

## Conclusions

Tout au long de l'histoire de l'informatique on ne peut s'empêcher de secouer la tête concernant toutes les décisions prises ou petites phrases qui semblent complètement idiotes aujourd'hui. Thomas Watson Jr qui voit « un marché mondial de cinq ordinateurs ». IBM qui a créé si peu de protections pour son PC qu'il en perd le contrôle. Steve Jobs qui, en 1984, refuse l'offre de Bill Gates de licencier MacOS. WordPerfect et Lotus qui misent sur OS/2 et loupent le tournant Windows. Les géants des moteurs de recherche qui se focalisent sur le "portail", laissant le champ libre à Google. Novell qui s'attaque à la bureautique plutôt que de s'étendre côté serveur. Le fondateur de DEC qui trouve qu'il n'existe aucune raison à aucun individu d'avoir un ordinateur à la maison. Etc. « *Hindsight is 20/20* » dit un proverbe américain. Autrement dit, avec du recul on a une vision parfaite.

Certaines de ces décisions malencontreuses étaient visiblement les mauvaises même à l'époque. Lorsque Be, le constructeur d'ordinateur créé par Jean-Louis Gassée, a sorti son ordinateur en 1995 (le BeBox), nombreux sont ceux qui ont réalisé que la société n'irait nulle part. Et de fait, Be a suivi les traces de NeXT, créé quelques années plus tôt : vouloir créer un micro-ordinateur de la nouvelle génération puis abandonner la partie matérielle pour se concentrer sur le système d'exploitation. Si M. Gassée est opérationnellement excellent (il a fait du très bon travail au sein d'Apple), n'est pas visionnaire qui veut (il n'en n'existe que très peu). M. Gassée aurait pu sauver la mise en vendant Be à Apple, mais il a pêché par arrogance en demandant beaucoup trop (15 % d'Apple) et en pensant qu'il était la seule option de la firme à la pomme. Lorsqu'il s'est présenté devant le comité de direction d'Apple en décembre 1996 pour plaider son cas, Gassée pensait que l'affaire était dans la poche et n'avait rien préparé. Face à Steve Jobs qui a su être brillant, il n'a pas fait le poids.

Mais dans la plupart des cas, les décisions qui ont été prises avaient énormément de sens à l'époque. Par exemple, on a souvent eu affaire à des compagnies "chemin de fer" qui n'ont pas su agir -ou même penser- comme des cow-boys : Novell qui a ignoré TCP/IP, Microsoft qui a ignoré Internet puis Google, DEC qui a ignoré la micro-informatique. Aucune de ces compagnies n'était gérée par des incompetents, bien au contraire. Mais si une compagnie de cow-boy a tendance à se transformer en une compagnie "chemin de fer" au fur et à mesure qu'elle grandit, la tendance inverse est quasiment impossible. C'est la même raison qui explique qu'énormément de standards qui se sont imposés ont été boudés par les experts parce qu'ils étaient imparfaits : IP qui fonctionne sous le principe de "meilleur effort" (au grand dam des télécoms), Ethernet qui accepte le principe de collisions (contrairement à Token Ring), HTML qui n'a pas de base de gestion de liens, HTTP contre CORBA, HTML contre Java ou ActiveX, PHP contre Java, le PC contre le reste de la micro-informatique, etc.

Il est très difficile lorsque l'on a développé une attitude "chemin de fer" pour des raisons tout à fait nobles (les télécoms ont dû apporter très tôt un certain niveau de qualité) d'avoir une attitude de cow-boy et de renier des réflexes renforcés pendant des années.

Toute industrie est souvent créée par des individus (la plupart du temps amateurs) et évolue par la suite en un monde où seules les grosses entreprises peuvent survivre. Au départ, les "cow-boys" règnent sur un marché où tout est à construire. Mais petit à petit des lignes de chemins de fer sont mises en place, amenant avec elles un bouleversement du paysage. C'est par exemple le cas de l'industrie automobile. Louis et Marcel Renault, Henry Ford ou Karl Benz ont créé leur compagnie en concevant une voiture dans leur garage. Certains, comme Henry Ford, ont ignoré la logique conventionnelle de l'époque à un tel point qu'ils ont dû passer pour des fous. Par exemple, Henry Ford a été un patron qui a voulu faire dans le social — en 1908 ! Non seulement il a eu la vision d'une voiture pour le peuple (la célèbre Ford T), mais il a payé ses ouvriers au double du prix couramment pratiqué, et a constamment baissé le prix de ses voitures alors que les carnets de commandes étaient pleins pendant des années. A tel point que Ford a été pris pour un ahuri (voire

Cow-boys contre chemin de fer ou que savez-vous vraiment de l'histoire de l'informatique ?

un fou dangereux) dans les milieux d'affaire de l'époque et a même été poursuivi en justice par ses actionnaires.

Mais, au fur et à mesure que cette industrie s'est développée, les professionnels ont remplacé les amateurs, les sociétés sont devenues de plus en plus grandes, si bien qu'à l'heure actuelle il est quasiment impossible de se tailler une place dans cette industrie tellement les investissements sont importants. Quant aux PDG des grands groupes automobiles, aucun n'a le centième de l'audace d'un Henry Ford. Tous les géants actuels de l'automobile sont arrivés très tôt sur ce marché. Les meilleures chances de nouveaux venus proviennent désormais de pays émergents tels que l'Inde ou la Chine. L'automobile n'a pas eu son Microsoft ou son Google.

L'informatique par contre a suivi une approche différente. En effet, elle a commencé par être menée par des grosses entreprises et avec le temps a été de plus en plus influencée par des nouveaux venus, voire même par des amateurs. En d'autres termes, elle a évolué d'un marché de professionnels à parfois un monde de cow-boys. Car l'informatique, contrairement à de nombreuses industries telles que l'automobile, a vu ses coûts de production diminuer avec le temps, permettant l'apparition de nombreuses innovations disruptives. En parallèle, si l'informatique est un domaine complexe comme toutes les industries hi-tech, elle offre un type de complexité différent. La biotechnologie demande par exemple de très grosses connaissances pour faire quoi que ce soit -d'où l'importance qu'attache ce secteur aux diplômés.

L'informatique par contre est plus comme le jeu d'échec : souvent facile de saisir les concepts de base, mais beaucoup plus ardu de maîtriser. Certes, certains langages de programmation demandent une formation, mais il n'y a pas besoin d'avoir une thèse pour programmer en C++ ou en Java. La complexité se trouve au niveau de la pratique. Écrire un programme qui puisse évoluer facilement sans avoir besoin d'être réécrit ou hideusement déformé. Écrire un programme qui puisse monter en charge, etc.

Parfois même, le challenge se trouve du côté fonctionnel et non du côté programmation. Hotmail ou YouTube n'ont pas demandé des miracles technologiques, mais plus une bonne idée de départ. Ces deux caractéristiques ont permis l'arrivée d'amateurs sur le marché, de plus en plus nombreux avec le temps.

Maintenant, pourquoi les professionnels les ont-ils laissés faire ?

« Je vois un marché mondial de cinq ordinateurs » aurait dit Thomas Watson en 1943. Si certains contestent le fait que le fondateur d'IBM ait dit cette phrase, elle n'en reste pas moins caractéristique de la vision de chaque vague informatique. Ces dernières, alors dirigées par des professionnels, n'ont imaginé qu'un monde où les ordinateurs devenaient forcément de plus en plus puissants et n'ont cherché qu'à étendre leur marché plutôt que d'en créer de nouveaux. Pour l'industrie des mainframes, la vision de l'avenir était sans doute de produire des machines toujours plus grosses auxquelles de plus en plus d'utilisateurs auraient accès. Mais un accès qui resterait en mode caractère et toujours dans un contexte strictement professionnel (hors de l'informatique d'entreprise point de salut). Si bien qu'ils ont complètement ignoré l'arrivée des minis, un monde créé par des amateurs s'adressant à des universitaires. En d'autres termes du menu fretin.

Et le monde de la mini-informatique a commis la même erreur. S'étant creusé son marché et dominant l'informatique d'entreprise grâce aux bases de données relationnelles, le monde Unix a ignoré l'arrivée de la micro-informatique (pensez donc, des machines surtout utilisées pour des jeux !) Ils ont donc laissé le champ libre à des amateurs tels que Steve Jobs, Steve Wozniak et autres Clive Sinclair d'utiliser des microprocesseurs bon marché et de créer Apple ou Sinclair. Ce phénomène s'est accéléré avec l'augmentation de l'importance du logiciel.

Ce dernier ayant des coûts de production très faibles, il a été encore plus facile de voir des amateurs créer leur société. Bill Gates et Paul Allen ont créé Microsoft. Mitch Kapor et Jonathan Sachs ont fondé Lotus. Un étudiant finlandais nommé Linus Torvalds a créé Linux. Richard Stallman (un universitaire du MIT), quant à lui, a lancé les bases du mouvement du logiciel libre.

Le paroxysme a été atteint avec le Web -créé là encore par un cow-boy dans son coin, Tim-Berners Lee, qui a osé ne pas écouter les recommandations de ses pairs. Le Web permettant non seulement de créer une application à des coûts très faibles (on peut développer un site Web uniquement avec des logiciels disponibles gratuitement) mais de distri-

buer une application le plus simplement du monde, l'informatique a été bouleversée par un monde de cow-boys se lançant à l'assaut d'un nouveau marché ressemblant au far-west. Des milliers d'amateurs (dont beaucoup d'étudiants) ont créé leur entreprise, beaucoup sans modèle économique qui tienne la route. Ce qui n'a pas empêché quelques-uns tels que Hotmail ou YouTube d'être rachetés à prix d'or. On a vu l'arrivée de nombreux géants, tous créés par des individus dans leur coin, et ce même après l'éclatement de la bulle Internet : Yahoo, eBay, Amazon.com, Google, Facebook, Twitter. Certains d'entre eux étaient des professionnels (comme Jeff Bezos d'Amazon.com), mais beaucoup étaient des étudiants.

L'émergence de l'informatique mobile a cependant affiché une tendance inverse et a tout de suite été professionnelle. On remarque en effet que les géants (Apple, Google, RIM, Motorola, ARM) proviennent d'autres industries. L'une des raisons est que l'informatique mobile est une évolution de la téléphonie mobile. Si c'est une technologie disruptive par rapport à la micro-informatique cela reste une technologie soutenant par rapport à la téléphonie mobile. C'est pour cette raison que des compagnies comme RIM, Nokia, Motorola ont tout de suite sauté sur le marché de l'informatique mobile car plus lucrative que la téléphonie mobile : un smartphone rapportant bien plus qu'un portable normal. L'informatique mobile a également servi de refuge à des compagnies telles qu'Apple ou ARM qui avaient perdu la bataille de la micro-informatique face au tandem Wintel. Google, quant à lui, est arrivé tard sur ce marché (tout comme IBM sur la micro-informatique).

De manière générale, le domaine du logiciel est plus fertile au domaine du matériel pour les cow-boys. Du côté matériel, les cow-boys n'ont en effet une chance que lorsqu'un nouveau type de machine apparaît (minis, micros). La valeur ajoutée du logiciel étant avant tout les fonctionnalités, n'importe qui peut potentiellement trouver la prochaine killer app. Il est au contraire de plus en plus difficile de voir des nouveaux géants du matériel. Il est apparu une quantité limitée de nouveaux marchés du matériel. Au-delà de l'informatique mobile, où peuvent se développer de nouveaux venus ?

Mais au-delà de l'anecdote, le fait que l'informatique ait tant de fois été influencée par des amateurs a eu de profondes conséquences. L'une d'entre elles est que cette industrie a attiré de très nombreux entrepreneurs, même plusieurs décennies après sa création. « Un entrepreneur », a une fois dit Reid Hoffman, le fondateur de LinkedIn.com, « est quelqu'un qui saute d'une falaise et qui assemble un avion pendant la chute... ». Ces cow-boys modernes, souvent un rien tête brûlée, sont en effet idéals pour lancer de nouveaux concepts et créer de nouveaux marchés. Il faut en effet une sacrée dose de confiance en soi (certains diront de folie) et de vision pour dépenser autant d'énergie sur un marché que personne d'autre ne voit.

L'autre conséquence s'appelle la simplicité. Car l'informatique, comme toute industrie hi-tech, va en se complexifiant. Si jusque dans les années quatre-vingt on pouvait encore entretenir soi-même sa voiture, c'est aujourd'hui impossible étant donné la quantité d'électronique désormais embarquée. De même, s'il était possible de bidouiller Windows 3.1 en éditant manuellement quelques fichiers systèmes, le système de registre de Windows XP/Vista/7 est devenu tellement complexe qu'il est très difficile pour un individu de s'y retrouver. La nature humaine est telle que l'on va pousser la limite d'un système, quitte à le complexifier le plus possible. Cette pente naturelle vers la complexité a tout de même été à la source de nombreux échecs. Les megaprojets d'entreprises ont trop souvent eu d'énormes retards et des surcoûts faramineux. CORBA, Java et ActiveX n'ont pas réussi à "améliorer" le Web. Trop de projets de CRM n'ont pas apporté le retour sur investissement espéré car leur cahier des charges trop épais a créé trop de complexité au-delà du raisonnable. Et les soi-disant "autoroutes de l'information" des années quatre-vingt-dix (pre-Web) s'orientaient vers des solutions propriétaires comme CompuServe ou America Online.

Les cow-boys ont réussi à influencer l'informatique en insufflant une dose de simplicité. Ken Thomson et Dennis Richie ont créé Unix comme une alternative plus simple au système d'exploitation Multics sur lequel ils avaient travaillé et qui était trop complexe. Le nom Unix est d'ailleurs un jeu de mot sur le nom Multics car il ne pouvait à l'origine supporter qu'un seul utilisateur simultané. L'Apple// a apporté aux particuliers (certes un peu fortunés à l'époque) un micro-ordinateur prêt à l'emploi. Le Web, quant à lui, a représenté quelque chose que l'on aurait pensé irréalisable : une plate-forme indépendante du lan-