



HAL
open science

Les développeurs, une nouvelle classe sociale?

Paris Chrysos

► **To cite this version:**

Paris Chrysos. Les développeurs, une nouvelle classe sociale?. Paris Innovation Review, 2017. halshs-01615472

HAL Id: halshs-01615472

<https://shs.hal.science/halshs-01615472>

Submitted on 18 Oct 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Formerly ParisTech Review

Les développeurs, une nouvelle classe sociale?



Paris Chrysos / Professor de management de l'innovation, ISC Paris / September 28th, 2017

talents and competences

Icônes de l'innovation, les développeurs travaillent au cœur de la révolution numérique depuis ses débuts et continuent à animer les transformations sociales et économiques induites par les technologies émergentes. Pourtant, ces acteurs ont été injustement négligés par la recherche en sciences humaines et sociales. Cette sous-représentation tient, en partie, au fait que les développeurs ne sauraient être classés dans les statuts sociaux ou économiques habituels sans ignorer les dynamiques sous-jacentes qui font leur spécificité. Pour saisir ces dynamiques, il faut distinguer « travail officiel » et « projets personnels », une division fondamentale dans leur modus operandi. Ce « double jeu » a des conséquences à la fois sur la manière dont nous concevons l'action générale et sur la manière dont le potentiel technologique est exploré dans le domaine industriel. Dans le même mouvement, les développeurs créent de nouvelles formes d'organisation telles que les BarCamps et les Hackathons, pour soutenir leurs projets d'exploration personnels.

Pour les développeurs, le développement personnel et technologique sont étroitement liés. Trois portraits illustrent cette dynamique statutaire : Jay, Michael and Lefteris.

Projets parallèles et auto-développement

Jay a développé une application pour eBay. Alors que de nombreux utilisateurs de cette plateforme populaire effectuent le plus souvent des transactions ponctuelles (en vendant ou en achetant des articles d'occasion), Jay l'utilise plus massivement, en effectuant de nombreuses transactions d'articles similaires, avec un bénéfice marginal. Pour gérer ces transactions, il développe, pendant son temps libre, une application destinée à automatiser les tâches de logistique et de comptabilité liées à son activité. En développant son application pour un usage personnel, Jay envisage une nouvelle possibilité : celle de vendre l'application elle-même. Il devient ainsi un utilisateur-développeur-entrepreneur (UDE). Toutefois, l'utilisation personnelle diffère de l'exploitation commerciale car les consommateurs éventuels exigeront toutes les fonctionnalités d'un produit développé par une entreprise (interface conviviale, support clientèle, etc.). Non disposé à investir plus de

temps personnel et confronté à une concurrence émergente, Jay décide finalement d'abandonner le projet. Cependant, il l'utilise comme étude de cas pour enseigner la gestion de projet dans le cadre de son travail officiel, à l'université de Wichita, au Kansas.

Michael est mathématicien. Je l'ai rencontré dans un séminaire gratuit au siège de Twitter, à San Francisco, où il donnait une conférence sur un nouveau langage de programmation pendant son temps libre. Il a commencé à travailler dans le secteur avant la bulle Internet, dans une start-up fondée par un professeur de Stanford. Quand la start-up a été emportée, comme tant autres, par l'éclatement de la bulle, Michael est recruté par Amazon pour devenir architecte de plateforme pendant plusieurs années, avant de partir chez Google. Pour Michael, le développement des nouvelles technologies tient plus d'une activité professionnelle que personnelle, même s'il n'a pas suivi un parcours classique. Ainsi, il peut être décrit comme un développeur-entrepreneur (DE), qui acquiert de nouvelles compétences et les utilise pour gagner sa vie.

Lefteris a fait des études de soins infirmiers, sa deuxième passion après les ordinateurs. Il est parvenu à conjuguer ses deux passions en travaillant dans une clinique pour les technologies du sommeil. Avec quelques amis, il anime hackerspace.gr, un sous-sol de la banlieue proche d'Athènes, où les gens peuvent passer du temps et utiliser le matériel. Pour répondre au *Space App Challenge* de la NASA, certains membres de ce hackerspace ont développé SATNOGS (*Satellite Networked Open Ground Station*), un réseau ouvert et DIY (*do it yourself*) de stations terrestres destinées aux satellites. Lauréat d'un prix d'une autre compétition, le groupe utilise ces ressources pour fonder la *Free Space Foundation*, poursuivant l'exploration spatiale à sa propre manière. Lefteris s'est engagé dans cette démarche pour son plaisir personnel, en marge de son travail officiel, devant ainsi un utilisateur-développeur (UD).

Comme le montrent ces trois portraits, les développeurs se définissent souvent plus par leurs projets personnels que par leurs diplômes officiels ou leur travail officiel. En sciences de gestion, mesurer l'action collective est d'une importance capitale. Mais l'impact de l'action des développeurs est difficile à mesurer. Par exemple, une telle mesure ne peut être effectuée par des méthodes traditionnelles comme la mesure de la population de la classe créative définie par Richard Florida (c.-à-d. l'analyse des données démographiques relatives à une liste de métiers). Au cours d'une présentation récente de mon travail dans le cadre de l'OCDE, j'ai suggéré qu'une mesure pertinente pourrait provenir des statistiques liées aux instruments spécifiques que les développeurs utilisent pour créer des applications, les interfaces de programmation (*Application Programming Interface* ou API). Mais une telle mesure ne fournirait que des résultats approximatifs, car le développeur est libre d'utiliser des comptes multiples pour une API donnée.

Ce problème de la précision est en même temps l'une des sources du phénomène : la grande accessibilité des instruments destinés aux développeurs est l'un des facteurs clés permettant l'expansion de cette population, car le processus pour devenir un développeur est ouvert. Cependant, l'importance de cette « classe » tient essentiellement à ses caractéristiques qualitatives, plutôt que quantitatives, car elle semble défier la manière dont l'action collective a été pensée jusqu'ici.

Statuts complexes

De nos jours, les activités réalisées pendant le temps libre sont souvent liées aux loisirs et, par conséquent, elles ne sont pas vraiment prises au sérieux. Cependant, une exploration des origines de la théorie statutaire fournit une compréhension qualitative de l'importance des projets entrepris pendant le temps libre, au-delà de leur statut formel. La théorie statutaire remonte aux bouleversements du XV^e siècle et aux travaux du juriste français Jean de Terrevermeille (1370-1430). Soutenant que le roi Charles VI ne peut refuser l'accession au trône de son fils, Charles VII, Terrevermeille établit dans son *Tractatus* une distinction entre propriété privée et statut : être roi consiste à assumer les responsabilités correspondant à une fonction publique, et non à jouir des avantages d'un héritage privé. Ainsi, la couronne doit être attribuée au fils du roi, en dépit du refus du roi, car elle appartient au public, pas au roi lui-même. Bien évidemment, les arguments de Terrevermeille n'étaient pas suffisants pour entraîner un changement historique aussi important et il a fallu une guerre, la guerre de Cent Ans, pour entériner cette nouvelle règle.

Au cours de l'Histoire, l'introduction de statuts différents a contribué à la formation de l'État moderne, ainsi que des entreprises modernes, en permettant une séparation des rôles et la constitution de grandes organisations. Par exemple, le fait qu'un membre du conseil d'administration ne puisse donner des ordres à un employé de l'organisation, à moins d'y être explicitement autorisé par les statuts des dirigeants, est une condition préalable pour mettre en œuvre une division complexe du travail, où chaque niveau hiérarchique correspond à des responsabilités spécifiques. En somme, aujourd'hui, nous acceptons globalement qu'un statut fournisse à la fois la *légitimité* pour exercer une autorité et le cadre de *responsabilités* correspondantes.

Ainsi, l'exercice d'une activité au-delà d'un statut n'a pas toujours été une possibilité évidente. De fait, les plus importantes théories de l'action sont fondées sur le présupposé que les statuts sont déjà en place.

Max Weber, l'un des fondateurs de la sociologie, a largement bâti ses catégories sociologiques sur ce qu'il dénomme la « *Legitimitätsglaube* », c'est-à-dire la croyance dans la légitimité. Pour Weber, une forme de légitimité correspond à une croyance dans la légalité, qu'il conçoit comme soumission à des statuts corrects, d'un point de vue formel. Fait intéressant, pour Henri Fayol, l'un des pères du management, la personnalité joue également un rôle crucial. Plus précisément, celui-ci explique que pour devenir un bon chef, l'autorité personnelle est un complément indispensable de l'autorité statutaire, une dimension du travail qui est souvent négligée dans les modèles de management.

Quoi qu'il en soit, l'émergence des technologies contemporaines met à mal la vision statutaire traditionnelle des acteurs. D'un côté, les technologies avancées sont de plus en plus disponibles, à des coûts abordables pour les individus. De l'autre, le temps libre n'est pas soumis aux contraintes d'une fonction publique ou professionnelle.

Cependant, une telle liberté requiert un cadre de travail qui soutienne les aspirations personnelles et les nouvelles compétences de développement. C'est ainsi que les développeurs ont inventé de nouvelles formes d'organisation, adaptées à leur capacité à explorer les nouvelles technologies. Les *BarCamps* et les *Hackathons* correspondent parfaitement à ces nouvelles formes.

BarCamps et Hackathons

Les *BarCamps* sont des réunions éphémères qui peuvent se dérouler dans différents lieux, en fonction des organisateurs : le siège d'une entreprise ou d'une institution, un espace de coworking, un café... En général, les organisateurs choisissent un sujet large qui sera exploré par un appel ouvert à la participation. L'événement débute par une auto-présentation des participants réunis qui formulent collectivement l'ordre du jour des discussions. Des petits groupes de travail se forment autour de thématiques choisies, en fonction des salles et des créneaux horaires disponibles. À la fin de l'événement, tous les participants se réunissent une dernière fois et chaque groupe présente une synthèse des discussions.

Les *Hackathons* sont des espaces d'exploration. Comme pour les *BarCamps*, les participants sont invités à participer par des appels ouverts. Plus orientés et plus intenses que les *BarCamps*, ils commencent généralement par un séminaire de présentation sur les nouvelles innovations technologiques. Puis, les participants proposent des concepts à explorer en fonction de cette technologie et invitent les autres à les rejoindre dans un exercice de développement intensif qui dure pratiquement trois jours (et nuits). Au cours de cet exercice, les fournisseurs technologiques peuvent vérifier la manière dont celle-ci est utilisée par les développeurs tandis que les participants sont libres de développer le concept qu'ils souhaitent. Enfin, toutes les équipes présentent leurs applications ou leurs prototypes. Chaque création illustre un certain aspect du potentiel de la technologie. Enfin, les meilleurs prototypes sont récompensés.

Les *BarCamps* et les *Hackathons* n'exigent aucun statut particulier pour devenir organisateur ou participant. Par ailleurs, il n'existe aucune autorité pour coordonner la formation des groupes de personnes ou définir les concepts qui seront explorés. Les deux événements fournissent des opportunités, au détriment du temps investi par les participants. Tandis que dans les *BarCamps*, la responsabilité des résultats possibles est partagée par l'ensemble des participants, dans les *Hackathons*, ce sont les organisateurs qui doivent « nourrir » l'exploration par leurs connaissances d'une technologie spécifique. Par ailleurs, le décernement d'un prix permet de stimuler la concurrence entre les différents groupes.

Les résultats possibles des deux types de réunions sont à la fois d'ordre cognitif et social. Dans les *BarCamps*, les participants peuvent trouver des personnes intéressées par les mêmes sujets et réseaux émergents ou encore créer une communauté sur un thème spécifique. En parallèle, ils peuvent s'informer des tendances récentes dans un domaine et tenter de prédire son évolution (en termes d'usages, de marchés ou de technologies). Les *Hackathons* sont l'occasion d'un réseautage plus ciblé entre ceux qui gèrent les plateformes technologiques et ceux qui l'utilisent, mais également de créer une communauté de développeurs. Par ailleurs, ils permettent d'entendre des analyses au sujet des opportunités offertes par les technologies et donnent des chances d'acquérir des compétences pertinentes... Les organisateurs et les participants peuvent apporter un regard neuf sur les possibilités offertes par la technologie en question.

Les deux types d'événements semblent ignorer les règles de base des environnements sociaux habituels. Même dans les formes de coordination libre à travers les réseaux sociaux, on s'attend toujours à pouvoir identifier les connaissances. Les *BarCamps* et les

Hackathons sont conçus pour créer des surprises, en réunissant un public d'inconnus, au-delà des réseaux sociaux spécifiques. Par ailleurs, ils génèrent une sorte d'intimité, une ambiance amicale où tout le monde peut s'exposer librement. Ainsi, les participants peuvent proposer des concepts qui ne seront probablement pas faisables, ni même utiles, tester des interfaces incomplètes ou des ébauches de plateforme, discuter d'idées utopiques ou naïves. Un tel environnement s'accorde parfaitement avec les projets personnels des développeurs et le passage d'un statut à un autre (par exemple, du statut d'utilisateur à celui d'entrepreneur). Pouvoir exposer, c'est aussi pouvoir tester et explorer une identité potentielle.

La *foggy economy*: comment rationaliser tout en continuant à explorer

Loin d'être un problème marginal ou tout simplement personnel, les processus explorés par les développeurs sont au cœur des transformations industrielles et sociétales actuelles. Je propose le concept de *foggy economy* (économie « brumeuse ») pour désigner la condition contemporaine d'incertitude qui pèse sur la localisation de la valeur. Par exemple, nous sentons bien que les technologies *blockchain* ont un fort potentiel, mais nous ignorons comment les exploiter dans différents contextes. Une partie du potentiel des nouvelles technologies peut ainsi être dévoilé et le processus de rationalisation s'effectue par l'introduction de nouvelles formes de statuts. En même temps, une autre partie du même potentiel reste à explorer, en continuant à nous engager dans des processus d'exploration ouverts, susceptibles de conduire à des découvertes surprenantes.

Cet article a fait l'objet d'une présentation dans le cadre du séminaire « Le Modèle californien », organisé par Monique Dagnaud et Olivier Alexandre à l'EHESS.

More on [paris innovation review](#)

By the author

[Les développeurs, une nouvelle classe sociale?](#) on September 28th, 2017



This content is licensed under a Creative Commons Attribution 3.0 License
You are free to share, copy, distribute and transmit this content



5 quai Voltaire 75007 Paris, France - Email : contact@parisinnovationreview.com / Landline : +33 1 44 50 32 89