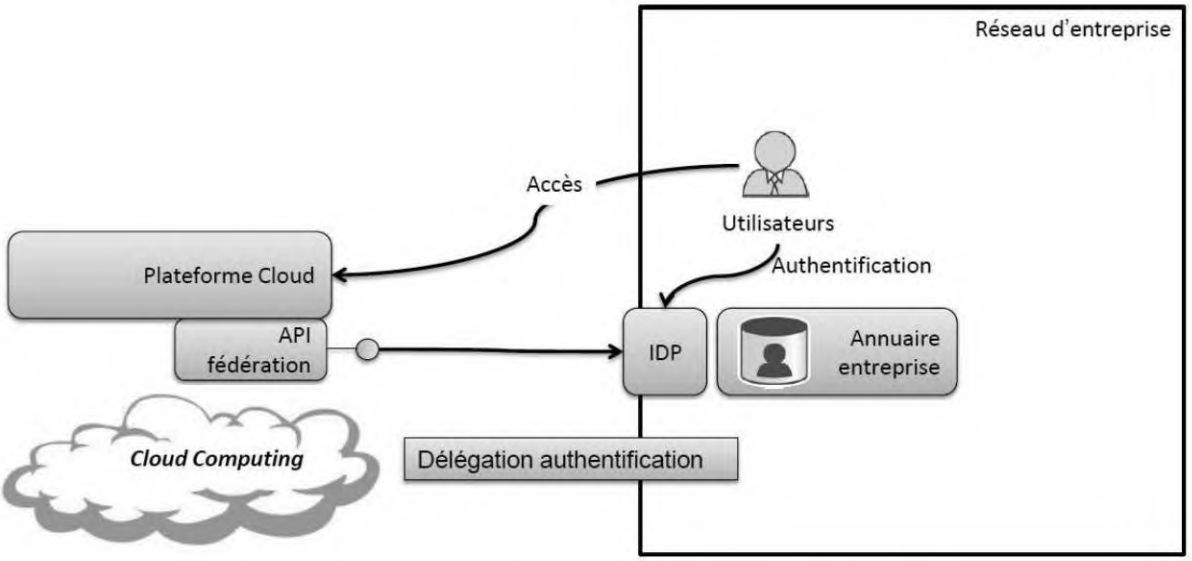

Figure 10.4 — Fédération d'identité avec SaaS/PaaS.

À propos de la fédération d'identité

La fédération d'identité est une approche d'architecture qui propose d'établir des liens de confiance de manière distribuée entre des services applicatifs et un annuaire utilisateurs, appelés *serveur d'identité*. Le serveur d'identité est une sorte d'annuaire LDAP amélioré, capable d'authentifier des accédants pour le compte d'applications internes ou externes à l'entreprise. Il fournit aussi des fonctions de Single Sign On.

Le principal objectif de la fédération d'identité est de faciliter les échanges entre partenaires sans avoir à dupliquer les annuaires de sécurité et donc en faisant de la délégation d'authentification sur la base de liens de confiance entre entreprises.

Intégration de données

Pour les mêmes raisons que pour la gestion des comptes utilisateurs, ce paragraphe concerne les plateformes SaaS et PaaS, mais pas les IaaS.

Lors de la mise en production d'une application cloud, il peut être nécessaire d'y importer un certain nombre de données préexistantes, comme un carnet d'adresse clients dans une application de CRM, des archives d'e-mails dans une application de messagerie, des archives de documents dans une application de partage documentaire, etc. **Ce cas n'est pas le cas général : bien souvent, on peut démarrer une application à partir d'un contenu vide.**

Pour intégrer des données préexistantes, on peut à nouveau considérer deux scénarios : gestion manuelle ou gestion automatique.

Dans le cadre de la migration de données de manière manuelle, les utilisateurs tirent parti de fonctionnalités d'import/export disponibles dans les interfaces des applications SaaS/PaaS. L'import se fera la plupart du temps au travers du format CSV[1].

Dans le cadre de la migration de données de manière automatisée, les informations sont importées depuis le système d'information d'entreprise. Dans ce cas, la DSI va utiliser les API d'intégration mises à disposition par l'opérateur SaaS/PaaS pour la gestion de l'import. Deux types de processus pourront être déployés :

- envoi automatique de données depuis le SI d'entreprise vers l'opérateur cloud, lorsque de nouvelles informations sont disponibles ;
- collecte automatique de données depuis l'opérateur cloud, lorsque des informations sont traitées chez ce dernier. Il convient d'être prudent dans ce second scénario, car l'import automatique de données dans le SI depuis une source extérieure peut comporter un risque sur l'intégrité des données d'entreprise.

Nous verrons dans le chapitre 10 que cette intégration peut se révéler plus complexe dans le cas de plusieurs clouds, et d'une industrialisation de la gestion des échanges de données.

1. *Coma Separated Values.*

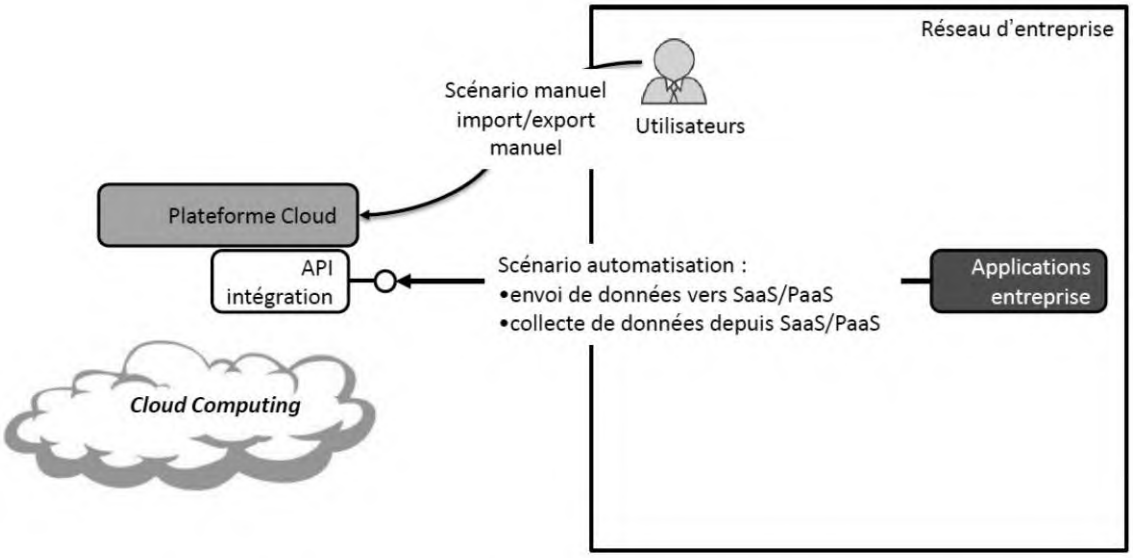

Figure 10.5 — Scénarios d'échanges de données avec opérateurs SaaS/PaaS.

Cas du déploiement d'une nouvelle application

Dans le cas d'une nouvelle application, la mise en production peut se révéler triviale : elle se réduit généralement à la création manuelle des comptes utilisateurs chez l'opérateur cloud, et à l'import manuel de données préexistantes.

> **Le cas d'INDUS**
>
> Pour le déploiement de sa CRM chez Salesforce, la direction informatique a décidé de procéder à des imports manuels.
>
> En effet, compte tenu de la taille réduite de l'équipe commerciale (50 collaborateurs), le nombre de comptes utilisateurs à gérer ne justifie pas le développement de processus automatisés.
>
> Pour ce qui concerne l'import des données clients, il se trouve que chaque commercial avait jusqu'à présent sa méthode et ses outils de gestion de clientèle. De fait, un import de masse est impossible, et chaque commercial se chargera d'importer lui-même son historique de relation client.

Cas du remplacement d'une application

Dans le cadre du remplacement d'une application, la mise en production est plus ardue. Un certain nombre d'étapes doivent en effet être franchies :

- **résolution des « effets de bords » :** le changement de solution implique des différences dans la gestion de certaines fonctionnalités. Ces modifications impliquent parfois de nouveaux modes de travail. Prenons l'exemple d'une migration de messagerie : la gestion des fonctions annexes comme les listes de diffusion, la lutte anti-spam, le mailing marketing peut se révéler différente et impliquer une nouvelle organisation. Si l'opérateur cloud ne fournit pas de

fonction de mailing de masse, il faudra se doter d'une plateforme tierce pour gérer cette fonctionnalité ;

- **cohabitation des deux solutions :** il est nécessaire de faire cohabiter les deux solutions pendant la phase de transition, une partie des utilisateurs accédant à l'ancienne solution et l'autre à la nouvelle. Cette cohabitation est complexe à gérer dans le cadre d'une messagerie, où les messages devront être routés vers l'une et l'autre des solutions par un service de *dispatching* (en l'occurrence, un service de proxy) ;
- **synchronisation des données pendant la phase de transition :** il est nécessaire de synchroniser les données entre les deux solutions pendant la phase de transition, afin de permettre aux deux populations d'utilisateurs de travailler ensemble dans des conditions normales. L'exemple du partage de calendrier illustre bien cette problématique ;
- **basculement par lots :** pour une grande entreprise, il est impossible de basculer en un seul coup tous les utilisateurs vers la nouvelle solution. Plusieurs raisons expliquent cette incapacité : le risque technique, le besoin de faire monter en charge progressivement la nouvelle solution, le besoin d'accompagner progressivement les utilisateurs ;
- **fin de service pour la solution historique :** lorsque le basculement est achevé, il est possible de désactiver la solution historique (solution « software »). Il est cependant déconseillé de la démembrer immédiatement : il est prudent de la conserver pendant six mois, en cas d'émergence d'un problème non anticipé.

Dans le cas d'une migration, il est donc nécessaire de mener un « **plan de migration** » sur plusieurs mois, afin de gérer correctement les étapes décrites ci-dessus.

La migration est très comparable à celle d'une application vers une autre au sein du SI. Les différences tiennent au modèle cloud : problématiques d'intégration et sécurisation de flux hors SI.

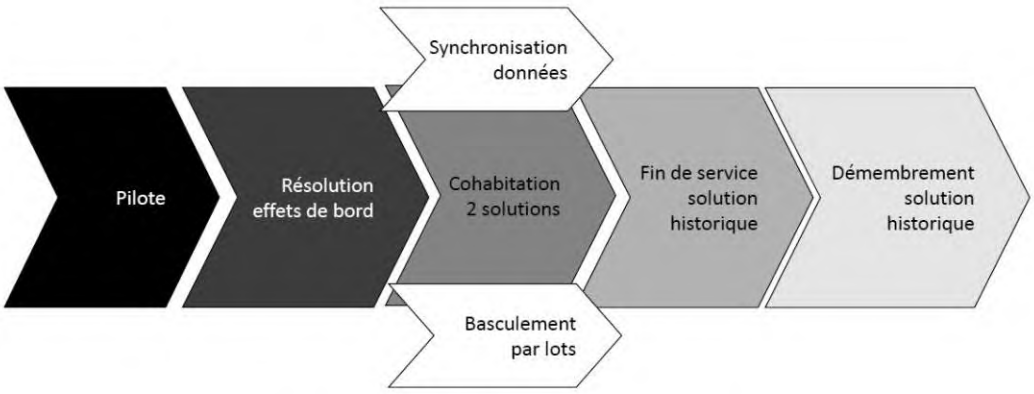

Figure 10.6 — Plan de migration vers le cloud.

Le cas d'INDUS

Après une petite étude, INDUS a décidé de se faire accompagner par un cabinet de conseil pour la migration de ses outils collaboratifs vers Google Apps.

Il a été décidé de mettre en œuvre un *provisioning* automatique des comptes entre l'annuaire réseau INDUS et Google Apps. Les API fournies par l'opérateur permettent de déployer les trois processus automatisés : création de compte, mise à jour de mot de passe, suppression de compte.

L'import des archives e-mails et des documents chez Google sera effectué par les utilisateurs eux-mêmes au travers d'outils fournis par l'opérateur.

Les deux messageries cohabiteront pendant 6 mois, en maintenant les calendriers synchronisés.

La migration des utilisateurs se fera site par site, à raison de groupes de 500 collaborateurs, pendant 18 mois.

10.2.2 La conduite du changement pour les utilisateurs

Comme on l'a vu au chapitre 6, le déploiement d'une solution cloud peut donner aux utilisateurs une impression de perte de possession. En effet, les données ne résident plus en local sur leur PC, mais sur un serveur distant sur lequel ils ont peu de contrôle. Les utilisateurs doivent accepter un changement de modèle : passer d'applications embarquées sur leur machine et parfois administrées par eux-mêmes, à des interfaces web, qui gèrent plus ou moins bien le mode déconnecté[1], dans un contexte nomade.

Si les interfaces HTML5 offrent un bon niveau de productivité, elles se présentent différemment des applications client/serveur. Il est donc nécessaire de documenter ces interfaces.

Enfin, avec le cloud, la DSI peut leur couper l'accès au service sur ordre de la direction générale et ils perdent instantanément tout accès à leurs données. Cette option est souhaitable pour l'entreprise qui veut protéger ses données d'un pillage par un collaborateur démissionnaire ; en revanche, elle donne aux employés un sentiment de « dictature ».

Il convient donc d'accompagner les utilisateurs sur ces changements :

- **perte de contrôle des applications et problématique du mode déconnecté :** il faut leur expliquer que le déploiement d'application poste par poste est une terrible charge pour la DSI et que les applications seront plus accessibles en mode SaaS (accès depuis leur domicile, un cybercafé, un aéroport, etc.) ; la problématique du mode déconnecté devra être gérée par une synchronisation ;
- **centralisation des données et possibilité de coupure de leur accès :** il faut leur expliquer que cette architecture assure un meilleur niveau de sécurité pour les données d'entreprise ; l'intégrité des données est mieux gérée dans un

1. On a vu dans le chapitre 1 que HTML5 permet de travailler en mode déconnecté ; mais cette solution n'est pas généralisée aujourd'hui.

Datacenter que sur un fragile PC ; la protection des données contre un accès illicite est mieux gérée dans un Datacenter que sur un PC éventuellement nomade.

Le cas d'INDUS

Pour accompagner l'équipe commerciale dans la prise en main de sa nouvelle solution de CRM, INDUS a profité d'un séminaire commercial pour organiser une journée de formation et d'échange sur Salesforce. Le déroulé du séminaire a été organisé par les expérimentateurs de la solution. Les commerciaux ont beaucoup apprécié le caractère interactif de ce séminaire.

Pour ce qui concerne la migration des outils collaboratifs vers Google Apps, le cabinet de conseil en charge de l'accompagnement a créé une formation en demi-journée, intégrant les spécificités d'INDUS. Cette formation a été donnée en parallèle du déploiement site par site. Chaque bascule utilisateur a été accompagnée afin de ne laisser personne découvrir seul les nouvelles interfaces.

En résumé

Ce chapitre a abordé les différentes étapes nécessaires au déploiement d'une première application sur le cloud :

– phase de pilote ;
– phase de déploiement et problématiques d'intégration avec le SI ;
– phase de conduite du changement.

11

La gestion d'un système d'information hybride

Objectif

Ce chapitre aborde la question d'un SI hybride constitué d'applications internes et d'applications cloud. Il met l'accent sur la cohabitation entre le cloud et les applications de la plateforme d'entreprise.

11.1 L'INDUSTRIALISATION DU DÉPLOIEMENT DES APPLICATIONS CLOUD

Une entreprise qui utilise le modèle cloud avec succès pour quelques applications pourra souhaiter étendre progressivement le modèle à une grande partie de son informatique de commodité, afin de réduire ses coûts et de recentrer son équipe informatique sur ses applications métiers.

Chaque externalisation applicative fera l'objet d'une évaluation. Dans certains cas, l'entreprise se contentera de faire héberger son application en l'état, suivant le modèle outsourcing. Dans d'autres cas, elle choisira le modèle cloud.

Le cas d'INDUS

INDUS utilise un ERP pour la gestion de ses échanges avec ses sous-traitants. Cet ERP a été fortement customisé pour tenir compte des spécificités métier d'INDUS. Il a donc perdu sa généricité et il est impossible à INDUS de faire appel à un opérateur SaaS pour lui fournir un service identique. INDUS a donc pris une option de l'exploiter sur une plateforme IaaS.

Pour ses applications de CRM et ses outils collaboratifs, on rappelle qu'INDUS a des besoins suffisamment génériques pour aller vers des opérateurs SaaS. Les solutions Salesforce et Google Apps ont été choisies.

La figure 11.1 présente l'évolution du SI d'INDUS. On verra plus loin, avec la figure 11.3, un scénario analogue pour les développements spécifiques.

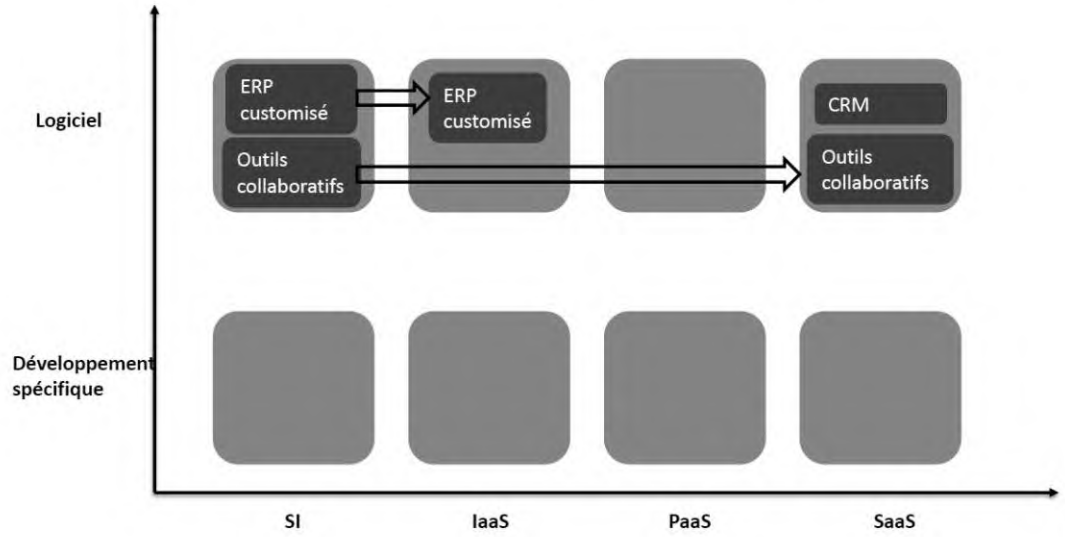

Figure 11.1 — Changements de modèles logiciels.

À chaque migration, il faudra traiter comme on l'a vu précédemment :

- le choix de la solution ;
- la phase pilote ;
- l'accompagnement au changement des équipes de la DSI et des utilisateurs.

À la suite de ses divers retours d'expérience, l'entreprise utilisatrice saura progressivement industrialiser ces pratiques. Elle va alors fatalement se poser la question d'une stratégie de choix des offres cloud :

- **Souhaite-t-elle utiliser au maximum un opérateur cloud unique ?** de la même manière que certaines entreprises utilisent exclusivement des solutions IBM ou des solutions Microsoft ? Cette première option fait courir le risque d'un enfermement, d'être « pieds et mains liés » à un seul acteur. En revanche, elle permet de disposer d'une base de comptes utilisateurs unique, d'une interface d'accès unifiée, et de la possibilité de partager des données entre plusieurs

applications chez l'opérateur. Cette option est d'autant plus envisageable que beaucoup d'opérateurs proposent à la fois plateforme SaaS et plateforme PaaS ou IaaS. Il est donc possible d'externaliser chez eux des applications génériques et des applications spécifiques.

- **Souhaite-t-elle utiliser le meilleur acteur du marché pour chaque fonction ?** selon une approche *best of breed* [1] ? On peut ainsi choisir Google Apps pour la collaboration, Basecamp pour la gestion de projet, Amazon Web Services pour le déport d'applications développées en interne, etc. Dans ce cas, la cohabitation des applications cloud va se révéler plus ardue. Il va falloir en effet trouver des solutions pour traiter les problématiques d'authentification unique, d'interface unifiée et d'intégration des données. C'est ce que nous allons voir dans le paragraphe suivant.

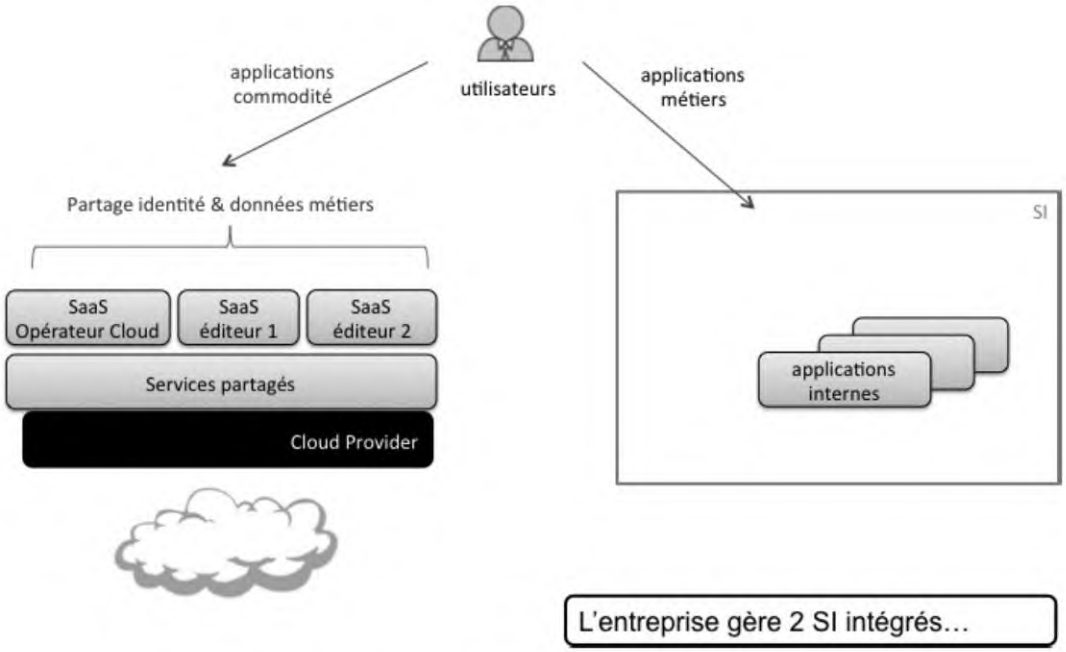

Figure 11.2 — Le recours à un écosystème intégré.

Le cas de Salesforce

Salesforce a eu très tôt l'idée d'ouvrir sa plateforme, intitulée App Exchange, à des éditeurs tiers afin qu'ils développent des applications qui seront hébergées et opérées par ses soins. Ces éditeurs vendent leurs applications aux entreprises intéressées et Salesforce leur loue un hébergement.

Salesforce propose ainsi, sur App Exchange, un vaste écosystème d'applications SaaS répondant à divers besoins : analyse et tableaux de bord décisionnels, gestion des ressources humaines, gestion financière, gestion d'acheminement, gestion de projet, gestion de centre d'appel, outils marketing, gestion des relations partenaires, etc.

1. En français, on traduirait cette expression par « meilleur de son espèce ».

Salesforce propose même des applications métiers pour certains secteurs d'activité. Par ailleurs, les applications hébergées sur App Exchange peuvent facilement collaborer entre elles.

11.2 L'INTÉGRATION DES CLOUDS

La plupart des entreprises utilisatrices du cloud sont confrontées à la problématique d'intégration des applications cloud aux applications suivant le modèle « software » situées à l'intérieur de leur réseau d'entreprise.

Si l'on prend le cas d'un logiciel de messagerie SaaS, il n'aura pas nécessairement à échanger des données métiers avec le réseau d'entreprise. En revanche, il devra authentifier les utilisateurs, et de préférence avec le même identifiant/mot de passe que les applications internes à l'entreprise. **Cela milite en faveur d'un service d'identité unique.**

Si l'on prend l'exemple de l'externalisation du logiciel de gestion de la relation client, il est possible que ce dernier doive échanger des informations avec le logiciel de gestion financière sans doute hébergé dans les murs de l'entreprise. Ces échanges pourront être fréquents et éventuellement à double sens. **Si ces échanges entre le SI et les clouds se multiplient, l'entreprise devra donc se doter d'un logiciel d'intégration ou d'un bus d'échange pour gérer ces flux de données.**

Enfin, pour optimiser la productivité des utilisateurs, l'entreprise peut recourir à un portail pointant vers toutes ses applications. Ce portail peut être une simple page de liens vers des applications web : ce n'est pas nécessairement un logiciel de portail à proprement parler. Dans ce cas, il est tout à fait heureux que **ce portail s'intègre aussi bien avec les applications cloud qu'avec les applications positionnées sur le réseau d'entreprise.**

Ce paragraphe fait donc émerger trois services d'intégration quasi indispensables entre les clouds et les applications situées dans l'entreprise :

- **le service d'identité,** pour permettre aux utilisateurs d'accéder en toute transparence aux applications issues des deux mondes ;
- **le bus d'intégration,** pour faire circuler les messages entre les deux mondes ;
- **le portail,** pour permettre aux utilisateurs de disposer d'une interface unique vers les applications issues des deux mondes.

La figure 11.3 récapitule ces trois services.

Ces services peuvent eux aussi être issus des deux modèles cloud et « software ». Il existe ainsi des services d'infrastructure, disponibles sur le cloud.

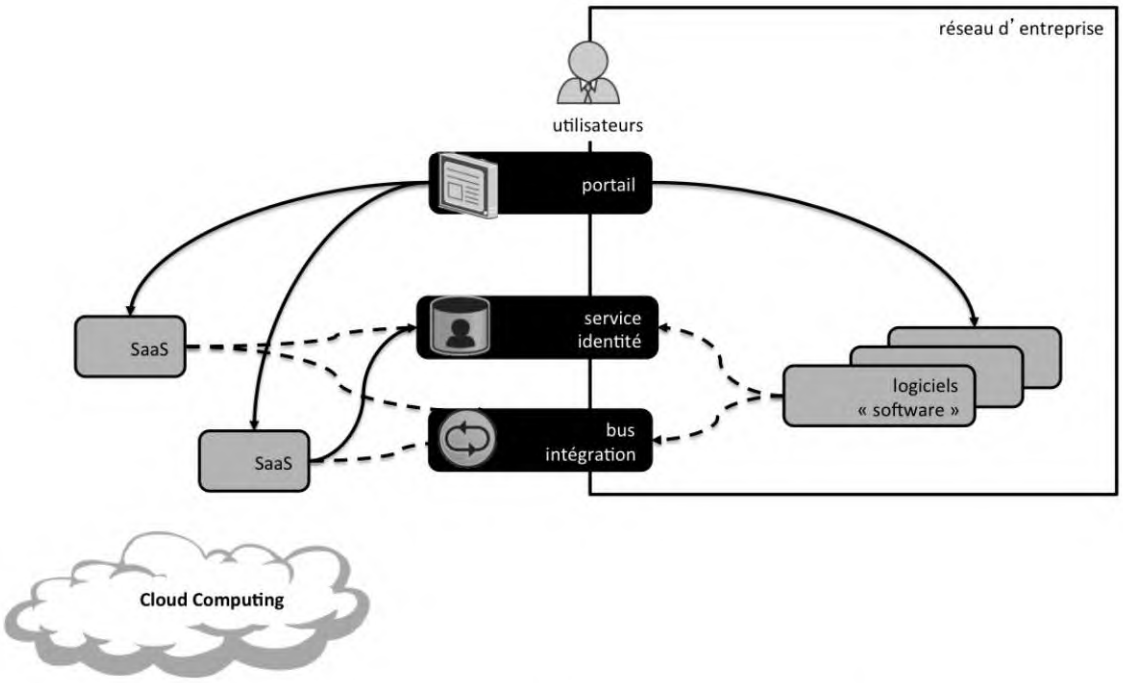

Figure 11.3 — Les trois services d'intégration entre cloud et réseau d'entreprise.

11.2.1 Le service d'identité

On a vu au chapitre 9 que trois options de gestion des comptes utilisateurs s'offraient aux entreprises qui déploient leur première application cloud : la gestion manuelle, le *provisioning* automatisé et la fédération d'identité.

Si le recours au cloud se généralise, l'entreprise utilisatrice va probablement s'orienter vers un système d'authentification centralisé, un **service de fédération d'identité**, plutôt que vers la propagation des identifiants/mots de passe vers ses opérateurs cloud, et ce, pour plusieurs raisons.

- **La sécurité des mots de passe :** la propagation de mots de passe vers divers partenaires pose un problème de sécurité. Si quelqu'un de malintentionné parvenait à les capturer pendant l'envoi ou depuis la plateforme de l'opérateur cloud, il existerait un gros risque d'usurpation d'identité sur l'ensemble des applications cloud de l'entreprise utilisatrice. Un service d'identité centralisé offre un bien meilleur niveau de sécurité.
- **Le risque d'une erreur de *provisioning* :** si une erreur se glisse dans l'un des processus de *provisioning*, l'entreprise peut avoir à gérer les dysfonctionnements suivants : un utilisateur ne peut pas travailler car son compte n'a pas été créé ou son mot de passe n'a pas été mis à jour ; un collaborateur ayant quitté l'entreprise continue à accéder à une application cloud car son compte n'a pas été supprimé. Ce dernier scénario est catastrophique en termes de sécurité.
- **Le renforcement de l'authentification :** un système de sécurité centralisé permet d'appliquer une politique de mot de passe efficace et d'apporter une solution aux carences de certains clouds sur le renouvellement des mots

de passe (*cf.* chapitre 7) ; il permet aussi de mettre en œuvre un système d'authentification forte.

- **La fourniture d'un système de *Single Sign On* (SSO) :** un service de SSO est un confort supplémentaire pour les utilisateurs ; il leur permet de basculer entre les applications cloud et les applications situées dans l'entreprise en toute transparence, sans avoir à se réauthentifier ; il représente un gain de productivité.

À propos de l'authentification forte

On utilise le terme d'authentification forte lorsqu'on combine deux types de facteurs lors de l'authentification. Ces combinaisons peuvent être :
– connaissance d'un mot de passe et possession d'un objet ;
– connaissance d'un mot de passe et identification biométrique ;
– possession d'un objet et identification biométrique.
La combinaison la plus courante est la première. L'objet peut être une Carte Bleue, une carte SIM[1], un certificat numérique[2], un générateur électronique de mots de passe à usage unique[3], etc.

Le service de fédération d'identité centralisé doit fournir une API permettant aux applications de lui déléguer leur sécurité, qu'elles soient en mode cloud ou dans l'intranet de l'entreprise.

Une bonne pratique, pour l'entreprise utilisatrice, consiste donc à déployer un service d'identité centralisé et utilisant une authentification forte. Ce service devra être accessible pour les applications cloud et pour les applications internes à l'entreprise.

Il peut être déployé selon deux scénarios :

- déploiement dans l'entreprise sur la base d'un logiciel « software » fourni par un éditeur (Ping Identity, Sun, Novell, IBM, Oracle, EMC, CA, etc.) ;
- usage d'un service cloud, comme PingOne.com ou OneLogin.com. Dans ce scénario, il est clair que la direction de la sûreté de l'entreprise mènera une étude poussée de la solution avant de l'adopter. Il s'agit ni plus ni moins que de déléguer sa sécurité à un tiers. Cette stratégie est admissible plutôt par des PME.

Il existe aujourd'hui un protocole standardisé pour la mise en œuvre de la fédération d'identité : le protocole SAML (*Security Assertion Markup Language*). C'est une grammaire XML normalisée par l'OASIS qui permet d'attester la bonne authentification d'un utilisateur souhaitant accéder à plusieurs services. Une alternative a été proposée

1. SIM signifie *Subscriber Identity Module*. Les cartes SIM sont utilisées pour l'authentification des téléphones portables vis-à-vis des réseaux télécoms mobiles.
2. Voir le système d'authentification pour la télédéclaration des impôts en France.
3. Voir la solution SecurID de RSA : un porte-clés électronique qui génère un mot de passe toutes les trois minutes.

par les acteurs du mode web 2.0 : il s'agit de la norme OpenID[1]. Cette norme est plus simple à prendre en main que SAML, mais elle est plutôt destinée aux sites web grand public. Elle laisse l'utilisateur libre de fédérer son identité avec les sites qui l'intéresse, ce qui n'est pas toujours souhaitable pour les entreprises qui préfèrent contrôler les activités de leurs collaborateurs.

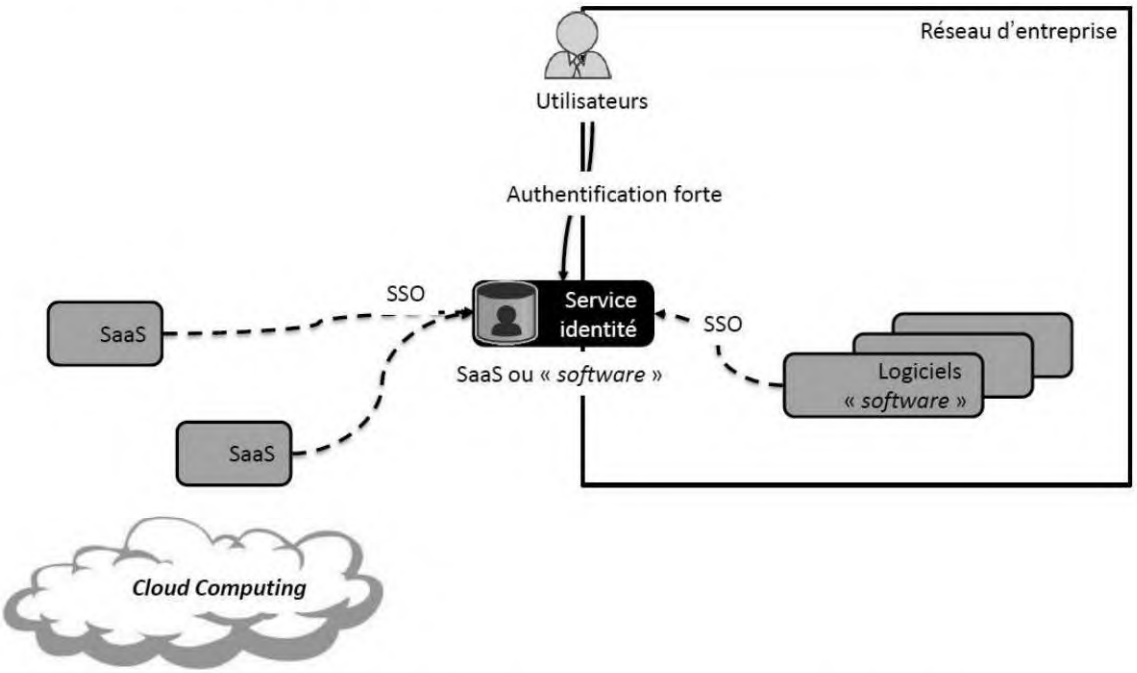

Figure 11.4 — Bonnes pratiques de service d'identité centralisé.

11.2.2 Le service d'identité cloud OneLogin

OneLogin est un service d'identité cloud simple à prendre en main : nous le conseillons dans le cadre d'un pilote sur la fédération d'identité.

Le service fournit :

- la fédération avec les protocoles SAML & OpenID ;
- un système d'authentification à deux facteurs, *via* un logiciel à installer sur téléphone Android ou iPhone ;
- la création automatique des comptes dans les plateformes SaaS ;
- des Logs d'activité sur la connexion des utilisateurs ;
- une gestion de groupes ;
- une gestion de politique de mot de passe ;
- un portail qui prend la forme d'une simple page de liens.

1. Il existe de nombreuses autres technologies de fédération d'identité. Nous n'avons pas souhaité les aborder pour éviter de noyer le lecteur.

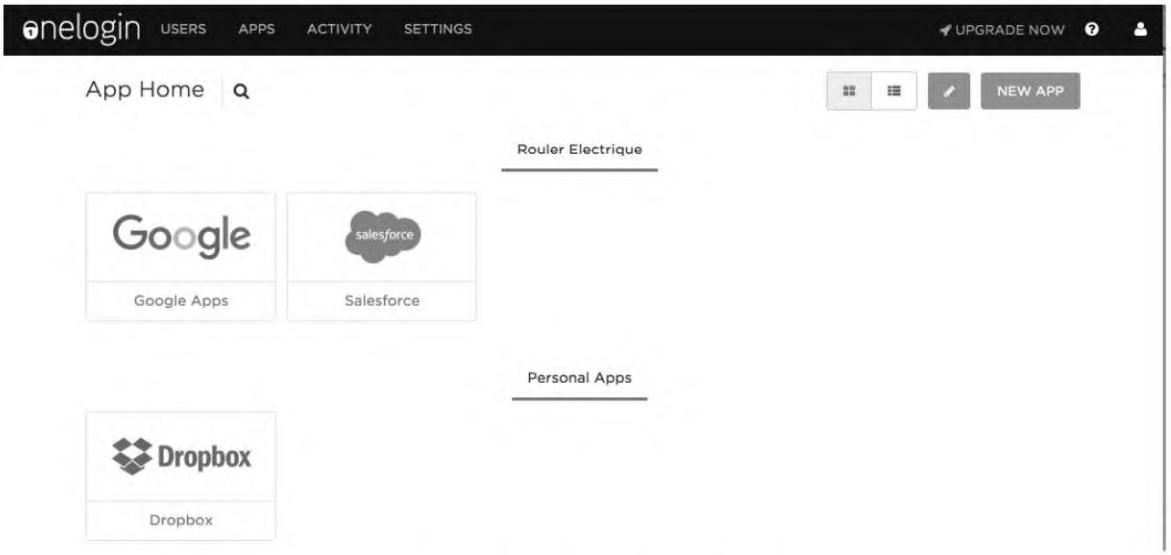

Figure 11.5 — Le portail OneLogin.

11.2.3 Le portail d'intégration

Rappel sur les portails

La métaphore du portail désigne la porte d'entrée du SI d'entreprise. De nombreuses entreprises déploient un portail pour proposer à leurs utilisateurs un point d'entrée unique et homogène vers toutes leurs applications. Un portail est donc intéressant en termes de productivité. Il permet aussi de pousser des informations importantes vers les utilisateurs.

On intègre généralement au sein d'un portail :

– des contenus de communication interne : le mot du président, les offres du comité d'entreprise, etc. ;

– des outils collaboratifs : calendriers partagés, espaces de partage de documents, etc. ;

– des indicateurs issus d'applications métiers : ventes de la semaine, audimat dans le cas d'une chaîne de télévision, etc. ;

– plus rarement, les interfaces des applications métiers ou des raccourcis vers ces interfaces.

Un portail se matérialise par une page web composite, regroupant des fragments d'interfaces qui pointent vers les différents applicatifs de l'entreprise.

Dans le cas d'une entreprise utilisatrice d'application cloud, le portail va devoir intégrer les applications internes et celles qui sont hébergées par des opérateurs cloud. Par exemple, sur la figure 11.5, un calendrier pourrait être issu de Google Apps, et des indicateurs de vente issus de Salesforce.

Notons au passage qu'un service de Single Sign On est un prérequis à un portail proposant un bon niveau d'ergonomie. On a vu comment déployer un tel service dans le paragraphe précédent.

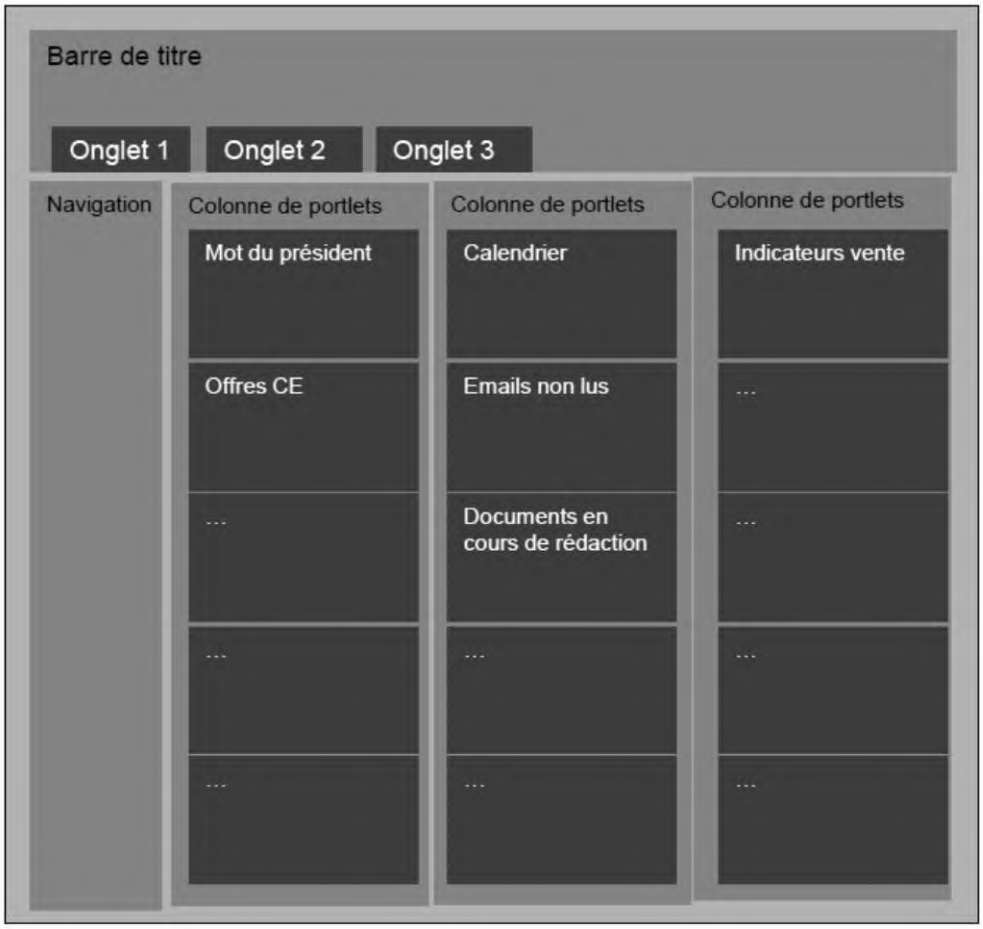

Figure 11.6 — Une interface de portail.

Notons que, pour des entreprises qui ne disposent pas de portail et n'ont pas de projet dans ce sens, la page proposée par le système de fédération d'identité (*cf.* figure 11.5) peut faire office de portail.

11.2.4 Le bus d'intégration

Rappel sur les problématiques d'intégration

Les entreprises font souvent face à des problématiques de partage d'informations entre leurs applications, ce qui les amène à créer des communications interapplicatives. Lorsque ces communications sont gérées au cas par cas, on aboutit à de multiples liens entre applications : c'est le fameux **syndrome du « plat de spaghettis »**. Pour résoudre ce syndrome, il est recommandé de mettre en œuvre un outil unique et dédié à la gestion des échanges d'informations entre applications. C'est l'objectif des EAI (*Enterprise Application Integration*) et de leurs successeurs les ESB (*Enterprise Service Bus*).

Dans le cas d'une entreprise utilisatrice d'applications cloud, la problématique d'intégration va au-delà de son réseau local : il est nécessaire d'intégrer les applications internes avec celles qui sont hébergées par des opérateurs cloud.

Le bus d'intégration devra donc se doter d'une connectivité vers les API cloud, ce qui se révélera assez simple car ces API sont basées sur le protocole HTTP. Il devra aussi être capable de sécuriser les échanges *via* SSL car ces derniers transitent par Internet. Enfin, si des flux de données émis depuis l'extérieur doivent pénétrer au sein du SI d'entreprise, il faudra s'assurer de la provenance de ces flux *via* un protocole d'authentification sûr. Le bus d'intégration devra ainsi gérer des spécificités pour fonctionner en bonne entente avec les plateformes cloud.

Il peut être déployé selon deux scénarios :

- **déploiement dans l'entreprise** sur la base d'un logiciel « software » fourni par un éditeur (IBM, Microsoft, Oracle, etc.) ;
- **usage d'un service cloud**, comme RunMyProcess, Amazon Simple Queue Service (SQS).

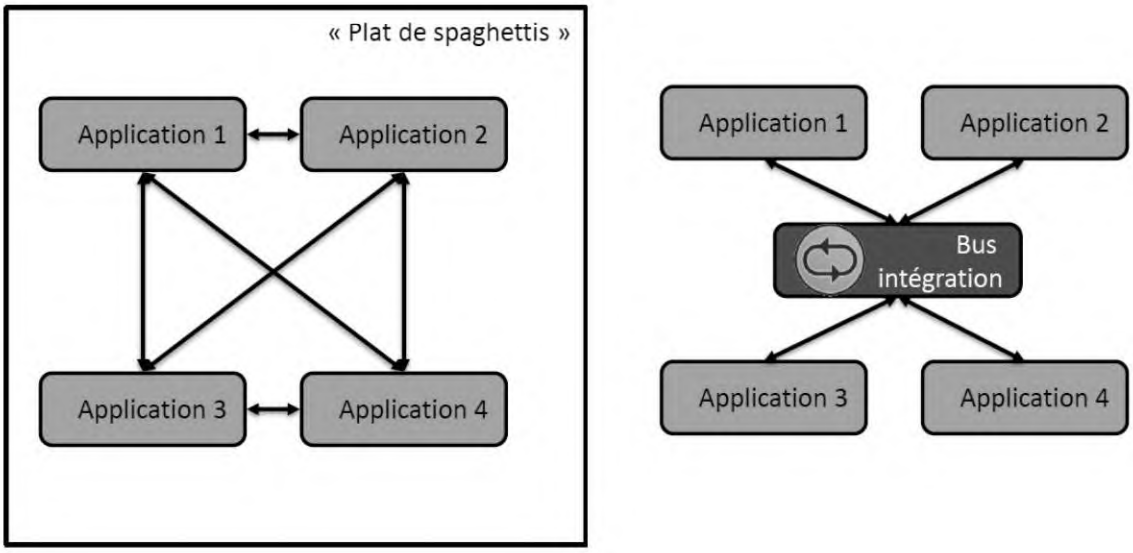

Figure 11.7 — Le bus d'intégration solutionne le syndrome du « plat de spaghettis ».

Les ISB

Les opérateurs de bus d'intégration proposés en mode cloud ont proposé une nouvelle terminologie, qui vient s'ajouter aux EAI et ESB.

Il s'agit de l'ISB, *Internet Service Bus*. Comme son nom l'indique, ce bus a pour fonction de véhiculer des informations au travers de l'Internet.

11.2.5 L'ISB RunMyProcess

L'objet de ce paragraphe est de présenter un ISB, indépendant d'une plateforme cloud particulière. En effet, le service SQS d'Amazon est très lié à ses services, il en est de même pour celui d'Azure, etc.

RunMyProcess est un ISB en mode SaaS proposé par une start-up française. Il propose un mode de facturation proche des PaaS : paiement à la consommation, en l'occurrence en fonction du nombre de messages véhiculés.

RunMyProcess repose sur quatre composants :

- **un outil de modélisation en ligne :** un outil graphique permet de modéliser des processus d'intégration ou d'échange au travers du formalisme BPMN[1]. Ces processus peuvent être modélisés par des profils non techniques, comme des maîtrises d'ouvrage. Il est ensuite possible de décrire les flux d'informations concernés par le processus, *via* des grammaires XML. Enfin, l'outil permet de « câbler » les processus sur les flux d'informations concernés ;
- **une connectivité autour de standards techniques :** SOAP, fichiers XML, HTTP/HTTPS, FTP, MSMQ, MQSeries, SMTP/SMTPS ;
- **un ensemble de connecteurs orientés vers les SaaS :** RunMyProcess propose des connecteurs vers les outils Salesforce, Google Apps, Zoho, Xignite (données financières), SAP by Design, ADP, Netsuite, etc. Ces connecteurs permettent d'accélérer l'intégration des flux d'informations. Le catalogue de connecteurs peut être enrichi par chaque client par simple paramétrage ;
- **un environnement d'exécution** qui permet d'exécuter les différents flux d'informations définis sur la plateforme. Une console de supervision permet de vérifier leur bon déroulement et d'accéder à toutes les données manipulées durant l'exécution ;
- **un portail de suivi et d'analyse** qui permet d'analyser les flux et les informations échangées à des fins de pilotage, de reporting ou d'optimisation. Toutes ces informations sont intégrables dans un *mashup* d'entreprise *via* des flux RSS sécurisés.

1. *Business Process Modeling Notation*.

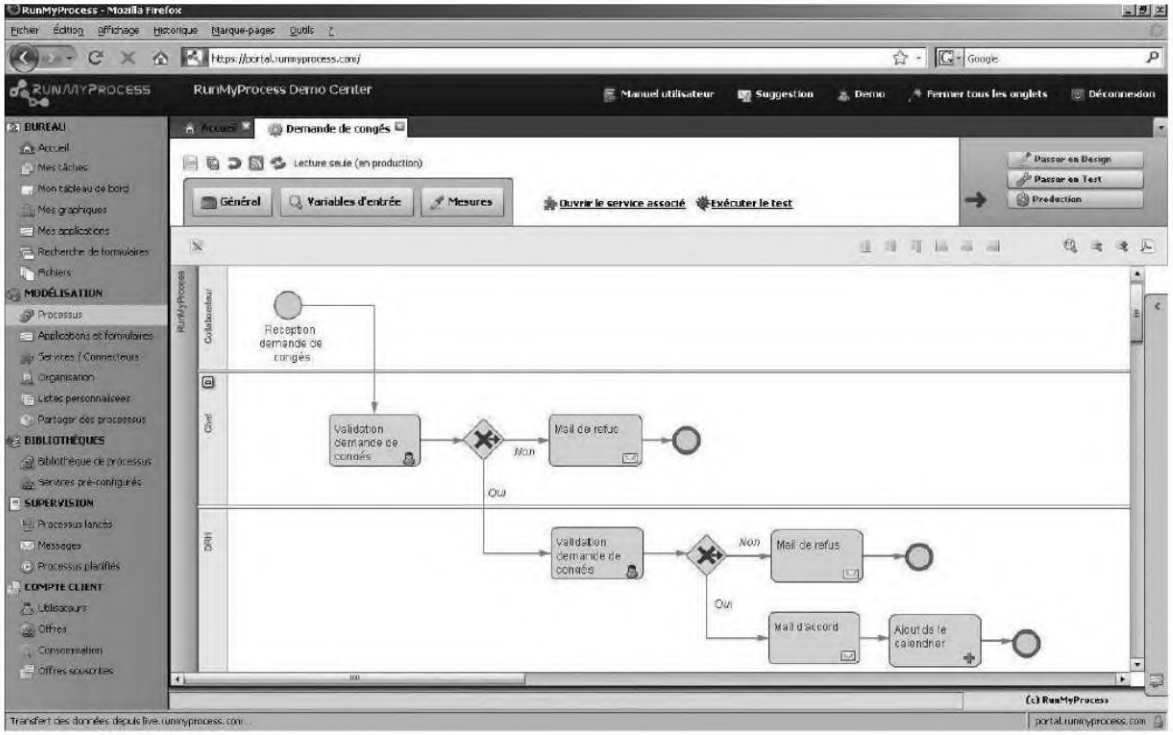

Figure 11.8 — L'interface de modélisation de processus RunMyProcess.

11.3 LA QUESTION DU DÉCISIONNEL DANS LE CLOUD

Les systèmes décisionnels reposent sur un entrepôt de données (*Data WareHouse*) qui collecte des informations issues des diverses applications du SI. En architecture hybride, ces données peuvent être issues d'applications cloud. On va donc se poser la question des modalités de collecte de ces données dans le SI décisionnel :

- Les données peuvent-elles circuler sur Internet sans chiffrement ?
- Leur volume rend-elle la collecte difficile, du fait de la latence réseau d'internet ? C'est problématique dans le cas de gigabits...
- L'entrepôt de donnés est-il à même de collecter des données externes ?

Comme pour les autres aspects de l'intégration, il est possible de recourir à un système décisionnel cloud, comme BIME ou Amazon Redshift.

11.4 LA QUESTION DES RÉFÉRENTIELS DANS LE CLOUD ?

Les référentiels sont les données structurantes et généralement invariantes du système d'information, comme :

- l'annuaire des utilisateurs de l'entreprise ;
- la base des clients ;

- la base de produits de l'entreprise ;
- les méthodes et processus de travail ;
- les bases de connaissance ;
- etc.

Du fait de leur aspect structurant, et souvent stratégique, on peut hésiter à les déplacer dans le cloud. La question d'un référentiel dans le cloud se pose si un pan complet du SI se trouve dans le cloud. Par exemple :

- si tout le SI RH est dans le cloud, on peut envisager de déplacer l'annuaire des utilisateurs ;
- si tout le SI client est dans le cloud, on peut envisager de déplacer la base clients.

Ces arbitrages se font au cas par cas, selon le degré d'hybridation du SI.

On peut cependant noter que Google Marketplace et Salesforce AppExchange proposent des écosystèmes applicatifs assez complets qui rendent possible l'externalisation d'un pan du SI.

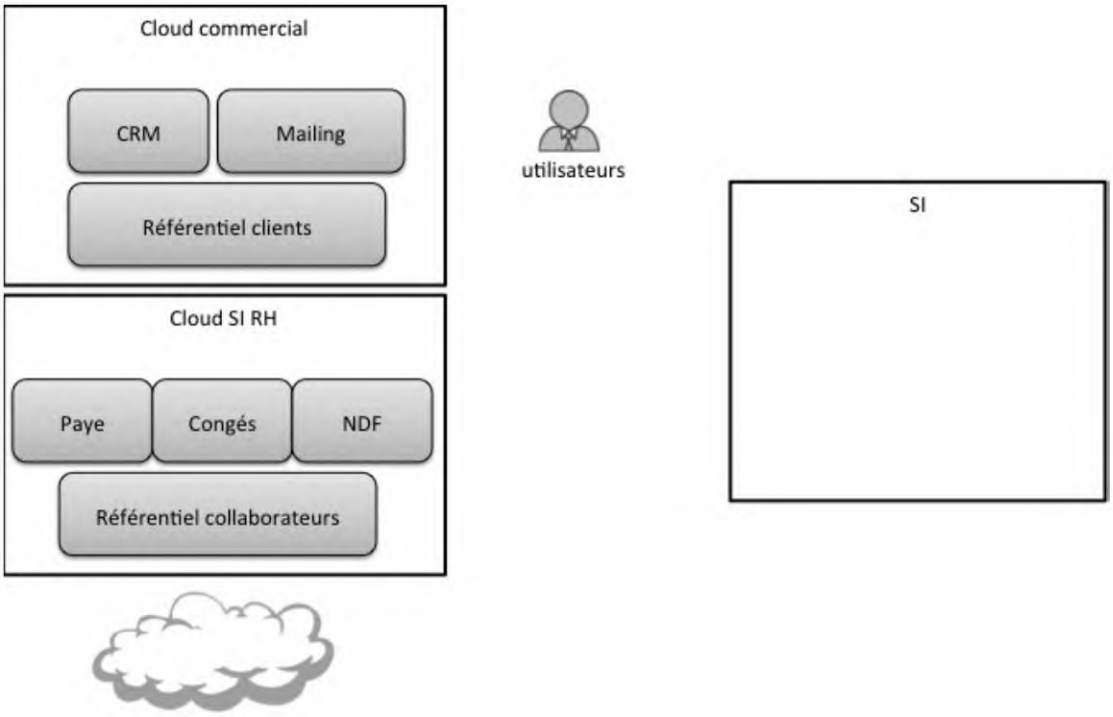

Figure 11.9 — L'externalisation de pans du SI dans le cloud.

11.5 COMMENT ARBITRER SES CHOIX D'HYBRIDATION ?

Plusieurs stratégies peuvent orienter les choix d'architecture hybride. Ce chapitre en présente trois. La pratique montre qu'on choisit généralement un compromis entre ces approches.

11.5.1 Pilotage par la cohérence

Pour l'architecte de système d'information, le maintien de la cohérence du SI est prioritaire. Il est indispensable d'éviter la multiplication des référentiels de même type (annuaires d'authentification par exemple), il faut empêcher les applications de fonctionner en silos, c'est-à-dire sans communication avec le reste du SI. Son objectif est de limiter l'entropie et d'« urbaniser » le SI. Il faut limiter le risque de latence lié à un éclatement géographique du SI dans des clouds sur tous les continents.

En suivant ces principes, l'architecte tentera de minimiser le nombre de clouds utilisés et privilégiera les écosystèmes intégrés.

11.5.2 Pilotage par la sécurité

L'approche sécuritaire (*cf.* chapitre 4) consiste à privilégier les opérateurs cloud qui :

- disposent de certifications ;
- localisent leurs datacenters en France ou en Europe ;
- offrent des garanties sur la vie privée (Safe Harbor) ;
- sont perçus comme pérennes.

11.5.3 Pilotage par les coûts

Le pilotage par les coûts implique de choisir l'architecture la plus économique.

Un expert des architectures cloud peut en effet orienter les choix sachant que les différentes offres du marché facturent ou non :

- le trafic réseau ;
- l'uptime, c'est-à-dire le temps pendant lequel les applications sont disponibles mais inutilisées ;
- le temps de calcul du processeur ;
- l'espace disque ;
- le nombre d'utilisateurs.

Il est ainsi possible de choisir ses plateformes cloud dans une optique d'optimisation des coûts. Des offres d'intermédiaires (*brokers*) commencent à émerger pour aider les entreprises à créer des « architectures orientées coûts ».

En résumé

Ce chapitre a présenté les problématiques liées à la multiplication de l'usage de plateformes cloud :

– cohabitation entre applications internes et externes à l'entreprise ;

– nécessité de déployer un service d'identité, et un service d'intégration, pour bien gérer cette cohabitation ;

– faire les bons choix d'hybridation.

12

Généralisation du cloud et impacts organisationnels

Objectif

Ce chapitre présente une vision dans laquelle le recours au cloud computing est poussé assez loin. Il en décrit les impacts organisationnels.

12.1 CLOUD ET APPLICATIONS CRITIQUES

12.1.1 Applications métier génériques

On a vu précédemment que l'usage du cloud était plutôt privilégié pour l'informatique de commodité. Néanmoins, il existe des secteurs d'activité pour lesquels **des applications métiers sont assez génériques, du fait de la maturité des usages métiers**. Certains opérateurs cloud commencent à proposer des SaaS métiers : c'est le cas de Salesforce ou de Netsuite. Le catalogue de la plateforme AppExchange de Salesforce dispose aujourd'hui de solutions dans les secteurs suivants :

- **Télécommunications :** par exemple, applications pour opérateurs télécoms...
- **Éducation :** application de gestion de planning de cours...
- **Santé :** application de logistique hospitalière...
- **Industrie :** application de logistique d'acheminement...
- **Média :** application de gestion de campagne publicitaire...
- **Organisations non gouvernementales (ONG) :** application de gestion de l'évacuation d'une région...
- **Pharmacie :** application de gestion des visiteurs médicaux...

- **Service :** application de gestion de feuilles de temps...
- Etc.

12.1.2 Développement métiers

Si le niveau de classification des données le permet, il est envisageable d'héberger sur le cloud toute application métier développée en spécifique.

En effet, l'hébergement sur une plateforme IaaS ou PaaS offre de nombreux avantages en termes d'agilité.

12.2 LA POSITION DE LA DIRECTION DE LA SÉCURITÉ

Le directeur de la sûreté joue un rôle critique dans le cas d'externalisation des fonctions critiques. C'est en effet à lui que revient la charge de valider l'externalisation des données vers les différents opérateurs cloud et par conséquent de piloter des études et des audits sur les engagements de confidentialité de ces différents acteurs. Il acquiert un rôle central pour l'entreprise.

12.3 LE CLOUD BROKER

Dès lors que l'on accepte comme possible la multiplication des applications sur le cloud, il est nécessaire de s'équiper d'un outil pour faciliter la démarche. C'est le rôle du cloud *broker*. Le cloud *broker* repose sur une plateforme d'intermédiation entre les utilisateurs du SI hybride et les infrastructures du SI.

Il fournit les services suivants aux utilisateurs :

- un catalogue de service, à la manière d'un AppStore, permettant aux utilisateurs métiers de souscrire aux services dont ils ont besoin, de manière transparente, que ceux-ci soient dans le cloud ou dans les murs de l'entreprise ;

un service de *Single Sign On*, généralement basé sur la fédération d'identité (présentée au chapitre 11) ;

une facturation unifiée des services cloud publics et privés ;

un « lanceur » d'application, ou portail d'accès.

La valeur ajoutée du cloud *broker* est de simplifier la souscription et l'accès aux services cloud, et surtout de créer une couche d'abstraction qui masque le fait qu'un service soit interne ou externe au SI.

On peut voir des exemples de facturation unifiée et *Single Sign On* chez les opérateurs télécoms : le cloud pro Orange (lecloudpro.orange.fr) ou le cloud business store SFR (store.saas.sfrbusinessteam.fr).

Le cloud *broker* propose aussi des couches techniques :

- bus d'intégration entre applications ;
- service d'authentification fédérée ;
- service de monitoring des cloud ;
- services réseaux pour assurer la qualité de service et la sécurisation des flux.

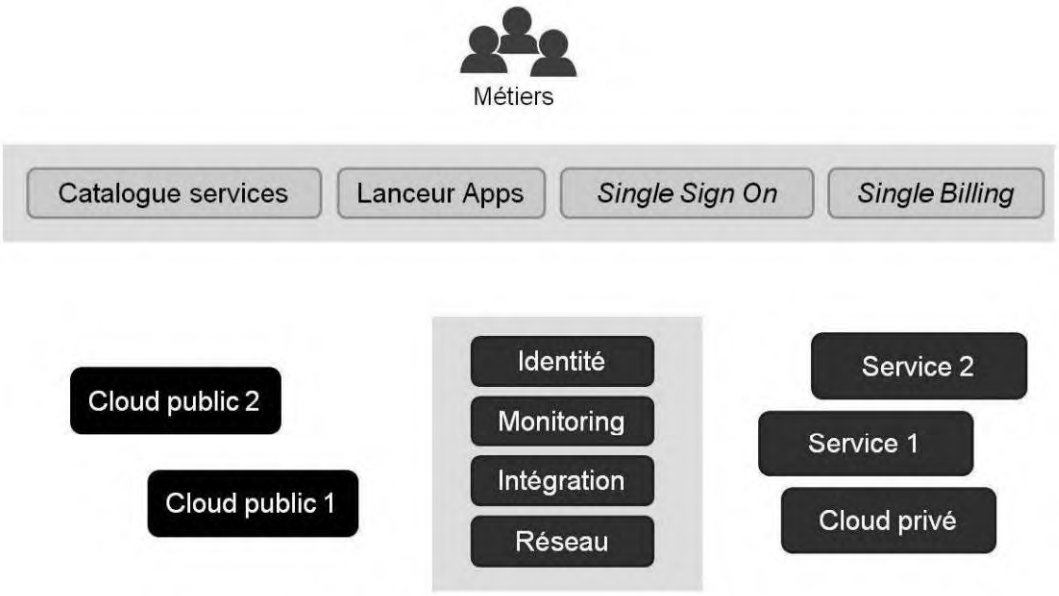

Figure 12.1 — Composants du cloud *broker*.

12.4 LA GOUVERNANCE D'UN PROJET SAAS

12.4.1 La position de la DSI

Dans ce nouveau contexte, le rôle premier de la DSI est d'opérer le cloud *broker* pour les métiers. Elle doit par ailleurs opérer la réversibilité des données en cas de changement d'opérateur cloud.

Elle doit assurer sa responsabilité « régalienne » d'urbanisation et de cohérence du SI, à savoir éviter la multiplication des annuaires d'identité, des référentiels métiers.

Dans le cadre de la gouvernance cloud, la DSI peut prendre plusieurs postures.

La DSI absente

Cette posture n'est pas acceptable selon nous : elle consiste à laisser les métiers utiliser le cloud sans donner le moindre cadre. Elle peut déboucher sur le vol ou la perte de données importantes pour l'entreprise.

La DSI bastion

Cette posture n'est pas non plus acceptable selon nous : c'est celle d'une DSI qui veut choisir tous les opérateurs cloud, qui donne des vetos sur l'usage de services, et donne aux métiers l'impression d'être un frein à l'innovation. Cette DSI sera souvent contournée dans les faits.

La DSI broker

Cette DSI propose de créer un centre de compétence cloud qui sélectionne les opérateurs éligibles et propose un catalogue de services hybrides.

Elle reste le point d'accès unique au cloud pour assurer la cohérence du SI hybride.

La DSI facilitatrice

Cette DSI se contente de proposer ses services : elle aide le métier à maîtriser la consommation d'offres SaaS, et elle veille à la cohérence du SI hybride.

Notre point de vue

Nous pensons que la meilleure posture est celle de la DSI facilitatrice. Il nous semble vain de vouloir être le seul acteur compétent sur le cloud. Nous pensons qu'un centre de compétence cloud a vocation à transférer sa compétence aux autres entités de l'entreprise et à disparaître, de même qu'un centre de compétence e-business ou médias sociaux est voué à se dissoudre, dès que les collaborateurs seront tous sensibilisés.

Nous allons considérer cette posture dans les paragraphes suivants.

12.4.2 Un transfert de compétence cloud

Il nous semble important de sensibiliser les acteurs de l'entreprise aux spécificités du cloud : les métiers, le département juridique, le service achat, en particulier. Cette formation pourra être montée par la DSI. Elle abordera les points évoqués dans la suite de ce paragraphe.

12.4.3 Choix de l'opérateur SaaS

Le choix du service SaaS doit être effectué par les métiers, car ils sont les plus à même de comprendre les fonctionnalités offertes. Ils doivent, selon nous, consulter :

- la direction juridique et la direction de la sécurité pour valider le niveau de protection des données de l'entreprise ;
- le service des achats pour évaluer les risques de dérive des coûts en cas de paiement à l'usage ;
- la DSI pour vérifier l'intégrabilité avec le SaaS et la réversibilité des données.

Il nous semble important de sensibiliser les métiers à la nécessité de cette consultation.

12.4.4 Déploiement

Dans la phase de déploiement, les métiers solliciteront la DSI afin de déployer le cloud *broker*, en particulier la fédération d'identité et le bus d'intégration.

12.4.5 Exploitation

Les métiers peuvent rester autonomes pour gérer leur environnement SaaS.

Support

Il faudra alors les sensibiliser au fait que le service desk[1] ne pourra répondre à leurs questions, car le service SaaS ne fera pas partie de leur périmètre. Ils devront alors s'organiser pour s'entraider en créant une FAQ de manière collective, ou en demandant du support directement à l'opérateur SaaS. Ce dernier risque alors de ne répondre qu'aux questions avancées, leur demandant de se débrouiller seuls pour les questions élémentaires.

Gestion du changement

De nombreux services SaaS mettent en ligne des évolutions au fil de l'eau, sans faire de change management au sens d'ITIL. Les métiers peuvent être étonnés par cette pratique. Dans certains cas, il est possible de retarder une évolution pendant quelques mois (exemple de GoogleApps), mais ce n'est pas toujours le cas. Il faudra les informer sur cet aspect.

Comme les API évoluent au même titre que les interfaces, il peut arriver que les intégrations avec le SI nécessitent une mise à jour. Encore un aspect à côté duquel les métiers peuvent passer.

12.4.6 La fin de service

Dans le cas du changement d'opérateur, la DSI sera sollicitée pour la réversibilité des données, s'assurer qu'aucune trace ne subsiste chez l'opérateur SaaS. Elle devra aussi réimporter les données dans un système tiers.

12.5 LA GOUVERNANCE D'UN PROJET PAAS/IAAS

À propos de DevOps

La communauté « DevOps » nous invite à repenser la frontière classique dans les DSI entre d'un côté les études, c'est-à-dire ceux qui écrivent le code, et de l'autre la production, c'est-à-dire ceux qui déploient et exploitent ces applications.

1. C'est le service qui répond aux utilisateurs en cas de problème dans la norme ITIL.

L'idée de DevOps est de casser cette frontière, afin de travailler en meilleure entente, en partageant les responsabilités sur le produit fini. Les études vont convaincre la production de les laisser mettre en ligne plus souvent, tandis que la production va sensibiliser les études sur les problématiques de stabilité.

Dans la pratique, avec DevOps, la production automatise le déploiement des applications, les études mettent en place le déploiement en continu, et tous partagent des rituels réguliers.

Le cloud computing est un formidable accélérateur pour appliquer les idées de DevOps.

12.5.1 Choix de l'opérateur PaaS/IaaS

Le choix du service PaaS/IaaS doit être effectué en collaboration entre les Devs et les Ops.

Pour ce faire, les Devs devront évaluer l'architecture, les spécificités des plateformes pour les développeurs. Les Ops devront étudier les plateformes en termes de monitoring, de disponibilité, de robustesse, de sécurité. Il est important qu'un consensus se fasse entre les deux équipes. La DSI doit selon nous consulter :

- la direction juridique et la direction de la sécurité pour valider le niveau de protection des données de l'entreprise ;
- le service des achats pour évaluer les risques de dérive des coûts en cas de paiement à l'usage.

12.5.2 Conception de l'architecture

Dans cette phase, les Devs prendront en compte les spécificités architecturales de la plateforme PaaS ou IaaS. Ces dernières peuvent être très fortes comme on le verra dans le chapitre 20. Elles peuvent nécessiter une montée en compétence de l'équipe de développement. En particulier, on pourra s'orienter vers une architecture orientée coûts (*cf.* chapitre 11).

12.5.3 Développement

Le cloud est un accélérateur pour les développements applicatifs. Il permet de disposer en quelques minutes de ses environnements de développement, test, et production. Il en est de même pour la création de l'usine logicielle.

À propos de l'usine logicielle

Les équipes de développement travaillant dans les environnements Java, .NET ou autres utilisent aujourd'hui des « usines logicielles ». Cette terminologie recouvre un ensemble d'outils permettant d'effectuer ces tâches :

– centralisation des sources ;

– compilation ;

– build de l'application ;

– génération de la documentation ;
– tests de non-régression ;
– tests fonctionnels ;
– etc.

L'usine logicielle est ni plus ni moins une plateforme collaborative orientée vers le développement d'applications.

L'usine logicielle est un prérequis dans le cadre du développement piloté par les tests (*Test Driven Development*), pratiques au centre des méthodes agiles.

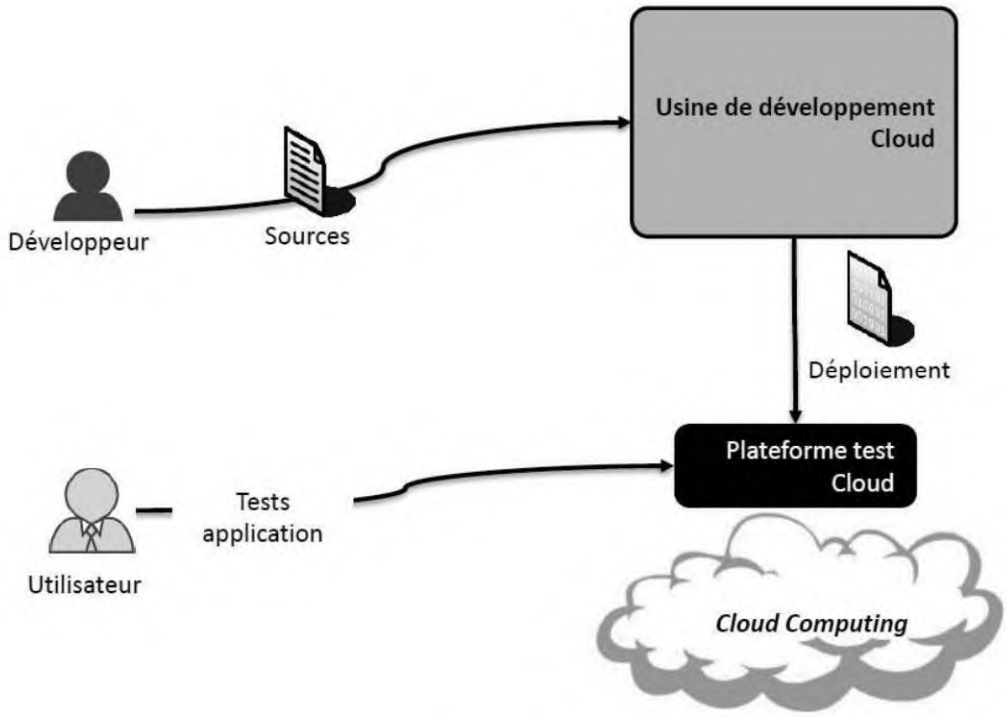

Figure 12.2 — L'usine logicielle dans le cadre du cloud computing.

Il existe plusieurs offres intéressantes pour bâtir une usine logicielle dans le cloud :

- **GitHub et BitBucket** permettent la gestion de source centralisée.
- **snap-ci ou semaphore-ci** permettent l'intégration continue.

Ces outils permettent le déploiement en continu vers des PaaS comme Heroku.

Ainsi le cloud offre un outillage idéal pour la mise en œuvre des méthodes agiles.

12.5.4 Mise en production

Deux pratiques DevOps nécessitent un outillage technique :

- le « Continuous Delivery » est outillé par l'usine logicielle, présentée ci-dessus, mise à disposition par les Devs ;

- l'« infrastructure as code » est outillée par des scripts de déploiement, mis à disposition par les Ops. Ces scripts permettent aux Devs d'effectuer des mises en production après le passage de batteries de tests.

Il existe plusieurs outils pour mettre en place l'« infrastructure as code » : chef et ses fameuses recettes (scripts), Puppet ou Capistrano. **Amazon propose OpsWorks sur sa plateforme : ce service permet de lancer des recettes chef de déploiement applicatif. Il est aussi possible de piloter Google Compute Engine avec chef.**

À nouveau, le cloud fournit des outils sur étagère pour utiliser les pratiques DevOps.

12.5.5 Tests

Le cloud simplifie la création de la plateforme de test en IaaS ou PaaS. Dans le cas du PaaS, les environnements de test/production feront partie des services de base offerts par la plateforme.

12.5.6 Exploitation

La notion d'exploitation est ici très ténue. En effet, l'équipe d'exploitation ne va pas gérer l'application à proprement parler, mais **elle va se former à un nouveau métier : le pilotage d'opérateur cloud.**

Ce nouveau métier va la rapprocher de la direction des études, rompue depuis longtemps au pilotage de SSII[1] pour la réalisation de projets d'intégration. Le modèle cloud va donc conduire l'équipe d'exploitation à cesser de gérer elle-même ses serveurs et la dichotomie usuelle dans les DSI entre études et production va se réduire.

L'équipe d'exploitation assiste la direction des études dans le déploiement d'applications vers les plateformes PaaS, selon les principes de DevOps.

Ainsi, la direction de l'exploitation va se former à la contractualisation et au pilotage d'opérateurs pour la gestion de ses applicatifs : ce pilotage se rapprochera de la relation avec un opérateur télécom ou un prestataire d'hébergement. En collaboration avec le RSSI, elle suivra les engagements de l'opérateur cloud sur les aspects suivants :

- **garantie sur la disponibilité des applications :** engagement à maintenir une disponibilité de 99,9 %, par exemple. Publication d'informations sur les pannes, opérateurs de maintenance, et donc disponibilité réelle ;
- **garantie sur le temps de rétablissement :** engagement à rétablir les applications en cas de plantage ou désastre dans un Datacenter (inondation, incendie, etc.) ;
- **garantie sur les sauvegardes :** engagement à mener des sauvegardes régulières des données, à conserver une copie dans un site distant ;
- **garantie sur le rétablissement des sauvegardes :** capacité à extraire et à rétablir une sauvegarde sur l'infrastructure de production ;

1. Société de service en ingénierie informatique.

- **garantie de support technique :** support technique à l'équipe d'exploitation, au travers d'un système de tickets d'incidents ;
- **garantie de support aux utilisateurs dans le cas SaaS :** support aux utilisateurs, au travers d'un système de tickets d'incidents ;
- **garantie de support aux développeurs dans le cas PaaS :** support aux développeurs, au travers d'un système de tickets d'incidents.

Elle va mener des études et audits sur la capacité technique des plateformes PaaS à assurer le niveau de disponibilité et de robustesse attendu par les utilisateurs.

Rappels sur le monitoring[1]

Le monitoring consiste à collecter des informations pour créer des indicateurs de performance. Des seuils sont définis pour chaque indicateur et les franchissements de seuils peuvent déclencher des événements. On distingue habituellement deux types de monitoring :

– **le monitoring de disponibilité** qui permet de mesurer la disponibilité des applications ;

– **le monitoring des temps de réponse** qui permet de mesurer les temps de réponse aux requêtes des utilisateurs.

Les équipes de production ont la responsabilité de faire le monitoring des applications pour s'assurer que les engagements de service (SLA, *Service Level Agreement*) définis avec les utilisateurs sont respectés. Il faudra donc que les opérateurs cloud s'engagent sur des SLA (*cf.* l'engagement de 99,9 % de disponibilité généralement proposé).

Il faudra aussi que ces derniers leur fournissent un outillage de vérification de la disponibilité du service, c'est-à-dire une console de monitoring cloud ou bien des traces exploitables par la DSI. Certains opérateurs cloud privilégient la seconde option en fournissant des traces au travers d'une API. Les équipes de production seront donc amenées à exploiter ces traces à l'aide d'un outil d'analyse comparable à ceux qui permettent de suivre la fréquentation des sites web.

Pour le monitoring des temps de réponse, la DSI s'appuiera sur une application interne qui simule le comportement des utilisateurs et mesure les temps de réponse. Elle pourra aussi recourir à un opérateur cloud, spécialisé dans la mesure des temps de réponse, par exemple Xiti.

À propos du support des opérateurs cloud

Comme on l'a vu dans le chapitre 5, les opérateurs cloud tendent à automatiser le maximum de tâches pour permettre de réduire leurs coûts d'exploitation. Par conséquent, le dialogue avec l'équipe d'exploitation de l'entreprise est centré sur des consoles web et des échanges d'e-mails pour traiter les incidents. Les échanges par

1. Voir Grojean P., Morel M., Nolin S-P., Plouin G., *Performance des architectures IT*, 2e édition, Dunod, 2011.

téléphone ou lors de rencontres physiques sont généralement réduits aux phases commerciales. Cette relation impersonnelle est liée à un souci d'efficacité : elle peut rebuter certaines DSI habituées à avoir une relation de proximité avec leurs fournisseurs.

L'équipe **reporte ses engagements de disponibilité vers ses prestataires cloud.** Ainsi, son rôle revient à transmettre les exigences des métiers vers les opérateurs cloud, à les contractualiser et à les surveiller. On pourrait ainsi envisager de changer le terme « direction de l'exploitation » pour une terminologie du type « direction de la disponibilité ».

Le cas d'INDUS

INDUS dispose d'un extranet métier, destiné aux échanges avec ses sous-traitants. Cet extranet souffre de problèmes de disponibilité liés à une trop forte charge à certaines périodes de l'année. Par ailleurs, il est développé sur la base d'une technologie dépassée, pour laquelle il n'existe plus de compétence sur le marché.

Les fonctionnalités de cet extranet sont assez simples. INDUS a donc décidé de le redévelopper sur des technologies actuelles et de faire ce développement sur une plateforme PaaS afin de disposer d'une forte capacité à montée en charge.

INDUS a par ailleurs développé un site web présentant son activité dont l'hébergement a été confié à un opérateur IaaS.

Enfin INDUS a décidé de se munir d'un « bac à sable innovation » pour incuber les idées de ses équipes métiers.

La figure 12.3 présente l'évolution du SI d'INDUS. Cette figure est complémentaire de la figure 11.1, qui portait sur les progiciels.

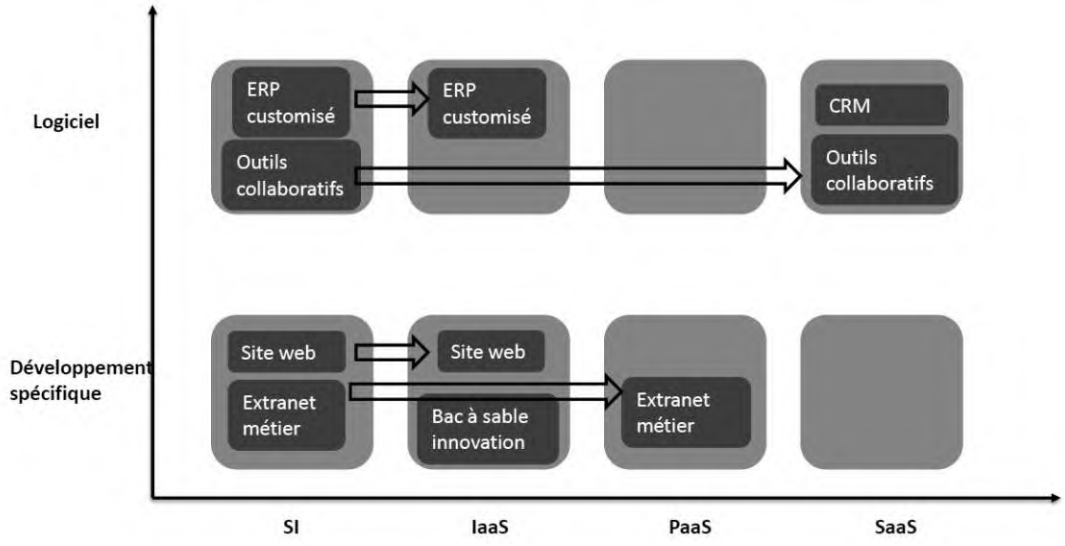

Figure 12.3 — Externalisation des développements spécifiques.

En résumé

Ce chapitre a présenté une vision d'une entreprise qui industrialise le recours au cloud computing pour son informatique.

Cette vision laisse entrevoir une organisation nouvelle pour la DSI.

QUATRIÈME PARTIE

Les services SaaS disponibles

Cette partie commence par évoquer les attitudes des différents acteurs informatiques vis-à-vis du cloud computing.

Elle propose un panorama des offres SaaS, ou progiciels en ligne prêts à l'emploi, classées suivant les catégories :

- services de collaboration ;
- services FrontOffice ;
- services BackOffice.

Ce panorama n'est pas exhaustif : nous avons choisi de présenter les offres qui nous paraissent les plus intéressantes.

Cette partie propose enfin une vision prospective : celle d'un cloud *desktop* constitué uniquement d'applications SaaS.

13

Le positionnement des grands acteurs de l'IT

Objectif

Ce chapitre présente les attitudes des acteurs du monde informatique vis-à-vis du cloud computing. Il présente en particulier les stratégies des éditeurs de logiciels traditionnels, des constructeurs, des opérateurs télécom et des géants du web.

13.1 LES ACTEURS HISTORIQUES

La plupart des grands acteurs du monde informatique voient le cloud computing comme un nouveau modèle incontournable, à développer en priorité dans leur offre. Ce développement prend plusieurs formes suivant leur positionnement dans le mode informatique. De par leur métier, certains tendent à devenir opérateurs SaaS, d'autres proposent des IaaS ou PaaS afin d'héberger des applications, d'autres enfin proposent un outillage pour développer des SaaS, PaaS, IaaS.

Les acteurs qui disposent d'une offre logicielle et matérielle (Sun/Oracle, HP, IBM) ont une opportunité à saisir : proposer un cloud clef en main à leurs clients. En effet, ils sont capables de maîtriser une infrastructure du sol au plafond comme Google ou Amazon.

13.1.1 Les opérateurs télécom

Positionnement vis-à-vis du SaaS

Les opérateurs télécom proposent des services SaaS de collaboration. Ils couvrent en général les fonctionnalités suivantes :

- une messagerie e-mail basée sur une adresse du type NuméroMobile @operateur.fr ;
- parfois une messagerie unifiée, c'est-à-dire intégration de la boîte vocale et de la messagerie e-mail ;
- un service d'agenda et de carnet d'adresses synchronisé avec le téléphone mobile ;
- un stockage en ligne de documents et de médias.

Ces fonctionnalités sont proposées par ailleurs par d'autres opérateurs SaaS spécialisés (*cf.* chapitre 14). Cette concurrence peut les desservir car il leur est difficile d'être meilleurs que les spécialistes dans chaque gamme de service.

Les opérateurs télécom proposent souvent des portails SaaS qui permettent à leurs clients d'utiliser des services opérés par eux-mêmes ou par des tiers. L'opérateur se charge de facturer l'ensemble de ces services *via* une facture unifiée. On peut citer en exemple « le Cloud Pro » d'Orange (http://lecloudpro.orange.fr).

Ces offres sont intéressantes pour les PME et TPE qui connaissent mal les offres américaines.

Positionnement vis-à-vis de l'IaaS

Les opérateurs télécom proposent aussi des offres IaaS, souvent basées sur le socle VMware (*cf.* chapitre 21).

Ils ont un atout différenciant : ils sont capables de proposer un cloud public accessible au travers d'un réseau privé virtuel, opéré par leurs soins, et par là de réduire les craintes de leurs clients vis-à-vis de l'exposition de leurs données et de la latence d'Internet.

13.1.2 Les clouds souverains CloudWatt & Numergy

Une des originalités de l'État français est d'avoir souhaité financer des clouds publics. Ainsi deux offres cloud sont cofinancées par l'État :

- CloudWatt, en partenariat avec Orange et Thalès ;
- Numergy, en partenariat avec SFR et Bull.

Ces clouds dits « souverains » fournissent des services IaaS pour permettre aux acteurs français de l'informatique de s'en servir comme socle technique. Ils n'ont pas démontré leur pertinence, et sont fortement remis en question.

13.1.3 Les hébergeurs

Le métier d'hébergeur mutualisé[1] évoluant naturellement vers celui d'opérateur cloud, on voit un certain nombre d'opérateurs proposer des offres IaaS.

On peut citer par exemple Atos, Claranet, LinkByNet, etc. L'acteur qui sort du lot dans cette catégorie est OVH, avec ses 220 000 serveurs (fin 2015). Un certain nombre d'entre eux a choisi l'offre VMware pour accélérer le déploiement de leur offre cloud. Certains n'offrent ni le *Pay As You Go*, ni le *Self Service*.

13.1.4 Les constructeurs

La plupart des constructeurs proposent aujourd'hui d'utiliser leurs solutions matérielles (serveurs, stockage, etc.) pour bâtir un cloud public ou privé. Ils complètent généralement cette offre en proposant du conseil sur la mise en œuvre d'un cloud.

Sun (Oracle)

Avant son rachat par Oracle, Sun disposait d'un outillage pour bâtir une PaaS, une plateforme de cloud computing. Cette offre intitulée « Sun Grid Engine » avait pour objectif de fournir une solution de création de Datacenter, basée sur de vastes grappes[2] de serveurs. Cette offre constituait le socle du Project Caroline, la plateforme PaaS de Sun.

Notons aussi que Sun proposait, au travers du projet BlackBox, des containers clefs en main contant des milliers de serveurs. Ces conteneurs étaient destinés aux entreprises qui souhaitent déployer rapidement un Datacenter de grande dimension.

Ces différentes offres ont intégré le catalogue Oracle.

HP Helion Cloud

HP a sorti en 2012 une offre IaaS publique basée sur la technologie OpenStack (*cf.* chapitre 21). L'offre s'est enrichie d'un service PaaS à la suite du rachat de la solution Open Source Eucalyptus. Mais, HP s'est retiré du cloud public fin 2015.

HP s'est aussi positionné sur l'Open Hardware, en se joignant à l'Open Compute Project pour créer sa gamme de serveurs Cloudline.

Dell

Dell a abandonné le marché du cloud public. Le constructeur propose aujourd'hui des clouds privés clef en main sur la base des technologies OpenStack.

1. Les offres d'hébergement mutualisé reposent sur des plateformes partagées entre plusieurs clients. Les hébergeurs ont généralement aussi une offre de serveurs dédiés : cette offre pourrait évoluer vers un service de cloud privé externalisé.
2. En anglais, on emploie le terme *cluster*.

IBM

IBM s'est positionné très tardivement sur le cloud. Le pionnier de l'informatique propose aujourd'hui une offre IaaS, compatible OpenStack, issue du rachat de SoftLayer en 2013. Son offre PaaS Bluemix est sortie seulement en 2014. Bluemix local est la déclinaison cloud privée de cette offre.

IBM propose aussi des déclinaisons PaaS de ses solutions de collaboration.

13.1.5 Les éditeurs de logiciels « software »

Selon les scénarios, les éditeurs peuvent se positionner sur :

- le portage de leurs offres « software » sur une plateforme SaaS opérée par leurs soins ;
- l'usage de leur offre de logiciels d'infrastructure pour créer des clouds publics (IaaS ou PaaS), opérés par eux-mêmes ;
- l'usage de leur offre de logiciels d'infrastructure pour créer des clouds publics (IaaS ou PaaS), opérés par des tiers ;
- l'usage de leur offre de logiciels d'infrastructure pour créer des clouds privés (IaaS ou PaaS), opérés par les entreprises utilisatrices.

VMware

VMware est le leader incontesté de la virtualisation dans le monde. L'éditeur propose en particulier vcloud, une solution logicielle permettant de bâtir une plateforme IaaS (*cf.* chapitre 21). La solution permet de mettre en œuvre le *Pay As You Go* et le *Self Service* dans le cadre de clouds publics et privés. Elle est en particulier adoptée par de nombreux hébergeurs qui souhaitent proposer des offres de cloud public de type IaaS.

Depuis le rachat de SpringSource en 2009, VMware dispose d'une plateforme d'exécution Java qui peut servir à bâtir une plateforme PaaS. Cette plateforme a été renommée Pivotal.

Oracle

Oracle a depuis longtemps décliné une partie de ses solutions progicielles en mode « outsourcing chez l'éditeur » : il s'agit de l'offre « **Oracle on Demand** » :

- Oracle On Demand for Siebel CRM ;
- People Soft Enterprise On Demand ;
- Oracle E-Business Suite On Demand ;
- Oracle Hyperion On Demand ;
- etc.

Le modèle « outsourcing chez l'éditeur » n'est pas tout à fait du SaaS, car il ne propose pas de *Self Service*.

Larry Ellison, PDG d'Oracle, a longuement décrié le cloud computing qu'il considérait comme un buzz sans fondement, pour finalement annoncer une offre cloud Oracle. La plupart des produits Oracle sont depuis peu déclinés en mode cloud.

SAP

SAP est schizophrène dans son approche du cloud : le modèle remet en cause son mode de fonctionnement.

SAP a lancé en 2010 une version simplifiée de sa suite en mode SaaS, intitulée « *SAP Business by Design* ». Cette offre a été un échec. Mais SAP a relancé son offre cloud fin 2014.

Adobe

Depuis 2013, Adobe a totalement revu son offre en mode cloud. Ses outils sont vendus selon un modèle d'abonnement de type SaaS, même si les logiciels sont toujours installés localement sur les postes de travail :

- la solution de gestion du cycle de vie des documents LiveCycle est disponible en mode SaaS ;
- le Creative Cloud permet aux graphistes de télécharger leurs logiciels et de stocker leurs travaux dans les Datacenters Adobe (*cf.* chapitre 14) ;
- le marketing cloud propose des outils de ciblage publicitaire et de suivi des campagnes en mode SaaS.

Microsoft

Nous avons positionné Microsoft à la charnière entre les paragraphes « acteurs historiques » et « géants du web ». En effet, Microsoft appartient à ces deux mondes :

- en tant qu'éditeur, il propose depuis des années des logiciels « classiques » qui migrent peu à peu en mode SaaS, PaaS, IaaS ;
- en tant qu'opérateur historique de MSN Messenger ou Hotmail, Microsoft a depuis longtemps des pratiques et des infrastructures comparables à celles de Google.

Positionnement vis-à-vis du SaaS

Comme Adobe, Microsoft a adopté pour Office un modèle d'abonnement qui rappelle le SaaS, en proposant l'installation des logiciels sur le poste de travail, avec un espace de stockage cloud : One Drive.

Ce modèle connaît le succès : Office est un outil incontournable dans le monde de l'entreprise. Et Microsoft a fait un gros effort pour porter Office sur iOS, Android, sur le web.

Dans le passé, le géant de Redmond a une approche un peu schizophrène du modèle cloud. Par exemple, un logiciel comme Exchange n'était pas conçu pour le cloud, ni pour gérer plusieurs entreprises utilisatrices. A contrario, des offres comme Google

Apps ou Salesforce sont nativement « multi-tenant », capables de gérer *N* utilisateurs dans M entreprises.

Microsoft proposait donc Exchange en mode hébergé dans ses Datacenters. Il a fallu plusieurs années de refactoring pour aboutir à Office365 qui est aujourd'hui une vraie plateforme cloud.

Positionnement vis-à-vis du PaaS/IaaS

Microsoft a beaucoup investi dans le PaaS et l'IaaS au travers de son offre **Azure**. Cette offre est très complète : elle fait de Microsoft le second acteur du cloud, juste derrière Amazon Web Services. On y reviendra au chapitre 20.

Positionnement vis-à-vis du cloud privé

Microsoft propose de bâtir des clouds privés au travers de son offre de virtualisation HyperV & System Center (*cf.* chapitre 21).

13.2 LES GÉANTS DU WEB

Les acteurs issus du web sont naturellement intéressés par le modèle cloud, qui représente la continuité de leur modèle grand public, les mêmes technologies et services étant déclinés pour les entreprises. En revanche, ils ont parfois du mal à adapter un modèle gratuit, basé sur la publicité, au monde de l'entreprise. Enfin, ils peuvent manquer d'expérience dans les relations avec les entreprises.

13.2.1 Google

La société Google a été créée autour d'un moteur de recherche gratuit. Au démarrage, Google n'avait donc pas de modèle économique. Puis, l'invention des « liens sponsorisés », des publicités ciblées sans être intrusives, a permis à la société de devenir un des acteurs les plus rentables du monde informatique.

Google offre essentiellement des services gratuits rentabilisés par la publicité. Seules **Google Geospatial Solutions** et **Google Apps**, offres SaaS destinées aux entreprises, sortent de son modèle économique habituel. L'offre s'est bâtie très progressivement en rachetant de très nombreuses start-up. Sans être exhaustif, on peut citer quelques solutions intégrées dans les offres entreprises du géant de Mountain View :

- Writely (racheté en mars 2006) est à la base du traitement de texte de Google Apps ;
- 2Web Technologies (racheté en juin 2006) est à la base du tableur de Google Apps ;
- Zenter (racheté en juin 2007) et Tonic Systems (racheté en avril 2007) sont à la base de l'outil de présentation de Google Apps ;
- Postini (racheté en juillet 2007) et Greenborder (racheté en mai 2007) permettent la protection antivirus et anti-spam des e-mails dans Google Apps ;

- YouTube (racheté en octobre 2006) et Omnisio (racheté en juillet 2008) sont à la base de l'intranet vidéo intégré à Google Apps ;
- Grand Central (racheté en juillet 2007) devrait permettre de gérer des lignes téléphoniques dans Google Apps ;
- Marratech (racheté en avril 2007) permet la visioconférence dans Google Apps ;
- Deja News (racheté en février 2001) est à la base Google Groups.

Avec cette offre SaaS, Google a pour ambition de détrôner les suites collaboratives historiques Microsoft Exchange et IBM Lotus. Les fonctionnalités de Google Apps évoluent progressivement selon le principe de la « bêta perpétuelle » évoqué au chapitre 3 et sont proches de celles des offres Microsoft et IBM.

Le spécialiste de la recherche propose aussi une plateforme IaaS/PaaS avec **Google Cloud Platform**. Nous reviendrons sur cette plateforme dans le chapitre 20.

Google dispose à ce jour du plus grand parc de Datacenters connus : ses capacités de stockage et de traitement sont gigantesques. Rappelons, par exemple, que :

- son algorithme de recherche nécessite l'aspiration et le stockage de l'ensemble du web ;
- avec Book Search, Google s'est lancé dans un projet pharaonique et très controversé de numérisation de tous les ouvrages de toutes les grandes bibliothèques du monde ;
- avec Street View, Google s'est lancé dans un projet pharaonique, lui aussi controversé, de photographie de toutes les rues des grandes villes ;
- Google Translate, son système de traduction, utilise des statistiques à grande échelle pour proposer la meilleure interprétation possible.

Google est donc un acteur totalement incontournable du cloud computing.

13.2.2 Amazon

Amazon a été la première société à proposer une plateforme de cloud computing avec **Amazon Web Services**. Il s'agit d'une offre de mise à disposition de ressources de traitement et de stockage.

De nombreuses start-up utilisent aujourd'hui la plateforme Amazon pour héberger leurs services : on peut citer par exemple slideshare.net, une solution de présentation de diapositives en ligne. Amazon est le premier acteur du cloud, fin 2015.

En résumé

Ce chapitre a présenté les stratégies des grands acteurs de l'informatique et de l'Internet : tous semblent vouloir aller vers le cloud computing.

Leurs approches sont très variées : certains opèrent une plateforme SaaS, PaaS ou IaaS, d'autres fournissent des outils et du conseil pour bâtir un cloud public ou privé.

14

Les services de collaboration

Objectif

Les services de collaboration sont les offres SaaS les plus développées aujourd'hui. Ce sont en effet des fonctions typiques de l'informatique de commodité : il y a peu de valeur ajoutée à les développer ou à les exploiter en interne.

Ces services manipulent généralement des données de fonctionnement, peu critiques, plutôt que des données métiers, plus sensibles. Leur ouverture facilite la collaboration hors de l'entreprise.

Ce sont souvent les premiers services SaaS adoptés par les entreprises.

14.1 LES CONCEPTS DE LA COLLABORATION UNIFIÉE

Les fournisseurs de solution de collaboration proposent aujourd'hui des solutions intégrées ou unifiées qui offrent en général des services de **communication synchrone**, de **communication asynchrone** et de **partage d'informations unifié**.

Un outil vient en support à cette collaboration d'information unifié : la recherche transverse permet de retrouver un contenu sur l'ensemble des supports.

14.1.1 La communication synchrone

On entend par service de **communication synchrone** un système qui offre diverses méthodes pour échanger ou débattre à distance avec des collègues.

Les fonctions offertes par ces outils sont généralement :

- la messagerie instantanée ;
- la téléphonie sur IP ;
- la webconférence ;
- le partage d'applications ou du bureau du poste de travail ;
- la coédition de documents ;
- le tableau blanc, qui permet de faire des schémas visibles instantanément par ses interlocuteurs ;
- le transfert instantané de fichier.

À propos des solutions de conférence vidéo

Nous entendons par **webconférence** un échange vidéo basé sur une interface web. Elle ne nécessite ni installation d'application lourde, ni réseau télécom spécifique. Elle propose souvent un système de partage de document ou d'application.

Par opposition, nous entendons par **visioconférence** un échange vidéo *via* une salle spécialement équipée avec caméra et rétroprojecteur. Ce type de système permet d'organiser une réunion entre des personnes situées dans deux salles distantes. Chaque salle dispose d'une caméra panoramique et d'un écran unique. Cet échange fait appel à des infrastructures spécifiques proposées par des équipementiers télécom.

Enfin, nous entendons par **téléprésence** un échange vidéo qui donne aux utilisateurs l'impression d'être dans la même pièce. Il s'agit de salles équipées de grands écrans plats et d'un dispositif sonore digne d'une salle de cinéma. Chacun y est présenté en grandeur nature. Ainsi, chaque utilisateur a l'impression d'être face à ses interlocuteurs. Un bâillement ou un manque d'intérêt pour la réunion est immédiatement remarqué par les autres. Cette solution très luxueuse empêche le syndrome de la rédaction d'e-mails pendant la réunion. Souvent issue de l'offre d'un équipementier télécom, elle est réservée à des grandes entreprises.

Seule la webconférence utilise une interface RIA et entre dans notre définition des SaaS.

Deux outils viennent en support à cette communication synchrone : **l'agenda partagé** permet d'organiser une séance de travail à distance ; **l'indicateur de présence** permet de connaître l'état de disponibilité de son interlocuteur afin de déclencher un échange à l'improviste sans le déranger. L'indicateur de présence est une version avancée de la « sonnerie occupée » des téléphones classiques.

14.1.2 La communication asynchrone

On entend par service de **communication asynchrone ou messagerie unifiée** un système qui offre diverses méthodes pour laisser des messages à des collègues.

La messagerie unifiée est constituée d'une boîte de réception collectant : les e-mails, les SMS et les messages vocaux.

Figure 14.1 — Google Hangout, un exemple d'outil de communication synchrone.

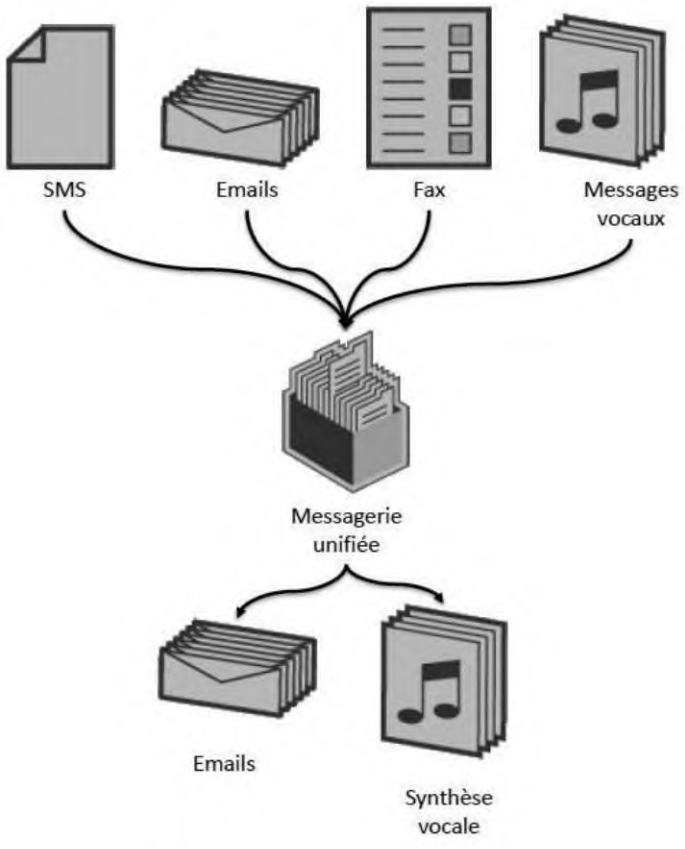

Figure 14.2 — Fonctionnement de la messagerie unifiée.

Ces contenus peuvent être consultés sous forme textuelle ou orale : des conversions entre ces modes sont parfois possibles grâce à la reconnaissance vocale et à la synthèse vocale.

Le recours à la communication asynchrone peut être décidé à la suite de la consultation de **l'agenda** ou de **l'indicateur de présence** de son interlocuteur, s'il est indisponible.

14.1.3 Le partage d'information

On entend par service de partage d'information, un système qui permet de collaborer de manière cohérente sur :

- des documents bureautiques, avec des fonctions de GED (gestion électronique de documents) ;
- des contenus web, avec des fonctions de CMS (*Content Management Solution* ou gestion de contenu web) ;
- des espaces de communication sur des projets sous la forme de blogs ;
- des espaces de capitalisation autour des compétences métiers, sous la forme de wikis ;
- des messages reçus issus des communications synchrone et asynchrone.

Ces outils proposent des fonctions de stockage, mais parfois aussi d'archivage. L'archivage consiste à préserver l'intégrité et la lisibilité des documents afin d'assurer une possibilité de restitution à longue échéance. Il s'agit de lutter contre l'usure et la dégradation des supports documentaires, de s'assurer que les formats utilisés ne sont pas frappés d'obsolescence et restent lisibles.

L'archivage peut être délégué à un prestataire : le tiers archiveur. C'est alors lui qui fournira des API pour permettre l'accès aux documents.

14.1.4 L'édition de documents

Les suites de collaboration intègrent aussi des solutions bureautiques. Elles permettent l'édition de documents de type feuille de calcul, présentation, texte. Certaines permettent la coédition en temps réel. D'autres outils, plus spécialisés, sont destinés à l'édition des documents graphiques : plans, schémas, carte heuristique ou MindMapping, dessin, etc.

14.1.5 Les outils de gestion de tâche et prise de note

Les outils de prise de note et de gestion de tâche sont très utiles pour compenser les défaillances de notre mémoire. Ils permettent de nous décharger l'esprit, de nous débarrasser de la crainte d'oublier quelque chose.

Les fonctions classiques attendues d'un outil de prise de note sont : saisie de texte, ajout de photo ou de note vocale, classement des notes. Celles attendues d'un outil de

gestion de tâche sont : saisie et classement des tâches, configuration d'une date de fin, priorisation, création de rappels.

14.1.6 Les réseaux sociaux d'entreprise

Les suites de collaboration classiques sont essentiellement centrées sur l'édition et l'échange de documents et d'informations : c'est pourquoi, nous les appelons ici « document centric ».

D'autres solutions émergent, des solutions centrées sur les collaborateurs et leurs compétences, afin de créer des communautés d'experts.

Elles sont dérivées des réseaux sociaux, proposent un graphe social (réseau de relation), permettant aux collaborateurs de se regrouper autour d'affinités, de centres d'intérêts communs, comme dans Facebook. L'émergence de communautés éventuellement multisites permet à l'entreprise d'identifier ses compétences et de faciliter les échanges entre experts. Nous évoquerons quelques-unes de ces solutions.

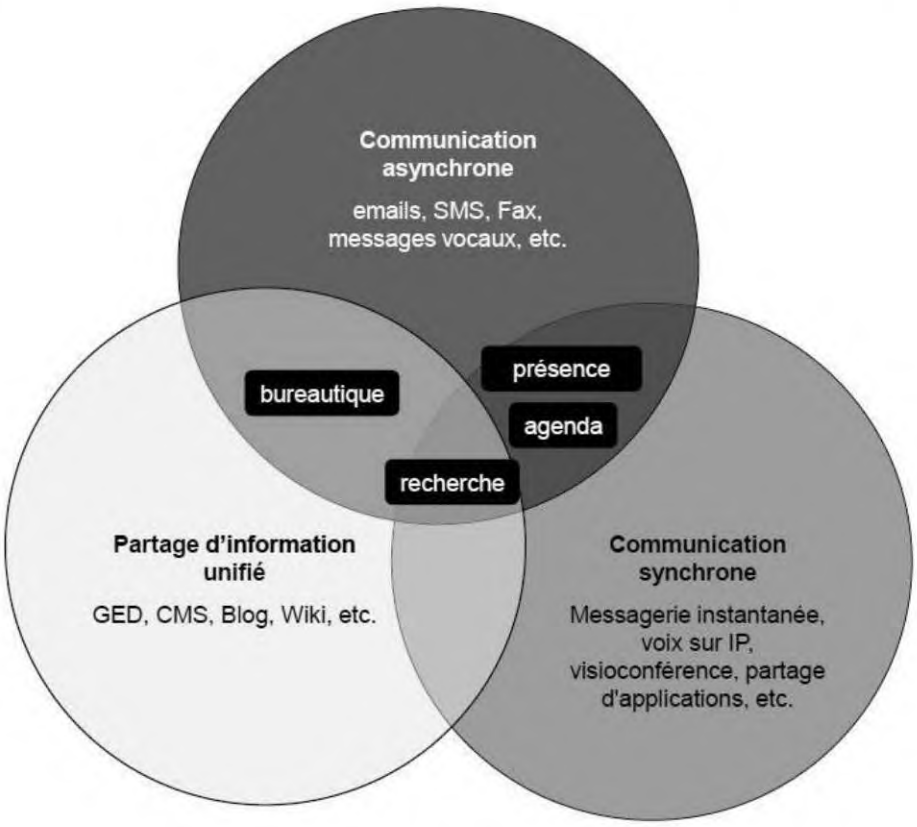

Figure 14.3 — La communication unifiée.

14.1.7 La pertinence des SaaS

Les outils SaaS sont particulièrement intéressants pour la mise en œuvre de communication unifiée, à plusieurs titres :

- La **coédition de documents** est possible, du fait que les données sont centralisées sur une plateforme cloud. Par exemple, Google Sheets permet à deux utilisateurs de modifier de manière simultanée deux cases d'une feuille de calcul.
- Le ***versioning* des documents** est géré de manière automatique, en tâche de fond, du fait que les données sont centralisées sur une plateforme *cloud* disposant de grandes capacités de stockage. L'accès direct aux documents supprime les risques de désynchronisation entre deux documents modifiés par deux personnes différentes.
- **Des fonctions de communication synchrone** sont disponibles lors de la consultation des documents.
- **Les services de collaboration** sont accessibles dans tous les contextes (bureau, domicile, cybercafé, aéroport ou gare, etc.).
- **La communication synchrone** est possible avec des personnes qui n'appartiennent pas à l'entreprise (clients, partenaires).
- **Le partage de documents** est également possible avec des personnes qui n'appartiennent pas à l'entreprise (clients, partenaires).
- **Les outils de collaboration** sont disponibles sur mobile (iPhone/iPad, Android, etc.).
- **L'intégrité des données** : on a vu précédemment que les opérateurs SaaS disposent de Datacenters redondants, offrant de bonnes garanties sur l'intégrité des documents en cas de sinistre. De plus, ils proposent des API qui facilitent l'accès aux documents depuis d'autres applications.
- **L'archivage** : certains opérateurs SaaS se sont spécialisés dans l'archivage et sont devenus tiers archiveurs.

Les fonctions de communication unifiée peuvent être intégrées avec d'autres applications grâce aux API fournies par les opérateurs SaaS (exemple : intégration Google Apps/Salesforce, évoquée dans le chapitre 2).

À propos des outils de présentation

Les outils de présentation ont une spécificité dans le domaine de la bureautique : ils s'apparentent à des outils d'infographie. Ils sont donc beaucoup plus exigeants que les traitements de texte ou tableurs en matière graphique. Ainsi, pour certains, il n'existe aucun outil de présentation satisfaisant dans le Cloud.

14.2 LES OUTILS DE COLLABORATION SAAS

14.2.1 Google Apps

L'offre de collaboration de Google couvre une large partie des concepts de la communication unifiée. Elle est en constante évolution. L'offre de Google se distingue par l'aspect épuré de ses interfaces, et par la mise à disposition régulière de nouvelles fonctionnalités, selon le principe de la bêta perpétuelle. Elle intègre aussi les labs, des fonctionnalités qui sont proposées et soumises à l'approbation des utilisateurs, par mesure de leur adoption.

Nous allons décrire Google Apps de manière assez détaillée car c'est l'offre SaaS de collaboration la plus complète : elle fait référence.

Google Apps permet de déléguer le nom de domaine de son entreprise et donc de recevoir directement sur la plateforme Google les e-mails en `prenom.nom@entreprise.com`. Il est aussi possible de personnaliser les adresses de services Google avec le nom de domaine de l'entreprise (exemples : `http://mail.entreprise.com` pour accéder à la messagerie, `http://calendrier.entreprise.com` pour accéder aux agendas, etc.).

Gmail

Gmail est l'outil de **communication asynchrone** de Google. C'est une messagerie développée en HTML5. L'application offre donc un bon niveau d'ergonomie au sein d'un navigateur. Elle peut aussi être accessible depuis un logiciel de messagerie classique comme Microsoft Outlook ou Mozilla Thunderbird. Elle offre des fonctions classiques de classement, carnet d'adresses, signature de mail, message d'absence de bureau, filtres automatiques, etc. et une intégration forte avec la communication synchrone.

Elle dispose d'un filtre anti-spam très efficace grâce à un système d'intelligence artificielle qui tient compte des signalements de spam de tous les utilisateurs.

Le classement de Gmail privilégie les mots clefs (libellés) plutôt que les répertoires pour classer les messages. L'avantage est qu'on peut affecter plusieurs libellés à un même message, tandis qu'on ne peut pas mettre un même message dans plusieurs répertoires. C'est parfois perturbant au démarrage et cet aspect peut nécessiter un **accompagnement au changement**.

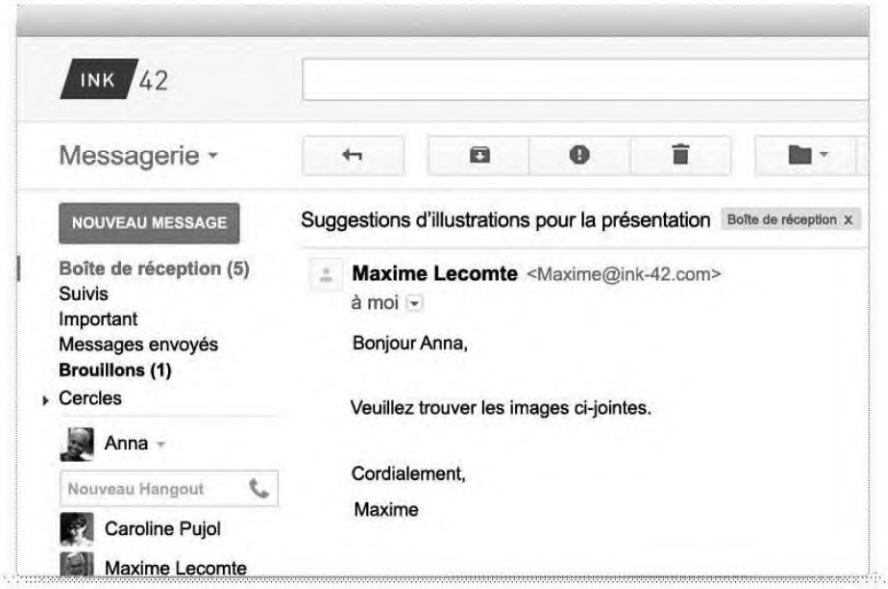

Figure 14.4 — L'interface de Gmail.

Google Agenda

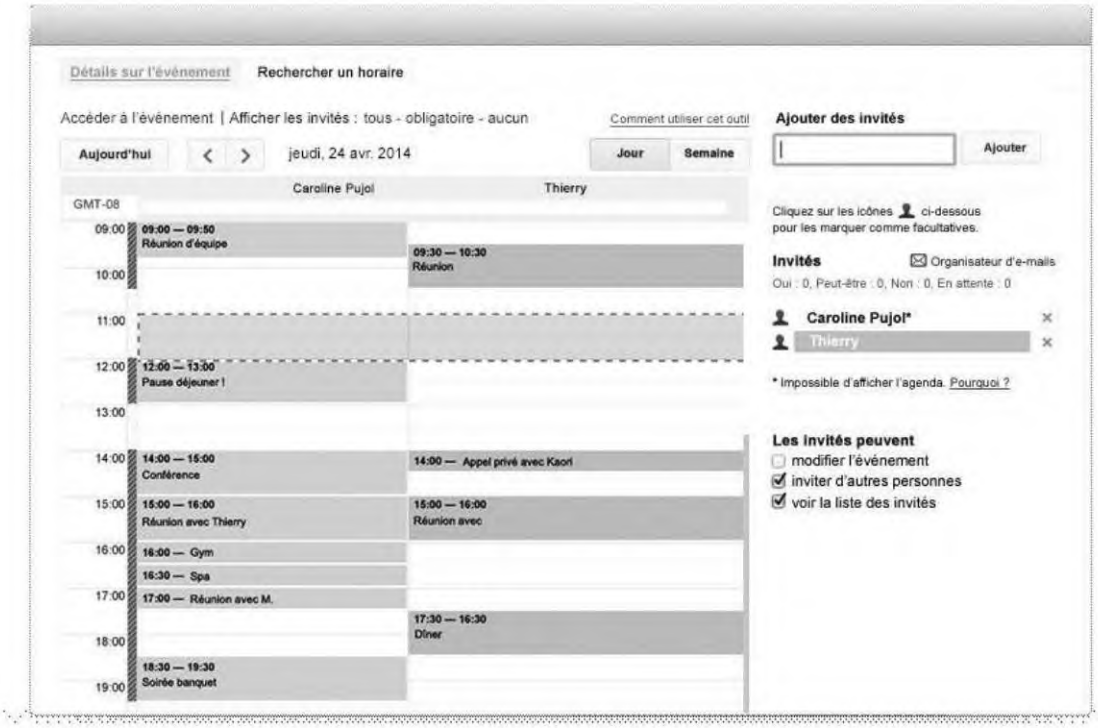

Figure 14.5 — L'interface de Google Agenda.

L'interface de Google Agenda est très facile à prendre en main : elle offre le même niveau d'ergonomie qu'une application client/serveur. Google Agenda permet de gérer et de partager plusieurs calendriers par utilisateur. L'application permet en particulier des calendriers publics, comme ceux des vacances scolaires ou des jours fériés. Grâce

au support du standard iCal[1], il est possible d'accéder à son calendrier Google depuis une autre interface.

Google Hangout

Hangout, l'outil de webconférence de Google, offre des fonctions de messagerie instantanée, téléphonie sur IP, visiophonie, partage d'écran, transfert instantané de fichier. Il permet de converser simultanément avec dix personnes et de communiquer avec des utilisateurs en dehors de l'entreprise.

Google Drive, Sites

Google Drive est un environnement web qui couvre des fonctions de GED et de bureautique. Il propose des fonctionnalités d'édition de documents un peu moins évoluées que celles d'Office. En revanche, ses **possibilités de coédition et de communication synchrone autour des documents sont révolutionnaires pour un utilisateur de logiciel bureautique classique.**

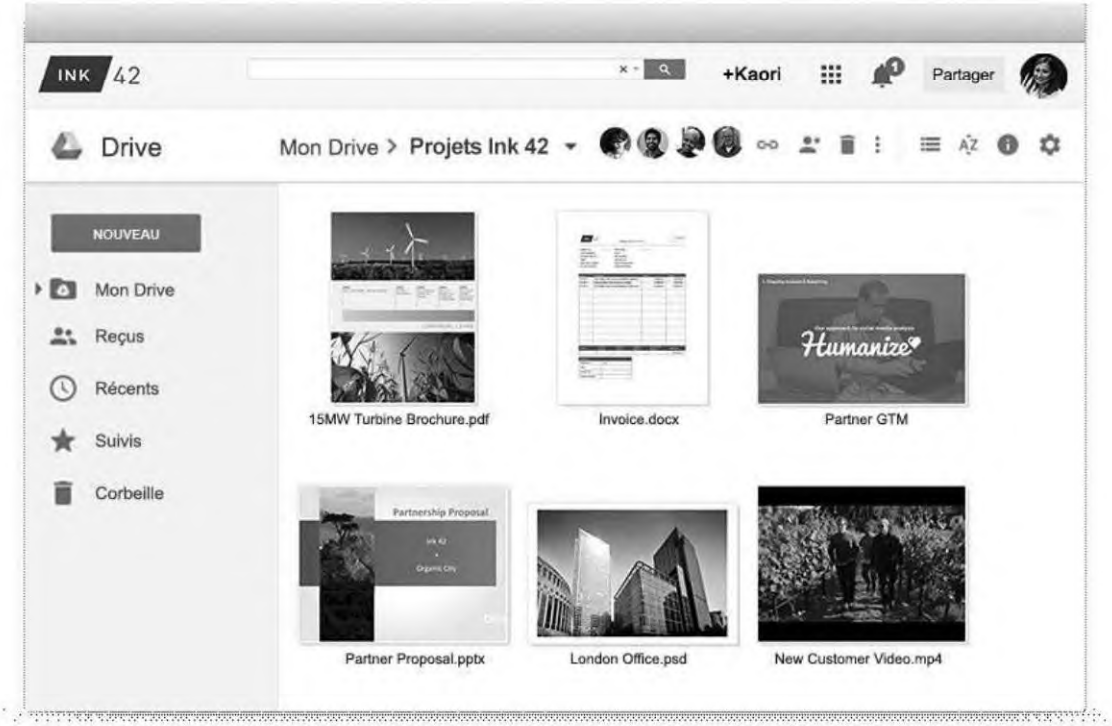

Figure 14.6 — L'interface de Google Drive.

L'outil permet aussi la recherche plein texte dans les documents, cœur de métier de Google.

1. iCal est un standard d'accès aux calendriers, il permet d'accéder à ces derniers depuis Outlook, Thunderbird.

Il offre une grande accessibilité aux documents :

- depuis n'importe quel PC ou terminal mobile ;
- par un utilisateur hors de l'entreprise ;
- *via* d'autres outils de bureautique en ligne comme Zoho.

Lorsqu'un document est importé dans Google Drive, il est transformé dans un format spécifique à Google, qui ne gère pas les fonctions bureautiques avancées. Cet import peut donc déboucher sur une perte de certaines propriétés du document : par exemple, l'import d'un fichier Excel sophistiqué fait perdre les macros. **Il faut donc considérer Google Drive comme un outil de collaboration/partage plutôt que comme une GED, destinée à stocker tous les documents d'entreprise.**

Google Sites est un Wiki avec des fonctions de GED. Il permet de créer un site web, ou à une équipe de créer un portail ou un intranet.

14.2.2 Microsoft Office 365

Il est possible de faire un parallèle entre les offres Google et Microsoft :

- **solutions grand public :** galaxie Gmail chez Google, galaxie Live chez Microsoft ;
- **solutions entreprises :** offre Google Apps for Work chez Google, offre Office 365 Business chez Microsoft.

Comme pour Google Apps, il est possible de gérer ses e-mails de type `prenom.nom@entreprise.com`.

Office 365 propose des versions SaaS de ses offres de référence :

- **Exchange :** il s'agit des fonctions de communication asynchrone proposées par Microsoft dans son offre « software » intitulée Exchange Server ;
- **Skype Entreprise :** il s'agit des fonctions de communication synchrone issues du rachat de Skype ;
- **SharePoint et Yammer :** il s'agit des fonctions de partage d'information unifié proposées par Microsoft.

Office 365 est donc totalement isofonctionnel avec l'offre « software » de Microsoft.

L'offre Office 365 de Microsoft privilégie un usage à partir des logiciels de la gamme Office (Outlook, Word, Excel, PowerPoint), mais elle offre aussi des interfaces web : les Office Web Apps. Ces Office Web Apps proposent moins de fonctionnalités que Google Drive. Par exemple : Excel Online ne permet pas de faire de scripts, Word Online ne permet pas de générer de table de matières. En revanche, elle gère Open XML en natif, et son interface est la même que celle d'Office, ce qui facilite le passage d'Office à Office Web.

Bien entendu, Microsoft propose des offres intégrant Office 365 et Office.

Microsoft a particulièrement soigné l'intégration de ses services Online avec le système d'information de l'entreprise : des outils de migration depuis Exchange et de synchronisation avec Active Directory sont proposés. Si ces outils adressent uniquement l'intégration avec les autres produits Microsoft, ils sont plus faciles à utiliser que ceux de Google. Ces derniers adressent des intégrations avec des technologies plus variées, mais nécessitent plus de travail d'intégration.

14.2.3 Zoho Online Office

La suite de collaboration Cloud de Zoho est assez proche de celle de Google ou Microsoft. Zoho a proposé de nombreuses innovations intéressantes et ses fonctionnalités ont souvent vu le jour avant celles du géant des moteurs de recherche. Zoho propose un vaste catalogue d'applications, au sein duquel d'autres sociétés peuvent déployer leurs SaaS. Cette offre est une bonne illustration des fonctions de collaboration possibles en SaaS.

En revanche, les interfaces de sa suite sont moins ergonomiques que celles de Google. Par ailleurs, la pérennité de Zoho reste à démontrer.

14.2.4 IBM Collaboration tools

Collaboration Tools est la proposition d'IBM de plateforme de collaboration SaaS pour les PME. On y trouve les fonctionnalités suivantes :

- messagerie et agenda ;
- messagerie instantanée ;
- webconférence avec partage de bureau et pilotage de présentation ;
- partage de fichiers ;
- gestion de projet ;
- partage de tâches avec « Activities » ;
- réseau social d'entreprise ;
- création de formulaires.

On peut remarquer qu'IBM est un vrai nouvel entrant dans le monde des SaaS, puisque la société n'avait jusqu'alors pas de solution de messagerie grand public, contrairement à Microsoft ou Google. En revanche, IBM met immédiatement le pied dans les réseaux sociaux, une fonction en laquelle Big Blue semble croire : IBM propose en effet une offre de réseau social d'entreprise en mode « software » intitulée Lotus Connexions.

14.2.5 Basecamp

37Signals était un acteur de référence dans le monde du web. La société a inventé Ruby On Rails et écrit de nombreux livres de références. Basecamp était leur outil phare. La société s'est donc rebaptisée Basecamp.

Basecamp est un **outil de partage d'information intégré**, orienté gestion de projet. Il permet en particulier de créer un espace projet entre une société de service et un client. Basecamp permet de :

- partager un des documents ;
- archiver des messages ;
- s'affecter des tâches ;
- partager un calendrier ;
- partager un planning projet.

Basecamp propose des API permettant d'accéder aux informations liées au projet *via* des flux RSS ou une interface iCal.

Il est simple à prendre en main et assez productif grâce à son interface HTML5. Il est très utilisé dans le monde des agences web.

14.2.6 Trello

Trello est un outil de gestion de projet SaaS, très visuel et très efficace. Il permet d'affecter des tâches aux membres d'une équipe. Il a été lancé en 2011 et connaît un grand succès.

Il utilise la métaphore du flux tendu, issue de Toyota (méthode Kanban). Les tâches avancent dans le temps de la gauche vers la droite, en changeant d'état : « à faire », « en cours », « terminé », etc.

14.2.7 CISCO WebEx

WebEx est une solution de référence pour la webconférence en mode SaaS, rachetée en 2007 par Cisco. Elle peut être utilisée sur abonnement ou en paiement à l'acte. Elle permet :

- l'organisation d'une conférence au travers d'un outil d'agenda partagé ;
- les échanges par messagerie instantanée ;
- la téléphonie sur IP ;
- la visiophonie ;
- le partage de bureau ;
- le partage d'applications ;
- l'annotation de documents ;
- le transfert instantané de fichier ;
- l'enregistrement vidéo de la conférence.

Au préalable, WebEx nécessite l'installation d'une extension de navigateur ou plug-in.

Parmi les concurrents directs de Webex, on peut citer Adobe Connect ou Citrix GoToMeeting.

Figure 14.7 — L'interface de WebEx.

14.2.8 Réseaux sociaux d'entreprise

Les réseaux sociaux d'entreprise proposent généralement les fonctions suivantes :

- profil : présenter son parcours, ses compétences et ses centres d'intérêt ;
- communautés : organisées autour d'une compétence ou d'un centre d'intérêt ;
- capitalisation : partage de la connaissance ;
- curation : partage de liens intéressants sur Internet ;
- boîte à idée pour des projets d'innovation.

Parmi les outils SaaS centrés sur les collaborateurs et leur réseau en entreprise, on peut citer : Microsoft Yammer, Jive, Dassault 3SD SwyM, Salesforce Chatter, Bluekiwi, ElCurator.

Et la solution Facebook for Work sort prochainement.

14.2.9 RememberTheMilk

Remember the Milk est un service Cloud de référence pour la gestion de tâches. Il permet la saisie et le classement des tâches, la configuration d'une date de fin, la priorisation et le partage avec des tiers.

Il permet d'outiller la démarche « Getting Things Done » de David Allen[1]. En deux mots, cette démarche consiste à noter toutes les choses qui nous traversent l'esprit pour se vider la tête, puis à les prioriser de manière efficace.

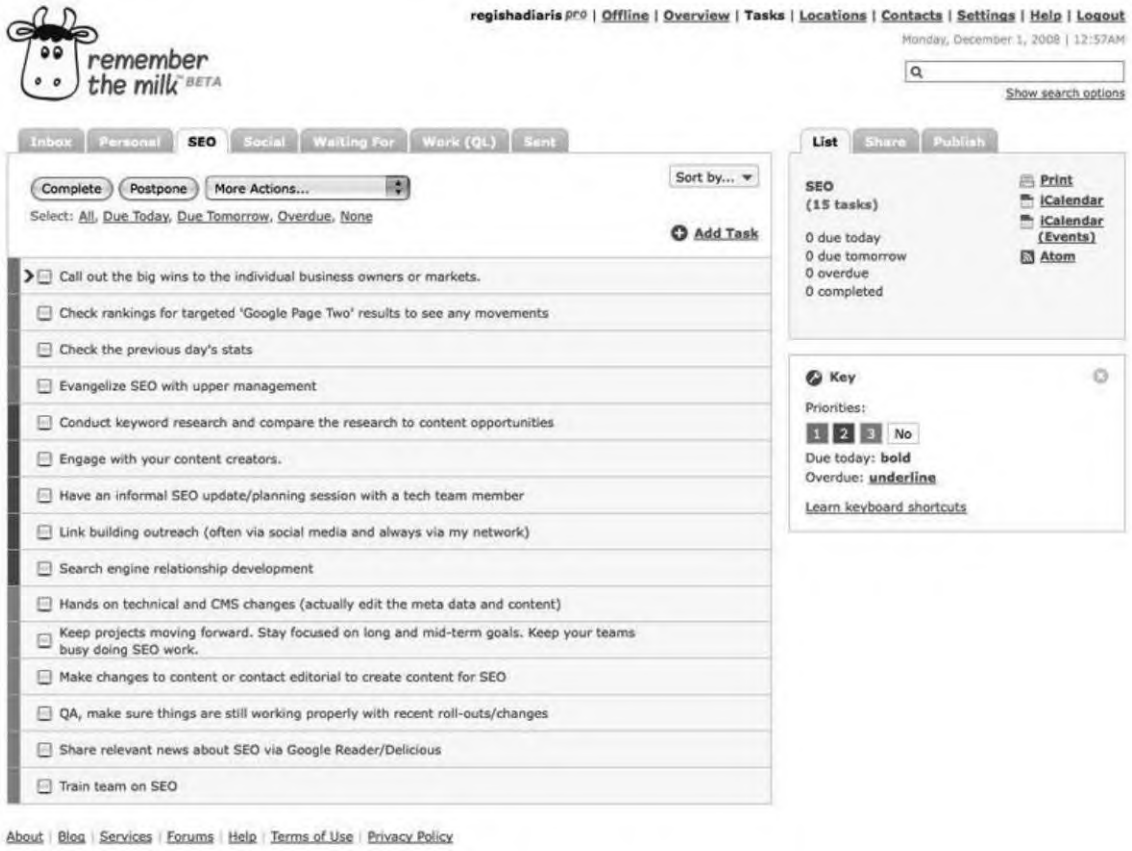

Figure 14.8 — L'outil RememberTheMilk.

Remember the Milk permet d'être notifié de la date de rendu d'une tâche *via* e-mail, SMS, ou *via* un signal sur téléphone ou sur tablette. On peut créer une tâche en envoyant un e-mail au service (chaque utilisateur dispose d'une adresse e-mail de type xxx@rmilk.com pour cet usage). On peut définir le lieu où la tâche devra être effectuée et consulter *via* son téléphone les tâches à effectuer près de l'endroit où l'on se trouve.

Remember the Milk sait aussi s'intégrer aux agendas, comme celui de Google.

Parmi les concurrents de RememberTheMilk, on peut citer : Wunderlist, Things, todoist, etc.

1. http://gettingthingsdone.com/

14.2.10 Evernote

Evernote est la référence en matière de prise de note cloud avec 100 millions d'utilisateurs en 2014. Il propose des interfaces adaptées à tous les écrans : PC, Mac, iPhone, Android, etc.

Le service permet de partager les notes avec des tiers. On peut créer une note en envoyant un e-mail au service (chaque utilisateur dispose d'une adresse e-mail de type xxx@m.evernote.com pour cet usage). Il permet de scanner des pages de texte et d'y faire des recherches grâce à une fonction de reconnaissance de caractères : par exemple, on peut photographier des cartes de visite depuis un mobile et les retrouver par une recherche sur le nom du contact.

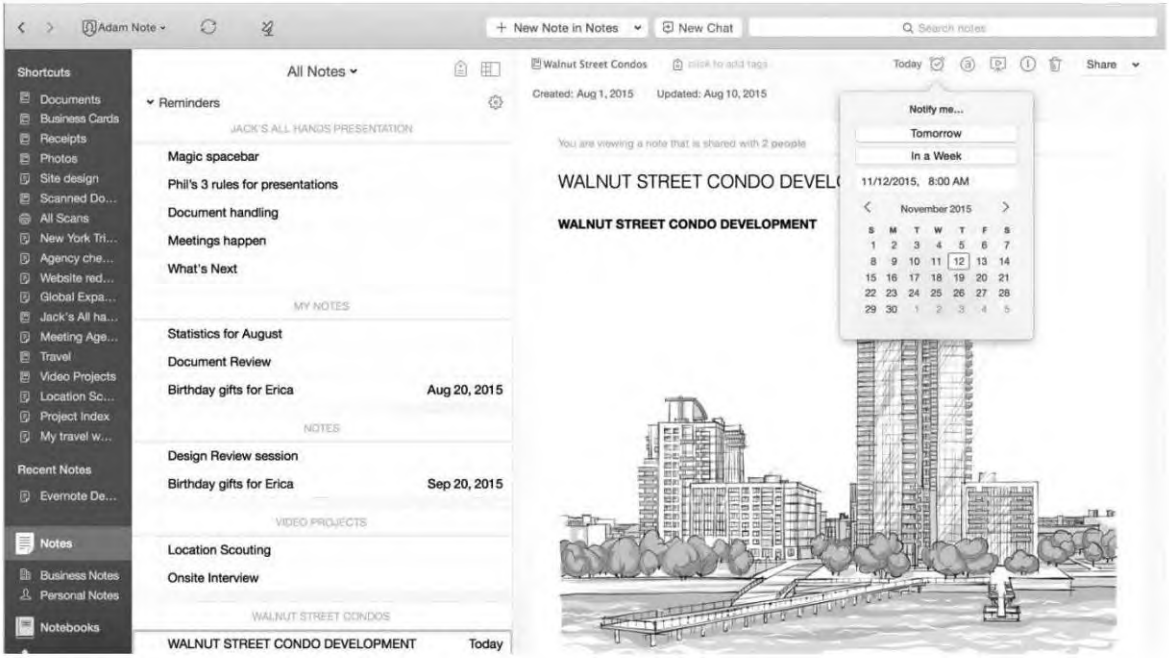

Figure 14.9 — L'outil Evernote.

Parmi les concurrents d'Evernote, on peut citer : Microsoft OneNote, Google Keep, Simple Note, etc.

14.2.11 Mindmeister.com

MindMeister est un outil de carte mentale ou « MindMapping » : concrètement, cela signifie qu'il permet de prendre des notes en les organisant sous la forme d'un arbre, avec des liens entre les notes qui évoquent des branches et des feuilles. L'avantage de cette représentation est qu'elle permet de hiérarchiser ses notes en sous-notes, sous-sous-notes, etc. Ce type d'outil est déconcertant au premier abord, mais il est très utilisé pour prendre des notes en réunion, sur un tableau blanc ou sur un ordinateur.

MindMeister permet de partager et de coéditer ses notes avec des tiers. Il utilise le principe du Wiki : conservation des versions successives d'un document pour pouvoir revenir à une version précédente.

Il dispose d'une interface web, mais aussi d'interface iPhone et iPad.

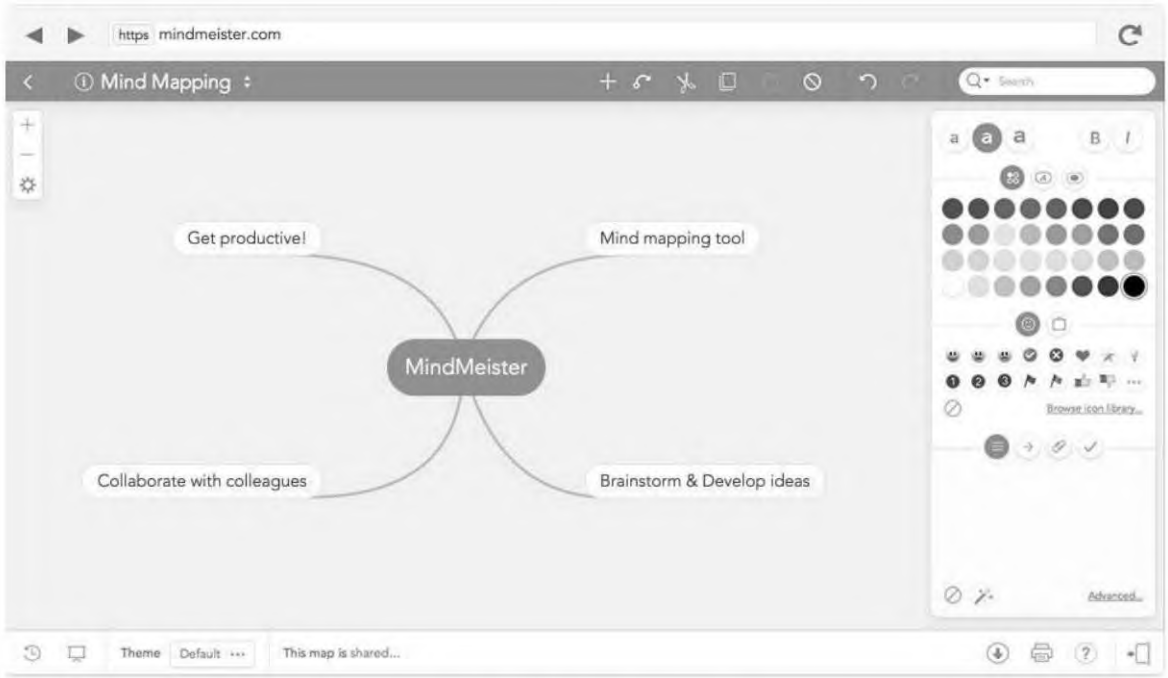

Figure 14.10 — L'outil MindMeister.

14.2.12 DropBox

DropBox est un outil SaaS de partage de documents. Il offre une interface web permettant de partager des fichiers avec toute personne identifiée par son adresse e-mail.

Sa particularité est d'offrir une possibilité de synchronisation entre l'espace DropBox et le poste de travail *via* un logiciel client lourd. Si un espace DropBox est partagé entre plusieurs collaborateurs, le système synchronise en tâche de fond le serveur DropBox avec tous les postes des collaborateurs. Cette fonction est précieuse pour disposer d'un référentiel identique, et ce, même en mode déconnecté. Elle introduit un risque : si plusieurs personnes éditent le même document en même temps, l'intégrité du document est menacée.

Google et Microsoft ont développé des offres équivalentes respectivement avec Drive et SkyDrive.

DropBox ne fournit pas de suite bureautique en ligne, mais permet l'édition de documents *via* Office Online.

D'autres solutions de stockage Cloud offrent l'édition *via* des suites tierces :

- box.net utilise Office Online ;
- oodrive utilise Zoho Docs.

14.2.13 CDC Arkhinéo

Arkhinéo est une offre SaaS du groupe CDC (Caisse des Dépôts), dédié à **l'archivage documentaire** en ligne. Son offre assure l'archivage longue durée de tous les types de documents électroniques (factures, contrats, images-chèques, bulletins de salaires, e-mails, etc.) sous forme scellée et sécurisée. Elle propose la mise en œuvre d'un tableau de conservation afin de décider de la durée d'archivage nécessaire (1 an, 10 ans, 30 ans, indéfinie). Elle s'appuie sur des Datacenters distants de plus de 450 km.

CDC Arkhinéo en qualité de tiers archiveur peut certifier, grâce au procédé de scellement utilisé, que le document restitué n'a pas été modifié. Le document peut ainsi servir de preuve vis-à-vis de la justice. Cette offre met donc très fortement l'accent sur la valeur juridique des documents.

14.2.14 Adobe Creative Cloud

Le Creative Cloud d'Adobe est une offre destinée aux graphistes professionnels. C'est un service Cloud de stockage et de partage de fichiers comparable à DropBox. Il permet, en particulier, de synchroniser les fichiers Photoshop, Illustrator, etc. entre son ordinateur, sa tablette et son Smartphone. Le Creative Cloud permet aussi de louer les logiciels Adobe au mois plutôt que de les acheter. Les logiciels sont cependant installés sur l'ordinateur et non utilisés directement en ligne.

Adobe propose aussi Photoshop.com, un outil Cloud de retouche photo destiné au grand public. Il permet de recadrer les photos, d'ajuster leur contraste et leurs couleurs, de supprimer les yeux rouges, de créer des effets graphiques, etc.

14.2.15 Moqups

Moqups est une solution SaaS permettant de collaborer autour de *storyboard*, de maquettes d'écrans applicatifs.

Cette solution est intéressante à plusieurs titres :

elle permet de créer un écran très vite grâce à des outils très efficaces ;

elle offre la collaboration en temps réel à plusieurs utilisateurs ;

elle démontre qu'il est possible de « dessiner » des choses simples au sein d'une interface HTML5 (on ne parle pas d'infographie sophistiquée ici).

En résumé

Ce chapitre a présenté des solutions SaaS offrant des fonctions de collaboration et partage d'information.

Les offres les plus avancées sur l'ensemble de ces fonctions sont celles de Google, Microsoft et Zoho. D'autres offres sont très pointues sur une fonction collaborative précise comme celles de Basecamp, WebEx, ou Adobe.

Des offres sont assez avancées pour le stockage et l'archivage : celles de box.net, Oodrive, CDC Arkhinéo.

15

Les services FrontOffice

Objectif

Ce chapitre présente des services FrontOffice. Il aborde des services destinés à la construction de sites web, ainsi que des applicatifs destinés à la gestion de la relation client.

15.1 LA RECHERCHE AU SEIN D'UN SITE WEB

Les grands moteurs de recherche proposent aujourd'hui la possibilité de s'intégrer dans l'interface d'un site web pour proposer une recherche au sein de ses pages. Dans ce domaine, on peut citer les offres Google Site Search, Microsoft Bing Search API.

Ces offres permettent de présenter les résultats de recherche dans l'interface du moteur de recherche, ou bien directement dans l'interface du site. Dans le second cas, il est nécessaire d'utiliser les API fournies par le moteur pour mener à bien l'intégration.

Ces offres sont généralement gratuites pour un faible volume de recherche ; des offres payantes sont proposées aux entreprises sur la base du nombre de requêtes par an.

15.2 LES SERVICES DE CARTOGRAPHIE

Un certain nombre de services de cartographie ont émergé ces dernières années, permettant de présenter des informations d'entreprise sur un fond de carte. On peut citer par exemple : Google Maps, Microsoft Bing Maps, Mappy, etc.

Ces services offrent des API permettant une intégration très simple. En effet, ils n'ont pas vocation à créer des systèmes d'information géographique (SIG) pour des besoins avancés, comme l'urbanisation ou la gestion des cadastres.

Ces services offrent en général plusieurs API :

- une API de « *geocoding* » qui permet la transformation d'adresses postales en coordonnées terrestres (latitude et longitude) ;
- une API de manipulation du fond cartographique, permettant d'afficher la carte d'un lieu donné avec un niveau de zoom adéquat ;
- une API de calcul d'itinéraire, permettant de calculer le trajet le plus cours entre deux points en voiture ou à pied ;
- parfois une API de dessin sur le fond cartographique, qui permet de présenter des points d'intérêt (sites de présence de l'entreprise), ou des polygones correspondant à des limites de zones (terrains inondés en cas de crue, par exemple).

Ces services sont fréquemment utilisés dans les sites web des entreprises pour localiser leurs points de présence. À la manière des services de recherche, ils sont généralement gratuits pour un accès limité ; des offres payantes sont proposées aux entreprises sur la base du nombre de requêtes par an.

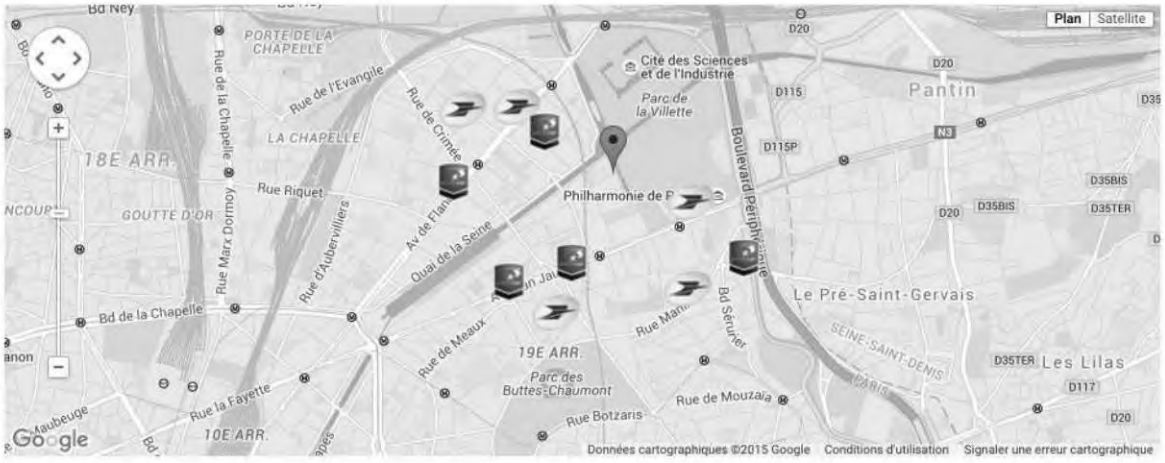

Figure 15.1 — Affichage de bureaux de poste sur un fond de carte.

15.3 LES SERVICES DE COMMERCE ÉLECTRONIQUE

Pour développer un site de commerce électronique et vendre des articles sur le web, les entreprises ont aujourd'hui plusieurs approches possibles :

- elles peuvent développer en spécifique l'ensemble des composants (catalogue, panier d'achat, paiement, livraison), éventuellement en utilisant des composants Open Source ;
- elles peuvent s'appuyer sur des offres SaaS clé en main ;
- elles peuvent s'appuyer sur des API pour l'un ou l'autre de ces composants.

Il existe des services en ligne pour la gestion de panier d'achat. Par exemple, Amazon Remote Shopping Cart permet aux sites de commerce électronique d'externaliser cette fonction. Là encore, une API permet l'intégration du service à un site web.

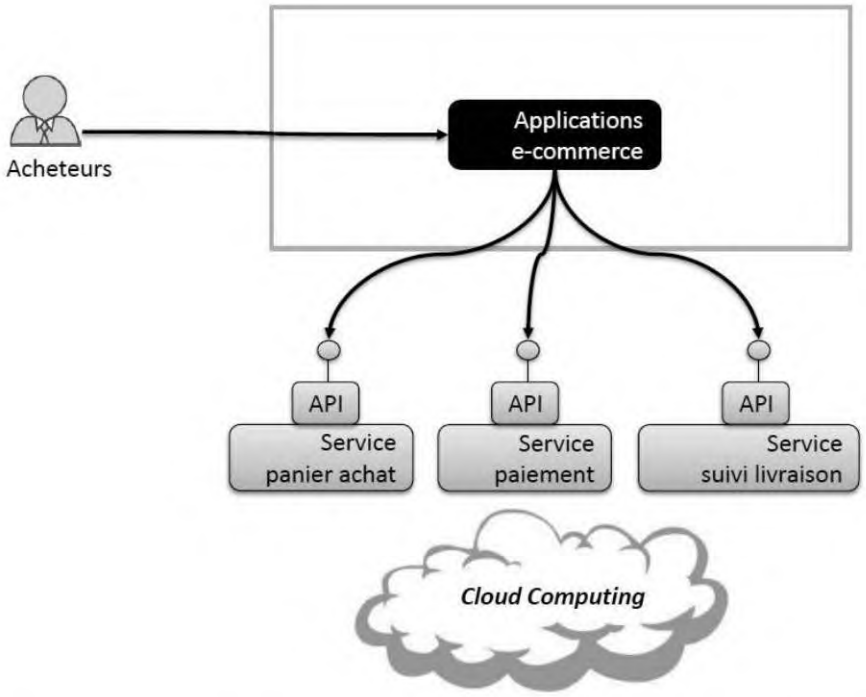

Figure 15.2 — L'intégration d'API dans le cadre du commerce électronique.

De nombreux services en ligne permettent la délégation de l'étape de paiement à un tiers. Cette délégation est particulièrement pertinente, car elle évite de devoir gérer des numéros de carte bancaire, ce qui est particulièrement critique en termes de sécurité. Le service de paiement valide la solvabilité de l'acheteur, effectue le paiement et notifie au commerçant que la transaction a bien eu lieu. Là encore l'intégration d'une API, cette fois sécurisée, est nécessaire. On peut citer les services de paiement suivants :

- Paypal ;
- Stripe ;
- des solutions proposées par des banques : Sogenactif de la Société Générale, CyberMUT P@iement du crédit Mutuel, etc.

Enfin, des services de gestion de livraison sont fournis par FedEx, UPS, ou la Poste. Ils proposent eux aussi des API pour permettre le suivi de l'acheminement des commandes au sein de l'interface du site de commerce électronique.

Le commerce électronique est ainsi un bon exemple d'utilisation avancée d'API fournis par des tiers.

15.3.1 Shopify

Shopify est une référence dans le domaine des boutiques en ligne SaaS. Ce service Cloud permet de créer un catalogue de produits avec photos, descriptions, prix et de les présenter par rubriques. Il sait gérer plusieurs langues et plusieurs modèles de TVA. Il accepte le paiement *via* son propre prestataire de paiement (généralement une banque), PayPal, Google Checkout. Il permet d'utiliser les transporteurs UPS, USPS, FedEx. Il propose de créer des coupons de réduction.

Bien entendu, on peut créer des pages pour présenter sa société, adapter le site à sa charte graphique et utiliser son nom de domaine de type : MaBoutique.fr

Shopify offre aussi des statistiques de visite. Il est possible de créer un site web adapté aux écrans de mobiles.

Enfin, Shopify propose une application iPad pour gérer sa boutique physique.

15.4 LES SERVICES DE MAILING

L'envoi de courrier électronique en masse à des fins marketings est une manière efficace et peu coûteuse de communiquer auprès de ses clients. Il pose cependant un certain nombre de problèmes. La problématique technique est aujourd'hui facile à gérer, les machines standards étant assez puissantes pour gérer la charge induite par l'envoi de nombreux e-mails. En revanche, d'autres aspects sont l'affaire de spécialistes :

- **la gestion de l'opt-in :** il s'agit d'obtenir l'accord des utilisateurs ciblés par les campagnes de mailing et de leur proposer une possibilité de se désinscrire à tout moment ;
- **la gestion des statistiques de campagne :** il s'agit de mesurer le nombre d'ouvertures de message, le nombre de clics à partir des messages ;
- **la gestion du spam :** les entreprises investissent massivement dans la lutte contre les e-mails non sollicités issus de pirates en tous genres, et il n'est pas simple pour un message marketing d'apparaître comme « honnête ». Le domaine (par exemple monentreprise.com) utilisé pour l'envoi des messages peut se retrouver catalogué comme domaine de spammeur, ce qui est catastrophique pour tous les e-mails issus de l'entreprise, qui seront automatiquement supprimés chez leurs clients et partenaires. La gestion de mailing marketing sans passer pour un spammeur est aujourd'hui une affaire d'experts.

Pour bien gérer ces différentes problématiques, il est très pertinent d'utiliser les services de sociétés spécialisées, comme MailChimp, SendInBlue, SendGrid, ou Sarbacane.

Ces sociétés offrent un service intégré de mailing : elles se rapprochent donc du modèle cloud.

Elles offrent aussi des API permettant :

- l'import/export des adresses de messagerie depuis le système d'information d'entreprise ;
- parfois, la collecte des résultats d'une campagne.

Ces offres fonctionnent par abonnement et sur la base du nombre de messages envoyés par campagne.

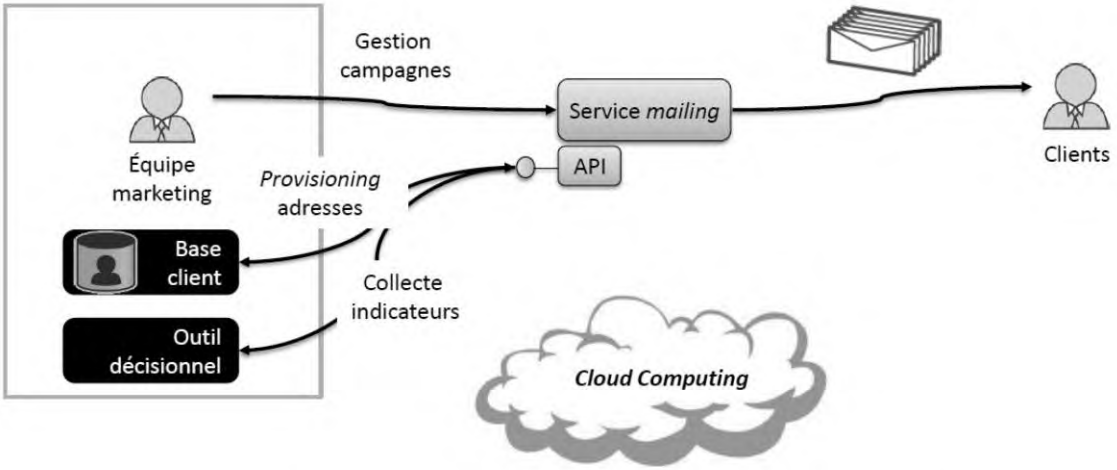

Figure 15.3 — Utilisation d'un outil de mail marketing.

15.5 L'ANALYSE DE TRAFIC

L'analyse de trafic est aussi un service proposé depuis longtemps en mode SaaS. Le mode SaaS est d'autant plus pertinent qu'il amène des chiffres crédibles, car mesurés par un tiers qui n'est pas juge et parti.

Nous présentons rapidement deux de ces offres dans ce paragraphe.

15.5.1 Xiti

Xiti est un outil historique dans le monde de l'analyse de trafic des sites web. Il a évolué vers des fonctions décisionnelles.

La gamme Xiti comporte en particulier :

- **un** outil permettant le suivi en temps réel du trafic et des indicateurs de performance, l'aide à la prise de décisions, le calcul de ROI des actions marketing online ;
- **un** outil permettant la segmentation et l'analyse des populations accédantes ;
- un outil permettant le monitoring en temps réel de l'accessibilité du site ;

- un outil pour la mesure de fréquentation des sites mobiles ;
- un outil qui permet de mesurer l'impact de sa marque sur les réseaux sociaux.

15.5.2 Google Analytics

Google Analytics est issu du rachat de la solution Urchin. L'usage de cette solution est gratuit. Elle permet la mesure du trafic web, l'analyse du comportement des utilisateurs, le suivi des objectifs marketing. Elle est destinée à fonctionner en association avec le système d'encarts publicitaires AdSense de Google. Elle peut cependant être utilisée pour d'autres types de suivi.

15.6 GESTION DE LA RELATION CLIENT

La CRM (gestion de la relation client) est une fonction relativement simple à externaliser, car les équipes commerciales travaillent souvent de manière très générique. L'intégration de la CRM au système d'information s'effectue généralement par une ressaisie manuelle des commandes par les commerciaux lorsqu'une affaire est gagnée. C'est ainsi une des premières gammes de progiciels à être allée vers le modèle cloud.

15.6.1 Salesforce

La principale application proposée par Salesforce est la CRM, mais la solution offre aussi des fonctionnalités de marketing, gestion de force de vente, support client, analyse, etc.

Il est non seulement possible d'intégrer les comptes utilisateurs (identifiants/mots de passe) à ceux de l'entreprise, mais aussi d'échanger des flux de données entre la CRM et les fonctions de gestion financière situées dans le système d'information.

Salesforce peut s'intégrer avec Outlook pour le rapatriement des contacts clients. Elle peut aussi s'intégrer avec Google Apps pour tirer parti de ses fonctions collaboratives.

Comme on l'a vu précédemment, Salesforce propose par ailleurs sur App Exchange un vaste écosystème d'applications SaaS répondant à divers besoins : analyse et tableaux de bord décisionnels, gestion des ressources humaines, gestion financière, gestion d'acheminement, gestion de projet, gestion de centre d'appel, outils marketing, gestion des relations partenaires, etc.

15.6.2 Microsoft Dynamics CRM

Microsoft propose une offre de CRM dans sa gamme Online : Dynamics CRM. Les fonctionnalités de la solution sont proches de celles de Salesforce. Elle s'intègre fortement avec Outlook. Les autres outils de la gamme Dynamics sont aussi disponibles en mode SaaS.

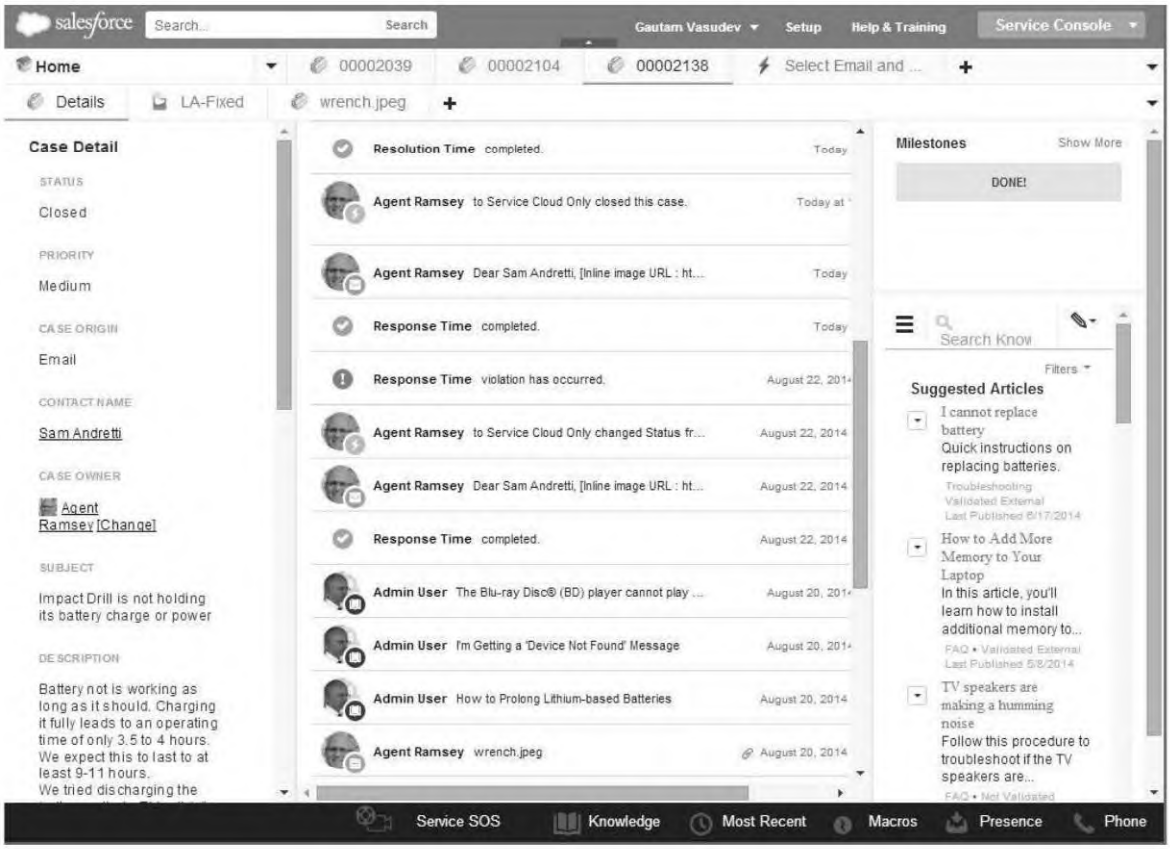

Figure 15.4 — La CRM Salesforce.

15.6.3 Sage CRM

La solution de Sage offre les fonctionnalités classiques. Elle se distingue par la richesse et l'ergonomie de son interface. C'est la première brique ERP que Sage a proposée en mode SaaS : la migration de l'ensemble de l'offre en mode Cloud est en cours.

15.6.4 Zoho CRM

L'offre de Zoho permet de gérer les contacts, devis, commandes, achats, factures, etc. Elle propose une page d'accueil personnalisable. Elle permet l'import de données fournisseurs, contacts, produits.

Highrise est un concurrent direct de Zoho CROM. C'est la solution issue des créateurs de Basecamp. Elle est réputée simple et intuitive.

En résumé

Ce chapitre a présenté les principales familles de services FrontOffice :

– services de recherche, de commerce électronique, de mailing de masse, de cartographie ;

– services de suivi de trafic ;

– services de CRM.

16

Les services BackOffice

Objectif

Ce chapitre passe rapidement en revue quelques offres de progiciel SaaS. Cette offre est vaste et en plein développement : il est donc difficile de la cerner de manière exhaustive et de la décrire complètement.

Il faut noter par ailleurs qu'il existe de nombreuses offres de progiciels en mode infogéré. Il ne s'agit pas alors de SaaS à proprement parler. Nous avons pris le parti de ne pas décrire ces offres.

16.1 GESTION RH

L'externalisation des fonctions RH est assez fréquente, à la manière de la CRM. Certaines fonctions, comme l'édition des bulletins de paie, sont en effet propices à l'externalisation.

En particulier, la « gestion des talents » est un domaine où le cloud est en plein essor, avec des offres comme CezanneSW.com, Workday.com, CornerStoneOnDemand.fr, TalentSoft.fr, SuccessFactors.com (racheté en 2012 par SAP).

16.1.1 SucccessFactors

SuccessFactors est une solution de référence pour la gestion des talents. Elle intéresse de nombreux grands comptes par sa promesse de mesurer l'expertise métier, le leadership et l'engagement. La solution incite à moderniser le management, en particulier en facilitant la communication entre les niveaux hiérarchiques.

Elle propose des modules de recrutement, gestion de la performance et des primes, gestion des formations. Elle est intégrée avec les réseaux sociaux, notamment LinkedIn.

16.1.2 ADP

ADP est un acteur positionné de longue date sur l'externalisation des fonctions RH. Il propose des solutions pour toutes les tailles d'entreprises.

Son offre est accessible au travers d'un portail. Il offre les fonctions suivantes :

- gestion de la paie ;
- gestion des déclarations sociales ;
- gestion de la formation ;
- gestion de l'administration du personnel ;
- gestion du reporting social et du décisionnel RH ;
- gestion des feuilles de temps et activités.

16.1.3 Zoho People

Zoho People est une solution basique de gestion des différents aspects liés aux ressources humaines : recrutement, organisation, départements, salaires, absences, vacances, etc.

16.1.4 Efolia

Efolia est une solution SaaS de dématérialisation des feuilles de paye. La conséquence de la dématérialisation est que le document numérique devient l'original, tandis que le papier n'est plus qu'une simple copie.

Efolia met à disposition des salariés un coffre-fort cloud dans lequel les documents sont gardés à vie.

16.2 ANALYSE DÉCISIONNELLE

Il existe des solutions SaaS pour l'analyse de données clients et la création de rapports décisionnels. On peut citer parmi elles :

- Amazon Web Services propose le Data Wharehouse RedShift, Data Pipeline pour intégrer les flux de données, EMR et Machine Learning pour analyser les données dans une approche Big Data ;
- JasperSoft et BusinessObjects proposent leur offre décisionnelle en SaaS sur la base d'Amazon Web Services ;
- Bime Analytics propose un service de reporting SaaS ;

- le service de reporting QlikView propose une solution très facile à prendre en main ;
- Lokad propose un logiciel de prévision de la demande et des ventes.

16.3 SUITES ERP

Ces offres SaaS couvrent, en version simplifiée, les fonctions :

- comptabilité et finances ;
- paye ;
- RH ;
- CRM ;
- gestion logistique ;
- gestion de projet ;
- etc.

On peut citer :

- **SAP Business by Design :** elle constitue les premiers pas dans le monde SaaS d'un éditeur issu d'un modèle très différent, celui des gros progiciels. Les observateurs affirment que la transition est difficile pour SAP ;
- **financialforce.com :** une solution intégrée à la plateforme Salesforce, et dont Salesforce est actionnaire ;
- **Netsuite :** une offre dont Larry Allison est actionnaire ;
- **Sage Online :** des déclinaisons SaaS des offres d'un acteur incontournable pour les PME.
- **OpenERP :** une offre Open Source avec une déclinaison SaaS ;
- **Your Cegid :** solution française ;
- **Esker.fr :** solution française de dématérialisation des factures.

En résumé

Le marché des progiciels en ligne est émergent et amené à se développer dans les prochaines années. Certaines solutions comme celles de Salesforce, ADP, Sage sont déjà largement adoptées.

17

Perspective : le modèle du cloud desktop

Objectif

La richesse de l'offre SaaS disponible à ce jour commence à rendre possible la création d'un environnement de travail entièrement en ligne, accessible uniquement *via* un navigateur.

La généralisation est en cours et on peut commencer à se projeter sur le mode de fonctionnement du « cloud *desktop* ».

17.1 LE MODÈLE DU BUREAU « CLASSIQUE »

Le concept du cloud *desktop* repose sur le même modèle que celui du poste de travail classique, il fournit à l'utilisateur :

- **une interface d'accueil :** le bureau avec ses fenêtres et sa corbeille ;
- **des espaces de stockage (stockage) :** le disque dur du PC et les serveurs de fichiers ;
- **des applications d'accès et d'édition de contenus** : les applications installées localement.

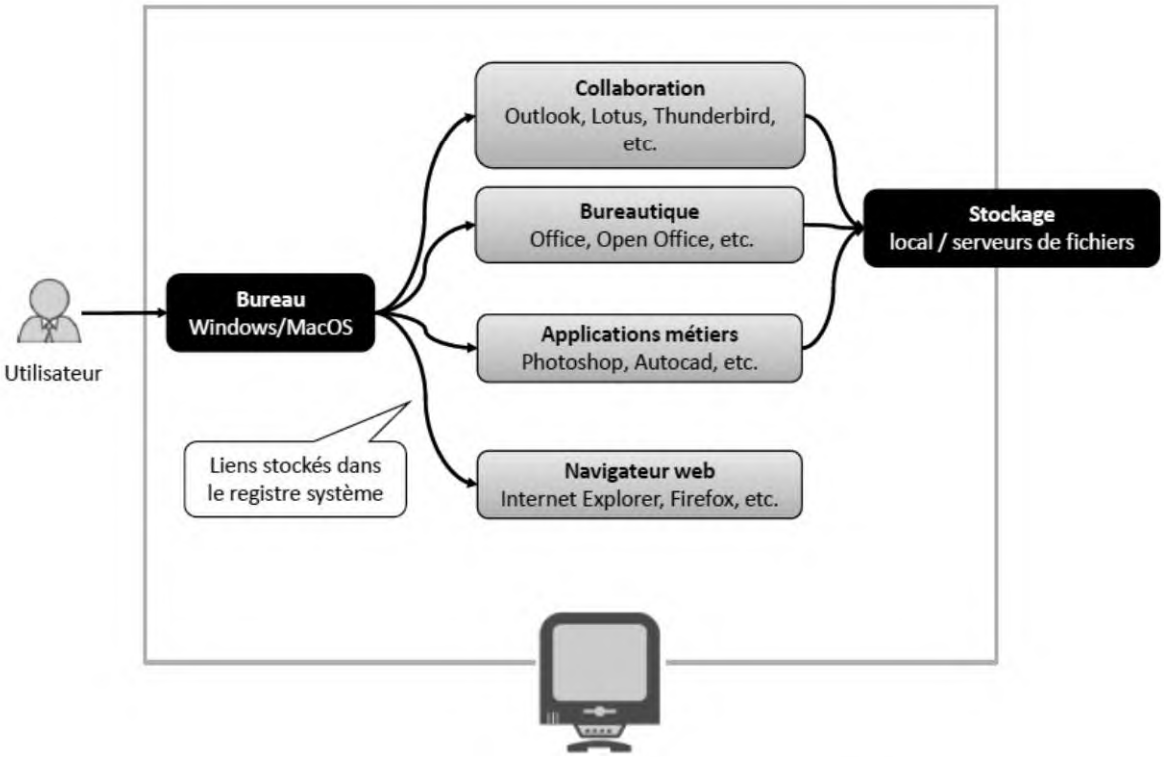

Figure 17.1 — Le bureau « classique ».

17.2 LE MODÈLE DU CLOUD *DESKTOP*

17.2.1 L'interface d'accueil

Dans le cadre du cloud *desktop*, l'interface d'accueil est une application HTML5 qui propose un menu démarrer à la manière de Windows (figures 17.2 et 17.3).

Il existe bien des interfaces web qui reprennent l'apparence du bureau classique avec fenêtres et corbeilles, comme eyeOS.com. Mais elles n'ont pas démontré leur pertinence.

17.2.2 Les applications d'accès et d'édition de contenus

On a aussi vu précédemment un panorama des applications disponibles en SaaS, en particulier :

- les applications de collaboration ;
- les applications de gestion du cycle de vie des documents ;
- les progiciels intégrés.

Le cloud *desktop* devra s'intégrer avec ces applications *via* leurs API. Il sera aussi souhaitable qu'il s'intègre avec les applications qui résident dans le système d'information d'entreprise.

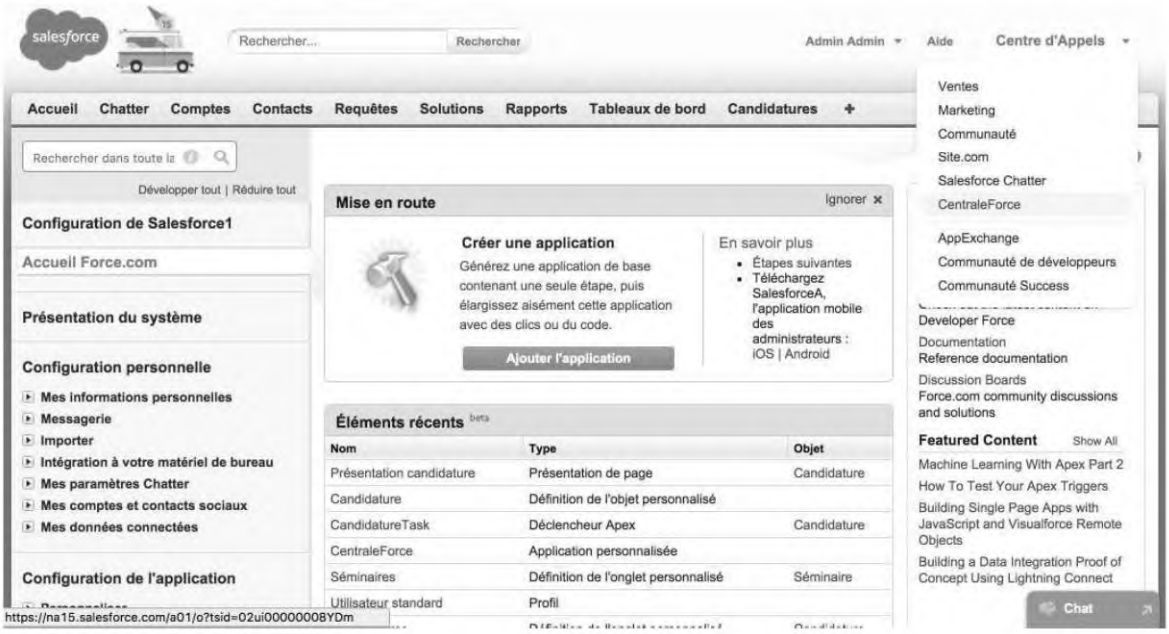

Figure 17.2 — Le bureau Salesforce.

Figure 17.3 — Le bureau Gmail.

17.2.3 La persistance des contenus

On a abordé dans le chapitre précédent quelques outils de stockage en ligne. On a vu qu'ils disposaient tous d'API, ce qui rend leur interfaçage avec les bureaux en ligne très simple.

17.2.4 Description du cloud desktop

Le *cloud desktop* reprend donc le fonctionnement du bureau classique avec son « registre » en bénéficiant des avantages du web. Il pose cependant un problème : celui d'une identité en ligne unique. En effet, accéder à *N* applications avec *N* identifiant/mot de passe est assez pénible. Deux solutions existent à ce problème : la

première consiste à utiliser un cloud *desktop* intégré comme Google Apps. Avec cette approche intégrée, l'identité est centralisée par Google Apps ou Salesforce. La seconde consiste à assembler soi-même ses applications et ses espaces de stockages autour d'une identité fédérée. Pour suivre la seconde approche, il faut maîtriser les mécanismes de la fédération d'identité. Elle est donc réservée à des utilisateurs aguerris. Dans le futur, on risque de retrouver avec Google Apps le syndrome d'enfermement qui a fait l'hégémonie de Windows depuis quinze ans. Et seuls les utilisateurs avancés, qui maîtriseront la fédération d'identité, pourront éviter l'enfermement, un peu comme les utilisateurs de Linux qui ont appris à se passer de Windows.

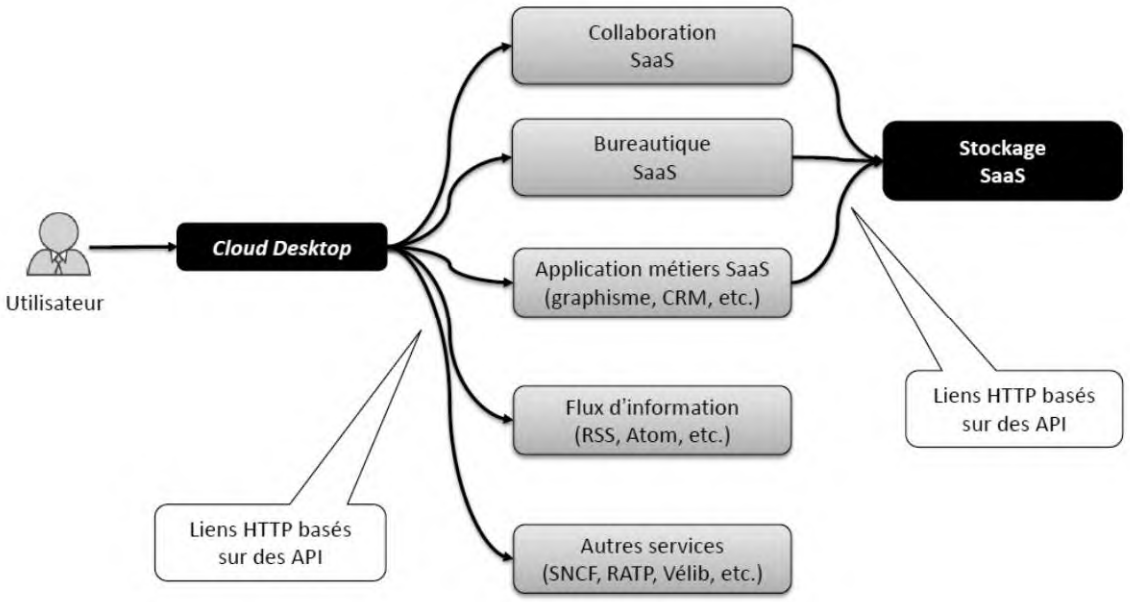

Figure 17.4 — Le cloud *desktop*.

Le cloud *desktop* permet d'installer, puis de supprimer des applications par simple création de liens HTTP. La multiplication des applications n'engorge donc pas le poste utilisateur et ne le rend pas instable, contrairement au bureau classique. Le cloud *desktop* est accessible depuis n'importe quel terminal, en particulier depuis un appareil mobile.

Enfin, les combinaisons possibles d'applications SaaS pour la personnalisation du cloud *desktop* sont infinies. En effet, celles-ci ne sont pas liées à un poste de travail en particulier, à la différence des applications développées spécifiquement pour Windows ou Mac OS.

Google propose avec ChromeOS, un système d'exploitation réduit à un simple navigateur, une première mise en pratique du cloud *desktop*. Ce système est disponible sur des ordinateurs portables appelés Chrome Books et des ordinateurs de bureau appelés Chrome Box.

En résumé

L'usage exclusif d'applications cloud permet de déporter le bureau de l'utilisateur, vers le cloud. Cette approche offre de vastes possibilités de coopération entre applications cloud.

CINQUIÈME PARTIE

Les plateformes de cloud computing

Cette partie aborde les aspects techniques du cloud computing. Elle décrit les architectures sous-jacentes aux plateformes cloud ainsi que leurs particularités.

Elle présente la structuration des PaaS et IaaS. Puis elle décrit les principales plateformes disponibles pour les entreprises qui souhaitent faire héberger leurs applications sur des plateformes cloud.

18

Les architectures du cloud computing

Objectif

Le propos de ce chapitre est de présenter les architectures techniques qui sous-tendent le cloud computing. Ces architectures sont comparées aux plateformes d'entreprises afin de permettre au lecteur de bien saisir leurs spécificités.

Le chapitre évoque aussi les cas d'utilisation de telles architectures.

18.1 CLOUD COMPUTING ET ARCHITECTURES MULTI-TIERS

Les architectures multi-tiers constituent l'état de l'art des architectures d'entreprise. Elles sont utilisées dans les systèmes informatiques depuis environ 15 ans. Cette pratique consiste à segmenter les couches techniques des applications en plusieurs « *tiers* » conçus et exploités de manière autonome. On considère généralement les tiers suivants :

- **le serveur de présentation,** dont le rôle est de produire les écrans visibles par les utilisateurs. Il peut reposer sur diverses technologies de génération d'interface utilisateurs ;

- **le serveur d'application,** dont le rôle est de servir de plateforme d'exécution pour les applications de l'entreprise. Il peut reposer sur un serveur JEE[1] (IBM WebSphere, Oracle Weblogic, Tomcat, etc.), sur la plateforme Microsoft .Net, etc.
- **le système de persistance,** dont le rôle est de stocker et de garantir l'intégrité des données métiers de l'entreprise. Il peut reposer sur une base de données relationnelle (Oracle, Microsoft SQL Server, MySQL, etc.) ou bien sur un serveur de fichiers, une base documentaire, etc.
- **le serveur d'authentification/gestion de droits,** dont le rôle est de fournir des services de sécurité aux applications de l'entreprise. Il peut reposer sur un annuaire LDAP[2], un système de SSO[3], etc.
- **le serveur d'intégration,** dont le rôle est de fournir une passerelle d'échange avec les autres applications de l'entreprise. Il peut reposer sur un middleware orienté messages (MOM), un ESB[4], etc.

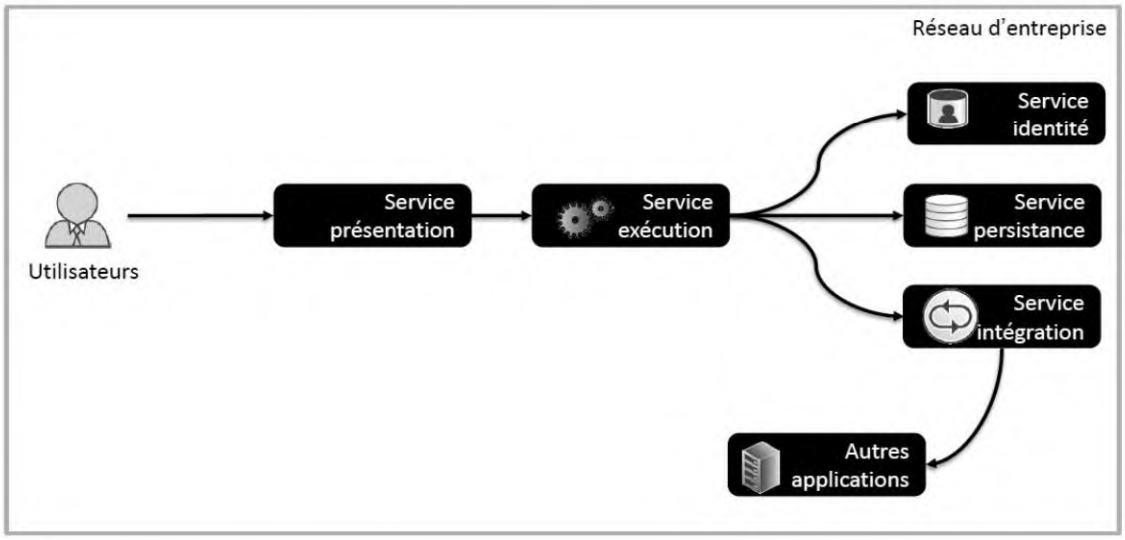

Figure 18.1 — Représentation schématique d'une architecture N-tiers.

Le nombre de tiers peut être plus important pour des applications complexes, mais la liste ci-dessus est représentative de ce type d'architecture.

Les bénéfices d'une architecture N-tiers sont les suivants :

- chaque tiers étant isolé et accédé au travers de protocoles standardisés, il est en théorie possible de les remplacer sans impacter l'ensemble de l'architecture. Ainsi, on pourra décider de remplacer une base Oracle par une base MySQL avec un impact limité. Ce type de pratique était impossible du temps des mainframes ;

1. *Java Enterprise Edition.*
2. *Lightweight Directory Access Protocol.*
3. *Single Sign On.*
4. *Enterprise Service Bus.*

- chaque tiers étant isolé, on peut gérer sa montée en charge de manière autonome. Ainsi, si la base de données peine à tenir la charge, on pourra augmenter ses ressources sans toucher à celle du serveur de présentation ;
- chaque tiers étant isolé, on peut introduire des règles et paramètres de sécurité à chaque interface. Ainsi, on peut définir des systèmes de sécurité différents pour la base de données, le serveur d'application, etc. Cette pratique permet de compliquer la tâche d'un éventuel pirate.

Les architectures N-tiers constituent à ce jour la meilleure solution en matière d'informatique d'entreprise. Elles forment **un premier pas vers l'élasticité**. Il est néanmoins nécessaire de concevoir des architectures spécifiques pour assurer le « scaling horizontal » propre au cloud computing.

Le cloud computing reprend ces bonnes pratiques et les plateformes PaaS sont structurées de manière similaire. Par exemple, les serveurs de présentation/exécution de composant se nomment Web Roles/Worker Roles sur Azure, Dyno/Workers sur Heroku.

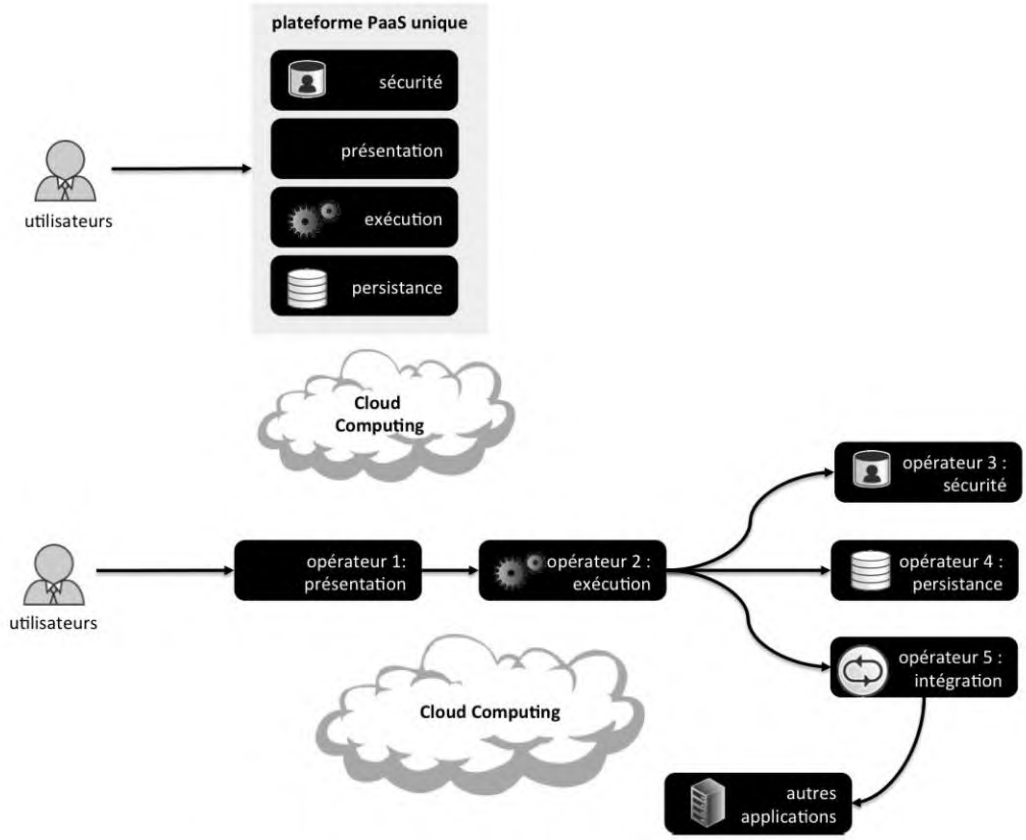

Figure 18.2 — Alternatives N-Tiers du cloud computing.

Avec le cloud, les tiers peuvent aussi être utilisés selon un mode distribué sur Internet. En effet, il est possible de reposer sur une plateforme PaaS unique, mais il est aussi possible d'utiliser (*cf.* figure 18.2) :

- un service de persistance proposé par une première PaaS ;
- un service d'exécution d'application proposé par une seconde PaaS ;
- un service d'intégration proposé par une troisième PaaS ;
- etc.

Dans ce scénario, il faudra bien entendu anticiper **les problématiques de temps de latence** : par exemple, la mise à jour de données depuis le service d'exécution vers le service de persistance pourra prendre quelques secondes. Il faudra aussi bien gérer la sécurité de bout en bout car les tiers communiqueront entre eux *via* Internet. Ce scénario peut malgré tout avoir sa pertinence dans une approche *best of breed*.

À propos de l'IaaS

L'IaaS présente une offre de plus bas niveau, qui propose des services système et réseau. Le propos ci-dessus ne s'applique donc pas. Nous reviendrons plus loin sur la structuration de l'IaaS.

18.2 CLOUD COMPUTING ET ARCHITECTURES DE SERVICES

Le propos de ce paragraphe est un parallèle entre le cloud computing et les architectures de service : SOA, REST, API ouvertes, micro-services... On a vu dans la première partie que les acteurs du web et du cloud ont un usage fréquent de services sous la forme d'API.

Le propos s'applique surtout au SaaS, qui fournit des services fonctionnels, un peu au PaaS qui fournit des services d'identité, et très peu à l'IaaS qui fournit essentiellement des services de configuration à distance.

Par le passé, les applications informatiques ont presque toujours été conçues sans anticiper leurs échanges avec des tiers. Cette démarche de conception isolationniste a produit des « silos », c'est-à-dire des applications monolithiques et peu communicantes. La problématique d'intégration interapplicative est un vieux défi de l'informatique. On a tenté de le résoudre avec diverses technologies : batch pilotés par des scripts, échanges de fichiers par FTP, *middleware* orientés message, EAI, etc.

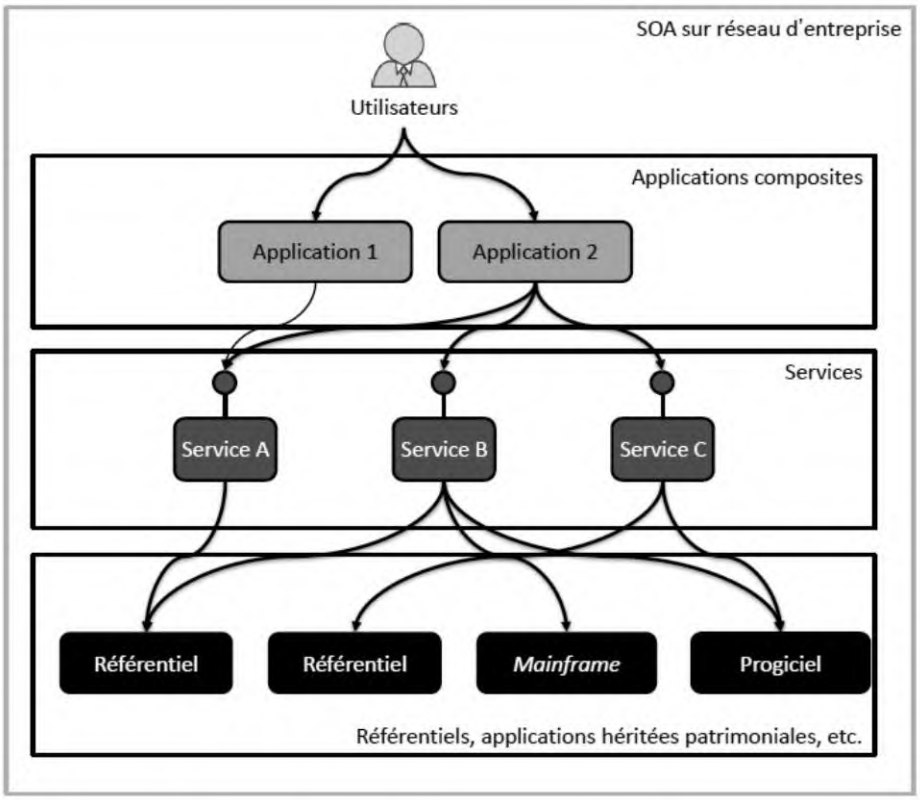

Figure 18.3 — L'architecture de référence SOA.

Les **architectures orientées services (SOA**[1]**)** offrent un modèle pour résoudre cette problématique. Elles proposent que chaque application mette à disposition ses fonctions et informations métier sous une forme utilisable par les autres : les **services**. Elles incitent également à remettre la gestion des processus métier au centre des enjeux des systèmes d'information, en simplifiant leur implémentation et en favorisant leur évolutivité. Les applications issues de l'assemblage des services sont appelées « **applications composites** ».

Les architectures SOA ont été conçues pour résoudre les problématiques d'architecture d'entreprise. Nous allons voir comment les applications cloud portent le modèle sur Internet. Les applications cloud reprennent le modèle d'architecture de service. En effet, elles exposent bien des services intégrables par des tiers. En revanche, elles utilisent généralement la sémantique issue du web 2.0 : les services sont intitulés « API ». De plus, elles privilégient les architectures de services simplifiées, suivant le style REST.

Le cloud computing reprend donc du modèle SOA et des pratiques du web 2.0. Il suit ainsi l'état de l'art en matière d'architecture distribuée.

1. *Service Oriented Architecture*.

Dans le monde du cloud computing, l'intégration s'appuie essentiellement sur REST. Ce style d'architecture est différent des modèles d'entreprises :

- le **temps de latence réseau** est à prendre en compte ;
- il ne permet pas les échanges asynchrones ;
- il n'offre pas de garantie transactionnelle (du fait des limites de HTTP) ;
- il pose des **problèmes de sécurité** liés à une architecture ouverte.

À propos de REST[1]

REST signifie *Representational State Transfer*. Le sens de cet acronyme n'est aujourd'hui plus très significatif. Il ne s'agit pas d'une technologie mais d'un style d'architecture, c'est-à-dire de pratiques, de partis pris en matière d'implémentation.

REST se définit par opposition aux architectures basées sur les Web Services, jugées trop complexes pour des usages simples. La philosophie de REST consiste à tirer parti des verbes de base du protocole HTTP. Ce protocole permet en effet de lire, mettre à jour ou supprimer des ressources sur le web. REST utilise donc tout simplement les fonctions CRUD[2] de HTTP plutôt que de passer par l'invocation de Web Services. REST utilise également les URL pour identifier les ressources. On parle parfois d'architecture orientée ressource (ROA, *Ressource Oriented Architecture*).

Ce principe connaît des limites lorsqu'on doit gérer des scénarios sophistiqués de sécurité, de contrôle transactionnel et de gestion des échanges asynchrones. Cependant, il est suffisant dans des configurations simples, pour des applications web. Il est ainsi largement utilisé par les acteurs du web 2.0 comme Google, Amazon, etc.

18.3 LES ARCHITECTURES « MULTI-TENANT »

La terminologie « multi-tenant » (expression anglaise qui signifie multilocataires) s'appuie sur une métaphore : celle des bailleurs et des locataires. L'intérêt du bailleur est de fournir un service identique à tous ses locataires. La mutualisation totale est intéressante pour lui car elle permet une rationalisation en termes de :

- souscription en ligne sur la base d'une offre totalement standard. On parle de « Customer as a commodity » pour dire que tous les clients sont tous identiques aux yeux de l'opérateur cloud ;
- automatisation de l'exploitation des applications (surveillance et monitoring) ;
- automatisation des mises à jour de composants et services (montées de version) ;
- réduction des coûts liée à des infrastructures mises en commun (la facture d'électricité d'une famille de cinq personnes est généralement inférieure à la somme des factures de cinq célibataires).

1. Pour des précisions techniques sur le style REST, voir Fournier-Morel X., Grojean P., Plouin G., Rognon C. SOA. *Le guide de l'architecte d'un SI agile*, 3e édition, Dunod, 2011.

2. CRUD signifie *Create, Read, Update, Delete*. Les commandes CRUD de HTTP sont HTTP GET, HTTP PUT, HTTP UPDATE, HTTP DELETE.

A contrario, la customisation permet de satisfaire les besoins spécifiques de chaque entreprise, lorsque les besoins ne sont pas totalement génériques.

Pour illustrer notre métaphore, on peut donner cette lecture :

- **Dans le cadre de l'externalisation selon le modèle « hébergeur »,** chaque entreprise dispose d'un environnement complètement indépendant chez l'hébergeur. L'application et les machines nécessaires à son exploitation sont purement et simplement déplacées chez l'hébergeur qui les exploite dans des armoires de serveurs indépendantes. Il y a donc très peu de mutualisation. On peut donc comparer cette approche à la création d'un lotissement immobilier où toutes les maisons sont autonomes.
- **Dans le cadre de l'utilisation d'une application web classique,** les ressources sont totalement mutualisées entre les utilisateurs. L'application est unique, non adaptable aux entreprises utilisatrices. On peut donc comparer cette approche à la création d'un dortoir de pensionnat où tout est partagé entre les étudiants.
- **Dans le cadre d'une application cloud,** la plupart des ressources sont partagées entre les entreprises utilisatrices, comme la plateforme matérielle, ou le système de persistance. Mais elles peuvent bénéficier d'un certain degré de customisation. Elles peuvent utiliser leurs logos, leurs noms de domaines, paramétrer finement les applications, publier leur propre code. On peut donc comparer cette approche à la création d'un immeuble de location où certaines ressources sont partagées (eau, électricité, accès à Internet, gardiennage, buanderie, piscine, etc.) et d'autres non (salle de bain et cuisine dans chaque appartement).

L'enjeu des architectures multi-tenant est donc de trouver le juste milieu entre mutualisation et possibilités de customisation.

Ainsi les opérateurs de plateformes de cloud computing cherchent tous le meilleur compromis entre rationalisation et possibilité de customisation. Chacun a sa propre stratégie et son propre modèle pour atteindre cette cible. Nous verrons quelques-unes de ces stratégies dans le chapitre 20.

Le tableau 18.1 indique les degrés de mutualisation selon les types de cloud.

Tableau 18.1 — La mutualisation des ressources dans le cadre du cloud computing.

Plateforme	IaaS	PaaS	SaaS
Applications	-	-	☑
Environnement exécution	-	☑	☑
Base de données	-	☑	☑
Système d'exploitation	-	☑	☑
Hyperviseur	☑	☑	☑
Stockage	☑	☑	☑
Machines	☑	☑	☑
Réseaux	☑	☑	☑

18.4 SPÉCIFICITÉS DES PLATEFORMES CLOUD VIS-À-VIS DES PLATEFORMES D'ENTREPRISE

18.4.1 Une différence de structuration

Comme on l'a évoqué dans la deuxième partie, les plateformes cloud reposent sur des Datacenters de grande envergure, comprenant des dizaines voire des centaines de milliers de serveurs. Ce gigantisme leur offre de grandes capacités d'exécution et de stockage. Ils sont donc capables de gérer des applications à forte charge utilisateurs, ou nécessitant beaucoup d'espace de persistance (médias : images, audio, vidéo). **Ils permettent en particulier de répondre à la promesse d'élasticité du cloud.**

Ces Datacenters utilisent les principes de la virtualisation, ce qui leur permet d'attribuer une grande capacité de traitement à une application donnée par simple réaffectation des ressources. Cette réaffectation est généralement effectuée de **manière automatique**, sans intervention humaine. Les opérateurs cloud ont pour cela développé des outils d'automatisation dont bien peu d'entreprises disposent. **Cette allocation automatique permet de répondre à la promesse du Self Service**. Ainsi, un ensemble de machines peut être alloué dynamiquement à une application en cas de pic d'activité, et désalloué dynamiquement à la fin de ce pic d'activité : cette fonction est appelée « auto-scaling » dans l'offre Amazon.

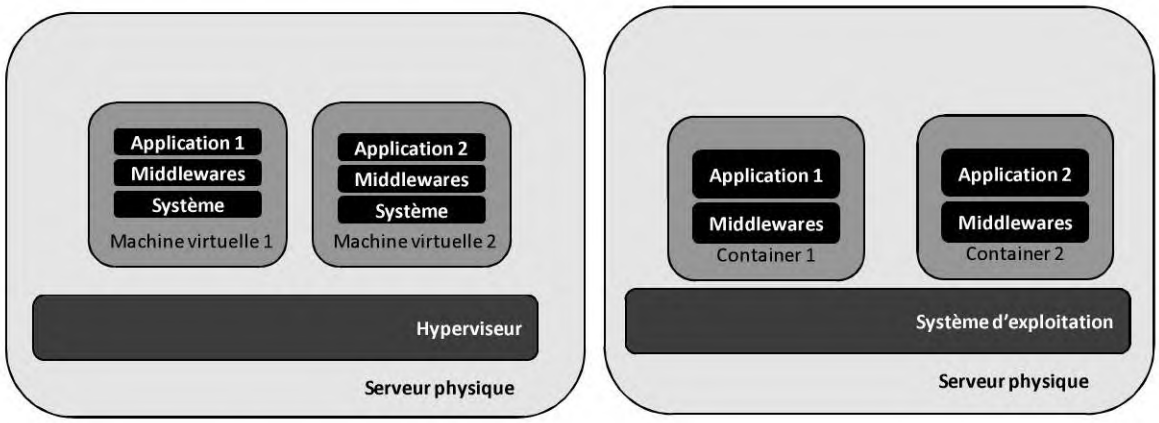

Figure 18.4 — La virtualisation basée sur hyperviseur ou container.

À propos de la virtualisation

La virtualisation permet de masquer la nature physique des ressources utilisées. Elle fournit une couche d'abstraction entre l'application et les serveurs physiques. Elle vise à augmenter la flexibilité de l'infrastructure en l'affranchissant de certaines limites physiques.

Ces solutions sont utilisées dans les DSI car elles permettent de rationaliser les coûts d'infrastructures matérielles en répartissant au mieux leurs ressources en fonction des besoins applicatifs.

La virtualisation la plus courante est celle qui repose sur des hyperviseurs. Les offres VMware ESX, Microsoft Hyper V, Xen, ou KVM sont les plus intéressantes sur les machines en architectures X86.
Un autre type de virtualisation est en train de monter en puissance : basées sur les containers. Les containers sont des environnements d'exécution qui embarquent les middlewares (serveur HTTP, bases de données...) et le code applicatif, mais pas le système d'exploitation. Ils sont exécutés sur des systèmes d'exploitation mutualisés. On parlera de la solution de container Docker au chapitre 19.

On ne connaît pas bien l'architecture détaillée des plateformes cloud, car cette information est jalousement gardée par leurs opérateurs : elle constitue le cœur de leur savoir-faire. Elles fournissent des interfaces de plus ou moins haut niveau, qui permettent de les utiliser sans entrer dans les détails de leur architecture. Ces interfaces, toutes spécifiques, demandent un effort d'apprentissage aux entreprises utilisatrices.

A contrario, les architectures d'entreprise sont très génériques : les plateformes Java/JEE et .NET et les architectures matérielles sous-jacentes sont bien connues des sociétés de services et des équipes d'exploitation des entreprises utilisatrices. De fait, elles sont plus simples à prendre en main.

Néanmoins, les architectures d'entreprise sont peu élastiques : une ou plusieurs machines sont affectées de manière figée à chaque fonction de l'entreprise. Par conséquent, il est fréquent que :

- les ressources soient surdimensionnées en gestion courante (en particulier, il est fréquent que 50 % de la capacité des disques durs soit inutilisée) ;
- et que ces ressources se révèlent insuffisantes lors de pics d'activités.

18.4.2 Le théorème de CAP

Le principe ou théorème de CAP donne un bon éclairage sur les problématiques complexes que gèrent les plateformes cloud, et sur leurs différences vis-à-vis des plateformes d'entreprise (on verra dans la suite que cloud et entreprise ne font pas les mêmes choix vis-à-vis de ce théorème).

L'acronyme CAP signifie *Consistency, Availability and Partition-Tolerant*. On peut expliquer ces termes de la manière suivante :

- **la Consistency**, ou *consistance*, signifie que tous les nœuds du système voient exactement les mêmes données au même moment ;
- **l'Availability**, ou disponibilité, concerne la capacité du système à offrir un temps de réponse optimal aux utilisateurs ;
- **la Partition-tolerance**, ou tolérance, concerne la capacité du système à fonctionner si le réseau reliant ses composants tombe en panne.

Le théorème de CAP énonce la propriété suivante :
« **Dans une architecture distribuée, il n'est possible de garantir que deux des trois propriétés CAP.** »

Ce théorème a été proposé par Eric Brewer, professeur à l'université de Berkeley.

Les plateformes cloud privilégient généralement les propriétés A et P, c'est-à-dire disponibilité et tolérance aux pannes. La consistance est garantie *in fine*, quelques fractions de seconde après la transaction.

A contrario, les clusters de serveurs dans les grandes banques garantissent la consistance des transactions et la haute disponibilité (A et C). Il n'est pas envisageable de tolérer une erreur dans une écriture bancaire. Mais, si un composant réseaux tombe en panne, le système peut en pâtir.

Le théorème de CAP souligne les différences fondamentales entre architectures d'entreprise et cloud computing. Ces dernières doivent en effet prendre en compte un risque de défaut de consistance pendant un temps court.

Les plateformes cloud manipulent de très grosses volumétries de données (*cf.* l'exemple de YouTube chez Google). Ces données doivent être partitionnées sur de nombreux serveurs, d'une part car leur taille interdit le stockage sur un simple cluster de base de données, d'autre part pour assurer leur intégrité grâce à la réplication. Elles doivent aussi être accessibles très rapidement en lecture comme en écriture. Ces besoins ne peuvent pas assurer/utiliser des mécanismes habituels des bases de données relationnelles en environnement distribués : le « *two phase commit* ».

Ainsi les plateformes cloud ont adopté des mécanismes de persistance inhabituels en entreprise. Ces mécanismes utilisent souvent des transactions optimistes et sont « **faiblement consistants** » par choix de privilégier les propriétés A et P. Il existe des exceptions à cette généralité : BigTable de Google privilégie les propriétés C et P.

18.4.3 Les systèmes de stockage NoSQL

NoSQL signifie *Not Only SQL* : les systèmes dits NoSQL proposent donc une alternative aux bases de données relationnelles, sans les remettre en cause. Ils suggèrent que d'autres modèles de persistance sont possibles hors tables relationnelles et hors transactions ACID.

D'autres systèmes plus anciens avaient déjà exploré ces voies :

- annuaires LDAP : non relationnels et non transactionnels ;
- bases de données objets : non relationnelles ;
- bases de données XML : non relationnelles ;
- etc.

Les systèmes NoSQL ont été conçus par les acteurs du cloud (Amazon et Google) pour répondre à leurs besoins spécifiques :

- haute disponibilité et reprise sur incident *via* réplications multiples ;
- systèmes décentralisés : capacité de fonctionnement autonome des sous-systèmes ;
- modèles de données évolutifs « à chaud » ;

- élasticité : capacité à ajouter ou à supprimer des serveurs « à chaud » ;
- grande performance en lecture et en écriture ;
- gestion de volumétries importantes de données ;
- la contrepartie de ces nouvelles propriétés est le déport vers la couche applicative du contrôle d'intégrité, de la gestion des relations entre entités. En effet, les systèmes NoSQL offrent moins de « services sur étagère » que les systèmes SQL.

Pour citer quelques exemples de systèmes NoSQL : Google a développé BigTable, Amazon a développé Dynamo, Facebook a développé Cassandra, LinkedIn a développé Voldemort, etc.

En complément des systèmes de stockage NoSQL, un certain nombre d'acteurs du cloud utilisent le pattern d'architecture Map/Reduce. Map/Reduce permet les calculs parallèles et distribués, sur des données à forte volumétrie. Il permet par exemple à Google de traiter l'indexation de son moteur de recherche, avec le calcul du PageRank (calcul du nombre de pages web qui pointent vers une page donnée). Il existe une déclinaison Open Source de Map/Reduce : Hadoop. Ce composant est très utilisé par les acteurs du cloud. Il est d'ailleurs proposé aux entreprises sous forme de service cloud par Amazon, Google, Microsoft.

18.4.4 Les modèles d'écosystèmes cloud

Marc Andreessen, ancien fondateur de Netscape, distingue trois types d'écosystèmes pour les plateformes cloud :

- **niveau 1 « Access API » :** ces plateformes fournissent une API permettant l'appel à un traitement métier sans fourniture d'interface homme/machine. Exemples : recherche de livres chez Amazon, geocoding chez Google ;
- **niveau 2 « Plug-In API »** : ces plateformes permettent d'intégrer une application à l'interface du fournisseur. Exemple : les applications de la Google Apps Market Place ;
- **niveau 3 « Runtime Environment »** : ces plateformes fournissent non seulement une API, une interface, mais aussi un environnement d'exécution. Exemple : les applications créées sur la plateforme Salesforce et accessibles depuis AppExchange.

Tableau 18.2 — Les typologies d'écosystèmes cloud.

Plateforme	Exécution	Intégration à l'IHM du fournisseur	Traitement
Runtime Environment	☑	☑	☑
Plug-In API	-	☑	☑
Access API	-	-	☑

Les plateformes cloud proposent ainsi divers modèles de consommation de services, correspondant à des modèles économiques très variés.

18.5 USAGE DES PLATEFORMES PAAS ET IAAS

Il existe deux grands types d'utilisateurs des plateformes de cloud computing : les entreprises installées et les start-up.

18.5.1 Usages par les entreprises installées

Les entreprises installées sont susceptibles d'utiliser les IaaS comme une plateforme d'hébergement pour tous types d'applications dès lors qu'ont été traitées les problématiques juridiques et de confidentialité.

Elles pourront à long terme décider d'externaliser leurs développements spécifiques vers des PaaS, en particulier leurs applications métiers. Ce type de stratégie peut se justifier si l'on souhaite réduire au maximum les charges d'exploitation. Les entreprises y viendront dans un second temps, lorsqu'elles auront acquis suffisamment d'expérience et de confiance dans ces plateformes.

Rappelons que **les plateformes PaaS offrent le meilleur *Time to market***, car l'architecture, les services de déploiement et de production sont fournis sur étagère. A contrario, l'IaaS nécessite de construire son architecture applicative et de gérer son déploiement et son exploitation. Cependant **l'IaaS est la seule solution dès lors qu'on souhaite une architecture non standard, et donc indisponible sur PaaS**.

Comme on a commencé à l'évoquer au chapitre 7, le recours à ces plateformes est particulièrement pertinent pour servir certains scénarios :

- **ajustement :** l'entreprise ne sait pas anticiper la charge d'un service, elle souhaite donc le lancer sur une plateforme élastique pour connaître sa charge en « vitesse de croisière ». Elle pourra éventuellement le ré-internaliser par la suite ;
- **débordement :** l'entreprise connaît un pic de trafic ponctuel (par exemple en fin de mois, ou en fin d'année). Elle souhaite gérer ce pic avec des ressources louées, plutôt que d'investir dans des machines qui seront éteintes la plupart du temps ;
- **calcul ponctuel consommateur en ressource :** sur certains types de traitement, les architectures cloud peuvent s'avérer plus efficaces que celle de l'entreprise (exemple : service de calcul GPU[1] fourni par Amazon) ;
- **haute disponibilité :** l'entreprise veut fournir un service offrant une très haute disponibilité (exemple : SLA de 99,95 %). Cette disponibilité aurait un coût prohibitif si le service était géré en interne, le cloud permet de l'offrir à un coût raisonnable ;
- **plan de reprise d'activité (PRA) :** l'entreprise utilise une plateforme cloud comme système de reprise d'activité en cas de sinistre grave. Ses applications et données sont répliquées sur la plateforme cloud, qui deviendra maître lors du sinistre ;

1. *Graphical Process Unit.*

- **besoin de forte intégrité :** l'entreprise souhaite disposer d'une très forte garantie d'intégrité sur certaines données, *via* une réplication intercontinentale. Une plateforme cloud permet d'assurer ce service à moindre coût ;
- **usine de développement et environnements :** les plateformes cloud permettent de déployer rapidement les environnements techniques nécessaires à la construction d'applications ;
- **bac à sable innovation :** les plateformes cloud sont idéales pour tester des concepts applicatifs dont on ignore la pérennité ;
- **tests en charge :** ce type de test doit être effectué lorsque le développement d'une application a été terminé et que l'on souhaite vérifier sa capacité à supporter un nombre suffisant d'utilisateurs. Les plateformes cloud constituent un socle idéal pour ce type de test : en effet, elles permettent d'avoir des machines à disposition pendant une période courte correspondant à la durée du test ;
- etc.

18.5.2 Start-ups

Les start-ups doivent généralement faire face aux problématiques suivantes : **lancer un service rapidement,** pour devancer la concurrence et conquérir des utilisateurs, **à moindre coût**, car leur succès est incertain**, gérer la montée en charge du service** liée à la croissance de son utilisation.

Dans ces situations, les plateformes cloud se révèlent intéressantes. En effet, elles permettent de mettre en production un service sans achat de machine, ni compétence sur les problématiques matérielles. De plus, si le service ne fonctionne pas, le fournisseur n'aura qu'à stopper son abonnement et il n'aura pas investi inutilement en CAPEX.

Par ailleurs, si le service connaît un succès croissant, il sera simple d'augmenter les ressources *via* l'interface de Self Service. Et ses dépenses suivront ses rentrées d'argent. Si l'architecture a été pensée pour un « scaling horizontal »[1], elle ne sera pas remise en cause. Les plateformes cloud constituent donc un socle idéal pour gérer la croissance d'un nouveau service.

Un certain nombre de start-up l'ont compris : on peut citer l'exemple de SlideShare qui utilise la plateforme Amazon ou de Github qui utilise RackSpace.

Enfin, les plateformes cloud proposent parfois un annuaire des services qu'elles hébergent (*cf.* Salesforce AppExchange ou Google Apps Market Place). Cet annuaire peut aider les nouveaux services à se faire connaître.

Le PaaS intéresse beaucoup les startups car il propose un scaling horizontal sur étagère, il permet la mise en ligne la plus rapide et permet de se passer d'équipe d'exploitation.

1. Capacité à augmenter la puissance d'une plateforme par ajout de serveurs.

En résumé

Les architectures de cloud computing reprennent certaines bonnes pratiques des architectures d'entreprise : multi-tiers et SOA. Elles ont cependant des spécificités marquées. Si elles offrent une grande souplesse dans l'allocation des ressources à coût réduit, elles pâtissent de problématiques de latence réseau et d'intégrité transactionnelle.

Elles sont très adaptées à des applications à fort trafic ou à des applications à trafic variable.

19

Les composants des plateformes PaaS et IaaS

Objectif

L'objectif de ce chapitre est d'ouvrir le capot d'une plateforme de cloud computing et de présenter les composantes d'une plateforme PaaS ou IaaS.

19.1 LES DATACENTERS

19.1.1 Principes

Les plateformes cloud reposent sur un ensemble de centres de données, dont le nombre est presque toujours supérieur à trois ; dans les cas de Microsoft ou de Google, les centres se comptent en dizaines. Cette redondance permet d'assurer la reprise de l'activité en cas de sinistre (incendie, inondation, etc.) : soit au sein d'un même site, soit entre plusieurs sites. En général, les fournisseurs de plateformes cloud maîtrisent et gèrent eux-mêmes leurs Datacenters, gardant leur architecture secrète.

Les plateformes cloud proposent de manière quasi standard un niveau de disponibilité de 99,9 %.

Les standards de Datacenters

La TIA (*Telecommunications Industry Association*) propose un modèle normalisé pour qualifier les Datacenters, les TIER 1, 2, 3 et 4.

– Le TIER I garantit 99,671 % de disponibilité (28,8 heures d'indisponibilité/an). Il dispose d'une seule source d'alimentation électrique. Il n'a pas de composant redondant.
– Le TIER 2 garantit 99,741 % de disponibilité (20 heures d'indisponibilité/an). Il dispose d'une redondance partielle.
– Le TIER 3 garantit 99,982 % de disponibilité (1,6 heure d'indisponibilité/an). Ses systèmes sont en double alimentation électrique. Tous ses composants sont redondants.
– Le TIER 4 garantit 99,995 % de disponibilité (0,8 heure d'indisponibilité/an). Il est totalement redondant, tolérant à la panne. Il dispose d'une alimentation en énergie en mode actif/actif.
Les grands acteurs du cloud utilisent des Datacenters de TIER 4 reposant sur des containers.

À titre de comparaison, les grandes entreprises déploient généralement des centres de données redondants dans le cadre de leur PRA (plan de reprise d'activité). Mais, elles utilisent souvent les services d'hébergeurs et le nombre de leurs centres dépasse rarement trois, pour des raisons évidentes de coûts.

Le site datacenterknowledge.com publie régulièrement des chiffres sur les plateformes des opérateurs cloud. Ces chiffres sont parfois des hypothèses, car tous les acteurs ne communiquent pas de manière transparente :

- Google : entre un et deux millions de serveurs ;
- Microsoft : plus d'un million de serveurs ;
- Amazon : 450 000 serveurs ;
- OVH : 220 000 serveurs ;
- Facebook : 180 000 serveurs ;
- Akamai Technologies : 100 000 serveurs ;
- Rackspace : 94 000 serveurs.

Amazon, Google et Microsoft ont des sites dédiés à la présentation de leurs datacenters :

- aws.amazon.com/fr/about-aws/global-infrastructure/
- www.google.com/about/datacenters/
- azure.microsoft.com/fr-fr/regions/

19.1.2 Architecture matérielle

Les plateformes cloud sont aujourd'hui fabriquées de manière industrielle. Les opérateurs cloud conçoivent et fabriquent parfois eux-mêmes leurs serveurs sur la base de composants élémentaires : processeur, disque dur... C'est le cas de Google.

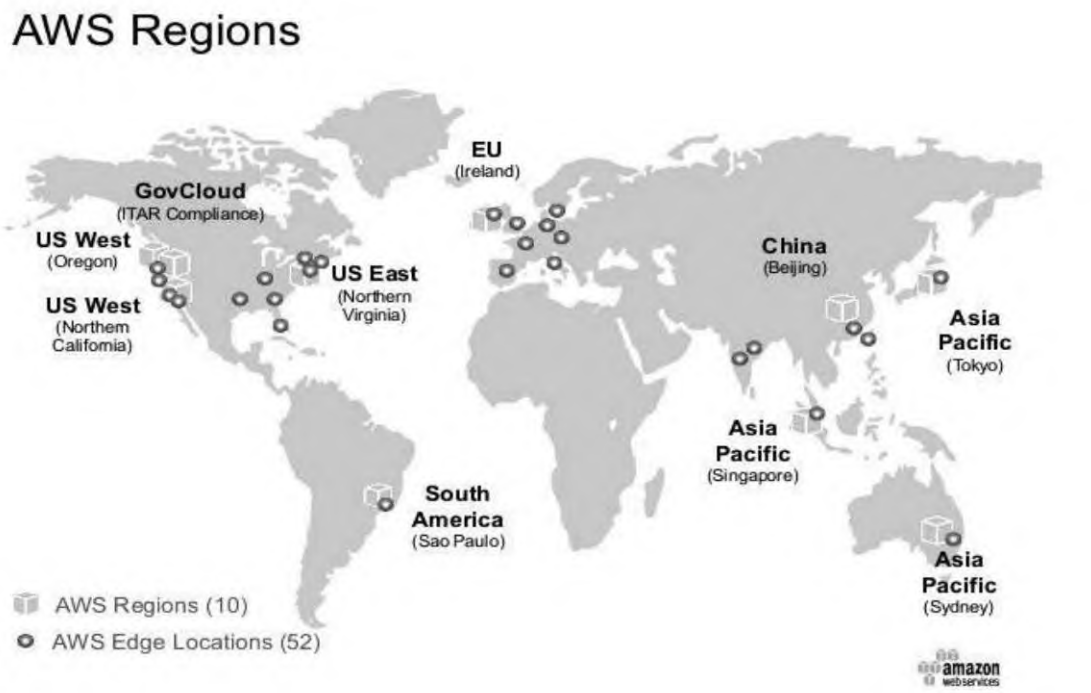

Figure 19.1 — La topologie d'Amazon Web Services

Le « commodity hardware »

Les opérateurs cloud utilisent des serveurs d'entrée de gamme (commodity hardware). Le stockage des données est effectué sur les disques durs des serveurs eux-mêmes, et non dans des réseaux de stockage de type SAN[1]. Ces disques durs n'utilisent pas de redondance RAID[2]. L'intégrité des données est assurée par les logiciels d'infrastructure et non par le matériel (*cf. Design for failure*). Cette approche fait suite à des études de ROI qui ont montré que ce type d'architecture est moins onéreux pour des datacenters d'envergure. Notons que ces solutions commencent à intéresser les entreprises et que VMware les a ajoutées à son catalogue d'offre.

Certains opérateurs procèdent par assemblage de containers (*cf.* les TIER 4) intégrant des milliers de serveurs et un système de climatisation, et qui offrent juste des interfaces réseaux et des prises de courant. Ces containers sont optimisés en termes d'encombrement et de circulation des flux d'air, à tel point qu'il est impossible de changer un composant d'un serveur en panne ; la philosophie étant que l'on change le container lorsque tous ses composants sont en panne.

1. *Storage Area Network*.
2. *Redundant Array of Independent Disks*.

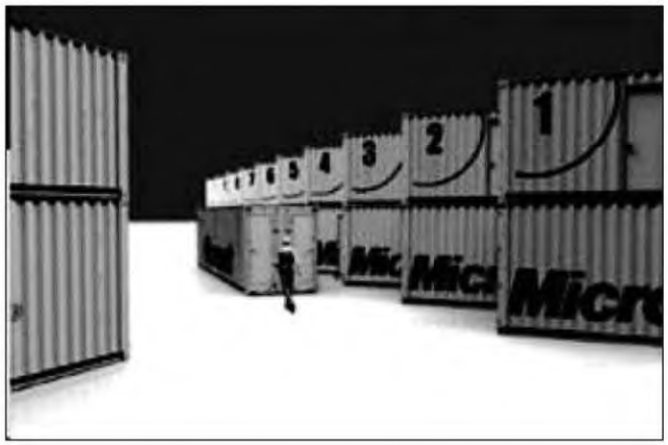

Figure 19.2 — Les containers C-Blox de Microsoft.

Le projet Open Compute

Lancé par Facebook, ce projet vise à partager de bonnes pratiques sur l'assemblage de cartes mères, de serveurs, de composants réseaux, de châssis, de datacenters... La documentation est en ligne sur opencompute.org. On a vu au chapitre 13 que HP l'utilisait pour concevoir ses serveurs. De plus en plus d'acteurs souhaitent en effet bénéficier des retours d'expérience des géants du web et du cloud.

19.1.3 Architecture logicielle

Les plateformes cloud reposent donc sur des grappes de serveurs de grandes envergures. Ces clusters embarquent généralement (à l'exception notable de ceux de Microsoft) des composants Open Source optimisés pour une performance maximale à un coût minimum. La valeur ajoutée d'une plateforme cloud vis-à-vis de ses concurrents repose sur les aspects suivants :

- **haute disponibilité**, pour satisfaire l'expérience client ;
- **tolérance aux pannes** et capacité à se réorganiser dynamiquement en cas de panne d'un composant ;
- **optimisation des performances** pour tirer le meilleur parti des ressources matérielles ;
- **automatisation des tâches d'exploitation et de maintenance.**

Le « design for failure »

Les plateformes cloud intégrant une grande quantité de serveurs considèrent comme inévitable, voire normal, que des composants serveurs tombent en panne régulièrement. Leurs concepteurs savent qu'il est statistiquement inévitable de perdre un (ou plusieurs) serveur(s) chaque jour. Les plateformes cloud sont donc conçues pour ne pas souffrir de ces pannes : les données sont répliquées sur plusieurs disques et lorsque l'un d'entre eux flanche, elles sont automatiquement recopiées sur un autre

disque opérationnel. Les exécutions d'application sont soumises à la même règle : lorsqu'un serveur flanche, le traitement est repris sur une autre machine.

Les architectes qui conçoivent ces plateformes capables de s'autogérer en cas de panne sont parmi les plus brillants sur le marché de l'informatique : les opérateurs cloud se les arrachent.

19.1.4 Architecture réseau

Les grands opérateurs cloud ont déployé un réseau international de fibres et de cache afin de distribuer leurs services au plus près de leurs utilisateurs. Ils ont des accords de peering avec les opérateurs télécom de tous les continents. Certains ont même des serveurs de cache dans les datacenters d'hébergeurs locaux.

Une partie de ces infrastructures est mise à la disposition de leurs clients. Ainsi Amazon et Microsoft ont des services de CDN dans leurs offres cloud.

Google communique sur son réseau à cette adresse peering.google.com, et une partie de cette infrastructure est mise à la disposition de ses clients, *via* l'offre cloud networking.

19.2 STRUCTURATION DES IAAS

Les plateformes IaaS fournissent principalement deux services :

- un hyperviseur, environnement d'exécution pour des machines virtuelles (VM) ;
- un système de stockage.

Les VM contiennent toute la pile applicative : code applicatif, système d'exploitation, serveur d'application, serveur de base de données, etc. Leur conception et implémentation sont à la charge de l'entreprise utilisatrice. Cependant, les IaaS proposent généralement un catalogue de VM sur étagère, prêtes à l'emploi. Les entreprises pourront les utiliser telles quelles, ou bien les étendre pour construire leur pile applicative. Le paramétrage des VM se fait généralement en ligne de commande *via* le protocole SSH (*Secure Shell*) pour Linux, et RDP (*Remote Desktop Protocol*) pour Windows.

L'entreprise a donc la liberté de choisir la technologie qui lui convient pour exécuter son application : Java, .Net, Python, PHP, etc. pour l'exécution ; Oracle, MySQL, PostGreSQL, etc. pour la persistance.

Les IaaS utilisent chacune une technologie de virtualisation propre à leur plateforme : il n'y a malheureusement pas de standard de virtualisation à l'échelle du cloud, même si l'initiative OpenStack va dans ce sens (*cf.* chapitre 20). Il est donc complexe de migrer une VM d'une plateforme IaaS vers une autre : cela reste parfois possible, car certains opérateurs proposent des utilitaires de conversion.

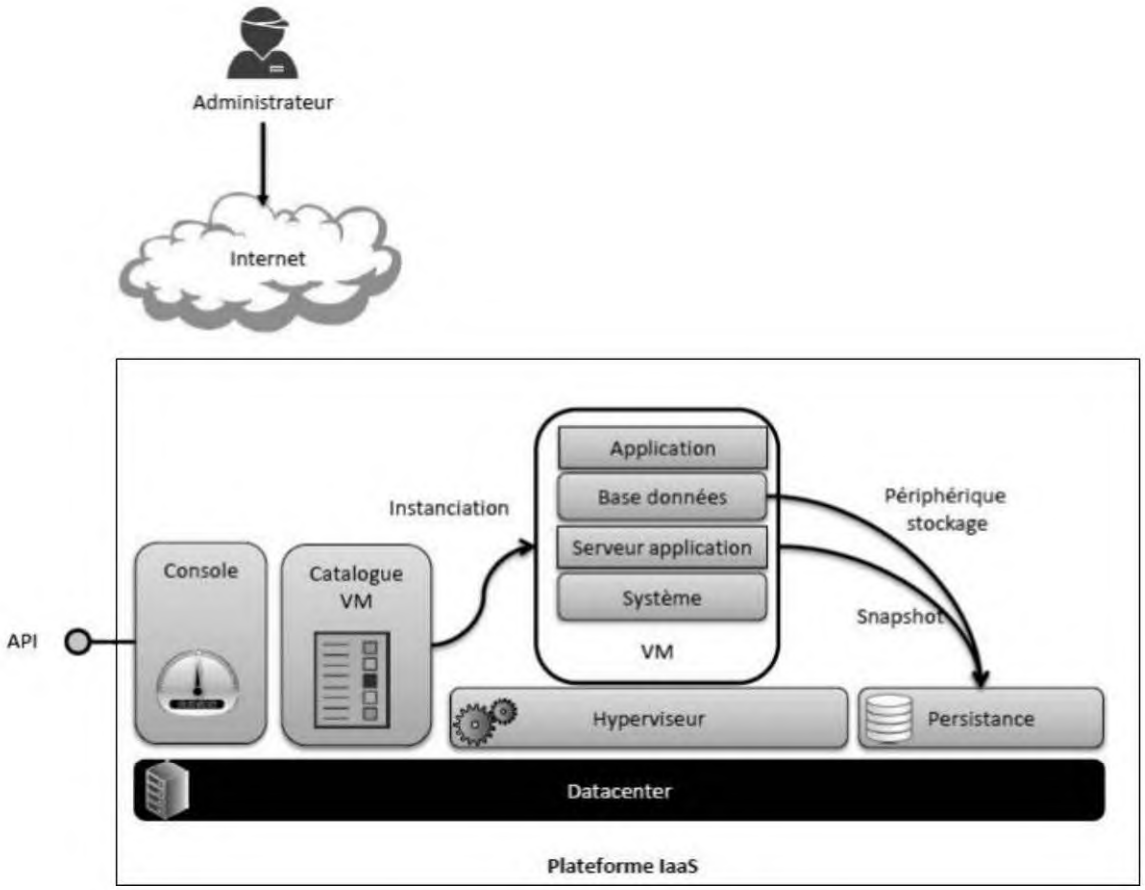

Figure 19.3 — Structuration des IaaS.

Le système de stockage des IaaS répond à deux besoins : du fait du caractère virtuel et non résilient des VM, il est indispensable de faire de temps à autre des sauvegardes (appelées « snapshot ») de leur contenu et configuration ; de plus, il est intéressant de stocker les données applicatives sur un système persistant pour assurer leur intégrité. Pour permettre cette intégrité, les IaaS proposent d'adosser les bases de données virtuelles à des périphériques de stockage persistants. Ainsi le système de persistance est externalisé à la machine virtuelle, afin que les données survivent à une éventuelle extinction de cette dernière.

Les IaaS proposent enfin une console permettant le *Self Service*, le *Pay As You Go* et la gestion du cycle de vie des VM : supervision, monitoring, déploiement, paiement, etc. Ces fonctions sont aussi accessibles *via* des API.

L'optimisation, la maintenance, la surveillance des VM est entièrement à la charge de l'entreprise utilisatrice. L'opérateur IaaS se charge de gérer les couches inférieures, c'est-à-dire la partie matérielle, l'hyperviseur et le stockage.

19.3 STRUCTURATION DES PAAS

Les plateformes PaaS fournissent un environnement d'exécution clef en main permettant aux entreprises utilisatrices de faire tourner leurs applications. Elles fournissent un *runtime* dédié à un sous-ensemble de fonctions dans un langage donné (Java, Ruby, Python, etc.). Elles fournissent un ou parfois plusieurs systèmes de persistance (fichiers, bases SQL, bases NoSQL). La pile logicielle est donc contrainte, et l'entreprise doit produire une application conforme à l'architecture de la PaaS. Elle doit se plier à un certain nombre de contraintes : le langage et les librairies (*frameworks*) fournies par la plateforme, le (ou les) système(s) de persistance proposé(s).

En général, les PaaS proposent deux types de composants applicatifs (les noms varient selon les offres, mais le concept reste le même) :

composant Web : il permet de développer des frontaux web interactifs. Son temps d'exécution maximum est généralement de 30 secondes. Au-delà, il est tué par la PaaS ;

composant Worker : il permet de développer des traitements asynchrones, en tâche de fond (génération d'un document, envoi d'un email, etc.).

L'architecture des plateformes PaaS repose sur des containers instanciés à la demande des clients. Chaque plateforme PaaS a son propre système de containers, dérivé de *Linux Containers* (LXC), de même que chaque plateforme IaaS a son hyperviseur. Ainsi l'architecture est assez opaque. Les choses sont néanmoins en train de changer, car la technologie Docker connaît un grand engouement. **Nous avons la conviction que Docker va créer un engouement pour le PaaS.**

> **À propos de Docker**
>
> Docker est une technologie de containers open source qui fait consensus sur le marché. Docker est supporté par Amazon, Google, Microsoft, VMware, etc. En particulier, Google a fortement investi sur le sujet en écrivant Kubernetes, une solution open source pour orchestrer les containers Docker. Cette technologie pourrait fournir à terme une interopérabilité entre les plateformes PaaS et le SI. Il sera possible d'utiliser le même container sur un poste de développement, un environnement de test, un environnement de production interne et sur le cloud. Dans un premier temps, Docker permettra au moins de rendre les architectures PaaS plus transparentes et de rassurer les entreprises utilisatrices.

Le bénéfice des architectures contraintes est une plus grande rapidité de développement, de la même manière qu'avec les frameworks logiciels : le cadre contraint, mais accélère.

Les PaaS proposent une console permettant le *Self Service*, le *Pay As You Go* et la gestion du cycle de vie des applications : génération de traces, paiement, etc. Ces fonctions sont généralement accessibles *via* des API.

Avec une plateforme PaaS bien utilisée, le développeur peut mettre directement en production : les tâches de déploiement, administration, optimisation sont entièrement automatisées.

Enfin, certaines PaaS proposent un catalogue de plug-ins (envoi de mails, sécurité, etc.) pour accélérer les développements.

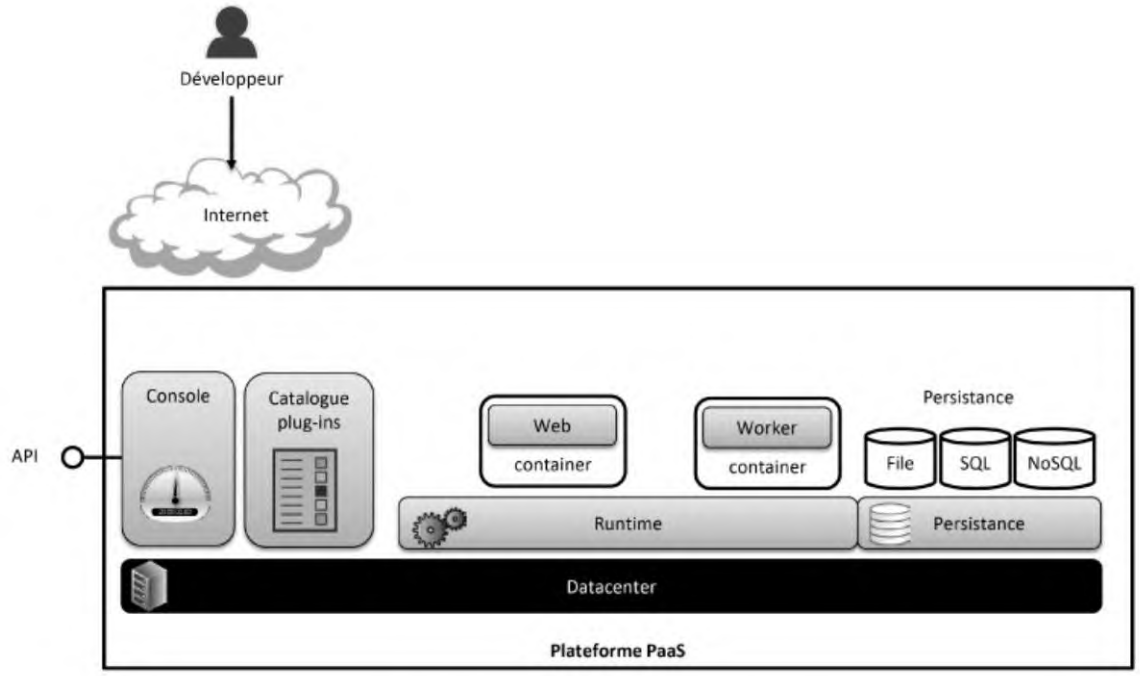

Figure 19.4 — Structuration des PaaS.

19.4 SYSTÈMES DE PERSISTANCE

Les plateformes PaaS fournissent au moins un système de persistance. À l'origine, certaines, comme Google App Engine, avaient fait le choix d'un système NoSQL, tout simplement car c'était le mode de persistance habituel de leur plateforme. Cependant, devant la demande des entreprises de disposer de stockage relationnel, elles ont proposé cette seconde option. D'autres plateformes, comme force.com ont commencé par le stockage relationnel, et ont proposé le NoSQL par la suite (après le rachat d'Heroku).

Ainsi, il semble aujourd'hui consensuel de proposer plusieurs modèles de stockage clef en main au sein d'une plateforme PaaS.

Certaines plateformes IaaS proposent aussi des services de stockage pour les entreprises qui ne souhaitent pas gérer la persistance au sein de leurs VM, comme Amazon Web Services avec SimpleDB et RDS.

Les puristes diront que ces services sont du niveau PaaS, car ils ne nécessitent pas d'installation, et que leur prise en main est possible directement pas les développeurs. Certaines plateformes IaaS proposent donc des services de niveau PaaS pour faciliter le travail de leurs clients.

19.5 LE SERVICE D'AUTHENTIFICATION

Les plateformes PaaS permettent l'authentification des utilisateurs au travers de diverses stratégies de sécurité. Il est toujours possible de reposer sur un système de persistance SQL ou NoSQL, et de stocker des identifiants/mots de passe dans une table. Il est parfois aussi possible de reposer sur un annuaire de sécurité (de type LDAP) intégré à la plateforme. Enfin, certaines PaaS proposent la possibilité de déléguer l'authentification à l'annuaire de l'entreprise, selon les principes de la fédération d'identité (*cf.* chapitre 9).

Dans le cadre de l'IaaS, Amazon propose IAM, une gestion des identités pour les administrateurs de la plateforme. Amazon propose aussi Directory Services, un annuaire utilisateurs, mais nous considérons que c'est un service PaaS.

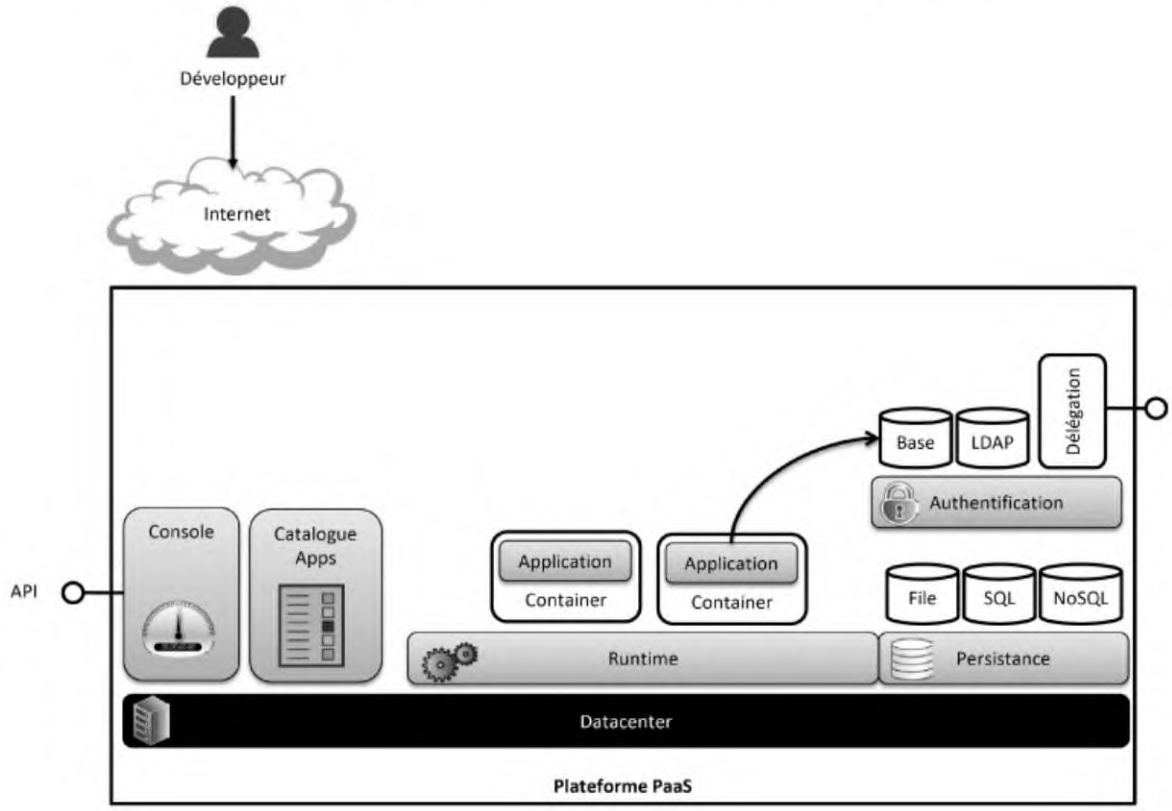

Figure 19.5 — Stratégies d'authentification des PaaS.

19.6 LE BUS D'INTÉGRATION

Il est fréquent que les applications hébergées sur les plateformes cloud aient besoin d'échanger des données avec la plateforme d'entreprise. Les IaaS et PaaS doivent donc offrir un système d'intégration. Ce système peut prendre plusieurs formes :

- une simple API permettant de lire et d'écrire des données sur la plateforme depuis le système d'information d'entreprise ;

- un vrai bus d'intégration capable de piloter les échanges avec le SI d'entreprise en modes synchrone, asynchrone ou suivant un processus d'intégration.

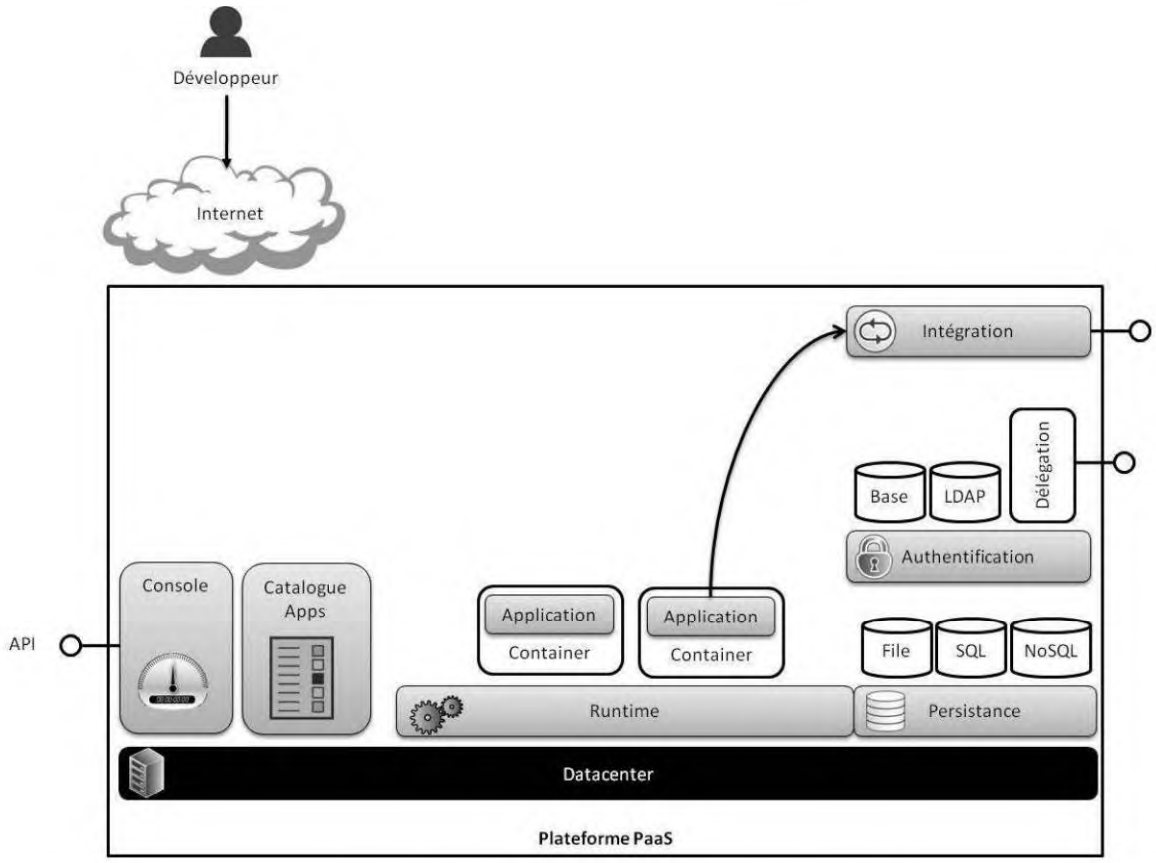

Figure 19.6 — Stratégies d'intégration des PaaS.

À nouveau, les puristes diront que ces services d'intégration sont du niveau PaaS, car ils sont proposés hors des VM et ne nécessitent pas d'installation. Certaines plateformes IaaS proposent donc des services de niveau PaaS pour faciliter le travail de leurs clients.

Le bus d'intégration peut aussi être un service d'infrastructure en ligne hébergé par une autre plateforme : nous verrons des exemples de ce type de service dans le chapitre 20.

À propos des bus d'intégration

Un bus d'intégration doit être capable de véhiculer des informations entre les applications selon différents modes d'échange :
– **Échange synchrone :** dans ce cas, l'application appelante attend la réponse à sa requête avant de continuer ses traitements.
– **Échange asynchrone :** dans ce cas, la demande est placée dans une queue d'attente et le traitement en cours n'est pas interrompu. Il prendra en compte la réponse à sa requête lorsque celle-ci sera disponible.
– **Processus d'intégration :** il s'agit là d'un mode d'intégration plus complexe. Le

bus d'intégration doit orchestrer un enchaînement d'échanges suivant un processus préétabli.

19.7 LA SÉCURISATION DES FLUX ENTRE SI ET CLOUD

Il est important de pouvoir sécuriser de manière forte les échanges d'information qui ont lieu entre le SI et le cloud (*cf.* fédération d'identité et bus d'intégration).

Certaines plateformes cloud proposent des solutions, avec un niveau de sécurité plus ou moins élevé :

- **tunnel SSL** entre la plateforme cloud et le SI. Cette solution est souvent proposée par les PaaS ;
- **VPN IPSEC** entre la plateforme cloud et le SI. Cette solution a l'avantage de créer un sous-réseau privé au sein de la plateforme cloud. Elle est plutôt proposée par les IaaS, qui permettent de paramétrer la couche réseau (voir plus haut) ;
- **réseau privé** en partenariat avec un opérateur télécom. Dans ce cas, les flux ne transitent pas par Internet.

19.8 L'ENVIRONNEMENT DE DÉVELOPPEMENT

Les PaaS proposent parfois un environnement de développement associé, permettant l'écriture du code, son débogage, sa compilation, son packaging et son déploiement (*cf.* la présentation de l'usine logicielle au chapitre 11). Deux options peuvent alors être proposées :

- **l'environnement de développement est proposé sous la forme d'un kit de développement** ou SDK[1] à télécharger. Ce SDK intègre les fonctions de construction du code source évoquées plus haut. Il propose aussi un *runtime* simulant la plateforme PaaS et permettant de tester l'application en local, sur le poste du développeur. Il intègre parfois aussi un logiciel de dessin des interfaces hommes machines ;
- **l'environnement de développement est proposé sous forme de service en ligne**. Dans ce cas, l'écriture du code se fait au travers d'un éditeur SaaS, à la manière d'un traitement de texte en ligne. C'est le cas avec Salesforce et son langage Apex. Puis, l'application est testée par le développeur directement sur la plateforme.

L'architecture de l'application peut être plus ou moins guidée, en fonction des librairies de code (*framework*) fournies et du degré d'abstraction de l'API.

Dans le cadre des plateformes IaaS, la création et la gestion de l'environnement de développement sont à la charge de l'entreprise utilisatrice.

1. *Software Development Kit*.

19.9 LA GESTION DU CYCLE DE VIE DES APPLICATIONS

Tout projet de développement logiciel nécessite divers environnements et au minimum :

- environnement de développement ;
- environnement de test/recette ;
- environnement de production.

La gestion de ces environnements est souvent proposée sur les plateformes PaaS qui fournissent au moins l'environnement de test et celui de production. Lorsque l'environnement de développement est proposé sous forme de service en ligne, il est aussi intégré à la plateforme.

Les PaaS offrent souvent un *versioning* des applications, c'est-à-dire que l'on peut stocker sur la plateforme différentes versions de l'application et choisir laquelle mettre en production. Il est ainsi possible de faire un retour en arrière si l'on constate un problème dans une nouvelle version.

Dans le cadre des plateformes IaaS, la gestion du cycle de vie est à la charge de l'entreprise utilisatrice.

19.10 LE MONITORING

Il est essentiel pour l'entreprise de pouvoir surveiller les applications hébergées sur les plateformes cloud. Cette surveillance a deux objectifs :

- vérifier que le niveau de service est bien conforme aux engagements de l'opérateur cloud ;
- observer la montée en charge de l'application (usage de la puissance de traitement et de l'espace disque) afin de statuer sur l'achat d'un complément de ressources auprès de l'opérateur cloud. Cette démarche est la même que pour un opérateur télécom : lorsque la bande passante pour l'accès à Internet devient insuffisante, les entreprises sont amenées à acheter une extension de cette bande passante.

Les plateformes cloud répondent à ce besoin de deux manières :

- elles peuvent proposer une API pour extraire les données et les analyser *via* un outil interne à l'entreprise ;
- elles peuvent proposer directement une interface graphique de suivi en temps réel des statistiques.

Il est peu probable que les plateformes cloud offriront un jour une interface SNMP[1], car ce protocole est conçu pour un suivi des infrastructures sur réseau local et non *via* Internet.

Enfin, les entreprises utiliseront certainement un outil tiers et indépendant de la plateforme cloud, comme Xiti Monitor, pour pouvoir s'assurer de la disponibilité de leur application de manière incontestable.

19.11 SYNTHÈSE SUR L'ARCHITECTURE DES IAAS ET PAAS

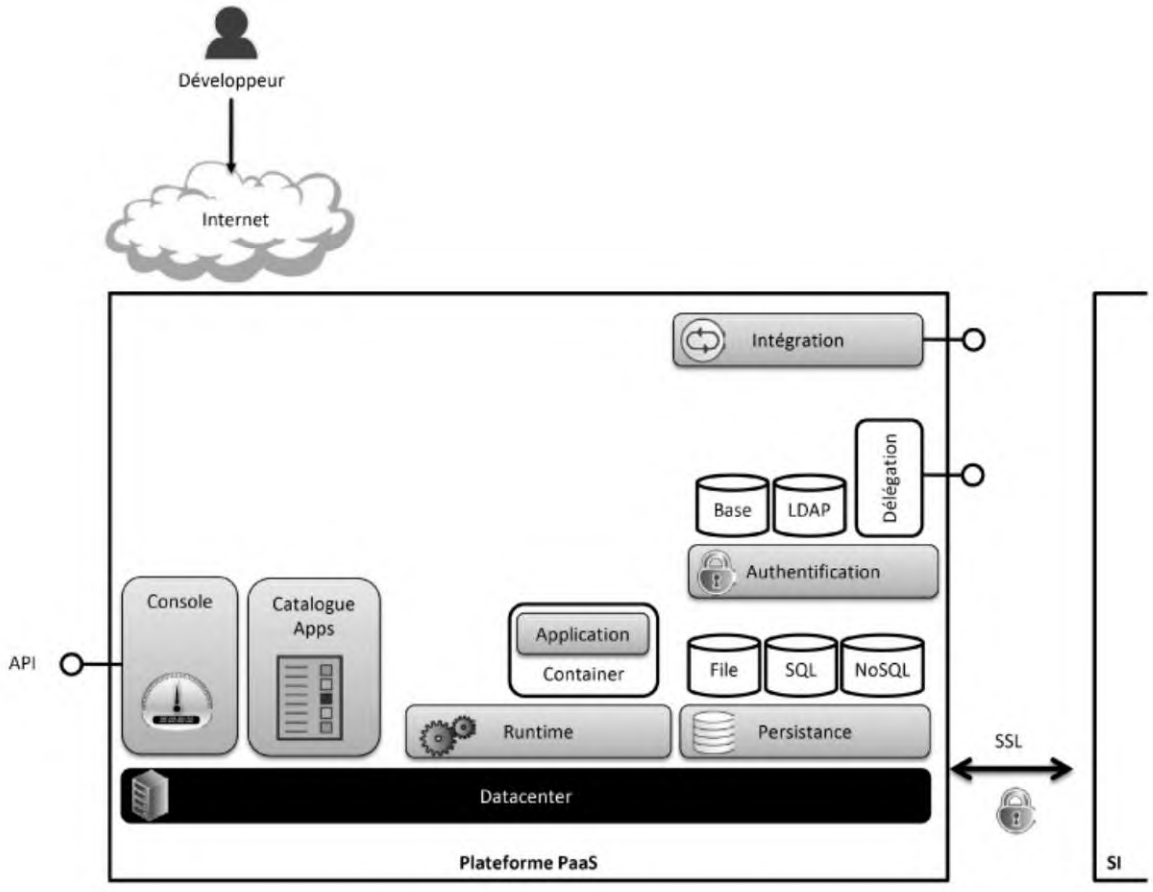

Figure 19.7 — Synthèse sur l'architecture des IaaS.

La figure 19.7 fait apparaître les composants classiques d'une plateforme IaaS, évoqués plus haut. On retrouve :

- le Datacenter ;
- l'hyperviseur, couche d'exécution des machines virtuelles ;

1. *Simple Network Management Protocol*, il s'agit du standard pour le suivi des métriques de fonctionnement des plateformes d'entreprises.

- le catalogue de machines virtuelles sur étagères ;
- le système de stockage persistant ;
- le tunnel IPSEC entre le SI et le cloud ;
- la console d'administration et les API permettant d'accéder à ses services.

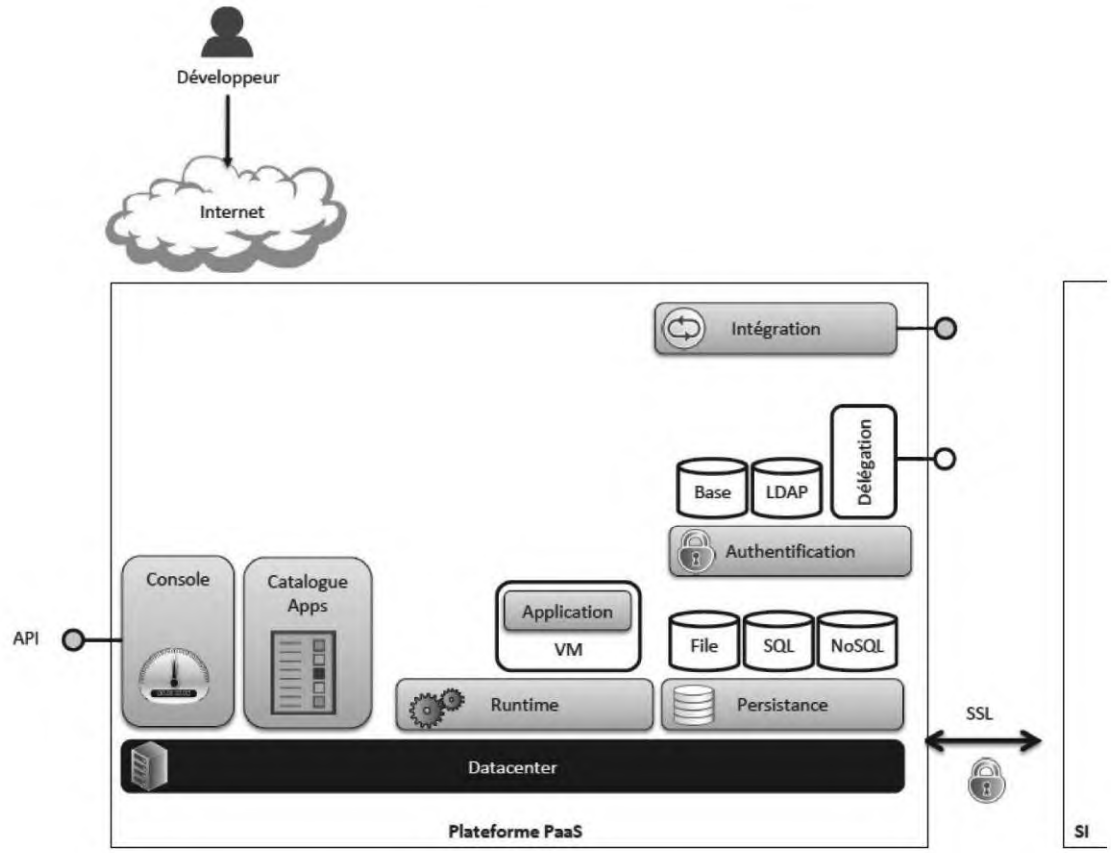

Figure 19.8 — Synthèse sur l'architecture des PaaS.

La figure 19.8 fait apparaître les composants classiques d'une plateforme PaaS, évoqués plus haut. On retrouve :

- le Datacenter ;
- la couche d'exécution des applications ;
- les systèmes de persistance sous forme de base de données SQL, NoSQL, ou de fichiers ;
- les systèmes d'authentification locale ou déléguée ;
- le système d'intégration : API ou bus d'intégration ;
- le tunnel SSL entre le SI et le cloud ;
- la console d'administration et les API permettant d'accéder à ses services.

On rappelle que certaines plateformes IaaS fournissent les services PaaS de persistance et d'intégration.

En résumé

Ce chapitre a présenté les différents services attendus d'une plateforme IaaS ou PaaS : en particulier, hébergement, exécution, persistance, sécurité, intégration. Il a aussi abordé les options possibles pour assurer ces services.

20

Les plateformes de cloud public (PaaS & IaaS)

Objectif

Ce chapitre présente les principales plateformes IaaS et PaaS publiques disponibles aujourd'hui.

20.1 LES OFFRES À 360° (PAAS & IAAS)

Ce paragraphe présente les offres les plus complètes du marché : celles qui embrassent PaaS et IaaS. Amazon, Microsoft et Google ont pour objectif de proposer les plateformes cloud le plus complètes possible.

20.1.1 Amazon Web Services

L'offre AWS (*Amazon Web Services*) a été la première offre IaaS disponible dès 2002. Le service EC2, pour *Elastic Cloud Computing*, a popularisé le terme cloud computing. L'offre est accessible sur aws.amazon.com. Amazon est parti de l'IaaS pour proposer progressivement des services PaaS, comme nous le verrons dans la suite. AWS a la plus grosse part de marché du cloud aujourd'hui (fin 2015).

Amazon exploite ses datacenters sur 11 « régions ». Chaque région dispose de trois Datacenters en miroir pour assurer la résilience des applications et des données. Il y a quatre régions aux États-Unis, une au Brésil, en Irlande, en Allemagne, à Singapour, au Japon, en Australie et en Chine. **Il est possible de choisir la région sur laquelle**

les données de l'entreprise seront stockées, ce qui permet de résoudre certaines problématiques juridiques (*cf.* chapitre 4).

Les principaux services de l'offre AWS sont les suivants :

- **EC2 est le service d'exécution des machines virtuelles.** Il est basé sur la technologie Open Source Xen. EC2 supporte les machines virtuelles sous Open Solaris, Linux et Windows Server. Amazon propose un catalogue de plus de 800 images systèmes, intitulées AMI (*Amazon Machines Images*). Les entreprises peuvent choisir la configuration de leurs VM à la carte. Amazon met à leur disposition des instances petites, larges, extra-larges, double extra-large, etc. caractérisées par leur puissance CPU et leur espace mémoire et disque.
- **S3, Simple Storage Service,** est le service de stockage persistant d'Amazon. Il stocke les données dans des « compartiments », espaces de stockage limités à 5 To. Ces données sont accessibles *via* une interface REST.
- **Glacier,** un service d'archivage en ligne.
- Amazon propose aussi **EBS (*Elastic Block Store*)** : une solution de périphérique de stockage intégrée à EC2 afin de rendre persistantes les données des machines virtuelles sur S3.
- Le service **Virtual Private Cloud** permet de créer un tunnel IPSEC entre le SI et le cloud Amazon.
- **AWS Direct Connect** permet un lien réseau direct entre le SI d'entreprise et le cloud Amazon.
- **Route 53** permet de configurer les DNS d'Amazon.
- Le service **AutoScaling** permet aux entreprises d'automatiser la mise à disposition de nouvelles ressources en cas de pic de charge sur une application. Il offre le meilleur de l'élasticité d'Amazon, mais il convient de l'utiliser prudemment car il fait perdre la maîtrise de la facture qui sera présentée en fin de mois.
- Par ailleurs, Amazon propose **CloudFront**, un service de cache distribué. Cette fonction ne rentre pas dans notre architecture de référence, présentée plus haut. Elle concurrence l'offre d'Akamai, un acteur historique du cache distribué, antérieur à l'émergence du cloud computing. Son objectif est de dupliquer les données au plus près des utilisateurs afin de réduire les temps de chargement.
- Etc.

Amazon offre aussi des services que nous qualifierons de services PaaS :

- **SimpleDB et DynamoDB** sont des services de persistance de type NoSQL.
- Relational Database Service **(RDS)** est un service de persistance de type SQL.
- **Elastic Beanstalk** est une plateforme d'exécution PaaS permettant le déploiement d'applications Java, PHP, Python, Ruby, .Net et Node.js.
- **SQS, *Simple Queue Service* est un service d'intégration**. Il permet l'échange de messages entre AWS et d'autres infrastructures. Il intègre une queue de messages et peut donc fonctionner en mode asynchrone. Il permet enfin d'enchaîner des échanges sous forme de workflow.
- Etc.

Pour l'administration, Amazon fournit AWS console : elle permet de piloter l'ensemble des services.

Nous avons pris le parti de ne pas détailler tous les services Amazon Web Services pour rester sur une vision d'ensemble. Nous pensons qu'il est techniquement possible d'externaliser tout un SI chez Amazon. En effet, il y existe même une offre poste de travail : Workspaces...

Nous souhaitons souligner un point : il est difficile d'anticiper le coût mensuel des services Amazon car leur modèle de calcul est complexe. Et même si Amazon propose une calculatrice prenant en compte toutes les subtilités de l'offre, l'estimation reste une tâche complexe pour les non-initiés. Il faut avoir en tête que **toute instance EC2 active est payée à l'heure, qu'elle soit accédée par des utilisateurs ou non**. Dans le cadre d'un service destiné aux heures de bureaux, il peut être pertinent de créer un script afin d'éteindre les VM le soir et de les rallumer le matin.

Figure 20.1 — La console AWS.

20.1.2 Microsoft Azure

Microsoft est arrivé tardivement sur le marché des plateformes cloud. Son offre Azure a été annoncée en octobre 2008 et ses composants sont encore en pleine évolution. Microsoft a tiré les enseignements des plateformes lancées avant la sienne : l'éditeur de Redmond s'est inspiré d'Amazon Web Service et de Google App Engine (*cf.* ci-après). Azure a la seconde plus grosse part de marché du cloud aujourd'hui (fin 2015).

La plateforme cloud repose donc sur les technologies habituelles de Microsoft pour l'entreprise : elle n'a pas été conçue spécifiquement pour le cloud, ce qui est intéressant en termes de compatibilité avec l'existant. Les logiciels Microsoft ont peu à peu évolué

pour devenir « *cloud ready* » : cette évolution bénéficie à Azure et aux déploiements dans les SI d'entreprise.

Azure offre un catalogue intitulé **Azure Marketplace** : il propose un annuaire des applications disponibles sur la plateforme, mais aussi un **DataMart**, qui permet de s'abonner à des flux de données métiers.

Selon le site datacenterknowledge.com, Microsoft déploie des Datacenters à partir de centaines de containers de serveurs intitulés C-Blox (*cf.* chapitre 18). Ces containers seraient totalement autonomes et disposés en plein air.

Azure compte 20 régions : 8 aux États-Unis, 3 en Inde, une à Dublin, à Amsterdam, à Singapour, à Hongkong, 2 au Japon, 2 en Australie, et une au Brésil. Ces datacenters ont des relations de réplication 2 à 2 pour assurer l'intégrité des données.

Les principaux services IaaS Azure sont :

- **Virtual Machines :** un service d'exécution des machines virtuelles. Il est basé sur la technologie HyperV et supporte les machines virtuelles sous Linux et Windows Server ;
- **Azure Storage :** un système de stockage non relationnel, en blobs ;
- **Azure CDN :** un système de cache géographique (*Content Delivery Network*) dans 18 pays ;
- **Azure Connect,** une solution de sécurisation des flux en IPSec entre Azure et le SI (*cf.* http://aws.amazon.com/vpc) ;
- etc.

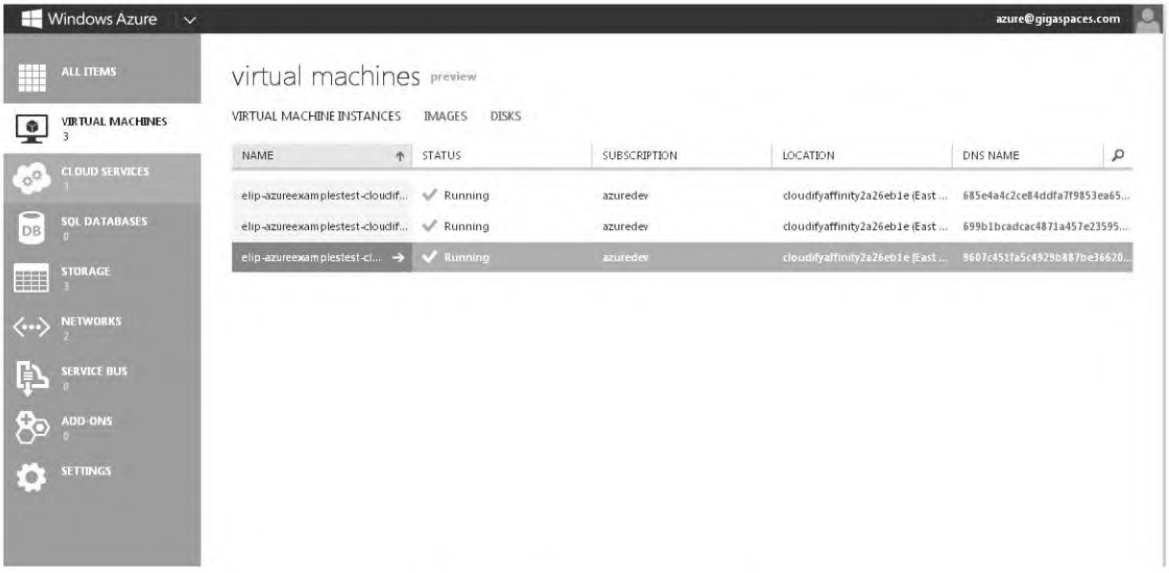

Figure 20.2 — La console Azure.

Les services PaaS Azure sont les suivants :

- **Azure AppService :** c'est le socle d'exécution, basé sur le serveur IIS et le runtime .NET. Il est à même de faire tourner des applications .NET adaptées

à Azure, c'est-à-dire respectant les spécifications Web Roles (pour les frontaux web) et Worker Roles (pour les traitements asynchrones). Les applications Azure sont contraintes par la taille de la VM qui leur est affectée. Azure est aussi capable d'exécuter des applications Java, Python, Ruby, Node.JS ou PHP, car il existe des *runtimes* pour Windows/IIS dans ces langages ;

- **SQL Azure :** un système de stockage relationnel ;
- **Azure Service Bus :** un bus applicatif pour le cloud ;
- **Azure Access Control Service :** une solution de fédération d'identité capable de fédérer Azure avec des annuaires d'entreprise *via* ADFS et SAML, ou avec des annuaires grands publics comme Google ID, Live ID, Facebook, Yahoo! ID, OpenID. Azure propose aussi un annuaire au sein de la plateforme : **Live ID** ;
- etc.

Nous avons pris le parti de ne pas détailler tous les services Azure pour rester sur une vision d'ensemble. Nous pensons qu'Azure est aujourd'hui proche d'Amazon Web Services en termes de maturité.

Les développements Azure utilisent **l'environnement Visual Studio**, bien connu des entreprises utilisatrices des technologies Microsoft. Les développeurs retrouvent donc leurs langages habituels. Cependant, il n'est pas possible de déployer sur Azure le même code que celui utilisé dans l'entreprise : des aménagements seront nécessaires.

La console d'administration permet aux développeurs d'accéder à la plateforme, de déployer les applications et de suivre leur activité.

20.1.3 Google Cloud Platform

Google est très secret sur la structure de ses datacenters : ces derniers constituent sa force et permettent une certaine avance sur ses concurrents. Nous allons donc parler au conditionnel dans ce paragraphe...

Google serait ainsi la société exploitant le plus grand nombre de serveurs dans le monde et sa compétence sur le sujet serait la plus avancée à ce jour. La plateforme Google serait commune pour tous les services Google : services de recherche, de cartographie, services collaboratifs Google Apps, etc., et le service PaaS Google App Engine. Ainsi, Google ne ferait pas de distinguo entre les socles de ses applications grand public et ceux des applications d'entreprise : la plateforme physique et logicielle de Google serait unifiée, ce qui lui offrirait une grande cohérence, une très grande puissance, une très grande disponibilité et une très grande capacité à gérer les pannes. En revanche, cette unification nécessiterait la maîtrise des composants déployés sur la plateforme. En effet, si une entreprise développait une application instable sur le cloud Google, cette instabilité pourrait affecter l'ensemble de son écosystème.

Selon son site, Google exploite treize Datacenters : six aux États-Unis, un au Chili, un en Belgique, un en Irlande, un en Finlande, un en Hollande, un à Singapour et un à Taïwan.

L'offre IaaS Google propose depuis quelques années les services suivants, peu adoptés par les entreprises :

- **Compute Engine** est le service d'exécution des machines virtuelles, uniquement sous Linux ;
- **Cloud Storage** est le service de stockage persistant. Il propose des « compartiments » dont la taille est de l'ordre d'un To.

La nouveauté en 2015 est que Google propose un « **container engine** » qui propose l'exécution de **containers Dockers, orchestrés par Kubernetes**. Comme on l'a vu au chapitre 19, Google investit fortement sur cette offre, qui le différencie d'Amazon ou Microsoft. Nous pensons que Google pourrait rattraper ces deux leaders grâce à cette nouvelle vision.

Depuis peu, Google propose *cloud networking*, un service d'interconnexion, d'accélération et de cache, basé sur son immense cœur de réseau.

L'offre PaaS propose principalement les services suivants :

- **Google App Engine**, service d'exécution Python, Java, PHP, et Go ;
- **DataStore**, un service de persistance NoSQL ;
- **Cloud SQL**, le service de persistance relationnel basé sur MySQL.

Enfin, Google propose des services d'analyse orientés Big Data, ce qui constitue son cœur de métier.

Notre présentation de l'offre Google n'est pas exhaustive, de même que pour Amazon et Microsoft.

20.2 LES OFFRES IAAS SEUL

20.2.1 Rackspace cloud

L'offre de RackSpace est assez proche de l'offre IaaS d'Amazon, dont elle s'est inspirée. Elle connaît un important succès aux États-Unis.

Elle est composée des services :

- **Cloud Servers, le service d'exécution des machines virtuelles.** Il est basé sur la technologie Xen et supporte les VM sous Linux et Windows Server. La persistance des données manipulées par les VM est assurée de manière automatique ;
- **Cloud Files, le service de stockage persistant.** Il a inspiré le standard OpenStack, dont nous reparlerons au chapitre 21.

20.2.2 Quelques autres...

Parmi les offres IaaS matures, on peut aussi citer GoGrid, Joyent. En France, OVH et Gandi proposent des offres intéressantes.

Orange, SFR et Bouygues Telecom proposent aussi des plateformes IaaS.

20.3 LES PAAS « MULTI-PURPOSE »

Ce paragraphe présente l'offre de Salesforce, puis des offres PaaS qui supportent plusieurs langages de programmation et plusieurs systèmes de persistance. Ce sont des PaaS à usages multiples, *multi-purpose* en anglais.

20.3.1 Salesforce

Le terme PaaS a été popularisé par Salesforce. La société a d'abord édité un logiciel de CRM en mode SaaS, puis elle a proposé son infrastructure comme plateforme de développement afin de créer un écosystème de progiciels en ligne, avec le concours d'autres entreprises. Sa stratégie consistait en effet à rendre son offre plus attractive en l'enrichissant avec des offres tierces étroitement liées et accessibles dans une interface commune.

La plateforme propose **App Exchange**, un catalogue des applications disponibles sur salesforce.com. C'est le premier catalogue de ce type à avoir été créé : Salesforce a été particulièrement innovant sur cette offre. Ainsi la plateforme PaaS a d'abord été créée pour enrichir l'offre SaaS.

Le modèle tarifaire de salesforce.com diffère de celui de la plupart des plateformes PaaS et IaaS : **la facturation est effectuée selon le nombre d'utilisateurs mensuels, et non selon les ressources consommées**.

Selon son site web, Salesforce exploite quatre datacenters aux États-Unis, un au Royaume Uni et un au Japon.

L'offre salesforce.com propose les services PaaS suivants :

- **Le service d'exécution force.com basé sur le langage Apex.** Ce langage spécifique a été créé par Salesforce pour accélérer le développement des applications sur force.com. Apex est un langage de haut niveau qui permet de créer rapidement des applications de type CRUD[1] : il n'est pas adapté à des traitements algorithmiques. Apex permet aussi de créer rapidement des workflows utilisateurs. Le développement des écrans peut se faire par clic et glisser/déposer à partir d'interfaces graphiques : coder en Apex n'est pas toujours nécessaire.

1. *Create Read Update Delete*. L'expression signifie ici que le langage permet de faire des écrans pour visualiser et mettre à jour des tables en base de données.

- **Database.com** est le service de persistance de force.com accessible *via* des API. Il propose plusieurs stratégies de persistance : fichiers et bases relationnelles.
- « **Proven real-time integration** » **semble être le nom du service d'intégration.** Il propose des connecteurs natifs pour ERP, comme SAP et Oracle Business ; il propose des connecteurs vers des bus d'entreprise comme TIBCO ; il supporte l'intégration Web Services, REST, JEE et .NET ; il propose des connecteurs pour Office et Lotus.
- **Site.com** est une plateforme de développement rapide de sites web.
- **Salesforce Identity** permet à Salesforce de proposer la fédération d'identité SAML dans deux types de scénarios : Salesforce délègue l'authentification à l'annuaire de l'entreprise, ou bien Salesforce est fournisseur d'identité principale.

20.3.2 Heroku

Salesforce a racheté en 2010 la société Heroku. Heroku est une référence incontournable dans les plateformes PaaS en langage Ruby. La plateforme est très populaire auprès des développeurs et des start up.

Heroku propose :

- un service **d'exécution** des langages Node.JS, Ruby, Java, PHP, Python, Go, Scala, Clojure ;
- de nombreux **services de persistance** basés sur PostGreSQL, MongoDB, Redis, Cassandra, Amazon RDS, Neo4J, etc. ;
- de **très nombreux plugins** : pour la recherche, l'e-mail marketing, le paiement, les statistiques, le monitoring, etc.

Heroku est resté très autonome vis-à-vis de Salesforce. Il existe cependant une passerelle technique entre les deux mondes. Elle permet de synchroniser la base client Salesforce avec une base Heroku.

À noter : Heroku repose sur la plateforme Amazon Web Services.

20.3.3 EngineYard

EngineYard est comparable à Heroku : cette PaaS est surtout utilisée pour les développements Ruby. Elle propose :

- un service **d'exécution** des langages Ruby, PHP, Node.JS. ;
- des services de **persistance** basés sur PostGreSQL, MySQL, Redis, etc.

À noter : EngineYard repose aussi sur la plateforme Amazon Web Services.

20.3.4 OpenShift

OpenShift est comparable à EngineYard. Cette PaaS propose :

- un service **d'exécution** des langages Ruby, PHP, Python, Java, Node.JS ;
- des services de **persistance** basés sur PostGreSQL, MySQL, MongoDB, Amazon RDS, etc.

À noter : OpenShift repose aussi sur la plateforme Amazon Web Services.

En résumé

Ce chapitre a présenté les principales IaaS et PaaS disponibles aujourd'hui :
– les offres à 360° d'Amazon, Google, Microsoft ;
– quelques offres IaaS ;
– les PaaS multipurpose : Salesforce, Heroku, EngineYard, OpenShift.

21

Les plateformes de cloud privé

Objectif

Ce chapitre présente les principales plateformes IaaS destinées au cloud privé.

21.1 LES SOLUTIONS ÉDITEURS

21.1.1 VMware vCloud

VMware est leader du marché des solutions de virtualisation. L'éditeur dispose d'une offre destinée à la création d'IaaS intitulée vCloud.

Cette offre est destinée aux entreprises qui souhaitent disposer en interne d'une plateforme élastique offrant le *Pay As You Go* et le *Self Service*.

Elle est aussi utilisable par les hébergeurs qui souhaitent évoluer vers le modèle cloud : ainsi Orange, SFR, Claranet, OVH, LinkByNet, etc. ont des offres reposant sur les logiciels VMware.

La stratégie VMware va au-delà de la simple fourniture de logiciel. L'éditeur a en effet passé des accords avec des hébergeurs afin que ces derniers mettent à sa disposition leurs Datacenters en mettant en avant sa marque : **vCloud Air Network Service Providers**.

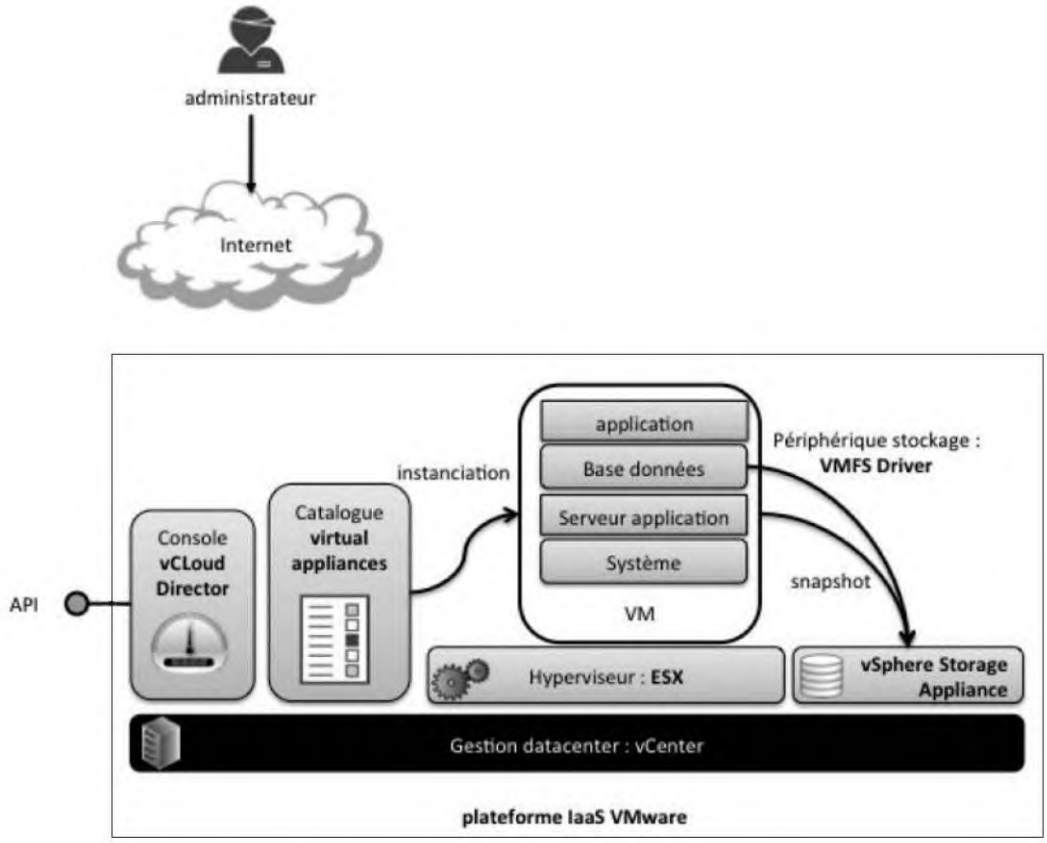

Figure 21.1 — L'architecture vCloud.

La suite vCloud propose les services IaaS suivants :

- **vSphere, le service d'exécution des machines virtuelles.** Il est basé sur la technologie VMware ESX et supporte les VM sous Linux, Netware, Solaris et Windows Server ;
- **vSphere Storage Appliance est un service de stockage persistant distribué.** Il tire parti des disques durs d'une ferme de serveurs pour créer un espace de stockage unifié. Le logiciel copie automatiquement les données sur plusieurs disques pour assurer leur intégrité ;
- **vCenter** est l'offre VMware de gestion de Datacenter. Elle permet de gérer le parc de serveurs physiques, le catalogue de VM sur étagères, etc. **vCenter Chargeback** permet en particulier de refacturer les ressources informatiques selon les consommations réelles ;
- **vCloud Automation et vFabric Application Director** permettent de mettre à disposition de manière automatique des environnements IaaS et PaaS ;
- **vCloud Director** est la console de gestion cloud permettant le Self Service.

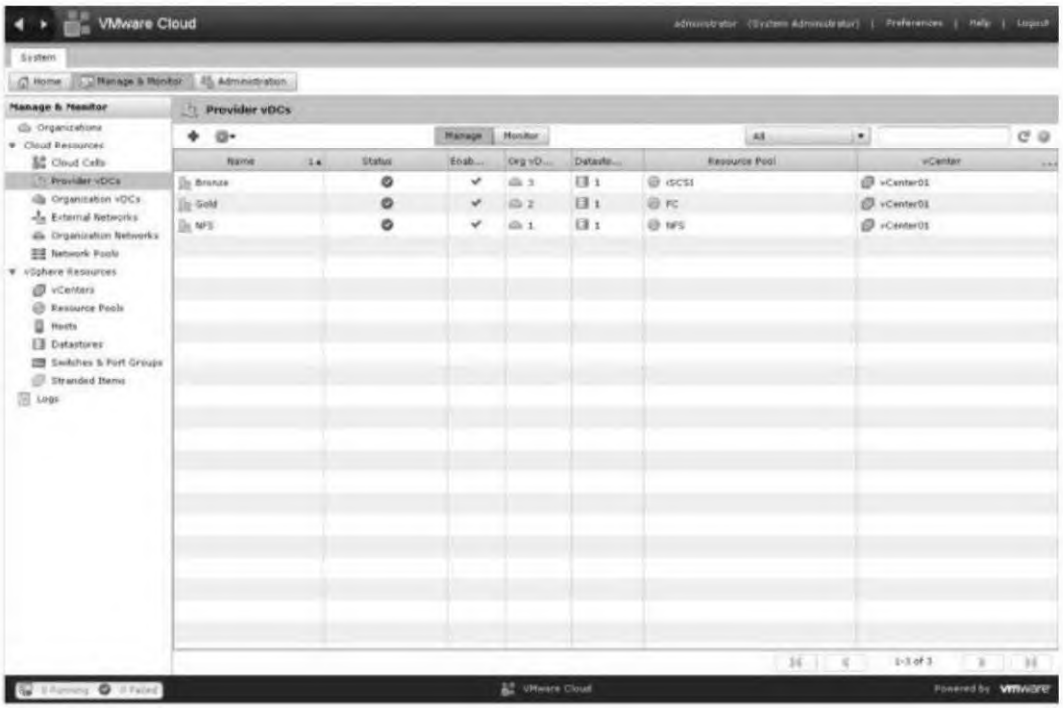

Figure 21.2 — La console vCloud Director.

21.1.2 Microsoft HyperV

Microsoft s'est fortement positionné sur la virtualisation depuis 2008. L'hyperviseur HyperV est d'ailleurs intégré à Windows Server. Il permet la virtualisation de Windows et Linux et supporte les VM au format VMware.

System Center 2012 est l'offre de gestion de cloud. Il propose :

- un système de refacturation interne ;
- le Self Service ;
- la gestion de cluster ;
- la migration à chaud ;
- un portail de *provisioning* avec un catalogue de machines virtuelles sur étagères.

C'est une offre de cloud privée complète pour l'écosystème Windows.

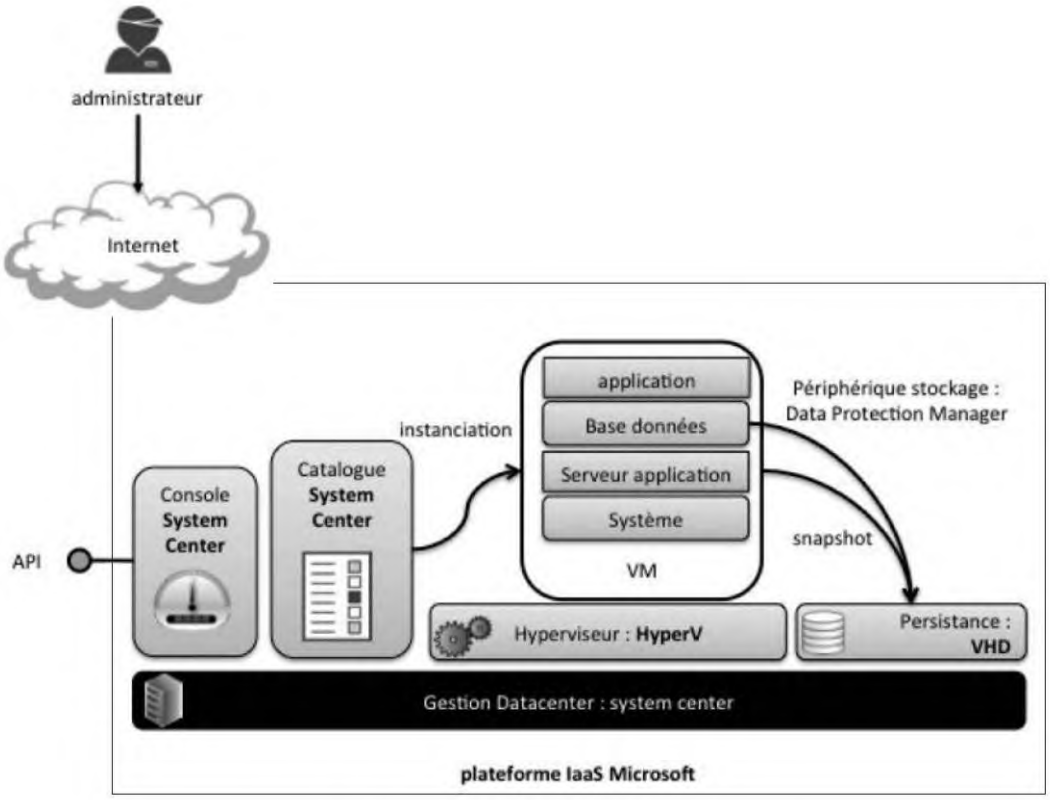

Figure 21.3 — L'architecture System Center.

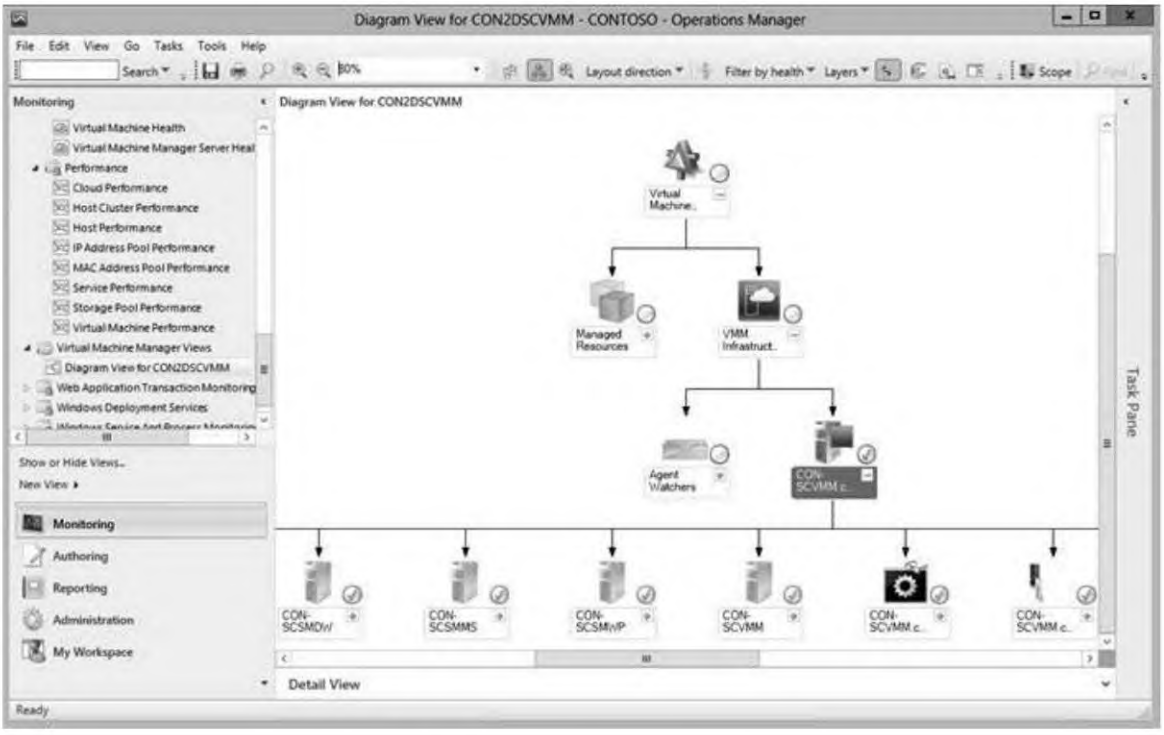

Figure 21.4 — La console System Center.

21.2 LES SOLUTIONS OPEN SOURCE

On a constaté depuis 2012 une maturation des concepts de l'IaaS. En effet, toutes les offres reposent sur :

- un système d'exécution ;
- un système de persistance ;
- un portail pour créer des machines virtuelles et facturer leur consommation ;
- un catalogue de machines virtuelles sur étagère.

Dans l'histoire de l'informatique, on constate que lorsqu'un sujet arrive à maturation, des standards émergent et on assiste à une poussée de l'Open Source. C'est le cas de l'IaaS depuis 2012, mais pas encore du PaaS.

21.2.1 OpenStack

OpenStack est un Projet Open Source sous licence Apache. Plus de 150 sociétés contribuent au projet : AMD, Intel, Canonical, SUSE Linux, Red Hat, Cisco, Dell, HP, IBM, Yahoo! (et donc ni Amazon, ni Google, ni Microsoft). Parmi les grands contributeurs, on peut citer Rackspace (*cf.* chapitre 20) et la NASA.

OpenStack propose les services suivants :

- **OpenStack Compute :** un service d'exécution des machines virtuelles compatible avec Xen, KVM, VMware, HyperV ;
- **OpenStack Storage :** un service de stockage persistant ;
- **OpenStack Networking :** un service de gestion de réseau virtuel ;
- **OpenStack Dashboard :** un portail de self service ;
- **Image Service :** un catalogue de machines virtuelles ;
- **Identity :** un service d'indentification des utilisateurs ;
- **OpenStack API :** pour l'administration des machines virtuelles, compatible Amazon Web Services.

Son architecture est modulaire et ouverte. Le projet se développe rapidement. Il est devenu une référence pour construire un cloud IaaS.

Il est utilisé par CloudWatt, Dell, HP, RackSpace, Piston Cloud, etc.

OpenStack ambitionne de concurrencer Amazon Web Services qui reste aujourd'hui la référence en matière d'IaaS.

21.2.2 OpenNebula

OpenNebula est une alternative Open Source à OpenStack plus légère et plus simple à prendre en main. La solution propose :

- un service d'exécution des machines virtuelles compatible avec Xen, KVM, VMware ;

- une API compatible avec EC2 Query, OGF OCCI et vCloud.

C'est une solution facile à mettre en œuvre, intéressante pour les besoins de développeurs, ou dans le cadre d'une PME.

21.2.3 CloudStack

Cloud Stack est une autre alternative Open Source sous licence Apache, propriété de Citrix. Elle a été rachetée avec la société cloud.com.

La solution supporte les hyperviseurs vSphere, Oracle VM, KVM et Xen.

21.3 LE CLOUD PRIVÉ « CLEF EN MAIN »

Un certain nombre de fournisseurs proposent depuis longtemps leurs services sous forme de boîtiers : c'est le cas de Google avec Search Appliance, de Fortinet avec ses firewalls, etc.

Les « Cloud in a Box » ou « Systèmes convergés » suivent cette tendance en proposant des technologies de datacenters sous forme de containers ou de machines packagées.

Elles facilitent la création d'un cloud privé.

21.3.1 L'offre VCE

VCE vBlock est le premier produit de ce type apparu sur le marché. Il repose sur les technologies de VMware, Cisco, EMC. Le produit prend la forme d'une baie de serveurs packagée.

Il est proposé sous deux modèles économiques :

- à l'achat : l'entreprise doit alors amortir son achat de matériel ;
- en paiement à l'usage : cette approche est plus cohérente avec le modèle cloud.

21.3.2 Oracle Exalogic/Exadata

Oracle dispose d'une offre de matériel serveur depuis le rachat de Sun. L'éditeur a ainsi créé l'offre Exalogic Elastic Cloud, présentée comme une solution de cloud privée sur technologies Oracle et Java.

21.3.3 Les autres

Selon le Gartner, les acteurs à suivre (fin 2015) dans ce domaine sont HP, IBM, Cisco, NetApp, Nutanix.

Figure 21.5 — Les baies VCE vBlock.

En résumé

Ce chapitre a présenté les architectures des principales IaaS pour les cloud privés : offres d'éditeur, offres Open Source et offres sous forme de box clé en main.

22

Les perspectives du cloud

Objectif

Ce chapitre de conclusion propose quelques pistes pour le futur du cloud computing. Il aborde la problématique des standards. Partant d'idées actuelles, il avance des hypothèses extrêmes.

22.1 ALLER AU BOUT DE LA LOGIQUE DE COMMODITÉ

À long terme, les plateformes cloud pourraient se banaliser et devenir de véritables commodités sans grande valeur ajoutée, au même titre que l'eau et l'électricité.

On pourrait alors imaginer que la migration d'une machine virtuelle se fasse d'une plateforme IaaS à une autre sans couture, ou bien qu'une application Java puisse migrer simplement d'une plateforme PaaS à une autre (idem pour Ruby, .NET, etc.). Les PaaS deviendraient alors des serveurs d'applications dans les nuages, aussi standardisés que les serveurs d'applications d'entreprise.

Les entreprises pourraient alors déployer leurs applications chez plusieurs opérateurs de plateformes cloud et ainsi répartir les risques techniques sur plusieurs acteurs. C'était la promesse des serveurs d'application JEE : il serait souhaitable pour nous, utilisateurs, qu'elle se concrétise un jour dans le domaine du cloud computing.

En allant plus loin, on pourrait imaginer que des « brokers cloud » gèrent l'achat de ressources au même titre que l'énergie. Ainsi, une entreprise achèterait ses ressources d'exécution au « broker », qui se chargerait de placer les applications sur la meilleure plateforme selon des critères définis avec son client. Ces critères pourraient être le coût, la disponibilité, la localisation dans tel pays, etc.

On pourrait même imaginer que le « broker » déplace les applications plusieurs fois par mois, afin d'optimiser les coûts en fonction de tarifications dynamiques. Des « clouds *low cost* » pourraient alors apparaître selon le modèle des compagnies aériennes.

Côté opérateurs cloud, la logique de commodité pourrait se traduire par des accords de « *roaming* » au même titre que les opérateurs télécom. Un opérateur cloud aurait ainsi des accords avec un partenaire pour que celui-ci l'aide à absorber un pic de charge trop important pour sa plateforme.

De fait, le cloud privé disparaîtrait, ce qui nous semble souhaitable tant ce concept est fumeux.

22.2 VERS DES STANDARDS ?

Il serait souhaitable que des standards émergent pour faciliter le scénario de commodité ci-dessus. Les standards permettraient de rassurer les entreprises sur la réversibilité de leurs actifs, la capacité à changer d'opérateur. Les standards permettraient aussi l'émergence d'un vaste écosystème d'opérateurs de toutes tailles, adaptés à divers besoins et diverses contraintes (exemple : cloud global ou cloud local). Enfin, ils permettraient probablement de faire baisser les prix.

Les membres de l'Open Cloud Manifesto travaillent sur le sujet : mais, tant qu'ils n'auront pas convaincu les grands acteurs de cloud de les rejoindre, leur démarche restera vide de sens.

Dans le cadre des **plateformes IaaS**, des standards signifieraient :

- un format standard pour les machines virtuelles. Dans ce domaine, la spécification **OpenStack Compute** propose un système capable d'exécuter des VM issues de plusieurs technologies d'hyperviseur (Xen, KVM, HyperV) ;
- un format standard de système de stockage persistant. **OpenStack Object Storage** devrait aller dans ce sens ;
- des API standards pour l'administration des VM. **OpenStack Compute** propose de telles API, proposées pour normalisation par RackSpace.

OpenStack est un standard assez mature aujourd'hui, il est très prometteur comme standard universel.

Dans le cadre des plateformes PaaS, l'écosystème Docker offre de bonnes perspectives de standardisation.

Dans le cadre des plateformes SaaS, les standards portent sur l'intégration avec le SI dans le cadre de la gestion d'identité et des échanges de flux métiers. Pour la fédération d'identité, un large consensus est apparu autour de SAML. Pour les échanges de flux métiers, on pourrait imaginer plus de normalisation des API REST et des flux d'échange XML/JSON.

Un standard de monitoring serait aussi souhaitable pour permettre aux entreprises de monitorer la performance de leurs applications hors les murs, sur les plateformes SaaS/PaaS/IaaS.

22.3 VERS LE CLOUD PERVASIF

Pour terminer cet ouvrage, nous avons voulu aborder une pure prospective : le cloud pervasif. Derrière cette expression se cache l'idée d'une infrastructure unifiée capable de gérer sept milliards d'utilisateurs, c'est-à-dire la totalité de l'humanité. On imagine ici une plateforme informatique qui serait le pendant de l'Internet : gigantesque, dotée d'une élasticité quasi infinie, massivement redondante, contrôlée par aucune entreprise ou État, hétérogène mais capable de fonctionner en bonne entente. Ses éléments banalisés seraient placés aussi bien dans des Datacenters, que chez des particuliers. Les données seraient chiffrées et dupliquées de telle manière que l'espionnage soit quasi impossible.

Une telle infrastructure permettrait à toute personne et toute entreprise de déléguer en toute confiance ses traitements et le stockage de ses données. Elle autoriserait des applications utilisables par l'humanité tout entière. Et le terme cloud prendrait vraiment tout son sens.

En résumé

Ce chapitre de conclusion a abordé la « commodification » ultime, les standards à venir autour du cloud et le cloud pervasif.

Références bibliographiques

Bibliographie

Nicholas G. CARR, Does IT Matter ? – *Information Technology and the Corrosion of Competitive Advantage*, Harvard Business School Press, Boston, Massachusetts, 2004.

Nicholas G. CARR, The *Big Switch, Rewriting the World, from Edison to Google*, Norton, 2008.

Jeff JARVIS, *What Would Google Do ?*, HarperBusiness, 2009.

David-A VISE, Mark MALSEED, *Google Story*, Dunod, 2006.

Francis PISANI, Dominique PIOTET, *Comment le web change le monde*, Pearson, 2008.

Pierre PEZZIARDI, Laurent BRISSE, Gilles LABORDERIE, Christophe THIBAUT, Julien CABOT, *Une politique pour le système d'information : Descartes*, Wittgenstein, (XML), OCTO Technology, 2006.

Guillaume PLOUIN, Julien SOYER, Marc-Éric TRIOULLIER, *Sécurité des architectures web*, Dunod, 2004.

Xavier FOURNIER-MOREL, Pascal GROJEAN, Guillaume PLOUIN, Cyril ROGNON, *SOA. Le guide de l'architecte du SI*, 3e édition, Dunod, 2011.

Pascal GROJEAN, Médéric MOREL, Guillaume PLOUIN, *Performance des architectures IT*, 2e édition, Dunod, 2011.

Jean-Marc RIETSCH, Marie-Anne CHABIN, Éric CAPRIOLI, *Dématérialisation et archivage électronique. Mise en œuvre de l'ILM (Information Lifecycle Mangement)*, Dunod, 2006.

Christophe LONGÉPÉ, *Le projet d'urbanisation du SI, Démarche pratique avec cas concret*, 4e édition, Dunod, 2009.

Webographie

Techcrunch : http://www.techcrunch.com

ReadWriteCloud : http://www.readwriteweb.com/cloud

Datacenter Knowledge : http://www.datacenterknowledge.com

High Scalability : http://highscalability.com

Cloud Magazine : http://www.cloudmagazine.fr

Cloud News : http://www.zdnet.fr/blogs/cloud-news

La lettre du cloud : http://lalettreducloud.com

Silicon Cloud : http://www.silicon.fr/categorie/cloud

Blog Amazon Web Services : http://aws.typepad.com/aws

Blog Google Apps : http://googleappsupdates.blogspot.com

Blog Google Cloud Platform : http://googlecloudplatform.blogspot.fr/

Blog Cloud Computing Microsoft France : http://blogs.msdn.com/b/cloudcomputing

Blog Ysance : http://decrypt.ysance.com

Blog OCTO : http://blog.octo.com

O'Reilly Radar : http://radar.oreilly.com

InternetActu : http://www.internetactu.net

L'atelier BNP Paribas : http://www.atelier.net

Blog de Frédéric Cavazza : http://www.fredcavazza.net

Blog Green IT : http://www.greenit.fr

Index

Symboles

.NET 29
37Signals 183

A

A/B testing 47
accessibilité 93
accompagner les utilisateurs 132
achats 113
administrateur héroïque 85
ADP 200
agilité 84
ajustement 99, 222
Amazon 28, 171, 241
analyse de risque 70, 111
analyse décisionnelle 200
Apex 247
API 13
    ouvertes 24
App Exchange 247
application
    composite 14, 215
    métier 73
    SaaS 76
approche *best of breed* 137
architecture
    multi-tiers 211
    N-tiers 213
    orientée service 215
archivage 176
Arkhinéo 189
ASP 5, 24, 120
authentification 60, 233
AutoScaling 242
AWS 241
Azure 170, 243
    Connect 244
    Storage 244

B

bac à sable
    d'incubation 98
    innovation 125
basculement par lots 131
Basecamp 183
besoin 109
bêta perpétuelle 47, 91
BigTable 221
BlackBox 167
brokers cloud 259
bureau
    classique 203
    en ligne 91, 204
bus d'intégration 143, 233
Business by Design 169

C

CAPEX 49, 223
cartographie 191
catalogue de VM 229
Cegid Business on Demand 201
cellule architecture 113, 125
centres de données 225
chromebooks 16, 45
classification des données 70, 111
client
    léger 44
client/serveur 3
cloud
    broker 152
    computing 19
    desktop 205
    in a box 256
    pervasif 261
    privé 48
    public 48
    washing 49
CloudFront 242
CNIL 58
coédition 178
coexistence 124
cohabitation 131
collaboration 92
    unifiée 173
commerce électronique 192
commodité 259
communication
    asynchrone 174
    synchrone 173
conduite du changement 125
confiance 57
confidentialité 62
consistance 219
console d'administration 126
containers 227
contraintes architecturales 80
CRM 196
Customer driven roadmap 48, 91
cycle
    de vie 236
    en V 89

D

Datacenters 225
DataStore 246
débordement 99
décideurs 81
déontologique 87
département innovation 122
dépendance au réseau 102
déploiement 4, 44
    d'applications sur IaaS 77
dépossession 95
développement d'applications sur PaaS 79
DevOps 155
directeur de la sûreté 152
direction 113
    des études 114
disponibilité 66, 219
DNS 229
données confidentielles 69
droit applicable 58
DropBox 188
Dynamics CRM Online 196
Dynamo 221

E

early adopters 121
EC2 241
Efolia 200
élasticité 21
EngineYard 248
environnement de développement 235
environnements 98
ERP 201
études 99
expérimentation 121, 123

expérimenter 119
externalisation 30

F

fail fast 98
fédération d'identité 129
financialforce.com 201
Flash 8
fonctions RH 199
Freemium 39
FrontOffice 191

G

garantie contractuelle 57
geocoding 192
gestion des comptes utilisateurs 126
Gmail 179
Google 170
  Analytics 196
  App Engine 245
  Apps 179
  Apps Marketplace 93
green IT 83
grille de critères 105
Gtalk 181

H

haute disponibilité 228
hébergeurs 167
helpdesk 99
Heroku 248
HTML5 7, 8
HyperV 170
hyperviseur 27, 229

I

IaaS 22, 26, 152, 229
IBM 168
indicateur de présence 174
industrialisation 135
informatique
  de commodité 73, 84, 119
  métier 84
Infrastructure as a Service 26
intégration de données 129
iPhone 16
ISB 144
ISO 27001 59, 71

J

JEE 29

L

licence perpétuelle 42
Live ID 245

M

mailing 194
mainframes 3
Map/Reduce 221
Mappy 191
mashup 14, 92, 215
messagerie unifiée 174
Microsoft 243
migration 94
mises en production 99
modèle
  cloud public 39, 42
  grandsystème 33
  hébergeur 37
  Open Source 35
  outsourcing 37
  software 34, 42
  web 38
monitoring 159, 236
montée en charge 223
multi-tenant 24, 39, 216
mutualisation 43

N

nomades 63
NoSQL 220, 232

O

obsolescence 44
offshore 100
Open cloud 51
Open Source 50
OpenID 141
OpenShift 249
OpenStack 229
opérateur
    SaaS 23
    SaaS unique 136
OPEX 49
optimisation 228
Oracle on Demand 168

P

PaaS 22, 25, 152, 213, 231
panier d'achat 193
partage d'information unifié 176
Patriot Act 59
Pay As You Go 21
Paypal 193
pénalités 86
pérennité 111
persistance 232
perte de pouvoir 100
perversion des versions 47
pilotage d'opérateur SaaS 158
pilote 113
    IaaS 124
    SaaS 121
plan de reprise d'activité 82, 222, 223
plate-forme d'entreprise 218
Platform as a Service 25
politique
    de mot de passe 61
    de sécurité 69
portail d'intégration 142
PRA 226
préfabriqués 226
problématiques
    d'achat 86
    d'intégration 112
    juridiques 58
progiciel 29, 199
Project Caroline 167
provisioning 76
Python 246

R

RackSpace 246, 247, 251
RDS 242
recentrage 99
recherche 191
    transverse 173
réduction des coûts 81, 84
règles de confidentialité 56
rejet 87
Representational State Transfer 216
responsable de la sûreté 69, 111, 113
Ressource Oriented Architecture 216
REST 216
réversibilité 101
RIA 6
ROA 216
RSSI 69, 70
RunMyProcess 145

S

S3 242
SaaS 22, 23
Safe Harbour 58
Sage 197
Salesforce 196, 247
salles blanches 45
SAML 140
SAP 169
Sarbanes-Oxley 58

SAS 70 Type II 71
SDK 235
secret industriel 69
secteur d'activité 74
sécurité 84, 101
    des accès 62
Self Service 10, 21
serveur d'identité 129
service d'identité 139
SIG 192
Silverlight 8
SimpleDB 242
Single Sign On 140
SLA 159
smartphones 16
snapshot 230
SNMP 237
SOA 23, 84, 214
Software as a Service 23
SpringSource 168
SQL Azure 245
SQS 242
SSO (*Single Sign On*) 62
standards 260
start-ups 223
SuccessFactors 199
Sun 167
synchronisation 131

T

tablettes 16
TCO 36, 46, 84
téléprésence 174
temps de latence 214
test de montée en charge 223
théorème de CAP 219
tiers archiveur 189
Tim Berners-Lee 5
Time to market 89
Time to market 27, 109
tolérance 219
    aux pannes 228
traçabilité 68
trafic réseau 103
Tunnel SSL 235

U

uptime 26
urbanisation 100
usine
    de développement 97
    logicielle 156
utilisateurs 89, 113

V

vCenter 252
vCloud Director 252
versioning 178, 236
virtualisation 46, 218
visioconférence 174
Visual Studio 245
VMware 168
VPN 63
VPN IPSEC 235
vSphere 252

W

web 5
web 2.0 10, 39
webconférence 174
WebEx 184
Wikipedia 10
Windows
    Azure 244
    Live 182

X

Xiti 195

Z

Zoho 183
    CRM 197
Zoho People 200