

Cahiers de
UICONOMIE
Numéro 1, mai 2015



xerfi

Table des matières

Introduction, p. 3

Vincent Lorphelin et Francis Jacq, De la compétence enrichie à l'entreprise hyper-coopérative, p. 7

Pierre-Jean Benghozi, Nouveaux modèles d'affaires ou modèles iconomiques ? p. 17

Laurent Bloch, De Multics à Linux et au logiciel libre, p. 29

Jacques Printz, Parallélisme massif, Big Data et Intégration, p. 43

Michel Volle, Éléments de théorie « iconomique », p. 55

Introduction

Ce numéro des *Cahiers de l'iconomie* inaugure la publication des travaux de l'institut de l'iconomie¹.

*Qu'est-ce que
l'iconomie ?*

L'iconomie, rappelons-le, est le modèle d'« une société dont l'économie s'appuie sur la synergie de la microélectronique, du logiciel et de l'Internet² » et qui, étant par hypothèse parvenue à l'efficacité, serait sortie de la crise de transition actuelle. Il ne s'agit donc pas d'une prévision mais d'un *repère* proposé aux intentions et aux stratégies.

Le phénomène de l'informatisation se déploie sur les plans scientifique, technique, économique, psychologique, sociologique, culturel, philosophique, politique, géopolitique, etc : l'iconomie comporte donc l'ensemble de ces dimensions.

Nombreux sont cependant ceux dont la pensée résiste devant une telle diversité. Ceux qui s'intéressent aux conséquences psychosociales de l'informatisation, à ces « usages » que l'on a coutume de désigner par le mot *numérique*, sont en effet trop souvent tentés de mépriser la « technique », tandis que celle-ci accapare l'attention des spécialistes de l'informatique.

Il faut pourtant que le raisonnement embrasse toutes les dimensions du phénomène car la décision, qu'elle soit stratégique, opérationnelle ou tactique, ne peut faire abstraction d'aucune d'entre elles. Celui qui décide doit donc en posséder, à défaut d'une expertise spéciale approfondie, l'*intuition exacte* que les militaires

1. L'institut de l'iconomie (iconomie.org) est un think tank indépendant. Il bénéficie du soutien de Xerfi, qui édite les *Cahiers de l'iconomie*. Ses textes sont publiés sous la licence #FairlyShare (www.fairlyshare.com).

2. Michel Volle, *iconomie*, Economica, 2014, p. 200.

nomment « coup d'œil » et qui lui permettra d'agir de façon judicieuse alors qu'il se trouve confronté à un monde complexe et dangereux.

Adopter le point de vue de l'historien Pour y parvenir, il faut qu'il adopte le point de vue de l'historien :

- considérer le phénomène dans la dynamique qui le propulse vers une prospective, et non dans son seul état actuel ;

- saisir la diversité de ses dimensions, ce qui suppose de savoir raisonner selon le *modèle en couches* qui explicite la logique propre à chacune ainsi que les relations qui traversent leurs interfaces ;

- assumer le fait que l'on ne peut *penser* cet être organique qu'à travers des schémas simples et donc incomplets : bien choisis, ils révéleront sa nature mieux que ne peut le faire une description qui aspirerait à l'exhaustivité.

Cette démarche suppose une ouverture intellectuelle et une curiosité qui respectent les disciplines concernées sans s'enfermer dans leur formalisme ni rien céder au corporatisme des spécialistes.

* *

Des coups de projecteur

Chacun des articles réunis dans ce numéro éclaire, comme un coup de projecteur, un des aspects du phénomène. La juxtaposition des images que cela fournit aidera l'intuition du lecteur à progresser avec nous dans la compréhension des possibilités et des dangers que l'informatisation apporte à la société et dont la maîtrise fera enfin, nous l'espérons, émerger une économie.

L'évolution des entreprises

Les deux premiers articles sont consacrés à l'effet de l'informatisation sur la stratégie des entreprises : Vincent Lorphelin et Francis Jacq décrivent l'émergence de l'*entreprise hyper-coopérative*, point de convergence des modèles de plate-forme et d'entreprise 2.0. Pierre-Jean Benghozi déploie l'éventail des divers *modèles d'affaire*.

Enjeux prospectifs de l'informatique

L'informatique, *cause matérielle* de l'informatisation, subit une évolution rapide. Laurent Bloch explore sa dynamique et son conditionnement sociologique en évoquant l'histoire d'Unix et de Linux, Jacques Printz montre la complexité et les défis que présentent les exigences de l'intégration et celles, nouvelles, de la programmation parallèle.

Ces deux articles soulignent une évidence : un pays ne pourra

atteindre l'économie et réussir sa « transformation numérique » que s'il sait prendre l'informatique au sérieux.

Le dernier article, dont je suis l'auteur, est consacré à la théorie économique pure : il montre que dans l'économie informatisée les marchés des divers produits n'obéissent ni au régime de la concurrence parfaite, ni à celui du monopole naturel, mais à celui de la *concurrence monopolistique* qui est donc le *régime de référence* de l'économie. *Le régime de la concurrence monopolistique*

Les trois derniers articles auront une suite dans les numéros suivants de ces *Cahiers*.

Michel Volle

De la compétence enrichie à l'entreprise hyper-coopérative

Vincent Lorphelin et Francis Jacq

Au-delà de l'apport de productivité lié aux technologies relationnelles, un saut qualitatif transforme les ressources humaines et réinvente l'entreprise. Le modèle de plate-forme a fait le succès des géants d'Internet mais ses dérives sont dommageables pour la collectivité. Un nouveau modèle plus souhaitable se dessine cependant au point de convergence de la plate-forme et de l'entreprise 2.0.

* *

Dans le courant du XIX^e siècle la machine à vapeur apporte une force domestique considérable. En s'alliant à la mécanique, elle provoque le développement de l'usine aux dépens de l'atelier. Étant peu flexible, elle impose son rythme à la production. Avec l'organisation scientifique du travail, la compétence se spécialise jusqu'à ce que la main-d'œuvre devienne un auxiliaire de la machine.

La machine électrique est par contre flexible. Chaque poste de travail peut disposer de son propre moteur électrique, ce qui lui donne davantage d'autonomie dans la maîtrise de la force mécanique. Formé à l'utilisation d'une machine à coudre, un artisan couturier voit par exemple s'ouvrir sensiblement les possibilités de son métier. Sa compétence est enrichie, son savoir-faire est amélioré par la connaissance de nouvelles pratiques ou techniques. En revanche, si sa productivité est multipliée par 85, cela n'influence pas considérablement l'organisation de l'entreprise.

La téléphonie a un impact plus important. Tandis que le modèle dominant était celui de la vente en gros, demi-gros puis détail, accompagné par de lourds stocks intermédiaires, le téléphone fluidifie la relation avec les fournisseurs et les clients. Le métier et les compétences relationnelles des vendeurs, dont le rôle se bornait à remplir des carnets de commandes, sont enrichis par des pratiques collaboratives nouvelles. La coopération entre les entreprises en est améliorée au point de faire apparaître les directions commerciales et de réduire la chaîne des intermédiaires : organisée auparavant autour de l'outil de fabrication, l'entreprise se reconstruit progressivement autour de sa fonction commerciale.

Cette présentation est bien entendu simpliste puisque l'utilisation de chaque technologie a les trois types d'influence sur les compétences. Pour la téléphonie actuelle, par exemple, les appels à la chaîne depuis des centres d'appels pour vendre des fenêtres ou des assurances ne sont qu'une extension du taylorisme et une spécialisation des compétences. La télé-assistance, lorsqu'elle est effectuée par un technicien assisté d'un ordinateur, relève de la compétence enrichie. La conciergerie par téléphone proposée aux propriétaires de voitures de luxe est fournie par la coopération enrichie d'un réseau social de compétences mobilisables.

*Moins de
compétences
spécialisées, plus de
coopération enrichie*

Quelle est alors la différence entre la révolution industrielle précédente et celle d'aujourd'hui ? C'est une question de proportions entre les trois formes de compétences. Aujourd'hui les tâches spécialisées, répétitives, sont remplacées par des automates : le tri postal est réalisé par la reconnaissance automatique des adresses, les cabines de peinture d'automobiles sont robotisées, la saisie des données comptables est informatisée.

Les tâches fonctionnelles sont davantage aidées par des assistants virtuels : pilote automatique d'un avion ou d'une centrale nucléaire, outil de recherche de jurisprudence pour les avocats, logiciel de dessin en 3D ou anticipation de requête par un moteur de recherche sur Internet. Ces assistants déchargent les professionnels des tâches courantes tout en enrichissant leurs capacités d'initiative, de discernement ou de débrouillardise.

Les tâches relationnelles sont pour leur part considérablement améliorées. La relation client devient multicanal (SMS, téléphone, web, réseaux sociaux) et permet d'accompagner le consommateur

de manière pertinente et non invasive tout au long du cycle de vente. Dans leur dialogue avec leurs patients hospitalisés les médecins gagnent du temps et de la précision grâce à la consultation du dossier médical sur tablette. Les formateurs utilisent des murs interactifs pour faciliter la compréhension et l'apprentissage.

La coopération entre les entreprises devient plus imbriquée : blogs métiers pour la diffusion et le partage des meilleures pratiques, regroupement de communautés professionnelles interentreprises, ressources partagées (vitrine web, paiement ou logistique sur Amazon), plates-formes de logiciels *Open source*, coopération. Les fournisseurs et les clients participent à l'innovation, plus ouverte. Les coopérations de l'entreprise avec son écosystème sont ainsi enrichies par les nouvelles techniques.

Un nouvel équilibre des compétences réinvente l'entreprise

Le changement de proportion entre les trois formes de compétences a des conséquences radicalement différentes. Du point de vue de la nature du travail, la révolution industrielle du XIX^e siècle a essentiellement provoqué le développement de la compétence spécialisée, ou *main-d'œuvre*. Celle que nous vivons provoque surtout celui du *cerveau d'œuvre* dont les compétences et coopérations sont enrichies par les techniques relationnelles.

*De la main d'œuvre
au cerveau d'œuvre*

Du point de vue des ressources humaines, l'intelligence relationnelle est mieux reconnue, ainsi que les services à la personne pour leur apport au bien-être de la population, donc à l'économie. Les formations professionnelles, concentrées auparavant sur la relation avec la matière (esprit de géométrie), s'étendent à la relation avec des personnes (esprit de finesse).

Du point de vue de la compétitivité, les gains de productivité résidaient au XX^e siècle dans la transformation de la matière, à laquelle l'organisation hiérarchique et taylorienne de l'entreprise était adaptée. Or celle-ci est confrontée maintenant à davantage de diversité, d'instabilité et de complexité. Leurs répercussions en interne provoquent incohérences stratégiques, inflation procédurale, spirale bureaucratique et perte d'efficacité.

*Le réseau des
partenaires*

Dans ce contexte, l'essentiel des gains de productivité réside dans le bon fonctionnement du réseau de partenaires qui coopèrent

à l'élaboration des produits. La coopération enrichie en est le facteur-clé.

Cette coopération favorise une organisation en mode projet qui fédère les compétences en réseau à l'intérieur ou à l'extérieur de l'entreprise et utilise les fonctions support (DRH, DSI, administration) comme des ressources partagées. Le décloisonnement, l'utilité, la traçabilité et la réputation remplacent la hiérarchie, le temps ouvré, la mission et l'évaluation.

Ce changement de proportion engendre une nouvelle économie de la coopération : l'entreprise se réinvente en tant que lieu d'organisation des coopérations pour valoriser le capital relationnel des individus et le capital social collectif.

Le modèle de plate-forme est au cœur de l'ère Google

Cette réinvention s'inspire mais dépasse le modèle de plate-forme, qui fait par exemple le succès de Google : *Google docs* permet à des collaborateurs de travailler sur un document partagé. *Youtube* et *Picasa* fournissent des ressources pour partager vidéos et photos. *AdSense* permet de monétiser l'audience d'un site web. De même pour *Facebook*, *Amazon*, *Flickr*, *SalesForce*, *Paypal*, etc.

En soi le concept de plate-forme n'est pas nouveau puisqu'il n'est qu'une forme moderne des marchés bifaces : un magazine organise une transaction tacite entre deux clients, l'annonceur et le lecteur. Une carte de crédit s'adresse à un commerçant et un consommateur. Une agence immobilière met en relation un vendeur et un acheteur.

La force d'une plate-forme est de multiplier les effets d'externalités : le développement de chaque face est bénéfique à l'autre face. Dès lors, l'avantage concurrentiel augmente avec la taille de la plate-forme jusqu'à créer un monopole de fait pour le leader d'un segment de marché.

En revanche, lorsque le marché des plates-formes est mondial, ce qui est le cas des plates-formes de données dont Google est emblématique, il conduit à des monopoles mondiaux qui sont tentés de monétiser les productions à leur unique profit. Du côté du consommateur, l'expression « *If you are not paying for it, you are the product* » est devenue classique. Du côté du producteur, le *crowdsourcing* est souvent dénoncé comme une nouvelle forme d'explo-

*Les plates-formes
sur internet, entre
prédation et
création de biens
communs*

tation de masse et le respect des droits de propriété intellectuelle des créateurs est un sujet de conflit majeur.

Comme les plates-formes de données déstabilisent et dominent souvent les secteurs qu'elles investissent (musique, tourisme), qu'elles récupèrent les biens communs, qu'elles entretiennent une pression à la baisse sur les prix, qu'elles créent peu d'emplois et qu'elles transfèrent leurs bénéfices vers les paradis fiscaux, certains n'hésitent pas à parler de *prédation* aux dépens des autres acteurs économiques.

Avec leur succès, les plates-formes de données ont cependant apporté en quelques années une diversification des applications et un enrichissement de la qualité des outils. Créer, exprimer, diffuser ou partager des informations est devenu un usage courant pour les internautes. Transposé dans le monde professionnel, cet usage a fait émerger de nouvelles pratiques managériales : l'animation de communautés sans hiérarchie, la gestion de coopérations synchrones ou différées, le dosage de la confiance, de la réputation et de la reconnaissance, l'équilibre entre objectif, règles et initiative. Dès lors, ces pratiques autour des *réseaux sociaux d'entreprise*, de la *gestion des connaissances* ou de l'*innovation ouverte* sont adoptées par ce qu'on appelle souvent les « *entreprises 2.0* ».

Pourtant, si les pratiques sont proches, les modèles de l'*entreprise 2.0* et de la plate-forme de données restent opposés sur le plan économique : les contributeurs sont rémunérés dans un cas, non rémunérés dans l'autre, la propriété intellectuelle est ou n'est pas revendiquée, les ressources sont ou ne sont pas subventionnées par de la publicité.

Ces modèles commencent néanmoins à se rapprocher par endroits (*entreprise étendue, open business, value-driven networks*). En anticipant leur point de convergence, on peut déjà décrire à grands traits celui de l'entreprise hyper-coopérative. En tant que plate-forme, cette entreprise met à disposition des ressources partagées parmi lesquelles des logiciels métiers qui permettent aux collaborateurs d'enrichir leurs compétences ; des outils collaboratifs, qui leurs permettent d'enrichir leurs coopérations ; des tutoriels, formations, services d'assistance, pour les accompagner ; des fonctions support stratégiques : ressources humaines, vision, objectifs, valeurs, règles et déontologie de collaboration, sécurité, secret industriel.

*L'entreprise
hyper-coopérative,
point de
convergence des
modèles de
plate-forme et
d'entreprise 2.0*

Les contributeurs travaillent pour plusieurs entreprises hyper-coopératives en fonction de leur disponibilité, intérêt et intérêt par projet. Le coût d'identification, de sollicitation, d'intégration et de fédération des talents est en effet largement compensé par les outils collaboratifs, ce qui permet une grande mobilité des ressources humaines. Le statut d'auto-entrepreneur unifié permet de cumuler des rémunérations diverses (auteur, salarié, inventeur, travailleur indépendant) sans complexité administrative. Pour gérer son temps de travail, le contributeur dispose d'un assistant virtuel, tableau de bord de ses savoir-faire, réputation, influence, opportunités, priorités, engagements, rémunération et développement personnel.

Vers l'économie de l'utilité ajoutée

L'économie de l'entreprise hyper-coopérative est fondée sur l'*utilité ajoutée*. L'utilité élémentaire de chaque contribution est calculée *a posteriori*, à partir du prix de vente d'un produit et du traçage de toutes les contributions à ce produit. La rémunération de chaque contributeur dépend de la valeur anticipée de l'utilité de ses contributions et de l'algorithme de rémunération. Celui-ci prend en compte des critères objectifs relatifs aux contenus (audience, référencement), à l'auteur (profil, réputation), au service associé (réactivité, rareté) et des critères subjectifs (clarté, originalité, convivialité).

L'entreprise hyper-coopérative évite deux écueils : celui de la grande entreprise actuelle, dont l'organisation hiérarchique provoque un coût exorbitant du « travailler ensemble » et un gâchis de compétences humaines. Ensuite celui du *crowdsourcing* des plates-formes de données, qui ignore la valeur économique des compétences.

L'entreprise hyper-coopérative offre une grande diversité de modes de collaboration : les décisions sont plus ou moins collégiales, les pouvoirs et responsabilités plus ou moins centralisés, les projets plus ou moins cadrés, les rémunérations plus ou moins dépendantes d'un succès collectif ou individuel, plus ou moins sécurisées ou incitatives.

Ce modèle est techniquement faisable : ses briques-clés sont toutes opérationnelles.

Il est socialement faisable : les précurseurs comme *MTurk*, qui

propose par exemple de supprimer les photos floues pour 1 cent par photo, existent depuis 2005. De manière plus récente, *Forbes* propose de publier un article contre une rémunération algorithmique dépendant de l'audience de l'article et de la fidélité des lecteurs à l'auteur. *Quirky* partage 10 % du prix entre plusieurs milliers de contributeurs à un produit, depuis celui qui a eu l'idée originale jusqu'à celui qui a voté pour le meilleur design, en application d'un algorithme fondé sur l'utilité des contributions au résultat final.

L'entreprise hyper-coopérative est faisable, souhaitable et dans l'air du temps

Il est souhaitable : il renforce l'esprit d'initiative, l'autonomie, la flexibilité du travail et la co-propriété industrielle. Il dénoue les tensions autour du temps de travail, du chômage, de l'employabilité, du marché du travail, du cloisonnement des activités, de la formation, de la hiérarchie, des baronnies et de la dérive bureaucratique. Il fédère l'énergie du *crowdsourcing* autour de la création de richesses partagées et d'emplois rémunérés.

Il est dans l'air du temps : depuis la révélation de la valeur économique des contributions bénévoles, notamment lors du rachat du *Huffington Post* par *AOL* en 2011, les contributeurs sont devenus exigeants. La rétribution est devenue une revendication forte, dans les milieux hacktivistes et au-delà, et cela engendre de multiples propositions comme le *micro-paiement universel*, la taxation des plates-formes qui tirent profit du travail gratuit ou la licence entre pairs.

L'avènement de ce modèle dépend donc de la seule innovation managériale. Les plates-formes nous ont déjà appris que le contributeur doit comprendre facilement le projet (vision, objectif, cadre,...), ce qu'on attend de lui (outils, processus, règles,...) et la contrepartie (réputation, influence, intégration sociale, sentiment de faire œuvre utile, plaisir de participer à un grand projet,...).

Les startups explorent le nouveau monde de l'entreprise hyper-coopérative

Le succès d'une plate-forme dépend de son « élégance », selon l'expression de Mark Zuckerberg, c'est-à-dire de la pertinence du mix projet/contribution/contrepartie/design. Ces règles restent valables pour l'entreprise hyper-coopérative avec l'ingrédient supplémentaire de la contrepartie monétaire, qui doit être jugée équitable.

Or si la mesure de celle-ci est évidente lorsque la contribution est le résultat d'une simple commande sur spécifications (*MTurk*), elle est en revanche plus délicate lorsque le mode de coopération devient

plus interactif. Cela soulève de nombreuses questions : où placer le curseur entre interne et externe, expertise et opinion collective, transparence et secret, création et perfectionnement, concours et partage de risque, propriété intellectuelle et transfert de droits ?

Ces questions stimulent l'imagination des explorateurs et startups, qui foisonnent pour proposer des mix innovants avec des contributeurs toujours plus professionnels et selon des modes toujours plus coopératifs : *Forbes.com*, *Quirky*, *Sensorica*, *Innocentive*, *Edison Nation*, *Local Motors*, *Nov'in*, *MyKompany*, *La Fabrique à Innovations*... Leurs résultats sont déjà substantiels : *Local Motors* regroupe 35 000 contributeurs et a levé 15 millions de dollars. *Quirky* fédère 900 000 contributeurs et génère un chiffre d'affaires de 100 millions de dollars, cinq fois plus qu'il y a deux ans.

Ainsi, grâce à l'accumulation de ces innovations managériales et malgré l'immensité du territoire qu'il reste à défricher, le modèle de l'entreprise hyper-coopérative se dessine rapidement ainsi que ses formidables implications sociales et économiques.

Bien qu'encore prospectif, ce modèle permet de fixer le regard sur un horizon probable pour orienter l'action des décideurs et les préparer - nous espérons utilement - à l'ère post-Google.

Bibliographie

Amandine Brugière. « Ce que le numérique fait au travail...et réciproquement ». *InternetActu*, 8 mars 2013

Amandine Brugière. « La métamorphose du travail ». *InternetActu*, 27 juin 2014

André-Yves Portnoff, *Le pari de l'intelligence... ensemble!*, Conférence du 24 mars 2011

Direction des Affaires Financières et des Entreprises, Comité de la Concurrence. *Table ronde sur les marchés bifaces*, Juin 2009.

FING. *Questions Numériques 2013/2014*, p.29

FING. *Repenser la place des individus au travail dans une société numérique*. Juin 2014

Jaron Lanier. *Who owns the future?*. Simon & Schuster, 2013

Jeff Jarvis. *La méthode Google*. Pocket, 2009

Lewis D'Vorkin. « Inside Forbes : amid the finger pointing, journalists need to explore new payment models ». *Forbes*, 18 mars 2013

Maria Mercanti Guérin, « La netnographie : outil de prospective des métiers, une application aux nouveaux métiers du Web », *Management & Avenir* 2009/5 (numéro 25)

Mathew Ingram. « Social Journalism and open platforms are the new normal - now we have to make them work ». 1er avril 2014

Michel Crémadez, *Organisations et Stratégie*, Dunod 2004

Michel Volle, *Iconomie*, Economica, 2014, p.103-106

P. Collin et N. Colin. *Mission d'expertise sur la fiscalité de l'économie numérique*. Ministère de l'Économie et des Finances, Janvier 2013.

P.Beraud et F. Cormerais. « Economie de la contribution et innovation sociétale ». *Innovations* numéro 34, Deboeck 2011

R. Parent et V. Chanal. « Quels business models pour les plateformes Web 2.0. : les apports de la théorie des marchés bi-faces ».

Ross Dawson. « Crowd business models ».

Ruth Simon. « One week, 3,000 product ideas ». *Wall Street Journal*, 3 Juillet 2014

SCC, « La mobilité change en profondeur les métiers », *ZDnet*, 16 juillet 2014

T. Claburn. « Oculus VR : crowdsourcing or mass exploitation? ». *Information Week*, 26 mars 2014

Vincent Lorphelin, *Le Rebond économique de la France*, Pearson 2012

Vincent Lorphelin. « L'ère post-Google a commencé ». *Les Échos*, 29 août 2013

Nouveaux modèles d'affaires ou modèles économiques ?

Pierre-Jean Benghozi

Directeur de recherche au CNRS et Professeur à l'École
polytechnique³

Le taux rapide de renouvellement des TIC est une évidence partagée et fréquemment soulignée. Les caractéristiques économiques plus spécifiques de ce trait sont moins souvent évoquées ou discutées. La structure particulière des changements techniques pèse pourtant d'un poids aussi important que leur fréquence.

L'émergence de l'économie numérique a consacré le terme de *business model*, qui fait désormais partie à part entière du langage du management et de la stratégie. Les évolutions qui se mettent en place autour des nouveaux réseaux montrent que ces modèles d'affaires ne sont plus simplement l'aboutissement et la concrétisation d'une succession de décisions stratégiques concurrentielles en matière de prix, de produit ou de relations clients.

C'est désormais dans la configuration de ces modèles d'affaires que se jouent, très directement, la concurrence entre firmes, la re-définition des chaînes de valeur et, plus profondément, la structuration des usages sociaux : modes de consommation, stimulation des échanges entre pairs, rapport au territoire. C'est cette dimension essentielle qui explique en particulier la multiplication, la diversité, mais aussi la forte instabilité des modèles d'affaires observables aujourd'hui sur l'Internet.

3. Ce texte reprend, avec leur aimable autorisation, une contribution déjà publiée dans les *Cahiers de l'Arcep* n° 10, avril-juin 2010

Classifier les modèles d'affaire

L'identification de ces modèles passe par la résolution de plusieurs questions qui, aussi simples puissent-elles être, fournissent les bases d'une caractérisation des différentes architectures de marché. Quels types de services sont offerts aux consommateurs ? Qui contrôle la relation finale avec le consommateur et qui peut vendre l'offre du service ? Faut-il un mécanisme de facturation, et si oui lequel utiliser pour procurer un revenu ? Quelles autres sources de revenu sont-elles envisageables (publicité, reversements) ? Comment se gère l'accès au réseau ? Qui assure la conception, le développement et la livraison de contenus attractifs ? Quelles sont les capacités d'investissement de chaque acteur ? Quelles synergies existe-t-il entre les activités existantes ?

*Le cas des modèles
low cost*

L'analyse du succès des modèles dits *low cost* permet par exemple de comprendre comment l'omniprésence, grâce aux TIC, des pré-occupations de contrôle de gestion peut permettre de gouverner la rentabilité et la performance des entreprises. La force de ces modèles tient moins à une recherche systématique d'économies et de réduction des coûts (« *cost killing* ») qu'à la capacité de penser rationnellement, mesurer, contrôler et optimiser dépenses, investissements et recettes grâce à l'utilisation méthodique des outils d'information.

C'est bien de modèles d'affaires d'ensemble dont il s'agit : leur spécificité tient justement dans la capacité de penser simultanément une réorganisation profonde de la production (*lean management*, usage intensif des technologies, gestion des systèmes d'information, traitement des *Big data*) et un positionnement stratégique tarifaire d'entrée de gamme. La capacité de repenser l'organisation et sa structure de coût est évidente dans des entreprises comme Easyjet ou Free, qui tirent leur force de l'évolution simultanée de l'offre proposée aux consommateurs et de la chaîne de valeur associée.

*Des possibilités
sans limite*

Se dessinent ainsi désormais des innovations au caractère à la fois modulaire et radical, facilitant des modalités élargies d'appropriation et ouvrant des possibilités d'action quasiment sans limite. Les différents acteurs économiques peuvent reconcevoir leurs offres et redessiner, dans tous les secteurs de l'industrie et des services, les structures traditionnelles de marché. Des produits et services analogues sont aujourd'hui conçus, envisagés, assurés et valorisés

à partir de positions et de ressources technologiques radicalement différentes.

Le renouvellement de l'offre de transports collectifs en donne un parfait exemple. A côté des sociétés de taxi traditionnelles, des services sont désormais proposés sous la forme de modes collaboratifs de covoiturage (Blablacar), en professionnalisant le recours aux voitures personnelles par le *crowdsourcing* (Uber), en développant des parcs de voitures partagées en location (Autolib), ou encore en élargissant et complétant l'offre de transports de voyageurs (SNCF).

Un bouleversement de l'offre

Ces possibilités illimitées de reconfiguration du système technique remettent brutalement en cause les modèles d'affaires et les structures compétitives des filières existantes au point qu'on les accuse de contourner leurs régulations et contraintes historiques. Le *crowdsourcing* et *crowdfunding* suscitent des formes de professionnalisation et de rémunération inédites (auto-entrepreneurs, auto-producteurs, etc.). L'économie collaborative élargit le marché de l'offre à des populations sans emploi ou en recherche de compléments de revenus et concurrentent, ce faisant, des modèles historiques parfois séculaires de prestation de services (hôtellerie avec AirBnB, taxis avec Uber, ingénierie et travail intellectuel avec Amazon Mechanical Turk, agences photographiques avec Istockphoto, etc.).

La substitution d'offres de service par abonnement aux ventes de produits permettent aux fabricants de garder sur la durée une relation traçable avec leurs clients au lieu de devoir reconstruire l'attractivité du produit, son positionnement concurrentiel et le déclenchement de l'acte d'achat pour chaque acquisition épisodique (opportunité recherchée notamment par les constructeurs automobiles grâce aux voitures électriques et connectées). Dans chacun de ces cas, la prise en compte de l'usage et du consommateur encourage des formes inédites d'intermédiation et de monétisation autour d'une plate-forme.

Dans l'économie informatisée, les innovations trouvent leur source dans des modes inédits d'articulation entre infrastructures, terminaux et applications. La labilité et la pervasivité des technologies favorisent en effet des recombinaisons constantes entre les divers re-

*La source des
innovations*

gistres techniques, les contenus, les services, les applications et les infrastructures. L'importance de cette articulation entre diverses couches techniques pour l'innovation et le développement de contenus n'est pas une nouveauté radicale : de récents travaux en management pointent le poids des écosystèmes industriels et de la complémentarité de leurs composants.

*Un antagonisme
inédit*

L'inventivité renouvelée de ces modèles est cependant aussi aujourd'hui le support d'un antagonisme inédit : il oppose les investissements et efforts d'innovation soutenant les modèles d'affaires, ceux qui s'opèrent sur la création de contenus et ceux qui portent sur le développement de la R&D et des infrastructures (réseaux, équipements, terminaux).

Ni les acteurs économiques, ni la puissance publique ne disposent toujours de moyens pour penser et anticiper ces évolutions⁴ et les divers pays agissent en ordre dispersé. C'est ce que montrent les débats récents autour d'Uber ou de la fiscalité des grandes entreprises dématérialisées (GAFA). De même des discussions européennes sont engagées autour du droit d'auteur pour répondre aux modes inédits de diffusion et de valorisation des services de *streaming* ou de vidéo dont le poids est croissant (Spotify, Deezer, Netflix...). Le débat engagé depuis une dizaine d'années - mais toujours vif - sur la « net-neutralité » témoigne également de la difficulté de tenir compte, dans la régulation, des nouvelles relations entre opérateurs et fournisseurs de services en ligne.

Les manières différentes de configurer des systèmes d'offre à partir de plates-formes ou bases techniques similaires contribuent ainsi au foisonnement et à l'instabilité des modèles d'affaires en concurrence : on trouve par exemple sur le seul marché français plusieurs centaines de sites qui proposent de la musique enregistrée en téléchargement ou en écoute, le nombre de sites de presse et médias est du même ordre de grandeur. Le développement des objets connectés en donne chaque jour de nouvelles illustrations.

Il existait déjà, avant l'arrivée de l'Internet, des prémises de cette économie de l'hyperoffre et, par voie de conséquence, de l'attention et de la prescription. Ce phénomène a plusieurs origines. Il résulte d'abord des stratégies éditoriales consistant à répondre au risque de production par la multiplication d'innovations testant

4. Benghozi, Gille et Vallée (2009)

auprès du public de nouveaux concepts et de nouvelles idées dans l'espoir que l'un d'eux décrochera le succès. Ces comportements s'accroissent sur l'Internet car ils s'y doublent d'autres effets. La numérisation et la quasi-absence de limitation physique au stockage et à la diffusion permettent une accumulation « mécanique » : l'ensemble des services et contenus disponibles ne fait que croître car chacun peut rester accessible. La masse des contributeurs amateurs ou aspirants professionnels disposés à mettre leurs productions en ligne pour se faire reconnaître contribue encore à l'explosion des contenus.

La valeur réside dans le modèle d'affaire

De manière paradoxale, face à une telle abondance de l'offre, ce ne sont plus nécessairement les contenus et les services qui ont une valeur en soi mais les modèles économiques de production et de consommation dans lesquels ils s'inscrivent. Dans le transport de voyageurs, par exemple, ce n'est pas seulement le déplacement entre A et B que cherche d'abord le consommateur, mais éventuellement une bonne qualité de transport, la fréquence du voyage, une prévisibilité permettant la meilleure anticipation des déplacements et des coûts, voire le simple niveau du prix⁵.

La multiplication de ces modèles et des manières de mettre des services à disposition n'est pas un phénomène transitoire lié à la phase d'émergence de l'économie informatisée : il résulte, plus largement, de stratégies systématiques d'innovation et d'exploration de modèles d'affaires à même d'assurer pérennité, rentabilité ou captation d'audience. D'où les mécanismes d'ajustement continu qu'opèrent les fournisseurs de contenus et de services en ligne pour renouveler leur offre et tester des solutions originales. Les cas de la presse ou du jeu vidéo sont tout à fait symptomatiques de ce point de vue : ils font régulièrement se succéder *paywall*, *freemium*, gratuité totale, micropaiements... La tentative récente d'Universal et d'autres majors du disque de remettre en cause le modèle *freemium* de Spotify au profit d'un basculement vers le tout payant est un

5. Certaines compagnies aériennes proposent ainsi des offres « à l'aveugle » sans préciser *a priori* la destination, et le succès de sites comme lastminute.com repose sur le choix d'une disponibilité et d'un prix plus que sur celui d'un lieu de villégiature.

bel exemple de la difficulté de maîtriser ces schémas radicalement nouveaux de la part des acteurs en place.

*Deux conceptions
du modèle d'affaire*

Une difficulté particulière tient à l'opposition entre deux conceptions radicalement différentes des modèles d'affaires, pour les startups notamment, selon que leur stratégie est de type industriel ou financier. Dans le premier cas, les entreprises recherchent une forme de rentabilité de l'activité permettant de générer marges commerciales et amortissement des investissements. Dans l'autre cas, au contraire, les entreprises, cherchent à assurer au premier chef leur croissance par la construction rapide d'une audience massive afin de stimuler une valorisation financière plus spéculative. C'est ainsi que la valorisation actuelle de Tesla, qui n'est encore qu'un petit constructeur innovant de véhicules électriques, égale celle de Renault, et que la valorisation de la plate-forme AirBnB a dépassé celle du groupe Accor.

Pour ces mêmes raisons les positions dominantes peuvent être très fragiles et la position des leaders connaître des renversements spectaculaires : pensons aux succès passés de Yahoo!, AltaVista, AOL, Blackberry, Nokia, etc. Cette situation a influencé la régulation de la concurrence : dans une jurisprudence de décembre 2013 à propos de l'acquisition de Skype par Microsoft le tribunal de l'Union européenne a déclaré que « dans la nouvelle économie des parts de marché très élevées et un fort degré de concentration – même 80 % – ne sont pas forcément des indices pertinents de pouvoir de marché, parce que ces parts de marché peuvent être éphémères ».

*L'apport de
l'Internet*

L'Internet a constitué un support incomparable pour favoriser, via l'émergence de ces nouveaux modèles d'affaires, l'apparition de nouveaux entrants, supports d'une inventivité constamment renouvelée et de positionnements inédits autour, par exemple, des fonctions d'agrégation et de recommandation. Dans ces mouvements, les industries de contenus ont été emblématiques de la manière dont les filières de distribution ont évolué dans d'autres secteurs industriels : par l'apparition de nouveaux entrants, puis par la concentration autour des acteurs historiques dominants du marché, ensuite par l'émergence de spécialistes dédiés à l'optimisation de certaines fonctions (achats et approvisionnements, flux transactionnels), enfin par la consolidation de plates-formes d'agrégation et de distribution usant d'une mobilisation intense des technologies.

Un bel exemple du succès et du caractère inédit de tels positionnements est donné par le groupe chinois Alibaba, fondé en 1999, introduit en Bourse à Wall Street en 2014 et valorisé à près de 200 milliards de dollars. Son succès a tenu à sa capacité de construire, autour d'un moteur de recherche et d'une plateforme de marché B2B⁶, un écosystème et modèle d'affaires permettant d'acheter ou de vendre en ligne partout dans le monde, en élargissant progressivement son offre au commerce électronique grand public, aux systèmes de paiement en ligne, au Cloud computing, etc.

De telles dynamiques conduisent à l'existence de structures alternatives concurrentes : des plates-formes issues des secteurs traditionnels de l'industrie et de la distribution d'un secteur donné, de nouveaux entrants construisant des positions puissantes à partir d'une offre d'agrégation et de la maîtrise des informations, des acteurs du commerce électronique se diversifiant à partir des fonctions logistiques et de vente en ligne. C'est tout l'enjeu, aujourd'hui, de la voiture connectée où des écosystèmes et consortiums concurrents se construisent autour d'Android ou de constructeurs automobiles.

De nouvelles formes de compétition s'esquissent ainsi autour des objets connectés : compétition à partir de la fourniture du service (EDF ou Veolia pour les compteurs intelligents), à partir de nouveaux entrants (effaceurs d'énergie proposant des compteurs), opérateurs télécoms intégrant la maîtrise des objets à partir de la gestion des réseaux et des abonnements, fournisseurs de matériels tels que les thermostats (Withings), plates-formes de données (Google⁷).

Les modèles économiques se structurent ainsi par une réorganisation des stratégies à l'intérieur des chaînes de valeur en affectant simultanément leur amont et leur aval, à l'image de Google, Face-

*Restructuration des
modèles
économiques*

6. L'importance des relations B2B soulignée par le succès d'Alibaba n'est pas exceptionnelle. Elle a été également la base du premier business model de Blablacar : des centaines de plateformes de covoiturage fournies aux entreprises en SaaS sur des Intranets. A côté de cette offre, le site grand public visait surtout, au départ, le covoiturage de longue distance le week-end. La société a ensuite évolué pour assurer la croissance et les coûts exponentiels du site grand public, qu'il était impossible de financer par la seule vente de plateformes de covoiturage au développement plus linéaire.

7. Google a ainsi récemment acheté pour 3,2 Mds\$ Nest, leader mondial des thermostats connectés.

book, Amazon ou Yahoo! qui investissent simultanément dans la fibre optique, des ballons ou des drones pour créer des infrastructures d'accès à l'Internet, et dans l'acquisition d'applications spécifiques à même d'enrichir leur plate-forme de services (WhatsApp, Waze, Tumblr, Instagram, etc.).

Dans un cas, il s'agit de contrôler une filière en construisant une position dominante d'agrégateur par la maîtrise amont, notamment, des cessions de droits d'exploitation et des accords d'exclusivité ou des infrastructures. Cette intégration des filières industrielles par l'amont s'explique par plusieurs facteurs : la sécurisation des approvisionnements exige d'instaurer des relations de longue durée entre fournisseurs et distributeurs et de rechercher l'exclusivité sur des produits à forte notoriété pour attirer des consommateurs et leur imposer des modalités de transaction (offres groupées, engagements de longue durée).

Dans l'autre cas, les acteurs cherchent à tirer profit de la dématérialisation et des effets d'échelles pour conquérir en aval de nouveaux marchés de niche, maîtriser et fidéliser les relations existantes avec les clients et réduire les coûts de transaction. Ce processus d'intégration par l'aval touche le cœur de l'activité des plateformes électroniques (base des données, gestion des catalogues électroniques, intégration informatique), les fonctions de personnalisation de l'offre au client (offres « customisées », propositions commerciales, ciblage) et la formalisation de la relation commerciale selon de nouveaux types de contrat qui favorisent la fidélisations et les abonnements.

L'enjeu du *Big data* est de valoriser la traçabilité, l'identification et l'historicisation des transactions, des utilisateurs, des usages, des ressources et données de tous ordres que permet l'informatisation. En rendant systématiques et automatiques la collecte et le croisement des données à une échelle radicalement nouvelle, ces outils statistiques ont élargi et transformé les méthodes du marketing : elles donnant désormais aux algorithmes et aux outils de suivi et de recommandation une toute première place dans les ressources stratégiques compétitives des entreprises.

Dans les industries de contenus plus encore que dans les secteurs industriels traditionnels, la capacité de concevoir de nouveaux services et de structurer de nouveaux modèles d'affaires tient à l'articu-

lation qui peut s'opérer entre les supports technologiques de l'information et les nouvelles formes de diffusion des contenus. L'arrivée actuelle sur le marché des biens culturels des offreurs de technologie ou d'opérateurs de télécommunication ne se traduit donc pas seulement par un rééquilibrage, elle bouleverse les modèles d'affaires et l'architecture des filières économiques du contenu : dématérialisation des supports, forfaitisation des achats et gratuité sont les pointes émergées de cet iceberg.

Le secteur des industries culturelles a été, à bien des égards, un laboratoire d'expérimentation précurseur des nouveaux modèles d'affaires. Citons celui, important et tout à fait singulier, de la gratuité dont les modalités existent depuis longtemps dans les médias (pensons à la radio ou la télévision) et grâce auquel la culture a largement exploré les solutions économiques associées.

*L'exemple donné
par les industries
culturelles*

Les raisons de ce caractère exemplaire tiennent à la fois aux spécificités de ces industries et à leur place dans l'économie. Si les industries culturelles apparaissent aussi inventives et « créatives » dans leurs modèles d'affaires, cela tient à certaines des caractéristiques structurelles qui en font un secteur emblématique : la dimension immatérielle de leurs contenus, l'articulation spécifique qu'elles opèrent entre valeur symbolique et valeur d'usage, le renouvellement permanent des formes et des contenus stylistiques, l'exceptionnel attrait des consommateurs pour des contenus ayant soutenu la croissance des GAFAs et motivant une large part des abonnements aux infrastructures de réseaux à très haut débit.

Cette position d'avant-garde se retrouve sous plusieurs formes : poids des nouveaux entrants dans des secteurs installés (Amazon dans l'édition, Apple dans la musique), caractère disruptif des technologies supports et des services associés (Google Books dans l'édition), radicalité des formes alternatives de valorisation (ventes de CD remplacées par le téléchargement de titres à l'unité, puis par des formes d'écoute en ligne gratuite ou par abonnement), rôle prépondérant des portails communautaires et plates-formes d'agrégation de contenus (sites de photographie ou d'audiovisuel comme Flickr ou Dailymotion), mise en avant d'une économie de la prescription mettant en cause l'économie des systèmes classiques de distribution des contenus, y compris dématérialisés (Google Actualités), variété des modes de commercialisation ouverts en parallèle (jeux vidéos), etc.

Bibliographie

Anderson C. (2009). *Free! : the Past and Future of a Radical Price*, NYC, Hyperion.

Benghozi P.-J. (2012), ed., « Entreprises culturelles et internet : contenus numériques et modèles d'affaires innovants », *Culture-Media & Numérique*, Ministère de la Culture et de la Communication, Paris.

Benghozi P.-J., Lyubareva I. (2013), « La presse française en ligne en 2012 : modèles d'affaires et pratiques de financement », Ministère de la Culture et de la Communication, *Culture Etudes*, n. 3, juin, Paris.

Benghozi Pierre-Jean (2010) « Pourquoi parler de nouveaux modèles d'affaires ? » *Les cahiers de l'ARCEP*, numéro 2, avril, mai juin 2010, pp. 20-21.

Benghozi P.J., S. Bureau et F. Massit-Follea (2009) *The Internet of Things : What challenges for Europe ?*, (publication bilingue), Éditions MSH-collection praTICs

Benghozi P.J. and T. Paris(2007) « The economics and business models of prescription in the Internet » in Eric Brousseau, Nicolas Curien [eds.] *Internet and Digital Economics – Principles, Methods and Applications*, Cambridge (Mass), Cambridge University Press , pp. 291-310.

Chantepie Philippe, Alain Le Diberder (2010, 2005 1ère éd.), *Révolution numérique et industries culturelles*, Paris, La Découverte.

Chesbrough, H.W. (2003), *Open Innovation, the new Imperative for Creating and Profiting from Technology*, Harvard Business Press, Boston, MA.

Colin N. et H. Verdier (2012), *L'âge de la multitude, Entreprendre et gouverner après la révolution numérique*, Ed. Armand Colin

De Prato, G., Sanz, E., Simon, J.P, (ed.) (2014), *Digital Media Worlds ; The new media economy*, Oxford, Palgrave.

Delmas D. et F. Gueroui, 2014. *Cas pédagogique de l'aéroport international de Genève*, Université de Genève

Flichy, P. (2010), *Le sacre de l'amateur*, Paris, Le Seuil : La République des idées.

Gawer, A. (2009), Ed., *Platforms, Markets and Innovation*, Chel-

tenham, Edward Elgar.

Huberman B. A., Romero D. M., Wu F., 2009, « Crowdsourcing, attention and productivity », *Journal of Information Science*, 35 (6) 2009, pp. 758–765

Rheingold, H. 2002. *Smart Mobs : The Next Social Revolution*. New York, Basic Books.

Robert H. Frank et Philip J. Cook, 1996, *The Winner-Take-All Society : Why the Few at the Top Get So Much More Than the Rest of Us*, Reed Business Information

Surowiecki, J. 2005. *The Wisdom of Crowds*. New York : Anchor Books.

Zott, C. : R. Amit : and L. Massa. 2010. « The Business Model : Recent Developments and Future Research ». *Journal of Management* 37(4) : 1019-1042.

De Multics à Linux et au logiciel libre

Laurent Bloch⁸

L'objectif du projet Multics⁹ était de réaliser un grand système informatique capable de fournir des services interactifs en temps partagé à un millier d'utilisateurs simultanés. Multics a comporté beaucoup d'innovations de grande portée : le langage de commande pour piloter le fonctionnement de la machine était un langage de programmation, le même que celui dont disposait l'utilisateur pour interagir avec le système, le *shell* inventé par Louis Pouzin pour le prédécesseur de Multics, CTSS (cf. ci-dessous). Le système d'exploitation était écrit en langage évolué (en l'occurrence PL/1), voie ouverte par les systèmes Burroughs écrits en Algol, mais encore peu fréquentée.

Des innovations de grande portée

Les concepts de mémoire centrale pour les données volatiles et de fichiers pour les données persistantes étaient fondus en un concept

8. Ce texte est adapté de mon livre *Systèmes d'exploitation des ordinateurs : histoire, fonctionnement, enjeux* (disponible en ligne librement) avec l'aimable autorisation des Éditions Vuibert.

9. Multics est un système d'exploitation né en 1964 au MIT (*Massachusetts Institute of Technology*) dans le cadre d'un projet de recherche nommé MAC, sous la direction de Fernando Corbató (La même équipe avait déjà créé le système CTSS). Multics était destiné aux ordinateurs *General Electric* de la famille GE 635 pour lesquels le constructeur fournissait de son côté un système d'exploitation plus conventionnel. Le projet associait le MIT, *General Electric* et les *Bell Telephone Laboratories* (filiale d'AT&T, *American Telegraph and Telephone*, qui détenait le monopole des télécommunications aux États-Unis).

unique de mémoire virtuelle segmentée, certains segments étant dotés de la qualité de persistance. Les segments persistants étaient catalogués dans des répertoires à l'organisation arborescente. Moins spectaculaire en apparence mais peut-être aussi importante était l'existence d'une collection d'outils programmables et combinables entre eux destinés à faciliter le travail d'un type d'utilisateur : le programmeur, et plus spécialement celui qui écrivait un système d'exploitation.

Causes de l'échec

Malgré ou à cause de ces qualités uniques et promises à un grand avenir, Multics fut en gros un échec du point de vue de sa diffusion. Ses mérites furent reconnus tardivement, et le plus souvent tacitement, par ceux qui en reprirent les idées à leur compte.

La méconnaissance prolongée des mérites de Multics a ses raisons : la technologie des ordinateurs disponible à l'époque de sa diffusion, dans les années 1970, ne permettait de mettre en œuvre les méthodes et les concepts élaborés de ce système qu'au prix d'une grande lourdeur : les ordinateurs exploités sous Multics étaient donc gros, chers et lents. L'institut de statistique qui employait à l'époque l'auteur de ces lignes en avait acquis un, fruit de la fusion CII-Honeywell-Bull, et ses experts en informatique démographique avaient essayé d'imaginer les moyens de traiter avec cet engin les recensements de la population mais ils avaient fini par regarder ce drôle de système un peu de l'œil de la poule qui a couvé un œuf de cane.

Leur perplexité n'avait d'ailleurs d'égale que celle des ingénieurs commerciaux qui essayaient de vendre la chose, ou celle des ingénieurs spécialistes de Multics auxquels nous posions des questions incongrues comme « Pourrions-nous traiter des fichiers sur bande magnétique en traitement par lots ? », méthode béotienne mais indispensable à l'époque du fait de la faible capacité des disques.

Il résulta de cette expérience peu de résultats nouveaux dans le travail statistique courant, mais un sursaut intellectuel parmi ceux des informaticiens qui n'avaient pas définitivement sombré dans la léthargie coboliste¹⁰ dont on n'émergerait que pour sombrer dans

10. Adjectif formé sur le nom de COBOL, langage de programmation peu prisé malgré de réelles qualités parce que destiné aux informaticiens de gestion, placés au bas de la hiérarchie implicite de la profession. Ses dignes successeurs sont SQL et Perl.

Windows.

Si changer de langage ou de système ne rend pas plus intelligent, certains systèmes et langages incitent à la réflexion, d'autres moins. Cela dépend pour une grande part du type d'initiative laissé à l'utilisateur et du niveau d'intelligibilité que le système exhibe. Les paramètres de fonctionnement de certains systèmes sont enfouis dans des fichiers binaires inaccessibles à l'utilisateur, qui n'a alors qu'une seule possibilité : cliquer sur des menus en espérant que cela finira par produire le résultat désiré ; c'est le cas de Windows. Unix au contraire place tous ses paramètres dans des fichiers texte lisibles et réunis dans un endroit connu (le répertoire `/etc`).

L'accessibilité du système

La première méthode n'est justifiable que si la réalisation du système est sans faille et autorise l'utilisateur à ne jamais se préoccuper de ces paramètres, ce qu'ont réussi les concepteurs de MacOS mais pas ceux de Windows ; elle suppose aussi que lesdits utilisateurs ne sont que cela, des utilisateurs, qu'ils ne nourrissent aucun intérêt pour le système et n'envisagent pas une seconde de le modifier, ou du moins de modifier son comportement en jouant sur les paramètres.

Multics, sous cet angle comme sous certains autres, était un précurseur d'Unix, système qui considère ses utilisateurs comme des personnes intelligentes et suffisamment intéressées par le système lui-même pour lui consacrer un travail dont les employeurs ne perçoivent pas toujours la fécondité. C'est ainsi que des industriels ont inventé pour Unix une interface nommée CDE (*Common Desktop Environment*) dont le principe est de plaquer sur le système une sorte de super-Windows dont le paramétrage, réservé à des administrateurs, est ensuite propagé à la masse des utilisateurs. Cette vision centralisée et hyper-organisée aurait sans doute réussi dans les années 1960 mais elle risque de ne pas résister aux sables mouvants de la sociologie réelle des organisations des années 2010.

Un système pour utilisateurs intelligents

Où l'on commence à rêver à Unix

Les créateurs d'Unix, Ken Thompson et Dennis M. Ritchie, avaient une forte expérience de Multics, et ils savaient aussi bien ce qu'ils voulaient en retenir que ce qu'ils en rejetaient. Ils en renaient notamment les aspects suivants :

- Le système est écrit non pas en assembleur, mais dans un lan-

gage de haut niveau (PL/1 pour Multics, C pour Unix), seuls quelques fragments du code du noyau intimement liés à un matériel particulier étant en assembleur. Ceci facilite le portage¹¹ du système sur un nouveau modèle d'ordinateur. On a pu dire que le langage C était un « assembleur portable ».

- Le système de commandes est le même interpréteur que celui qui permet à l'utilisateur d'exécuter des programmes et des commandes, et il donne accès à un langage de programmation. C'est le *shell*.
- Le système de fichiers d'Unix est inspiré de celui de Multics, d'où vient aussi l'idée d'exécuter chaque commande comme un processus distinct.
- Mais surtout, comme Dennis Ritchie l'a expliqué dans son article de 1979, ses collègues des Bell Laboratories et lui voulaient retrouver dans Unix un système qui, comme avec Multics, engendrerait pour ainsi dire spontanément la communication et l'échange d'expériences entre ses adeptes.

Le fait qu'Unix soit, comme Multics, propice à la création d'une communauté ouverte mérite que l'on s'y arrête. Lorsque Multics a été introduit dans l'institut statistique évoqué ci-dessus, il a immédiatement cristallisé la formation d'une petite communauté intellectuelle, que la Direction n'a d'ailleurs eu de cesse de résorber parce qu'elle n'en comprenait pas la fécondité et qu'elle percevait son activité comme un gaspillage. D'innombrables expériences similaires ont eu lieu autour de Multics et d'Unix sans qu'une cause unique puisse leur être attribuée.

Le fait que ces systèmes aient été créés par des chercheurs habitués à l'idée que la connaissance soit objet de partage gratuit et de communication désintéressée, mais assortie de plaisir, est un élément. L'existence de logiciels commodes pour la création de textes, le courrier électronique et les forums en ligne a aussi joué, mais cette existence était-elle une cause ou une conséquence ? La nature

*Un usage
collaboratif*

11. *Porter* un logiciel d'un ordinateur à un autre ou d'un système à un autre signifie l'adapter aux caractéristiques techniques particulières de ce nouveau contexte. L'opération s'appelle un *portage*. Un logiciel pour le portage duquel le travail à faire est nul ou négligeable est dit portable ; cette qualité, très recherchée, s'appelle *portabilité*. Unix est le système d'exploitation le plus portable parce qu'il est écrit pour l'essentiel dans un langage évolué (C) plutôt que dans l'assembleur d'un processeur particulier, et aussi grâce à la bonne abstraction de ses primitives et la simplicité et à l'élégance de son architecture.

programmable du *shell*, l'accès possible pour tous aux paramètres du système inscrits dans des fichiers de texte ordinaires, encourageaient un usage intelligent du système et l'intelligence va de pair avec l'échange.

Si l'on compare Multics et Unix aux systèmes industriels disponibles à l'époque, comme l'OS 360, GCOS 8 ou plus tard VMS de *Digital Equipment*, il apparaît que ces derniers n'ont pas été conçus dans le même esprit : l'utilisateur dispose d'un mode d'emploi du système, réputé contenir toutes les solutions pour les problèmes qu'il pourrait se poser. Lorsque tout ceci est bien fait, comme par exemple dans VMS, système à mémoire virtuelle conçu pour l'ordinateur VAX, cela suscite un usage du système efficace et commode, mais passif.

L'auteur de ces lignes a été un utilisateur longtemps réticent et sceptique d'Unix car il était dérouté par l'aspect « boîte à outils » du système. VMS, que je pratiquais simultanément, avait été conçu par une équipe de (très bons) ingénieurs, soucieux de livrer un produit homogène et cohérent, et ils avaient parfaitement réussi. La meilleure preuve de cette réussite était la documentation du système, souvent un aspect un peu négligé : celle de VMS était une merveille de clarté et d'exhaustivité, au prix d'un nombre impressionnant de mètres linéaires de rayonnage : quel que soit le problème à résoudre, on était sûr que la réponse était dans « la doc ». Lorsque *Digital Equipment* a produit sa propre version d'Unix, il a publié un petit manuel des commandes Unix baptisé « *The little grey book* » (la couleur canonique de la documentation Digital venait de virer de l'orange au gris). Par opposition, la documentation VMS a été baptisée « *The big grey wall* ».

*La simplicité
d'Unix*

Habitué donc à l'univers confortable et hyper-balisé de VMS, je découvrais avec Unix un système de prime abord beaucoup moins homogène, même si je devais voir plus tard que son homogénéité résidait ailleurs. Comme chaque commande Unix s'exécute sous le contrôle d'un processus distinct, elle peut être découplée du noyau du système. Cette modularité, qui est un avantage, avait permis de confier l'écriture de beaucoup de commandes à des étudiants en stage, quand elles n'étaient pas tout simplement des contributions spontanées, et alors leur qualité et celle de leur documentation pouvaient être assez inégales.

De Multics les créateurs d'Unix ont rejeté la lourdeur. La tentation fatale, pour des auteurs de systèmes informatiques en général et de systèmes d'exploitation ou d'architectures de processeurs en particulier, consiste à céder au perfectionnisme et à réaliser des dispositifs qui ajoutent au système global une complexité considérable pour résoudre des problèmes qui ne surgiront que très rarement. Or les problèmes rares peuvent se contenter de solutions inefficaces mais simples. Les auteurs de Multics n'avaient pas évité cette ornière, VMS non plus d'ailleurs, qui succédait aux merveilleux RSX-11M et autres IAS.

*Le syndrome du
second système*

Frederick P. Brooks, le concepteur de l'OS/360, dans le livre justement célèbre *The Mythical Man-Month*, décrit ce qu'il appelle le « syndrome du second système » et qui s'applique à Multics comme à l'OS/360 : une équipe constituée en grande partie des mêmes hommes autour de Fernando Corbató avait développé avec succès CTSS ; en s'attaquant à Multics ils ont voulu y introduire tous les perfectionnements coûteux qu'ils avaient, avec sagesse mais frustration, écartés de leur œuvre précédente. En informatique comme ailleurs, point de salut sans une part de renoncement.

En fait l'échec de Multics aux Bell Labs était patent à la fin des années 1960. L'équipe qui allait y concevoir Unix comprit que Multics ne serait pas utilisable pour un travail réel dans un délai raisonnable. De son côté le propriétaire de Multics, *General Electric*, se sentait assez peu concerné par ce système développé par des chercheurs universitaires et préférait commercialiser ses ordinateurs avec son système conventionnel, GECOS. Lorsque Multics deviendra utilisable, à la fin des années 1970, les ordinateurs qu'il pouvait piloter étaient définitivement périmés et sans espoir de succession.

Dans un article de 1979 Dennis Ritchie a décrit la période où les Bell Labs se retiraient du projet Multics. Ce processus s'accompagnait d'un autre facteur d'incertitude : une réorganisation visait à séparer les équipes de recherche en informatique des équipes de l'informatique opérationnelle ; ce genre de séparation, conforme aux vues des managers financiers à jaugeote courte, a pour conséquence habituelle de diminuer les moyens disponibles pour la recherche et de réduire la qualité de l'informatique opérationnelle, privée de stimulation intellectuelle.

Le groupe de D. Ritchie, K. Thompson, M. D. McIlroy et Joseph F. Ossanna souhaitait conserver l'environnement de travail luxueux que Multics leur procurait à un coût d'autant plus exorbitant qu'ils en étaient les derniers utilisateurs. Pour ce faire ils ont développé leur propre système sur un petit ordinateur bon marché et un peu inutilisé récupéré dans un couloir, un PDP 7 de *Digital Equipment* : Unix était sinon né, du moins conçu.

Les hommes d'Unix : sociologie des sciences et des hiérarchies

On rencontre peu de femmes dans cette histoire. Raison de plus pour mentionner Evi Nemeth (disparue en mer en 2013 entre l'Australie et la Nouvelle-Zélande), et, peut-être pas tout à fait dans le même domaine ni à la même époque, Radia Perlman, spécialiste des protocoles de réseau¹², et Elizabeth Zwicky.

Le lecteur, à la fin de l'alinéa précédent, se sera peut-être fait la réflexion que pour que des employés d'une grande entreprise puissent développer un système d'exploitation, même ascétique, pendant leur temps libre, il fallait que leur encadrement ne fût pas trop rigide. Ce même lecteur devrait maintenant être convaincu que le système d'exploitation est l'objet technique le plus complexe que l'homme ait conçu et réalisé au cours du XX^e siècle. Quelle était au fait la mission théorique de ces jeunes gens ? Qui contrôlait la réalisation de leurs objectifs ?

Le livre de Peter H. Salus, *A Quarter Century of UNIX*, met en scène les principaux acteurs de la naissance d'Unix. On découvre en le lisant que cette création, qui a eu des répercussions considérables dans les domaines technique autant qu'industriel et économique, n'a vraiment été décidée ni par un groupe industriel, ni par un gouvernement, ni par aucun organisme doté de pouvoir et de moyens financiers importants. On peut d'ailleurs en dire autant de l'Internet, autre création aux répercussions considérables et d'ailleurs très liée à Unix et issue du même milieu social.

*Une sociologie
étrange mais
féconde*

Quiconque a fréquenté les milieux scientifiques d'une part, les milieux industriels de l'autre, ne peut manquer d'être frappé par le

12. Notamment le *Spanning Tree Protocol* grâce auquel un réseau Ethernet fonctionne.

caractère décalé pour ne pas dire marginal de la plupart des acteurs de la genèse unixienne.

La hiérarchie des sciences

Le monde de la science a sa hiérarchie, où les disciplines spéculatives et abstraites ont le pas sur les recherches appliquées et les disciplines descriptives, et où bien sûr (surtout en France) les chercheurs sont patriciens et les ingénieurs et techniciens hilotes, entourés d'une population au statut incertain, les étudiants en thèse ou en post-doc dont une minorité d'élus accédera au patriciat mais dont la majorité ne deviendra même pas hilote car elle sera contrainte à descendre aux enfers, c'est-à-dire dans le monde réel des entreprises industrielles et commerciales.

Pourquoi les systèmes d'exploitation sont (bien à tort) méprisés

Dans cet univers social, l'informatique, discipline récente et mal identifiée, est perçue (au mépris de toute vraisemblance, mais qu'importe au sociologue) comme un vague sous-produit de la branche la moins noble des mathématiques (l'analyse numérique) et se situe plutôt vers le bas de l'échelle de la légitimité. Au sein de la discipline informatique, le haut du pavé est tenu par les domaines où existe une théorie de préférence mathématique ou à la rigueur physique : linguistique de la programmation, algorithmique (surtout numérique ou logique), traitement de l'image ou du signal en général. Les systèmes d'exploitation disposent de tout un arsenal de concepts, mais non d'une théorie, le fait est ; de surcroît ils sont bien près du matériel, qui a des relents de cambouis et de sueur. Donc ils sont classés tout en bas, ainsi que les ingénieurs qui s'en occupent.

Le monde industriel (nous nous plaçons à l'époque de la naissance d'Unix, avant la prise de pouvoir par les financiers) a (avait ?) un système de valeurs symétrique de celui de la science : on y respecte (au moins dans les entreprises dotées d'une tradition industrielle et qui ont su la conserver, ce qui est rare en France) celui qui fait des choses, de vraies choses. C'est un univers dominé par les ingénieurs censés se coltiner avec la matière. On sait bien qu'une industrie dynamique doit avoir des centres de recherche, et que dans ces endroits travaillent des êtres bizarres appelés chercheurs, mais même si on ne les méprise pas vraiment ils sont considérés avec une certaine distance.

Or que nous apprend Salus ? Thompson et Ritchie étaient chercheurs dans une entreprise industrielle. Au fur et à mesure de leur

apparition, les noms de ceux qui ont fait Unix, parmi eux Kirk McKusick, Bill Joy, Eric Allman, Keith Bostic, sont toujours accompagnés d'un commentaire de la même veine : ils étaient étudiants *undergraduates* ou en cours de PhD, et soudain ils ont découvert qu'Unix était bien plus passionnant que leurs études.

Bref, les auteurs d'Unix n'ont jamais emprunté ni la voie qui mène les ingénieurs vers un fauteuil de Directeur Général, ni celle que prennent les bons étudiants vers la tenure track, les chaires prestigieuses, voire le Nobel ¹³.

Introduction à la démarche unixienne

Comme le note Christian Queinnec aux premiers mots de son livre *ABC d'Unix* « UNIX est un système de production de programmes ». Conformément à l'esprit d'Unix, cette assertion est à prendre à la fois de façon extensive : ce système comporte tout ce dont peut rêver un auteur de programmes, et aussi de façon restrictive : malgré quelques concessions récentes, il ne comporte fondamentalement rien d'autre. Les Unix modernes tels que Linux sont certes dotés de logiciels utilisables par le commun des mortels, avec des interfaces graphiques, mais les vrais unixiens n'en abusent pas.

*Un système de
production de
programmes*

La documentation canonique d'Unix (les *man pages*) constitue une excellente entrée en matière : aucun effort pédagogique, aucune des redondances qui facilitent l'acquisition d'une notion. Les mots sont comptés, aucun ne manque mais aucun n'est de trop. La lecture attentive (très attentive) de ces pages délivre l'information nécessaire et suffisante à l'usage du système, d'où une locution proverbiale souvent proférée par les Unixiens expérimentés en réponse

13. On sait qu'Alfred Nobel, lorsqu'il créa ses Prix, ne voulut pas en attribuer aux Mathématiques. La légende dit que cette exclusion serait due à une trop grande sympathie de Mme Nobel pour un mathématicien. Pour se consoler les mathématiciens ont créé la médaille Fields, décernée tous les quatre ans. Les informaticiens ont encore plus besoin de consolation, puisqu'ils n'ont même pas réussi à séduire Mme Nobel. Ils ont créé le Turing Award qui a été décerné, parmi nos personnages, à Maurice V. Wilkes, E.W. Dijkstra, Donald E. Knuth, C. Antony R. Hoare, Frederick P. Brooks, Fernando Corbató, Ken Thompson, Dennis M. Ritchie et Leslie Lamport, sans oublier Joseph Sifakis, unique Français de ce palmarès. Voir le site de l'ACM pour plus de détails, <https://www.acm.org/>.

*Read the f...
manual!*

au néophyte qui demande de l'aide : « RTFM¹⁴ ! » (*Read the f... manual!*). On le voit, Unix est à l'opposé de ces logiciels à interface graphique dont les vendeurs laissent croire qu'ils peuvent être utilisés sans lire aucune documentation¹⁵.

À qui était, comme l'auteur, habitué aux systèmes des grands ordinateurs IBM des années 1970, ou au système VMS que *Digital Equipment Corporation* (DEC) avait développé pour ses ordinateurs VAX, la transition était rude. Les systèmes d'IBM et de DEC étaient conçus dans le but d'élargir l'audience de l'informatique à des utilisateurs moins professionnels, à une époque où les micro-ordinateurs n'existaient pas. Pour ce faire la syntaxe de leur langage de commandes cherchait à s'adoucir en utilisant des lexèmes plus proches du langage humain, en tolérant des abréviations ou au contraire des tournures plus bavardes mais plus faciles à mémoriser. La réponse du système à une commande était elle aussi édulcorée : présentation aérée, commentaires explicatifs.

*Une syntaxe austère
et efficace*

Pour un Unixien ces variations pédagogiques ne sont que concessions coupables à l'ignorance informatique des secrétaires et des comptables. L'initiation informatique des ces professions est un objectif louable, mais dont il ne veut rien savoir, et Unix non plus, parce que pour un développeur¹⁶ ces aides pédagogiques sont au-

14. L'équivalent français est « DESNRLN (Diantre, et si nous relisons la notice) ».

15. Cette prétention flatte la paresse naturelle de l'utilisateur mais elle est fallacieuse. Il est certes possible, avec tel logiciel de traitement de texte dont le nom signifie « mot » en anglais, de créer facilement un document laid et peu lisible, mais dès lors que l'on veut un résultat présentable il faut se plonger dans la documentation et découvrir que ce logiciel est complexe, tout simplement parce que la typographie est complexe.

16. C'est avec Unix que « développeur » a supplanté « programmeur ». Ces deux termes ont le même sens, « personne dont le métier est la création de programmes informatiques », mais « programmeur » avait été victime d'une dévalorisation injuste. Les premiers managers de l'informatique ont pensé y reproduire un schéma déjà en déclin dans l'industrie : l'ingénieur conçoit, l'ouvrier fait. En informatique l'analyste concevrait ce que le programmeur ferait. C'était une erreur.

L'ingénieur disposait, pour transmettre à l'ouvrier la description de ce qu'il devait faire, d'un outil précis et rigoureux, le dessin industriel. Rien de tel n'existe en informatique, malgré des efforts qui durent encore pour créer des systèmes de spécification infaillibles dont UML est le dernier avatar : un ordinateur doté de son système d'exploitation et de ses langages de programmation n'est pas seulement infiniment plus versatile et plus souple qu'un étoupeur.

tant d'obstacles à son travail.

La syntaxe des commandes Unix est sèche comme un coup de trique d'une part parce qu'elles sont destinées à des professionnels qui les connaissent par cœur à force de les utiliser à longueur de journée, d'autre part parce qu'elles constituent un langage de programmation (le *shell*) qui permettra d'automatiser des opérations répétitives et que pour un langage toute souple syntaxique se paye en espace et en temps (il faut bien écrire les instructions qui vont interpréter les commandes, et autant de variations possibles provoquent autant de dizaines de lignes de code en plus).

La réponse du système à l'utilisateur qui lui soumet une commande est tout aussi austère, le plus souvent d'ailleurs il n'y a pas de réponse. Ainsi, si vous voulez modifier le nom d'un fichier, et que le nouveau nom que vous souhaitez lui donner est déjà pris par un autre fichier, le fichier homonyme sera détruit si le renommage est effectué. Les systèmes à l'usage des secrétaires, comptables ou présidents d'université posent dans un tel cas la question à l'utilisateur : « veux-tu vraiment détruire l'autre fichier ? », ou renomment le fichier menacé. Avec Unix, rien de tel : le fichier est froidement détruit sans même une notification *post mortem*. C'est un système pour vrais hommes qui savent ce qu'ils font, assument leurs erreurs et savent les réparer.

Mais cet ascétisme a une raison encore plus dirimante et qui n'est pas de l'ordre du sado-masochisme. L'invention sans doute la plus géniale d'Unix est la possibilité, par la simple syntaxe du *shell*, de réaliser des opérations de composition de processus au sens algébrique du terme. *La composition de processus*

Les opérations les plus simples consistent à diriger les résultats de sortie d'une commande vers un fichier, et à donner en entrée à une commande des données stockées dans un fichier, ce qui n'est pas encore de la composition de processus. Mais pour les réaliser il

ou qu'une fraiseuse, il est surtout d'une toute autre nature. Il traite de l'information et l'information qui décrit le programme à réaliser est de la méta-information, puisque le programme est lui-même de l'information.

La vraie difficulté réside donc toujours dans l'écriture du programme, exercice incroyablement délicat et laborieux. E.W. Dijkstra se définissait lui-même comme programmeur. Mais rien n'y fait, le programmeur sent le cambouis et la sueur alors que son remplaçant le développeur peut arborer (jusqu'à la retraite et au-delà) l'uniforme seyant et l'ethos décontracté des étudiants californiens.

fallait standardiser les formats d'entrée et de sortie des commandes et introduire les notions d'entrée standard et de sortie standard, ce qui ouvrait la voie à des réalisations plus ambitieuses.

L'opérateur de composition de processus en séquence est « ; » : on remarque sa concision. « a ; b » se lit : exécuter la commande a, puis la commande b, la plupart des commandes s'exécutant comme un processus indépendant. Le lancement d'un programme obéit à la même syntaxe et aux mêmes règles, ce qui encourage les développeurs à utiliser les conventions des commandes Unix et contribue à l'enrichissement du système.

L'opérateur de composition de processus parallèles asynchrones est « & ». « a & b » se lit : lancer l'exécution de a, et lancer b sans attendre que a se termine. Les deux processus seront concomitants (et concurrents pour l'acquisition du contrôle du processeur).

L'opérateur de composition de processus parallèles synchrones est « | ». « a | b » se lit : lancer l'exécution de a, puis lancer b qui va attendre la première ligne de résultat issue de a, la traiter puis se bloquer en attente de la suivante, etc.

Un exemple

Prenons un exemple simple : je veux la liste de tous les processus en cours d'exécution qui exécutent le serveur Apache, avec leur numéro de processus.

La commande qui affiche la liste des processus s'appelle « ps », qui doit, pour afficher non seulement les processus de l'utilisateur mais tous les autres, être agrémentée des paramètres a et x, ce qui s'écrit donc « ps ax ». Cette commande va produire la liste de tous les processus avec leur numéro et le nom du programme exécuté, à raison d'une ligne par processus. Je veux filtrer cette liste pour n'en retenir que les lignes où le nom de programme qui apparaît est *apache*.

Parmi les plus belles commandes d'Unix il faut citer **grep** (pour *global regular expression print* retranscrit plus tard en *general regular expression parser*, analyseur général d'expressions régulières). Cette commande peut faire des choses très savantes, mais nous allons l'utiliser de façon très simple, pour retrouver une chaîne de caractères dans le texte soumis à son entrée standard. « **grep apache** » signifie : si la ligne de l'entrée standard contient le texte « *apache* », afficher le texte à l'écran, sinon passer à la suivante.

Nous allons composer les deux commandes « **ps ax** » et « **grep**

apache » par l'opérateur de composition parallèle synchrone « | » :

```
ps ax | grep apache
```

Chaque ligne issue de la première commande sera soumise à la seconde pour analyse, ce qui réalisera le filtre souhaité :

```
$ ps ax | grep apache
284? S 0:00 /usr/sbin/apache
295? S 0:00 /usr/sbin/apache
etc.
434 pts/0 S 0:00 grep apache
```

Je reçois ainsi la liste de tous les processus Apache avec leur numéro, et en prime le processus d'analyse puisque sa ligne de commande comporte elle aussi le texte « apache ».

La métaphore du filtre est promue au rang de paradigme par Unix : les programmes vraiment unixiens sont écrits comme des filtres, c'est-à-dire recevoir un flux de caractères sur leur entrée standard et émettre un un autre flux de caractères sur leur sortie standard, ce qui permet de les combiner *ad libitum*.

C'est le livre de Jean-Louis Nebut¹⁷ qui me semble-t-il explicite le mieux la syntaxe du *shell* en termes de filtres et de composition de processus. *Les vertus de la ligne de commande*

La syntaxe des commandes et leur mode d'interaction suppose de la part du développeur une discipline ascétique : comme elles sont les syntagmes d'un langage de programmation, dont les programmes sont usuellement appelés *shell scripts*, il n'est pas souhaitable que les commandes engagent un dialogue avec l'utilisateur qui, dans ce cas, n'est pas un humain mais le système d'exploitation. Vous avez dit interface graphique ? passez votre chemin ! seule vaut la ligne de commande.

(À suivre : dissémination d'Unix, sources du logiciel libre, Linux)

17. Jean-Louis Nebut. *UNIX pour l'utilisateur*. Éditions Technip, Paris, 1990. Face à l'océan des livres-modes d'emploi inodores et sans saveur, celui-ci introduit les concepts de la programmation dans un univers d'où ils sont souvent bannis. Unix y apparaît sous un jour nouveau, doté d'une cohérence non limitée à sa structure interne, et du coup compréhensible même à qui n'en a pas lu le noyau. Organiser cet apparent fouillis était difficile.

Parallélisme massif, Big Data et intégration

Jacques Printz

Le monde des systèmes informatisés, improprement qualifié de « numérique », que l'on voit se construire à grande vitesse autour de nous, est un monde où la matière première de cette construction n'est pas donnée par la nature mais entièrement élaborée par les cerveaux humains qui en sont la source créative unique. Cette « matière première » est le logiciel, matière « molle » s'il en est comme l'indique son nom anglo-saxon, *software*. Les entreprises phares de l'« économie », grandes comme petites, en donnent le témoignage quotidien.

Le patrimoine logiciel de ces entreprises se compte en dizaines de millions de lignes de code écrites dans des langages dont les noms sont aujourd'hui connus de tous. L'ingénierie requise pour le concevoir, le développer et le maintenir n'a rien à voir avec l'ingénierie au sens habituel du terme avec les usines, chaînes de montages et contrôles qualité, mais nécessite les savoir-faire et les compétences, bref l'*intelligence* de milliers d'architectes et de programmeurs rompus à l'exercice de l'abstraction qu'exige la pensée algorithmique, dont la qualité est une assurance et une construction collective.

*Le patrimoine
logiciel*

Une ligne de code est une ligne de texte qui s'adresse simultanément à l'ordinateur *et* au concepteur/lecteur de cette ligne. On peut exprimer la taille des programmes selon les normes de l'édition, soit en moyenne 50 lignes par page. Un million de lignes de code, c'est en gros 20 000 pages de texte brut auquel il faut ajouter table des matières, index, bibliographie, etc. soit au moins 30 % de

Volumétrie du code

plus pour avoir quelque chose de lisible et que les programmeurs qui en ont la compétence pourront s'échanger et/ou relire de façon croisée pour corriger les erreurs.

Notre million de lignes brutes s'est ainsi transformé en une centaine de volume de 250 pages. Sachant qu'un lecteur chevronné ne peut lire qu'une cinquantaine de pages à l'heure, mais beaucoup moins si le texte requiert un effort de compréhension en profondeur, on peut se faire une idée de la concentration nécessaire pour faire correctement son travail !

Un bon programmeur expérimenté (10 à 15 % de la population des programmeurs) sera dans le meilleur des cas capable de mémoriser un texte d'environ 20 000 lignes, soit un gros volume de 400 pages. Si ce qu'il fait n'est pas très clair dans sa tête et dans celle de ceux qui seront appelés à relire son texte pour le faire évoluer, on peut imaginer le chaos qui en résulte...

Une entreprise jeune comme Amadeus, leader dans la conception et la vente de services de réservation de toute nature aujourd'hui en ligne et accessibles à tous, annonce un patrimoine logiciel d'environ 150 millions de ligne de code avec un taux de croissance annuel de l'ordre de 8 à 10 % pour répondre à la demande de nouveaux services, satisfaire les besoins ses clients et maintenir son leadership.

Cela veut dire une capacité de production nette de nouvelles lignes de code de l'ordre de 10 millions de lignes par an. Sachant qu'un programmeur correctement formé et normalement expérimenté, ce qui suppose trois à cinq ans de pratique pour compléter une compétence acquise à Bac + 5, produira selon les statistiques les plus/mieux vérifiées 4 à 5 000 lignes satisfaisant aux critères de qualité du marché par année ouvrable (220 jours de travail/an sous nos latitudes), il est facile de se représenter la complexité à laquelle sont confrontées les équipes qui en assurent le développement.

Dans le secteur économique tout récent des jeux vidéos, en pleine expansion, un jeu « normal » compte tenu de l'état de l'art, du marché et de la demande des utilisateurs, nécessite sur une durée de deux à trois ans (car au delà il est pratiquement impossible d'imaginer ce que pourra être la demande) le travail coordonné d'équipes pouvant compter une centaine de programmeurs dont certains, très expérimentés (10 à 15 % de l'équipe), connaissent toutes les « ficelles » des technologies sous-jacentes au jeu proprement dit. Sous

Une charge de travail impressionnante

son aspect ludique l'ingénierie des jeux vidéo est un bon concentré des difficultés et complexités connues par ailleurs, comme celles des systèmes C4ISTAR du monde de la défense et de la sécurité.

S'organiser pour faire face à l'imprévisible, tel est le slogan d'apparence paradoxale de ce domaine exemplaire s'il en est de la société en cours d'informatisation massive qui se construit sous nos yeux à vitesse accélérée. Il est parfaitement illustratif du « syndrome de la Reine Rouge », obligée de courir pour ne pas reculer.

Le syndrome de la Reine Rouge

Avec l'Internet des objets¹⁸, tous les objets et/ou choses de notre vie courante vont être dotés d'une « intelligence », c'est-à-dire de logiciels, et de capacités d'interaction avec leur environnement. Gérard Simondon, penseur d'une société structurée par et pour l'information à la fois productrice et consommatrice de savoir, n'aurait pas pu imaginer meilleur exemple d'un monde de communautés d'acteurs organisés par l'« objet technique¹⁹ » et réciproquement. Son analyse, à peine réactualisée, est résumée par le schéma de la figure 1.

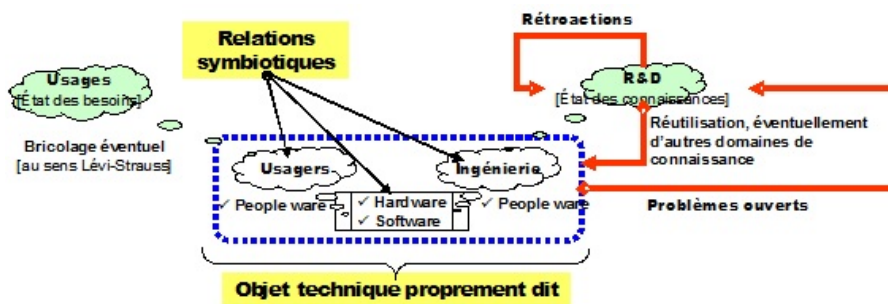


FIGURE 1 – L'objet technique et son environnement

Dans la configuration mondiale actuelle les utilisateurs peuvent se compter en centaines de millions ou même en milliards. Les objets techniques se partagent en deux grandes catégories :

- ceux qui se trouvent entre nos mains, visibles et concrets, par milliers,
- ceux qui sont invisibles et dont on peut simplement inférer l'existence, sauf à croire aux miracles : infrastructures techniques

18. En anglais « *Internet of Things* ».

19. Sa thèse s'intitule *Du mode d'existence des objets techniques*, 1960, rééditée chez Aubier 2012.

comme les parcs informatiques regroupant des milliers de serveurs, réseaux de communication, Internet, transport de l'énergie, moyens de paiement, etc.

*Compétence et
expérience,
ressources rares*

L'ingénierie de tout cela s'appuie sur des millions de programmeurs. Compétence et expérience sont des ressources rares que des géants du Net comme Google savent exploiter avec maestria au plan mondial. L'évolution suit un cours étrange : alors qu'en France le métier d'*ingénieur programmeur* est en déshérence et presque méprisé, il reste prestigieux dans la *Silicon Valley* où la formation mathématique des ingénieurs français est particulièrement appréciée. Comprenez qui pourra, alors que l'on nous rebat chaque jour les oreilles avec la compétitivité !

L'important dans le schéma de la figure 1 est la relation symbiotique entre l'aspect « matériel » de l'objet et les deux communautés qui le constituent comme tel, celle des utilisateurs et celle des ingénieurs. Qu'un déséquilibre apparaisse dans cette relation et c'en est fait de l'existence de l'objet, qui est en fait un système au sens exact du terme.

Chacun des ensembles ainsi en relation a une complexité et une trajectoire qui lui sont propres, mais ce qui fait la spécificité des objets informatisés est l'*intégration* de ces trois trajectoires, de ces trois complexités qu'il faut organiser en un tout cohérent. Cela requiert de les considérer deux à deux et globalement toutes les trois, soit au total selon sept combinaisons (les mathématiciens appellent cela « l'ensemble des parties »).

Les maîtres de cette symbiose ont été Steve Jobs aux manettes de Apple et aujourd'hui les dirigeants de Google et de Facebook. Ceux qui n'ont pas compris sa nécessité et sa raison d'être ne font pas de vieux os dans l'arène de l'économie informatisée : ainsi l'activité de terminaux et services de Nokia a disparu à la suite de son rachat par Microsoft.

Obstacles et opportunités

De ce rapide tour d'horizon on peut tirer des constats qui font apparaître des barrières de complexité que certains franchiront, d'autres non :

*Une complexité
extrême*

– complexité des infrastructures techniques qui doivent offrir un

service 24h/24 et 7j/7 sans interruption et sans limite de puissance : capacités de traitement, de stockage, d'interaction en temps réel ;

- complexité des projets d'ingénierie compte tenu du volume des patrimoines et des exigences parfois contradictoires des parties prenantes ;

- imprévisibilité des usages et complexité de la demande des utilisateurs qui souhaitent des services personnalisés, de la sécurité, etc., avec cependant un maximum d'interactivité et d'ouverture.

Face à cette complexité, il vaut mieux être lucide et surtout il faut réfléchir.

La demande en bons programmeurs, donc capables de raisonner dans des contextes diversifiés, va augmenter de façon inéluctable. *La barrière de compétence*

L'Internet des objets va siphonner les meilleurs. Pour s'en convaincre il suffit de regarder la stratégie en ressource humaine de sociétés comme Google qui « chassent » les talents au plan mondial, jusque dans les universités d'élite du programme 211 de la Chine (une centaine d'universités, l'équivalent de nos Grandes Écoles). L'Internet des objets repose sur des systèmes destinés au grand public : ils doivent donc être d'un emploi simple et « naturel », ce qui mécaniquement génère une grande complexité interne invisible à l'utilisateur mais dont il faut organiser l'ingénierie. Les gourous autoproclamés diront que c'est facile à faire, mais c'est un leurre fait pour abuser les naïfs.

L'expression « bon programmeur » est à prendre à la lettre car il s'agit d'*ingénierie de systèmes*, discipline transverse par essence. Elle est fort mal enseignée car elle n'entre pas dans les silos traditionnels de nos universités ni du CNRS, c'est tout juste si l'on en parle dans quelques Grandes Écoles... L'intégration en est la discipline principale, outre la connaissance des technologies de base qui reste le B.A.-BA du métier.

Le « combattant informatique » du XXI^e siècle est un ingénieur architecte/concepteur de systèmes, généraliste et polyvalent, capable d'intégrer les innovations technologiques, sachant jongler avec les abstractions et la sémantique, réceptif à l'air du temps et aux besoins des utilisateurs. Un tel profil ne peut pas provenir de formations à la va-vite, sans épaisseur, qui permettent tout au plus un bricolage cognitif dangereux.

Le « combattant informatique » du XXI^e siècle

Usages et utilisateurs exigent des solutions qu'ils perçoivent

comme personnalisées et non un galimatias pour utilisateur « moyen » ou « médian », mais sans rompre l'interactivité qui est l'essence de l'économie. Pouvoir dialoguer avec tous, tout en restant soi-même, tel est le dilemme de la nouvelle ingénierie.

Intégration

Si l'on analyse froidement ce qui se passe depuis quelques années dans les directions informatiques il est clair, pour ceux qui comprennent la problématique de l'évolution des systèmes informatisés, que l'enjeu principal de l'ingénierie est désormais d'intégrer non seulement les technologies disponibles et utiles, ce qui est déjà très difficile compte tenu de la complexité des infrastructures, mais aussi d'intégrer le fruit du travail de milliers de programmeurs dont il faut coordonner l'activité pour garantir la cohérence d'ensemble.

C'est là que réside la vraie difficulté compte tenu de la taille des patrimoines. Elle est aussi culturelle que technique car la perception de la complexité réelle est contre-intuitive. La complexité de deux systèmes en interaction n'est en effet ni la somme des complexités de chacun, ni leur minimum ou leur maximum, mais leur *produit* : la combinatoire de N systèmes (ou objets au sens de Simondon) en interaction ayant chacun k états distincts est de l'ordre de k^N . L'architecte intégrateur est donc confronté à une exponentielle de complexité : c'est la pire des situations.

Une exponentielle de complexité

Pour mesurer ce défi, considérons d'une part la notion de patrimoine, d'autre part ce que maîtrisent les programmeurs. Prenons une société de jeu vidéo qui aurait conçu et développé une cinquantaine de jeux : ce qui va suivre est un scénario fictif mais il reflète de façon non contestable l'état de l'art de l'ingénierie.

Un scénario

Avant de démarrer le développement de son cinquante-et-unième jeu, la société est à la tête d'un patrimoine de l'ordre d'une centaine de millions de lignes de code résultant de centaines de projets (plusieurs par jeu) plus ou moins coordonnés ou pas coordonnés du tout : la documentation technique et les tests jouent souvent le rôle de variable d'ajustement en cas de retard.

Nous avons vu ci-dessus qu'un bloc de code compréhensible, résultat du travail d'une petite équipe de deux ou trois programmeurs, comprend aux alentours de 20 000 lignes qui ont demandé quatre

ou cinq années ouvrées de travail (trois programmeurs pendant un an et demi).

Selon cette échelle de comptage le patrimoine de cette société va donc compter environ 5 000 blocs ou pièces, appelé dans le jargon « *Building Blocks* », « modules intégrables » ou encore « intégrats », néologisme au sens évident. Rappelons qu'aux normes de l'édition 20 000 lignes représentent un volume de 400 pages. S'il est bien géré, ce patrimoine est dont l'équivalent d'une bibliothèque de 5 000 volumes ; le plus souvent, il se présentera comme un tas de livres dispersé en vrac sur les machines servant au développement. La métaphore de la bibliothèque permet de comprendre l'importance de l'organisation de ce patrimoine sans qu'il soit besoin d'entrer dans les détails.

Pour créer un nouveau jeu, la connaissance de la bibliothèque permettra de ne pas refaire tout ou partie de ce qui a déjà été fait : cela aura un effet évident sur le coût, la qualité et le délai. Offrir un nouveau service reviendrait, à la limite, à assembler différemment des blocs existants : sa programmation serait réduite à celle d'un nouvel ordonnanceur, réalisé avec des langages et notations *ad hoc* comme celles du BPMN et du BPEL²⁰.

Comme pour les puzzles tout est alors dans l'art de la découpe, *L'art de la découpe* c'est-à-dire dans l'architecture du jeu, et elle ne dépend que du talent de l'architecte. À grande échelle on comprend pourquoi le découpage est beaucoup plus important que la pièce qui résulte du découpage : une pièce défectueuse dans un ensemble bien organisé peut toujours être remplacée, mais ce ne sera pas le cas si la découpe ne respecte pas la sémantique du jeu, c'est-à-dire son usage.

Pour simplifier le raisonnement, supposons que l'on puisse classer les blocs constitutifs d'un jeu en deux catégories :

- des *blocs spécifiques* matérialisent la sémantique du jeu : ils contiennent les innovations et ce qui va plaire si le jeu est réussi ;
- des *blocs génériques* venant de bibliothèques publiques ou propres à l'entreprise compte tenu de ses choix de plates-formes technologiques : ces blocs réalisent des services à la demande des blocs spécifiques.

Le patrimoine de l'entreprise est constitué de blocs spécifiques

20. « Business Process Model and Notation » et « Business Process Execution Language ».

et de blocs génériques. Il s'est enrichi progressivement : au début tout était spécifique mais si le management de l'ingénierie a été convenable la part du générique a crû progressivement, ce qui a permis de réduire le coût moyen d'un nouvel élément.

Ces techniques ont commencé à être bien maîtrisées dans les années 1970-80 avec l'ingénierie des systèmes d'exploitation et des « progiciels système » (SGBD, piles des protocoles des réseaux, interfaces graphiques, etc.). Tout cela a été systématisé et amélioré à l'occasion du développement des logiciels libres et de l'Open Source avec des initiatives comme celles de Richard Stallman et de la *GNU Software Foundation* au MIT, et aussi celles de beaucoup d'autres architectes moins connus.

Il faudra, pour satisfaire la demande de l'économie, pratiquer la technologie et les méthodes d'intégration à l'échelle industrielle, sachant que l'intégration doit être au service de la créativité et non l'inverse. Là sont la véritable innovation et la source de la compétitivité : c'est l'enjeu fondamental de la transformation en cours.

*L'enjeu
fondamental*

Mais un nouveau défi se présente à la programmation : celui du parallélisme.

Le défi du parallélisme massif

Jusque dans les années 2000 l'organisation de l'unité centrale de l'ordinateur (dans le jargon CPU, *Central Processing Unit*), devenu microprocesseur dans les années 1990 lorsque toute la logique de traitement a pu tenir sur une puce de 2 à 3 cm², a orienté le style de programmation de ces fantastiques machines selon le principe *séquentiel* : le processeur exécute une instruction après l'autre. Tous les langages de programmation usuels ont été bâtis sur ce principe qui permet aux programmeurs de contrôler parfaitement ce qu'ils font.

Avec le développement des bases de données et des réseaux dans les années 1970-80 le principe séquentiel a été aménagé pour pouvoir désynchroniser l'accès aux bases de données via divers types de réseaux de façon à protéger le cœur séquentiel de la « transaction » de la gestion des lectures et écritures qui, elles, se font indépendamment. C'est le modèle de la programmation transactionnelle,

« cheval de trait » de la programmation des systèmes d'information selon certains auteurs et qui représente 90 % de toutes les lignes de code écrites sur la planète.

La programmation séquentielle, complétée par sa variante transactionnelle, a été enseignée partout depuis les années 1960. Elle a permis l'extraordinaire développement des TIC. *Toucher à ce modèle, c'est automatiquement déclencher un séisme dans la population des programmeurs et dans l'enseignement de l'informatique.*

Un séisme dans la pratique et l'enseignement de la programmation

Tout le monde a entendu parler de la « loi » qu'a énoncée Gordon Moore, fondateur d'Intel et d'après laquelle la capacité de l'ordinateur double tous les 18-24 mois en raison des progrès techniques (essentiellement ceux de l'intégration des composants sur la puce de silicium). Elle a été assez bien vérifiée depuis les années 1960, mais personne n'évoque la façon dont cette performance est obtenue.

La finesse de gravure des circuits permet aujourd'hui de fabriquer des puces comptant plusieurs milliards de transistors et composants divers qui constituent la logique électronique de la puce. Ce qui était au départ et jusque dans les années 1980 un circuit sans mémoire, est devenu par la magie créatrice des architectes concepteurs de ces circuits un système complet comportant plusieurs processeurs, une hiérarchie de mémoires et des dispositifs d'entrée/sortie pour « attaquer » les organes périphériques : mémoire permanente, réseaux, etc. Dans le jargon, c'est ce qu'on appelle les SOC, *System On Chip*, ou encore les *multi-cœurs*. Cette évolution s'est faite sans toucher au modèle de la programmation, fût-ce au prix d'une complexification de la puce comme pour le processeur Pentium IV d'Intel.

Le problème, formulé dans sa brutalité, est donc : peut-on programmer ces SOC comme nos bons vieux processeurs séquentiels ? *La réponse est NON !*

On ne peut plus continuer à programmer comme avant

Dans la course à la puissance des SOC un problème technique connu depuis toujours mais masqué par d'autres phénomènes physiques est arrivé au premier plan dans les années 2000 : le *mur thermique*. Un circuit d'ordinateur, comme son nom l'indique, est d'abord un *circuit*, donc il chauffe. Il faut évacuer la chaleur au rythme de fonctionnement du circuit, or le silicium (c'est-à-dire le verre) est très mauvais conducteur de chaleur : d'où la nécessité de dispositifs de ventilation mécaniques et donc fragiles. Quand

Le mur thermique

le constructeur de SOC annonce un rythme d'horloge à 1 ou 2 GHz, cela veut dire que le circuit s'allume et s'éteint un milliard de fois par seconde, ce qui suppose une alimentation électrique quasi continue et une production de chaleur d'autant plus intense que le nombre de transistors est plus élevé.

Le dilemme est donc simple : il faudra ou bien continuer à augmenter le nombre de cœurs et la mémoire sur la puce et diminuer la vitesse de commutation en gérant l'énergie consommée par le SOC, ou bien trouver un autre dispositif de refroidissement, inconnu à ce jour.

L'obtention du gain de puissance est en conséquence renvoyée sur le programmeur, situation qui nous ramène au début du développement des TIC : jusque dans les années 1960 l'essentiel de la programmation se faisait en assembleur car il fallait piloter au plus près les organes de la machine, et à cette époque oubliée il n'était pas indécent pour un ingénieur issu du Top 10 des Grandes Écoles de s'adonner à cette activité ésotérique. Cette solution n'est pas généralisable aujourd'hui compte tenu de la rareté de la ressource.

Une rupture se prépare

Pour bien comprendre la rupture qui se prépare, il faut dire quelques mots sur la façon dont raisonnent les programmeurs qui, d'une certaine façon, sont des traducteurs et des navigateurs puisque que leur fonction essentielle est de traduire un processus de calcul, au sens le plus large du terme, en une procédure automatique que l'ordinateur pourra « exécuter », en lieu et place d'un processeur « humain », sur la base du plan de calcul proposé qui constitue une navigation : c'est ce qu'avaient compris les inventeurs des langages Fortran et COBOL.

Psychologie cognitive du programmeur

Programmer, c'est donc traduire, ce qui induit une psychologie cognitive particulière car pour bien traduire il faut parfaitement comprendre le sens de ce que l'on veut traduire dans la langue de la cible de traduction. Il s'agit donc de trouver des équivalences sémantiques, étant entendu qu'un cerveau humain n'est pas organisé comme un ordinateur. Cette ingénierie sémantique, « *human engineering* » comme disent les anglo-saxons, reste un domaine peu exploré bien que crucial dans la perspective qui s'ouvre.

L'esprit humain fonctionne bien quand on lui donne une tâche précise à effectuer. C'est le principe de la division scientifique du travail théorisée par Frederic Taylor et mis en œuvre par Henry Ford

dans ses usines au début du XX^e siècle. Mener à bien simultanément deux tâches qui interagissent demande par contre un apprentissage, avec trois tâches cela devient difficile et c'est virtuellement impossible à partir de quatre.

Il y a des raisons psychocognitives profondes à cela, liées à la structure de notre mémoire immédiate (mémoire dite de « travail ») dont la perte, dans la maladie d'Alzheimer, est un handicap majeur. Les observations effectuées par les psychologues montrent que cette mémoire est limitée à 7 ± 2 « morceaux » d'information²¹ (en anglais *chunks* comme dans « *chunk of bread* » ; ici il s'agit d'un « morceau de sens »). C'est pourquoi le bon sens populaire invite à ne pas « poursuivre plusieurs lièvres à la fois ».

Programmer en mode parallèle, donc effectuer simultanément plusieurs tâches à la fois, est beaucoup plus difficile que programmer en séquentiel, mode de programmation virtuellement accessible à tous. Cette difficulté ne peut être surmontée que par un outillage *ad hoc* permettant d'organiser la combinatoire qu'induit le parallélisme.

Si le programmeur dispose par exemple de trois processeurs il peut organiser son programme de façon à utiliser au mieux la ressource qui lui est offerte, soit trois flots d'exécution, en découpant les instructions et les données de façon *ad hoc*. À chaque décision de programmation, placement d'une action sur l'un des trois flots, il doit penser simultanément à ce qui se passe sur chacun des flots, mais aussi sur leurs combinaisons deux à deux et trois à trois, soit au total sept configurations à explorer avant de décider quoi faire.

*Comment maîtriser
plusieurs flots ?*

Ce nombre dénote l'ensemble des parties que l'on peut former avec trois informations venant de trois flots. S'il y avait quatre flots, il y aurait 15 informations à exploiter, bien au-delà du « nombre magique », la formule générale étant $2^N - 1$. La plupart des programmeurs, du moins ceux qui ont été bien formés, n'ont pas de problème avec deux flots de traitement, cas fréquent par exemple pour un calcul/transformation et pour les entrées/sorties du programme.

Le choix technologique retenu, compte tenu du « mur thermique », est l'architecture des multi-cœurs. Des smartphones et

21. 7 est le « nombre magique » du célèbre article de George A. Miller (voir l'entrée Wikipédia, plus complète dans sa version anglaise).

tablettes ont actuellement en standard des processeurs à quatre et huit cœurs, mais la technologie actuelle permet de faire des 256-cœurs comme la puce de Kalray. Pour programmer efficacement de tels engins il va falloir maîtriser la programmation parallèle en tenant compte de la limitation des capacités psychocognitives des programmeurs. C'est ce qu'ont parfaitement compris les industriels et les universités américaines qui s'y préparent depuis déjà quelques années.

La programmation des objets informatiques qui intègrent des multi-cœurs représente une rupture cognitive fondamentale par rapport au style de programmation auquel ont été formés 90 à 95 % des programmeurs.

(À suivre : éléments de méthode pour surmonter la barrière de la complexité)

Éléments de théorie

« iconomique »

Michel Volle

L'économie moderne s'est déployée à partir de la fin du XVIII^e siècle en s'appuyant sur la mécanique, la chimie, puis sur l'énergie à partir de la fin du XIX^e siècle.

Elle a fait place à partir des années 1970 à une économie informatisée²² qui s'appuie sur la synergie de la microélectronique, du logiciel et de l'Internet.

La mécanique, la chimie et l'énergie ne sont pas supprimées : elles s'informatisent, tout comme l'agriculture s'est mécanisée et chimisée aux XIX^e et XX^e siècles.

L'informatisation automatise les tâches répétitives physiques et mentales. Le flux de travail que demande la production devient faible en regard du stock de travail qui la prépare. Le coût de production tend à se réduire au coût du capital fixe initial.

Il en résulte une cascade de conséquences dans la nature des produits, le régime du marché, l'organisation des entreprises, la sociologie des pouvoirs et la psychologie des personnes.

Nous nommons iconomie une économie informatisée qui serait par hypothèse parvenue à la pleine efficacité ou, comme disent les économistes, à « l'équilibre ». Le modèle de l'iconomie met donc en évidence les conditions nécessaires de l'efficacité.

22. Nous n'utilisons pas ici le mot « numérique », qui est trop étroit pour désigner l'ensemble des phénomènes que comporte et provoque l'informatisation.

Le chômage de masse indique que l'économie informatisée actuelle n'est pas l'iconomie. Elle connaît une crise de transition due à l'inadéquation du comportement des agents économiques (entreprises, consommateurs, État) en regard des ressources et des dangers qu'apporte l'informatisation.

La stratégie pour sortir de cette crise s'appuie sur une conscience claire de ces ressources et de ces dangers pour orienter les agents économiques vers l'iconomie.

* *

L'économie informatisée, fondée sur la synergie de la microélectronique, du logiciel et de l'Internet, a succédé à partir des années 1970 à l'économie moderne, qui était fondée sur la synergie de la mécanique, de la chimie et de l'énergie (Gille, 1978).

Nous nommons *iconomie* (du grec *eikon*, image, et *nomos*, organisation) une société informatisée *efficace* (Saint-Etienne, 2013 et Volle, 2014).

L'économie de l'iconomie est donc par hypothèse *l'économie informatisée parvenue à l'efficacité* : elle utilise la totalité de ses ressources (en particulier la force de travail) et elle sait maîtriser les dangers qu'apporte l'informatisation.

Il s'agit d'une société future et non de la société actuelle, dont l'économie connaît une crise caractérisée notamment par le chômage de masse. Il s'agit plus exactement d'un repère posé à l'horizon du futur et en regard duquel la crise actuelle apparaît comme une *crise de transition*, car l'économie informatisée conserve encore des habitudes héritées de l'économie moderne.

Pour pouvoir « penser » l'iconomie nous construirons un *modèle* dont nous tirerons les conséquences. Comme tout modèle celui-ci est schématique mais le schéma qu'il propose sera correct s'il oriente l'intention vers l'action judicieuse.

* *

Une révolution industrielle comme celle que l'informatisation a provoquée transforme les fondations de l'économie. Pour comprendre ce que celle-ci est devenue il faut partir de ce que sa théorie a de plus fondamental.

Cette théorie se construit à partir de trois éléments²³ : (1) la distribution des ressources entre les acteurs, (2) la fonction de production des entreprises²⁴ et (3) la fonction d'utilité des consommateurs.

Selon ce modèle la production et l'échange conduisent, s'ils sont judicieux, de la distribution initiale des ressources à une situation *efficace* que les économistes nomment « optimum de Pareto » (1906) ou encore « équilibre » et qui est telle qu'il serait impossible d'accroître la satisfaction d'un consommateur sans diminuer celle d'un autre. La prise en compte du temps et de l'incertitude du futur introduit dans ce modèle une *dynamique* et le risque d'une inefficacité, ou *déséquilibre*.

La spécification des ressources, de la fonction de production et des besoins des consommateurs permet de poser un diagnostic sur une économie particulière.

Une révolution industrielle transforme les ressources, la fonction de production et jusqu'aux besoins des consommateurs car ceux-ci réagissent à l'offre dont ils ont connaissance. On ne peut donc conserver tels quels dans l'économie informatisée ni le diagnostic que les économistes ont porté sur l'économie moderne, ni les prescriptions qu'ils ont formulées. Il faut renouer, en amont de ces prescriptions, avec la réflexion des plus grands économistes.

*Revenir aux
fondations de la
théorie économique*

Or la théorie à l'œuvre, celle qu'ont en tête les dirigeants et les responsables de la politique économique, *n'est pas* la théorie savante : rares sont en effet ceux d'entre eux qui ont pris la peine de méditer les travaux des grands économistes.

Cette théorie est ce qui leur reste de cours d'économie écoutés d'une oreille distraite ou scolaire, complétés par des conversations et par la lecture épisodique du journal. Elle se condense en quelques théorèmes dont les conditions de validité sont trop complexes pour rester présentes à l'esprit.

Les théoriciens considèrent donc la théorie à l'œuvre comme un catalogue d'erreurs de débutant, elle ne les intéresse pas. Il est pourtant nécessaire d'en élaborer une critique car elle a des effets sur

*La théorie à l'œuvre
est un catalogue
d'erreurs*

23. La « boîte d'Edgeworth » (1881) en donne une représentation partielle mais éclairante.

24. Nous dirons « entreprises » pour désigner l'ensemble des institutions, dont l'institution « Entreprise » est un cas particulier.

l'économie réelle : l'illogisme du raisonnement provoque l'absurdité des décisions.

L'informatisation a en effet révélé dans le cerveau humain une ressource naturelle au potentiel *a priori* illimité – le logiciel est l'une de ses manifestations – et transformé la fonction de production.

Nous donnerons à cette dernière une nouvelle spécification. Il en résultera une représentation de l'équilibre (et de la dynamique) qui éclaire l'orientation stratégique des entreprises et la politique économique de l'État.

La production informatisée

L'agriculture, activité économique principale jusqu'au XVIII^e siècle, n'a pas disparu après la première révolution industrielle : elle s'est mécanisée et chimisée. De même l'informatisation ne fait pas disparaître la mécanique, la chimie et l'énergie : *elle les informatise*.

Or l'essentiel du coût de production d'un microprocesseur ou d'un logiciel est dépensé dans la phase de conception et d'investissement qui est antérieure à la production proprement dite : leur coût marginal est négligeable. Sur l'Internet le coût marginal du trafic est nul tant que celui-ci n'excède pas un seuil de dimensionnement qui est rarement atteint.

Le coût de ces produits étant indépendant du volume de la production, leur coût moyen décroît lorsque ce volume augmente : *le rendement d'échelle est croissant*. Il en est de même pour les ordinateurs, commutateurs, routeurs, etc. qui sont leurs applications les plus immédiates.

Il en est de même aussi pour la mécanique, la chimie et l'énergie pour qui l'informatique est devenue *la* technique principale : leur coût marginal n'est sans doute pas négligeable mais il est assez faible pour que leur rendement d'échelle soit croissant.

C'est là pour certains un phénomène bouleversant. John Hicks, qui fut l'un des plus grands économistes du XX^e siècle, estimait que renoncer aux rendements décroissants et à la tarification au coût marginal entraînerait le naufrage de la théorie économique²⁵.

*Le rendement
d'échelle est
croissant*

25. « On ne peut éviter le naufrage de la théorie de l'équilibre général qu'en supposant que pour la plupart des entreprises le régime du marché ne s'écarte pas beaucoup de la concurrence parfaite et que les prix ne s'écartent pas beau-

Il était trop pessimiste : il suffit de changer les hypothèses sur lesquelles s'appuie le raisonnement.

Comme le rendement d'échelle croissant se généralise dans l'économie informatisée, la tarification au coût marginal est désormais absurde, car elle obligerait les entreprises à vendre à perte, et les marchés des divers produits ne peuvent plus obéir au régime de la concurrence parfaite. Ils obéiront soit à celui du monopole naturel, soit à celui de la concurrence monopolistique.

Fonction de production

Pour connaître la fonction de coût d'une entreprise il faut partir de la *fonction de production* dont elle dérive, et qui décrit la relation entre le volume des facteurs mis en œuvre et le volume de la production qui en résulte.

Les facteurs retenus le plus souvent sont le capital et le travail, dont les volumes sont notés respectivement K (*Kapital*, notation inspirée de l'allemand) et L (*labour*). Le volume de la production est noté q . La forme générale de la fonction de production est donc :

$$q = f(K, L).$$

q et L sont des *flux* relatifs à une année tandis que K est un *stock* constitué avant le début de cette année.

K est le volume du « capital fixe » (bâtiments, équipements, logiciels etc.) dont la valeur figure à l'actif du bilan, et non le « capital » apporté par les actionnaires et qui figure au passif : c'est un *travail accumulé* pour rendre la production *possible*, tandis que L est le *flux de travail* nécessaire pour la *réaliser*. Fisher (1906) disait « stock » et « flux » pour « capital » et « travail » : il est salubre de partager cette intuition.

La spécification de la fonction de production est purement théorique. Dans la pratique il serait en effet difficile d'assigner un volume au flux de travail (comment pondérer les qualifications ?), ainsi qu'au stock de capital (comment pondérer les bâtiments, machines, logiciels, organisation, etc. ?) et même, souvent, à la production.

coup du coût marginal de production en niveau comme en évolution » (Hicks, 1939, p. 84).

Il ne faut pas s'attarder à de telles difficultés : le but d'un modèle n'est pas d'alimenter en détail une description réaliste, mais de favoriser un raisonnement qui soit *exact* en ce sens qu'il oriente la décision vers l'action judicieuse (Fixari, 1977).

Les coûts unitaires des facteurs de production sont notés w (*wage*) pour le travail et r (*return*) pour le capital. Il faut que la mesure des coûts soit, comme celle de la production, relative à une année.

r est donc le *coût d'usage* du capital. Il doit couvrir le coût d'un emprunt ou, ce qui revient au même, un coût d'opportunité (ce que l'entreprise aurait obtenu si elle avait placé ses fonds plutôt que d'investir), et couvrir aussi le risque que l'investissement comporte car il se peut que l'entreprise perde sa mise.

r est donc, si l'on note p_K le coût unitaire du capital fixe (en le supposant par exemple constitué de machines identiques) :

$$r = p_K(i + \pi),$$

où i est le taux d'intérêt du marché et π la prime de risque de l'activité considérée.

L'efficacité exige que pour chaque volume q de la production l'entreprise choisisse la combinaison des facteurs K et L qui minimise le coût de production : il s'agit donc de minimiser le coût annuel de production $c(q) = rK + wL$ sous la contrainte $q = f(K, L)$.

Il en résulte que, pour chaque valeur de q , K et L doivent être tels que :

$$\frac{\partial q}{\partial K} = \lambda r \text{ et } \frac{\partial q}{\partial L} = \lambda w, \lambda \text{ étant le multiplicateur de Lagrange,}$$

d'où :

$$w \frac{\partial q}{\partial K} = r \frac{\partial q}{\partial L}.$$

Nous allons voir que lorsque la fonction de production est spécifiée de façon explicite cette relation permet d'exprimer K et L en fonction de q et d'en déduire la *fonction de coût* :

$$c(q) = rK(q) + wL(q).$$

Fonction de coût

La théorie actuellement à l'œuvre à « Bruxelles », dans les ministères, à l'OMC, dans les médias, inspirée par la doctrine néo-libérale qui a tant de prestige auprès des conseils d'administration, affirme l'efficacité de la concurrence parfaite et du libre échange sans se soucier de savoir si l'économie considérée respecte les hypothèses des modèles de Arrow-Debreu (1959) et de Ricardo (1817). Elle recommande aussi la tarification au coût marginal, conseil qu'il est impossible de suivre lorsque le rendement d'échelle est croissant.

Cela résulte de la façon dont le cours élémentaire d'économie présente la fonction de coût des entreprises. Il commence, conformément à l'intuition, par dire que le rendement d'échelle est croissant (le coût moyen $c(q)/q$ ou, selon les cas, le coût marginal $c'(q)$ diminue) lorsque le volume produit q est faible, et qu'il devient décroissant à partir d'un certain volume de la production à cause de l'augmentation de la complexité des opérations. En complétant cette hypothèse par quelques autres, le cours démontre ensuite l'efficacité de la concurrence parfaite.

Puis il poursuit en proposant des spécifications de la fonction de production : fonction de Cobb-Douglas (1928), fonction à élasticité de substitution constante (CES, *Constant Elasticity of Substitution*) de Solow (1956), fonction à facteurs complémentaires de Leontief (1941) :

Spécifications usuelles de la fonction de production

$$q = aK^\alpha L^\beta \text{ (Cobb-Douglas),}$$

$$q = k(aK^r + (1 - a)L^r)^{1/r} \text{ (CES),}$$

$$q = \min(aK, bL) \text{ (Leontief).}$$

Ces fonctions (en fait, le plus souvent, la fonction de Cobb-Douglas) sont utilisées dans les modèles économiques et économétriques. Tout calcul fait, les fonctions de coût qui leur correspondent sont les suivantes (nous notons k_i un facteur constant qui diffère d'un cas à l'autre ; il dépend des coefficients de la fonction de production ainsi que de w et de r) :

$$c(q) = k_1 q^{1/(\alpha+\beta)} \text{ (Cobb-Douglas),}$$

$$c(q) = k_2 q \text{ (CES),}$$

$$c(q) = k_3q \text{ (Leontief).}$$

Avec la fonction de Cobb-Douglas, le rendement d'échelle est constant si $\alpha + \beta = 1$, croissant si $\alpha + \beta > 1$, décroissant si $\alpha + \beta < 1$. Avec la CES ou la fonction à facteurs complémentaires le rendement d'échelle est constant.

Dans ces trois cas, le rendement d'échelle est donc une fonction *monotone* du volume de la production, ce qui est contraire à la présentation initiale selon laquelle ce rendement est d'abord croissant, puis décroissant lorsque le volume produit dépasse un certain seuil.

Un fossé sépare ainsi l'introduction théorique du cours et la technique mobilisée par les modèles. Il en résulte un paradoxe : un modélisateur postulera l'efficacité de la concurrence parfaite tandis qu'il utilise la fonction de Cobb-Douglas, qui contredit l'une des hypothèses sur lesquelles s'appuie la démonstration de cette efficacité.

Pour combler le fossé entre l'intuition et les formulations usuelles de la fonction de production nous proposons une fonction à facteurs complémentaires qui généralise la fonction de Leontief :

Une fonction de production plus conforme à l'intuition

$$q = \min(aK^\alpha, bL^\beta) \text{ avec } \alpha > 1 > \beta.$$

Tout calcul fait, la fonction de coût qui lui correspond est :

$$c(q) = r \left(\frac{q}{a} \right)^{1/\alpha} + w \left(\frac{q}{b} \right)^{1/\beta}.$$

Supposons par exemple que :

$$q = \min(K^5, 5L^{0,2}) \text{ et que } w = r = 1.$$

La fonction de coût est alors :

$$c(q) = q^{0,2} + \left(\frac{q}{5} \right)^5.$$

Voici son graphe :

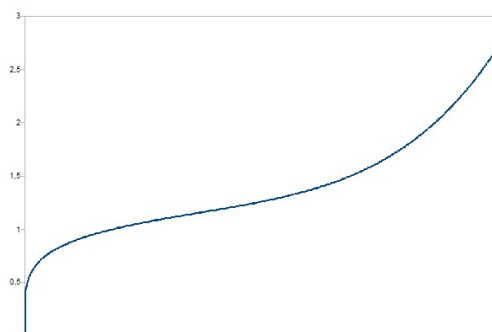


FIGURE 2 – Fonction de coût

Cette fonction de coût est conforme à l'intuition selon laquelle le rendement d'une entreprise est d'abord croissant, puis décroissant. Le coût moyen est minimal pour $q^* = 3,82$. Le rendement est croissant si $q < q^*$ et décroissant si $q > q^*$.

Régimes du marché

La théorie économique savante ne se limite pas à la concurrence parfaite : elle sait modéliser des régimes de concurrence imparfaite (Tirole, 1993) (monopole, oligopole, concurrence monopolistique), ainsi que la coopération entre des entreprises et les conséquences d'une asymétrie de l'information (sélection adverse, aléa moral, etc.).

Ces modèles ne sont pas plus compliqués que celui de la concurrence parfaite, mais seule celle-ci et le monopole, qui lui fait pendant, sont présentés dans le cours élémentaire d'économie. La théorie à l'œuvre ignore donc les autres régimes, notamment celui de la concurrence monopolistique.

Concurrence parfaite

Le régime de concurrence parfaite s'établit sur le marché d'un produit lorsque les hypothèses suivantes sont respectées :

- le rendement de la production du produit par une entreprise est d'abord croissant puis décroissant, de sorte qu'il existe un volume de production q^* tel que le coût moyen $c(q)/q$ soit minimal ;

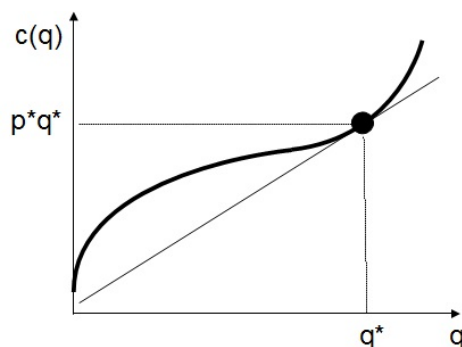


FIGURE 3 – Régime de concurrence parfaite

- pour le prix $p^* = c(q^*)/q^*$ la quantité $D(p^*)$ demandée sur le marché est beaucoup plus importante que q^* ;
- l'entrée d'une nouvelle entreprise sur le marché est libre.

Dans ces conditions on démontre qu'à l'équilibre le prix est p^* , chaque entreprise produit le volume q^* et le nombre des entreprises est $n^* = D(p^*)/q^*$.

Le profit des entreprises est nul mais l'économie est efficace car le coût de production $n^*c(q^*)$ de la branche d'activité est minimal.

Nota Bene : il faudrait dire « profit normal » plutôt que « profit nul » car les entreprises doivent compenser le risque que comporte leur activité.

On démontre aussi que le prix p^* est égal au coût marginal $c'(q^*)$ et que pour le volume $q = q^*$ la fonction $c'(q)$ est croissante : l'équilibre s'établit donc dans une zone de la fonction de coût où, le coût marginal étant croissant, *le rendement d'échelle est décroissant*.

Le raisonnement qui permet de démontrer l'efficacité de la concurrence parfaite est subtil sinon compliqué. La validité de ses résultats est suspendue au respect des hypothèses.

Monopole

Supposons que le volume de production qui correspond à la demande se trouve dans la zone des rendements croissants de la fonction de coût. Le marché obéit alors au régime du *monopole naturel* : une seule entreprise est en mesure de satisfaire toute la demande et elle peut évincer ses concurrents potentiels en pratiquant un prix plus bas que le leur. Si l'entreprise n'est soumise à aucune régulation, le prix du produit et la quantité produite seront ceux qui lui permettent de maximiser son profit.

Si l'on juge le monopole antipathique, c'est pour les raisons suivantes :

- contrairement à la concurrence parfaite, le monopole naturel procure à l'entreprise un profit supérieur à la prime de risque et le prix qu'elle pratique rationne la demande ;

- on peut craindre que l'entreprise monopoliste, sûre de faire du profit, ne soit pas incitée à innover ni même à se soucier de la qualité de son produit ;

- les monopoles ne sont pas tous « naturels » : il se peut qu'une entreprise s'empare d'une position de monopole, puis la conserve par des procédés violents.

Il faut donc que le marché soit soumis à une *régulation* qui interdise l'instauration d'un monopole par des procédés violents et qui, lorsque le monopole est naturel, sache préserver les intérêts des consommateurs et l'incitation à innover.

Il peut arriver qu'un régulateur mal renseigné ou trop dogmatique impose la concurrence sur un marché qui, sans son intervention, aurait obéi au régime du monopole naturel. Dans ce cas la régulation risque d'être contraire à l'efficacité.

Lorsqu'un changement de système technique a lieu, comme c'est le cas avec l'informatisation, la fonction de coût peut être transformée : le régime du marché peut alors changer.

Le monopole naturel est-il inévitable si le rendement d'échelle croissant s'instaure sur un marché qui obéissait auparavant au régime de la concurrence parfaite ? Non, car il existe une autre possibilité : le régime de la *concurrence monopolistique*.

Concurrence monopolistique

Supposons comme ci-dessus que le volume de production qui correspond à la demande se trouve dans la zone des rendements croissants de la fonction de coût, mais que les consommateurs aient des besoins divers (comme c'est depuis longtemps le cas sur le marché des livres, de la musique, des automobiles, etc.).

Le produit peut alors être diversifié en *variétés* qui se distinguent par leurs attributs qualitatifs et sont destinées chacune à un segment de la clientèle (Robinson, 1933 ; Chamberlin, 1933).

Diversification du produit en « variétés »

Cette différenciation permet à plusieurs entreprises de coexister sur le marché de ce produit alors même que le rendement d'échelle

est croissant. La différenciation aura pour effet d'accroître quelque peu la demande totale du produit car elle accroît son utilité. Cependant la demande adressée à une variété sera d'autant plus faible que le nombre des variétés offertes est plus élevé.

Chaque entreprise va se trouver en position de monopole sur le segment de clientèle auquel correspond la variété qu'elle produit (on suppose ici qu'une entreprise produit une variété et une seule). Elle se trouvera en concurrence par le prix envers les clients qui sont indifférents entre sa variété et une autre.

Tout comme les autres régimes celui de la concurrence monopolistique se prête à la modélisation mathématique : dans le cas où les variétés ont toutes le même coût de production le nombre n^* des variétés est déterminé à l'équilibre ainsi que la quantité q^* produite par chacune et leur prix unitaire p^* .

Un exemple

Voici un exemple simple pour illustrer la concurrence monopolistique.

Considérons une plage de longueur L où des vacanciers sont répartis selon la densité uniforme μ .

Un marchand de glaces s'installe. Il vend ses glaces au prix p . La consommation d'une glace procure à un vacancier le plaisir U mais l'aller-retour est d'autant plus pénible que la distance d qui le sépare du glacier est plus longue : nous supposons ce désagrément égal à kd .

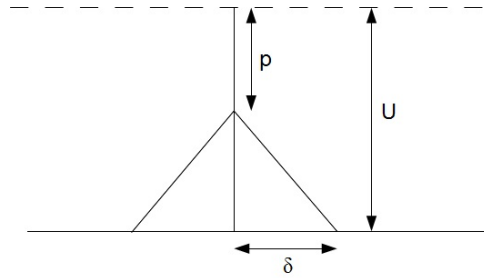
La satisfaction S que la consommation d'une glace procure à un vacancier est donc :

$$S = U - p - kd.$$

*Largeur du segment
de marché*

Un vacancier achète une glace (et, supposons-nous, une seule) si sa satisfaction est positive. Le glacier a donc pour clients les vacanciers qui se trouvent à une distance $d \leq (U - p)/k$. Notons δ la distance limite, $\delta = (U - p)/k$. Le nombre des glaces vendues est :

$$q = 2\mu\delta = 2\frac{\mu(U - p)}{k}.$$



Supposons le coût de production des glaces indépendant du nombre de glaces produites et donc réduit au coût fixe C des équipements nécessaires à leur production. Le profit que fait le glacier est $\Pi = 2\mu(U - p)p/k - C$, qui est maximal pour $p^0 = U/2$.

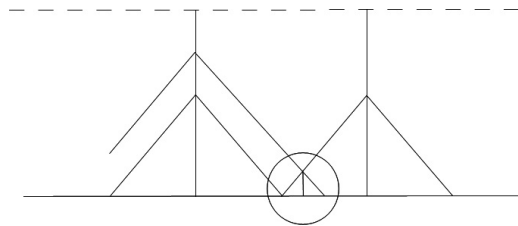
Si le glacier pratique le prix p^0 son profit est $\Pi^0 = \mu U^2/2k - C$: il ne peut être positif que si $U > \sqrt{2kC/\mu}$.

Si Π^0 était négatif aucun glacier ne s'installerait sur la plage. Nous supposons qu'il est positif, et aussi que la longueur L de la plage est beaucoup plus grande que la largeur 2δ du segment servi par un glacier.

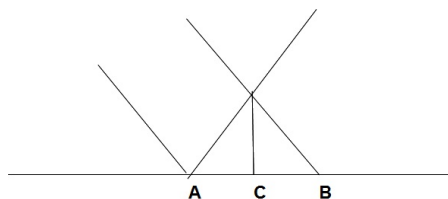
Le profit attire alors d'autres glaciers. Le deuxième s'installe loin du premier mais progressivement la plage entière est servie par des glaciers dont les « territoires » se touchent et qui font tous le même profit Π^0 .

Libre entrée des concurrents

Chaque glacier se trouve alors en concurrence par le prix avec ses deux voisins. Supposons en effet que ces voisins pratiquent tous le prix p^0 : si le glacier G pratique un prix inférieur à p^0 , il étend son territoire à leur détriment mais cet élargissement de son marché est deux fois moins sensible à la baisse de son prix qu'il ne l'aurait été si G avait été seul sur la plage.



Pour comprendre cela, il faut être attentif à l'intérieur du cercle dans le graphique ci-dessus, où l'on voit ce qui se passe entre G et le glacier G' qui se trouve à sa droite.



Quand G réduit son prix son territoire s'étend à droite et à gauche. Si G était seul sur la plage il gagnerait sur sa droite la longueur AB . Mais un vacancier qui se trouve sur le segment CB préférera G' car sur ce segment G' procure plus de satisfaction que G . G ne peut donc gagner en réduisant son prix que la longueur $AC = AB/2$ sur chacun de ses voisins.

Chaque glacier se trouve ainsi en position de monopole sur un segment de plage et en concurrence par les prix avec ses voisins : c'est pourquoi l'on dit que le régime de ce marché est la concurrence monopolistique.

L'évolution ne s'arrête cependant pas là. Le profit étant encore positif, de nouveaux glaciers sont incités à s'installer sur la plage, ce qui va comprimer le territoire et le profit des autres. L'installation de nouveaux glaciers va se poursuivre jusqu'à ce que le profit soit nul : le marché aura alors atteint l'équilibre de concurrence monopolistique.

Lorsqu'un glacier est seul sur le marché, la demande qui lui est adressée est $q = 2\mu(U - p)/k$. Le prix p qui lui permet de servir exactement le segment de largeur 2δ est égal à $U - k\delta$. Le volume de la demande est alors $q = 2\mu\delta$.

Si le glacier a des voisins avec lesquels il entre en concurrence par le prix, l'expression de la demande diffère de la précédente parce que l'effet d'une baisse du prix est deux fois moins fort que si le glacier était seul : on a donc alors $dq/dp = -\mu/k$.

Pour trouver les valeurs de p^* , n^* et q^* à l'équilibre de concurrence monopolistique il faut exprimer (1) que le profit est maximal, (2) que le profit est nul.

Le profit $pq - C$ est maximal si $pdq + qdp = 0$, soit $dq/dp = -q/p$, et il est nul si $pq = C$. On trouve donc :

$$p^* = \sqrt{kC/\mu}, \quad n^* = L\sqrt{k\mu/C}, \quad q^* = \sqrt{\mu C/k}.$$

Ce résultat est conforme au bon sens : le nombre des glaciers

est d'autant plus élevé que la plage est plus longue, la densité des vacanciers plus forte et leur sensibilité à distance plus grande ; il est d'autant moins élevé que le coût fixe est plus important. Le prix d'une glace est d'autant plus élevé que les vacanciers sont plus sensibles à la distance et que le coût fixe est plus important, d'autant moins élevé que la densité des vacanciers est plus forte.

Nota Bene 1 : $k\delta^* + p^* = (3/2)\sqrt{kC/\mu}$. Il faut que $U \geq k\delta^* + p^*$ pour que le consommateur qui se trouve à la même distance de deux glaciers bénéficie d'une utilité positive. La condition $U > \sqrt{2kC/\mu}$ pour qu'un premier glacier puisse s'installer sur la plage est alors respectée *ipso facto*.

Nota Bene 2 : Le surplus moyen d'un consommateur est $U - p^* - k\delta^*/2 = U - 5kL/4n^*$. Toutes choses égales d'ailleurs, la satisfaction d'un consommateur est donc d'autant plus élevée que le nombre n^* est plus grand.

* *

Nous pouvons généraliser les leçons que fournit cet exemple. *Généraliser la leçon*
 Considérons un produit susceptible d'être différencié en variétés dont la production demande le même coût fixe et qui se distinguent l'une de l'autre par la valeur x d'un paramètre qui « mesure » un attribut qualitatif.

Supposons que chaque consommateur ait une variété préférée x^0 dont la consommation lui procure le plaisir U , les autres variétés lui procurant un plaisir moindre $U - k|x - x^0|$. Supposons que l'étendue de la différenciation en variétés embrasse un intervalle de longueur L parmi les valeurs de x .

Si nous notons d la distance $|x - x^0|$, la satisfaction qu'une variété procure au consommateur s'écrit comme ci-dessus $S = U - p - kd$: chaque consommateur évalue de façon subjective la qualité des variétés du produit, selon ses propres besoins et préférences (si l'on prenait en compte la diversité des degrés de finition il faudrait dire que le consommateur évalue leur rapport qualité/prix).

On retrouve alors les résultats ci-dessus : le marché se divise en segments au centre desquels se trouve la variété offerte par une entreprise. Celle-ci jouit d'un monopole à l'intérieur de ce segment et se trouve en concurrence par le prix à sa frontière. À l'équilibre le nombre des variétés sera n^* et leur prix sera p^* .

Ce résultat s'étend au cas où les variétés se différencient selon deux attributs : le découpage du marché se fait alors non sur une droite mais sur un plan, chaque segment étant délimité par un hexagone dont le centre représente une variété offerte. L'entreprise est en position de monopole envers les clients dont le besoin est représenté par un point intérieur à l'hexagone, un client dont le besoin est situé sur la frontière de deux hexagones choisit indifféremment entre deux variétés (trois s'il se trouve au sommet d'un hexagone) et l'entreprise est en concurrence par le prix avec celles qui offrent les variétés placées au centre des hexagones voisins.

Le raisonnement s'étend *mutatis mutandis* au cas où les variétés se différencient selon plusieurs attributs x, y, z , etc.

Comme tout modèle celui de la concurrence monopolistique est essentiellement schématique : il montre seulement, en partant d'hypothèses simplificatrices, comment peut s'établir un équilibre de long terme sur un marché où sont offertes (et demandées) diverses variétés d'un même produit.

*Surmonter le
schématisme du
modèle*

Il n'éclaire donc pas la dynamique de l'entrée des nouvelles entreprises sur le marché (comment choisissent-elles la variété qu'elles vont offrir ? comment réagissent les autres entreprises ?), mais seulement l'aboutissement de cette dynamique, aboutissement qui se situe dans le long terme et qui peut donc reculer à mesure que le temps avance. Il n'éclaire pas non plus l'innovation qui, changeant le coût fixe et faisant apparaître de nouveaux paramètres qualitatifs (et donc de nouveaux besoins), transforme les conditions de l'équilibre.

Ce modèle, qui aboutit à un équilibre statique, appelle donc le dépassement qui permet de modéliser une dynamique (il en est de même, notons le, du modèle de la concurrence parfaite).

* *

Ce bref examen montre que d'un point de vue purement théorique un marché peut obéir à trois régimes différents :

– *concurrence parfaite*, si (1) le rendement d'échelle est d'abord croissant, puis décroissant, (2) pour un prix égal au minimum du coût moyen de production, la demande est beaucoup plus forte que la quantité qui correspond à ce minimum ;

– *monopole naturel*, si (1) le rendement d'échelle est croissant, (2) le produit ne se prête pas à une différenciation en variétés (exemple : lingot de cuivre pur) ;

– *concurrence monopolistique*, si (1) le rendement d'échelle est croissant, (2) le produit se prête à une différenciation en variétés.

Savoir si un marché obéit ou non au régime de la concurrence parfaite n'est donc pas une affaire de conviction ni d'idéologie, mais de *physique* : la question est tranchée par la position relative de la fonction de coût et de la fonction de demande.

Le régime du marché est une question de physique

L'affrontement idéologique entre la concurrence et le monopole occupe cependant tant de place dans la théorie à l'œuvre que l'expression « concurrence monopolistique » semble être un oxymore : parmi les économistes eux-mêmes, nombreux sont ceux qui ne sont pas familiers avec ce régime.

Il faut pourtant le considérer : nous montrerons que la concurrence monopolistique mérite de devenir, en lieu et place de la concurrence parfaite et du monopole, *la référence* pour les modèles qui formalisent l'économie informatisée.

Fonction de production macroéconomique

La fonction de production que nous avons considérée concerne *une* entreprise (plus précisément, la fraction d'une entreprise qui est consacrée à la fabrication d'un produit). Les modèles macroéconomiques utilisent la technique de l'*agent représentatif* pour raisonner sur une branche d'activité (ensemble des fractions d'entreprise qui fabriquent un même produit), assimilée à une entreprise unique.

La fonction de production d'un tel agrégat n'est cependant pas celle qui convient pour une entreprise. Supposons que le marché du produit considéré obéisse au régime de la concurrence parfaite ou de la concurrence monopolistique. Les n^* entreprises produisent chacune la quantité q^* et si la demande croît l'augmentation de la production se fera par croissance du nombre des entreprises. Il en résulte que la fonction de production d'une branche d'activité est à rendement constant.

q^* est par ailleurs fonction de w et de r : si le prix relatif des facteurs change, les quantités K et L mises en œuvre par chaque entreprise changent également. Il est donc légitime de représenter

la production d'une branche d'activité par une fonction de Cobb-Douglas à rendement constant : c'est ce qui explique le succès de cette fonction dans la modélisation macroéconomique.

Par ailleurs les modèles qui formalisent la croissance (Ramsey, 1928) et ceux qui fondent la théorie des échanges internationaux (Ricardo 1817, Heckscher-Ohlin 1933, Helpman 1985) assimilent un pays entier à une entreprise et l'ensemble de sa production à un ou deux produits.

Dans ce cas, w et r ne peuvent plus être considérés comme des paramètres du modèle car le prix d'un facteur de production augmente quand il est davantage demandé : il est donc légitime de postuler dans ces modèles un rendement d'échelle *décroissant*.

Les économistes ne devraient cependant pas utiliser, lorsqu'ils considèrent les entreprises, des fonctions de production qui ne peuvent être légitimes qu'au niveau d'un agrégat macroéconomique.

(À suivre : économie du dimensionnement, coût fixe, emploi et stratégie dans l'économie)

Bibliographie

Bertrand Gille, *Histoire des techniques*, Gallimard, Collection de la Pléiade, 1978.

Christian Saint-Etienne, *L'économie*, Odile Jacob, 2013.

Michel Volle, *économie*, Economica, 2014.

Francis Edgeworth, *Mathematical Psychics*, Kegan Paul, 1881.

Vilfredo Pareto, *Manuale di Economia politica*, Società editrice libraria, 1906.

John Hicks, *Value and Capital*, Oxford University Press, 1939.

Irving Fisher, *The Nature of Capital and Income*, Macmillan, 1906.

Daniel Fixari, « Le calcul économique, ou de l'utilisation des modèles irréalistes », *Annales des Mines*, avril 1977.

Gérard Debreu, *Theory of Value : An Axiomatic Analysis of Economic Equilibrium*, Yale University Press, 1959.

David Ricardo, *On the Principles of Political Economy and Taxation*, 1817.

Charles Cobb et Paul Douglas, « A Theory of Production », *American Economic Review*, 1928.

Robert Solow, « A contribution to the theory of economic growth », *The Quarterly Journal of Economics*, 1956.

Wassily Leontief, *The Structure of the American Economy 1919-1929*, Harvard University Press, 1941.

Jean Tirole, *Théorie de l'organisation industrielle*, Economica, 1993.

Joan Robinson, *The Economics of Imperfect Competition*, Macmillan, 1933.

Edward Chamberlin, *The Theory of Monopolistic Competition*, Harvard University Press, 1933.

Frank Ramsey, « A Mathematical Theory of Saving », *Economic Journal*, 1928.

Bertil Ohlin, *Interregional and International Trade*, 1933.

Elhanan Helpman et Paul Krugman, *Market Structure and Foreign Trade*, MIT Press, 1985.

Ce premier numéro des *Cahiers de l'iconomie* inaugure la publication des travaux de l'institut de l'iconomie (iconomie.org), think tank indépendant.

L'iconomie est la représentation schématique d'une société dont l'économie s'appuie sur la synergie de la microélectronique, du logiciel et de l'Internet et qui, étant par hypothèse parvenue à l'efficacité, serait sortie de la crise de transition actuelle. Il ne s'agit pas d'une prévision mais d'un *repère* proposé aux intentions et aux stratégies.

Le phénomène de l'informatisation se déploie sur les plans scientifique, technique, économique, psychologique, sociologique, culturel, philosophique, politique, géopolitique, etc : l'iconomie comporte donc l'ensemble de ces dimensions.

Chacun des articles réunis dans ce numéro éclaire, comme un coup de projecteur, un des aspects de ce phénomène.

10 €

ISBN 978-2-8190-2563-4



9 782819 025634