

numéro

6

1024

B U L L E T I N

de la société informatique
de France

juillet

2015

COMITÉ DE RÉDACTION

SYLVIE ALAYRANGUES
Université de Poitiers

VINCENT AUTEFAGE
Université de Bordeaux

OLIVIER BAUDON
Université de Bordeaux

COLIN DE LA HIGUERA
Université de Nantes

JEAN-PAUL DELAHAYE
Université Lille 1

OLIVIA DUMAS
Académie de Caen

CHRISTINE FROIDEVAUX
Université Paris-Sud

THIERRY GARCIA
*Université de Versailles Saint-Quentin-
en-Yvelines*

MARIE-CLAUDE GAUDEL
Université Paris-Sud

CLAUDIA MARINICA
Université de Cergy-Pontoise

PHILIPPE MARQUET
Université Lille 1

DENIS PALLEZ
Université Nice Sophia Antipolis

VINCENT RIBAUD
Université de Bretagne Occidentale

ÉRIC SOPENA
*Université de Bordeaux, rédacteur en
chef*

Contact : 1024@societe-informatique-de-france.fr



Cette œuvre est mise à disposition sous licence Attribution - Pas de Modification 4.0 France.
Pour voir une copie de cette licence, visitez

<http://creativecommons.org/licenses/by-nd/4.0/deed.fr>

ou écrivez à Creative Commons, 444 Castro Street, Suite 900, Mountain View, California,
94041, USA.

SOCIÉTÉ INFORMATIQUE DE FRANCE
Institut Henri Poincaré, 11 rue Pierre et Marie Curie, 75231 Paris Cedex 05

Prix public : 32 € (adhérents SIF : -30%)

Directeur de la publication : Jean-Marc Petit

ISSN : 2270-1419

Couverture : d'après une maquette réalisée par Lollygraph.com.

SOMMAIRE DU N° 6

SIF

Le mot du président, <i>Jean-Marc Petit</i>	3
Actualité de la SIF, <i>Sylvie Alayrangues</i>	7

SCIENCE

Il n'est pas impossible de résoudre le jeu d'échecs, <i>Julien Lemoine et Simon Viennot</i>	15
Humanités numériques, <i>Serge Abiteboul et Florence Hachez-Leroy</i>	41

ENTRETIENS

Du datagramme à la gouvernance de l'Internet, <i>Entretien avec Louis Pouzin</i>	59
--	----

TÉMOIGNAGE

Le plan Informatique pour tous dans l'académie de Créteil : une enquête d'évaluation, <i>Jean-Pierre Archambault</i>	77
--	----

PRIX ET DISTINCTIONS

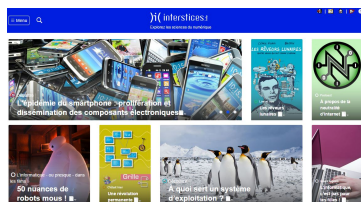
Michael Stonebraker, ACM Turing Award pour ses contributions au domaine de la gestion de données, <i>Patrick Valduriez</i>	85
Bilan du prix de thèse Gilles Kahn 2014, <i>Michel Riveill</i>	87
Conception et implémentation de cryptographie à base de réseaux, <i>Tancrède Lepoint</i>	89
Visualisation physique et tangible de l'information, <i>Yvonne Jansen</i>	93
Réponse à des requêtes conjonctives en présence de règles existentielles – décidabilité, complexité et algorithmes, <i>Michaël Thomazo</i>	97
Prix de thèse Gilles Kahn 2015, <i>Michel Riveill</i>	101

RÉCRÉATION

Moins de 20 cartes, <i>Jean-Paul Delahaye</i>	105
---	-----

Science informatique dites-vous ? Deux revues et un blog à votre service

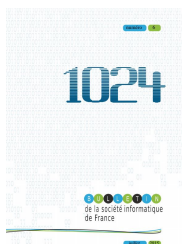
Il est important que, des actrices et acteurs... aux utilisateurs et utilisatrices du numérique, nous partagions une culture en sciences du numérique, pour comprendre et maîtriser les technologies qui en sont issues. Pour concrétiser ce partage avec le monde de la recherche, la Société informatique de France, Inria et le CNRS proposent deux revues et un blog : profitons-en !



I(n)terstices ? La revue de culture scientifique en ligne qui invite à explorer les sciences du numérique, à comprendre ses notions fondamentales, à mesurer ses enjeux pour la société, à rencontrer ses acteurs et actrices.



Binaire ? Le blog du monde.fr qui parle de l'informatique, de ses réussites, de son enseignement, de ses métiers, de ses risques, des cultures et des mondes numériques.



1024 ? Une revue pour les professionnelles et les professionnels du monde de l'enseignement, de la recherche et de l'industrie de l'informatique qui permet de découvrir les différentes facettes de cette science.



Le mot du président

Jean-Marc Petit¹

Chère adhérente, cher adhérent,

À quelques jours des vacances d'été, ce 7^e numéro de 1024 offre l'opportunité de faire le point sur les actions que la SIF a menées en 2015 et de partager avec vous quelques réflexions sur l'avenir.

o O o

Je voudrais tout d'abord aborder un sujet majeur pour la SIF, à savoir *l'enseignement de l'informatique* au primaire, au collège et au lycée. Rappelons que la discipline informatique est clairement reconnue en tant que telle dans l'enseignement supérieur, mais ne l'est toujours pas dans l'enseignement secondaire et primaire. Le corollaire est qu'il n'y a toujours ni CAPES ni agrégation d'informatique en France.

Ce début d'année 2015 aura pourtant marqué une rupture : le président de la République et la ministre de l'Éducation nationale ont fait de nombreuses annonces pour introduire de nouveaux programmes portant sur l'informatique au primaire et au collège, une nouvelle option d'enseignement d'exploration d'informatique et de création numérique² en classe de seconde générale et la généralisation de la spécialité ISN aux classes de terminale L et ES.

1. Président de la Société informatique de France, professeur à l'INSA de Lyon, LIRIS (UMR 5205 CNRS), jean-marc.petit@insa-lyon.fr.

2. ICN : <http://www.education.gouv.fr/cid89179/projet-de-programme-pour-un-enseignement-d-exploration-d-informatique-et-de-creation-numerique.html>

Ces annonces constituent une victoire pour toutes les personnes qui ont œuvré pour cette reconnaissance de la discipline informatique. La SIF a joué un rôle moteur dans cette prise de conscience au plus haut niveau de l'État et s'en est félicitée³. Ses arguments pour former nos plus jeunes concitoyens à l'informatique au XXI^e siècle ont convaincu les responsables politiques qu'il fallait donner à tous les français les clés pour appréhender le monde numérique dans lequel ils vivent. Les enfants de la République pourront enfin comprendre qu'il n'y a rien de magique à leurs activités journalières, comme envoyer un SMS ou publier une photo sur leur réseau social favori, juste la main de l'homme et de la machine !

Cela dit, il reste maintenant à mettre en œuvre concrètement ces réformes. Car si ces avancées sont bien réelles, reste l'épineux problème de l'existence du vivier d'enseignants qui vont enseigner dès la rentrée 2016 l'informatique aux élèves. Puisqu'il n'y a pas – et qu'il n'y aura pas à court terme – de professeurs formés à l'informatique, ces nouveaux cours seront enseignés par les professeurs de mathématiques et de technologie au collège. Plusieurs associations de professeurs appellent déjà à l'aide devant cet incroyable chantier.

Dans cette phase opérationnelle, deux mauvaises solutions s'offrent à nous : soit nous soutenons ces professeurs de mathématique ou de technologie pour qu'ils puissent faire leurs enseignements dans de moins mauvaises conditions, au risque d'être associés à l'échec attendu de ces enseignements ; soit nous refusons et nous risquons de nous isoler, voire de nous replier sur nous-mêmes.

Dans ce contexte, la volonté de la SIF est de promouvoir un enseignement de qualité prodigué par des professeurs formés à la discipline informatique, idéalement titulaires d'un CAPES ou d'une agrégation d'informatique mais, dans le même temps, de promouvoir aussi toutes les initiatives qui visent à aider celles et ceux qui se retrouvent sur le front de l'enseignement de l'informatique sans y avoir été préparés.



Sur le plan de l'organisation de la SIF, nous avons opéré récemment quelques changements structurels :

- *Nouvel organigramme au sein du CA* : une vice-présidente et un comité de direction sont venus compléter utilement l'organisation classique de l'association. Un adjoint a été nommé sur chaque poste important au sein du CA⁴.
- *Création d'une vingtaine de groupes de travail*⁵ (GT) regroupés au sein de quatre domaines : enseignement, recherche, médiation et transversal (pour

3. Voir le communiqué : <http://www.societe-informatique-de-france.fr/wp-content/uploads/2015/05/2015-05-Communique-SIF-annonces-NVB-et-FH.pdf>

4. Voir <http://www.societe-informatique-de-france.fr/la-sif/ca-2015/>

5. <http://www.societe-informatique-de-france.fr/la-sif/ca-2015/groupes-de-travail-de-la-sif/>

tout ce qui touche à l'organisation de la SIF, par exemple la communication, les personnes physiques ou les finances).

Ces GT se veulent des structures « agiles » qui vont bien au-delà du CA de la SIF. Leur ambition est de structurer, rationaliser et organiser les efforts de toutes celles et ceux qui ont pour ambition de promouvoir l'informatique comme une science et technique à part entière.

— *Lancement du conseil des associations* : cette action de la SIF est un travail de longue haleine qui contribue à unifier le paysage français sur les thématiques liées à l'informatique. Nous sommes en train de co-construire avec les associations partenaires un mode de fonctionnement pour unir les forces des uns et des autres, tout en respectant la richesse et l'autonomie de chacune d'elles.

o O o

Nous avons aussi engagé au sein du CA des réflexions sur deux sujets étroitement liés : le modèle économique et le nombre d'adhérents à l'association. Rappelons que nous avons deux catégories d'adhérents : les personnes « physiques » et les personnes morales, qui regroupent des centres INRIA, des UMR de l'INS2I du CNRS, des équipes d'accueil du ministère de l'enseignement supérieur, des départements d'IUT, des départements ou unités de formation d'université, des écoles d'ingénieurs, des associations thématiques ou professionnelles...

		Personnes morales	Personnes physiques
Budget	40 k€	90 %	10 %
Nombre d'adhérents	500	10 %	90 %

TABLE 1. Ordre de grandeur du budget et du nombre d'adhérents de la SIF

Dans ce contexte, nous avons deux objectifs devant nous : *augmenter notre budget*, notamment pour pouvoir rémunérer sur ressources propres un.e délégué.e général.e ou un.e secrétaire général.e, et *augmenter notre représentativité* afin de peser plus significativement sur le débat public lorsque nous prenons des positions. Sur ces deux objectifs, la Table 1 donne la tendance générale et pointe une inversion de valeurs sur la contribution respective des personnes morales et physiques. Nous pouvons en tirer divers enseignements, notamment que la SIF est financée principalement par ses personnes morales et que sa représentativité en terme de nombre est assurée par ses personnes physiques. Ces éléments nous amèneront à proposer

lors de la prochaine assemblée générale des évolutions pour relever ces deux défis à moyen terme.

o O o

Comme faits saillants de ce début d'été, la SIF est partenaire du projet Class'code, financé par le programme d'investissements d'avenir 2015. Ce projet s'attaque à la formation à l'informatique dès le plus jeune âge et regroupe un consortium des principaux acteurs académiques, industriels et associatifs du domaine. Il ambitionne de former sur 5 ans, en ligne et sur environ 1 500 lieux, les 300 000 formateurs dont le pays a besoin, y compris dans les milieux géographiquement et socialement écartés. La SIF a aussi été à l'initiative d'un communiqué commun avec les principales sociétés savantes scientifiques sur le projet de décret sur le doctorat⁶.

o O o

La rentrée 2015 s'annonce donc très riche avec la poursuite de l'ensemble de ces dossiers, mais aussi le lancement d'une nouvelle campagne d'adhésion, la réalisation d'un sondage auprès des personnes physiques pour avoir un retour sur nos actions et enfin le dépôt de la demande d'Association reconnue d'utilité publique (ARUP) pour changer le statut de notre association. Nous devrions recevoir un avis du Conseil d'État dans les deux années à venir sur cette action importante pour la SIF et ses adhérents, notamment pour l'exonération de taxes.

o O o

Dans ce numéro, vous trouverez enfin toutes les actualités de la SIF qui vous permettront de mesurer notre engagement sur des problèmes réels et concrets liés à la promotion de la discipline informatique.

En terme de rendez-vous, il faut noter dès à présent que le prochain congrès de la SIF se déroulera à Strasbourg du 28 au 29 janvier 2016. L'assemblée générale aura lieu comme usuellement la veille au soir. Les comités scientifique et d'organisation sont en train de préparer un programme tout à fait passionnant sur *Informatique et Europe*. Nous comptons sur votre présence pour ce moment important de la vie de notre association.

Bonne lecture de ce nouveau numéro et bonnes vacances à tous.

6. voir <http://www.societe-informatique-de-france.fr/recherche/communiqu%C3%A9-sur-le-doctorat/>, cosigné par la Société de mathématiques appliquées et industrielles (SMAI), la Société française de statistique (SFdS), la Société mathématique de France (SMF), la Société française de physique (SFP), la Société française d'optique (SFO), la Société chimique de France (SCF), ainsi que la Société professionnelle des enseignants et chercheurs en informatique de France (Specif-Campus).



Actualité de la SIF

Informations collectées par Sylvie Alayrangues

Une rubrique pour vous tenir informé de la vie de la SIF et vous inviter à y participer. Vous trouverez notamment ici des annonces de manifestations organisées ou soutenues par la SIF, un point sur les dossiers en cours, des appels à participation...

La SIF vous parle d'elle

Du côté du Conseil scientifique : séminaire des doctorants en informatique

Le Conseil scientifique de la SIF souhaite la mise en place d'un séminaire annuel organisé pour et par les doctorantes et doctorants en informatique, afin de découvrir des branches de l'informatique qu'ils connaissent peu ou mal et d'échanger sur les joies et les peines du doctorat. La ou les journée(s) serai(en)t consacrée(s) à des exposés scientifiques et la soirée à des échanges plus informels.

Si vous êtes en thèse et si vous souhaitez participer à l'organisation de la première édition de ce séminaire, contactez le président du Conseil scientifique, Gilles Dowek, gilles.dowek@inria.fr, en précisant qui vous êtes, le sujet sur lequel vous travaillez, le laboratoire auquel vous appartenez et le nombre d'années écoulées depuis le début de votre thèse. Le Conseil scientifique constituera ensuite un comité de programme et d'organisation, en veillant aux habituels équilibres thématiques, géographiques et démographiques.

La SIF et les associations en informatique

La dernière rencontre SIF - Associations a eu lieu le mardi 9 juin 2015, à Lyon. Lors de cette réunion, les associations présentes ont réaffirmé leur intérêt pour la création au sein de la SIF d'un Conseil des associations. Les représentants de quinze d'entre elles ont d'ores et déjà indiqué qu'ils souhaitent que leur association soit membre de ce conseil. Il s'agit de l'ACONIT, AFIG, AFIHM, AFPC, APMEP, ARIA, ASF, ATALA, ATIEF, EGC, INFORSID, PASCALINE, ROADEF, SPECIF-Campus et UPS.

La SIF espère bien entendu que d'autres les rejoindront afin que l'ensemble de la science informatique soit représentée dans toute sa diversité.

Toutes les informations relatives au Conseil des associations sont disponibles sur le site de la SIF¹.

La SIF vous propose

Retour sur les journées pédagogiques

Les journées pédagogiques 2015 ont eu lieu les 23 et 24 juin au CNAM, avec le soutien du CNRS et d'Inria, autour du thème : « L'enseignement de l'informatique pour les humanités et les sciences sociales ».

Le mardi 23 a débuté par une présentation en duo des aspects didactiques de l'informatique pour les humanités et les sciences sociales par Béatrice Drot-Delange et Éric Bruillard. Une table ronde a suivi sur la question « Quelle initiation à l'informatique en filières sciences humaines et sociales à l'Université ? ». Autour de la table, Isabelle Tellier, Olivier Baudon, David Janiszek et Nathalie Denos ont fait part de leur expérience dans des contextes différents et se sont prêtés au jeu des questions/réponses avec l'assistance.

L'après-midi était consacrée aux humanités numériques. Vous trouverez d'ailleurs dans ce numéro de 1024 un article, cosigné par Serge Abiteboul et Florence Hachez-Leroy, qui en présente la genèse et les enjeux. L'après-midi s'est ouverte sur une intervention de Serge Abiteboul inspirée de ce travail, puis les différents intervenants, Aurélien Berra, Isabelle Tellier, Colin de la Higuera, Arshia Cont, Nathalie Denos, et Stéphanie Netto, ont partagé leur expérience autour de la place de l'informatique dans leurs disciplines (traitement automatique des langues, langues et littératures anciennes, musique...) et/ou dans leurs pratiques pédagogiques.

Le lendemain, trois grands thèmes ont été abordés : l'informatique dans le travail et la formation des historiens, par Stéphane Lamassé, l'enseignement de l'informatique dans le secondaire en Europe, par Françoise Tort, et la formation à l'informatique dans les ESPE, par François Villemonteix.

1. <http://www.societe-informatique-de-france.fr/associations-partenaires/>

Comme à l'accoutumée, les échanges et débats entre les intervenants et l'assistance ont été riches et animés.

Les supports de ces présentations sont accessibles sur la page dédiée à ces journées².

Congrès 2016, « save the date » : 26-27-28 janvier 2016

Le congrès 2016 se déroulera à Strasbourg fin janvier 2016. Au menu, quels enjeux autour de la science informatique en Europe ? (Quels en sont les acteurs ? Comment l'informatique est-elle au cœur du développement industriel ? *quid* de l'informatique en tant que discipline scolaire ? Protection des données, une solution européenne ?...).

L'Assemblée générale de la SIF aura lieu le 26 janvier de 17 h à 19 h, et sera suivie du congrès lui-même, les 27 et 28 janvier.

La SIF soutient

Le prix de thèse Gilles Kahn

Comme chaque année, la période estivale voit l'ouverture des candidatures au prix de thèse Gilles Kahn. Les soumissions ouvrent le 15 juillet. Les dossiers de candidature doivent être remplis d'ici le 10 septembre 2015 et les candidats connaîtront le résultat le 1^{er} décembre 2015. L'annonce officielle des résultats et la remise des prix auront lieu lors du prochain congrès de la SIF.

Peut candidater tout(e) étudiant(e) ayant soutenu son doctorat d'informatique dans une école ou université française entre le 1^{er} septembre 2014 et le 31 août 2015. Toute candidature devra être explicitement soutenue par la directrice ou le directeur de thèse, l'une des co-directrices ou l'un des co-directeurs. Il n'est pas permis à un même encadrant de soutenir deux candidats.

Ce prix a vocation à récompenser de jeunes chercheuses ou chercheurs en informatique et à promouvoir l'ensemble des facettes de la science informatique.

Concours boost your code

En 2015, quatre projets ont atteint la phase finale. Deux d'entre eux ont plus particulièrement retenu l'attention du jury. Loïc Dauphin, passionné de robotique, étudiant à l'ENSEIRB-MATMECA, a été élu « lauréat de Boost your code 2015 » grâce au projet *Aversive++*. Il s'agit d'une bibliothèque qui s'adresse aux développeurs d'applications robotiques. Elle est née dans le club robotique de l'ENSEIRB-MATMECA. Son développement est déjà bien avancé, et Loïc profitera de son année à Inria pour le poursuivre et pourquoi pas créer ensuite une entreprise pour proposer des services aux industriels. Maximiliano Lopez, étudiant du master « Decision

2. <http://www.societe-informatique-de-france.fr/enseignement/j-pedago/j-pedago-2015-programme/>

Support and Business Intelligence » de l'École Centrale Paris a, quant à lui, reçu le « prix spécial du jury de Boost your code 2015 » pour son projet *ElectioVis*. Il s'agit de mettre à disposition du grand public, via une plateforme web, un outil d'aide à la décision utilisant les algorithmes les mieux adaptés à la situation décrite. Une première version a été mise en ligne récemment³.

Le palmarès de l'édition 2015 est accessible en ligne⁴.

C'est aussi l'occasion de donner des nouvelles du lauréat du concours 2013, Alexandre Gauthier-Foichat, qui a mis à profit son année à l'institut pour poursuivre le développement de son logiciel de compositing rebaptisé Natron. L'équipe de développement du logiciel s'est étoffée avec toujours l'objectif de fournir à tous (étudiants amateurs d'effets spéciaux, chercheurs, acteurs de l'industrie du cinéma...) un logiciel de qualité professionnelle. Natron a reçu en janvier dernier le prix de la « meilleure innovation de l'année ». N'hésitez pas à le télécharger⁵ !

Colloque annuel Femmes et Sciences : 13 novembre 2015

L'association Femmes et Sciences organise, pour la première fois à Toulouse, son colloque annuel sur le thème « Choisir et vivre une carrière scientifique ou technique au féminin : pourquoi ? comment ? », le vendredi 13 novembre prochain.

Cette journée est destinée au monde de l'enseignement (personnels de direction, enseignants, conseillers d'orientation) et de la recherche.

Témoignages, tables rondes et conférences permettront de faire le point d'une part sur l'intérêt des métiers scientifiques et techniques pour les jeunes filles et les freins à lever pour qu'elles s'y engagent, et d'autre part sur la place des femmes scientifiques dans le secteur public ou en entreprise, leurs motivations, leurs stratégies et les dernières actions menées en leur faveur.

Le programme complet est disponible sur le site de l'association⁶.

Cette journée de formation est proposée à travers le Plan Académique de Formation, à la rubrique Sciences / Maison des Sciences

Vous pouvez aussi vous inscrire directement sur le site du colloque⁷.

3. <http://www.electiovis.com/>

4. <http://www.inria.fr/actualite/actualites-inria/palmares-de-boost-your-code-2015>

5. <http://natron.inria.fr/>

6. <http://www.femmesetsciences.fr>

7. <http://www.femmesetsciences2015.eventbrite.fr>

La SIF prend position et agit

En enseignement

Dans le numéro précédent de 1024, nous vous parlions d'un projet de MOOC hybride porté par de multiples partenaires (académiques, industriels, collectivités territoriales, associations de terrain...) pour aider à former des éducateurs à l'Informatique pour notre société numérique. C'est maintenant officiel : le projet CLASS'CODE est accepté par le PIA (programme d'investissements d'avenir). Le but est de construire un MOOC enseignant les bases du code, de l'informatique et du numérique, et surtout les façons de transmettre ces connaissances aux enfants. Il s'agit également de créer un réseau de personnes relais sur le terrain pour être au plus près des éducateurs et les épauler dans leur apprentissage et dans leurs actions.

Le projet est piloté par Inria et comprend parmi ses partenaires des régions (dont la région Pays de la Loire), des associations professionnelles, des associations de terrain, les entreprises OpenClassrooms et Magic Makers, et la SIF.

Contact pour la SIF : Colin de la Higuera, cdlh@univ-nantes.fr

En avril, la SIF a réagi par un communiqué⁸ aux annonces du président de la République et de la ministre de l'Éducation nationale concernant le plan numérique pour l'école, depuis le primaire jusqu'au lycée en passant par le collège. Elle s'est également exprimée sur les annonces du CSP, via un communiqué⁹ signé par le président de la SIF, le président du Conseil scientifique et le vice-président enseignement.

Le groupe de travail de la CFEM (Commission française sur l'enseignement des mathématiques) sur les interactions entre enseignement des mathématiques et de l'informatique s'est réuni le 26 mai dernier et a réagi au projet de programmes pour l'école et le collège élaboré par le Conseil supérieur des programmes¹⁰.

Le 30 mai était également publié sur le blog BINAIRE un billet sur la réforme du collège¹¹.

La SIF a également été sollicitée à plusieurs reprises pour débattre de questions d'enseignement, le 17 mars lors d'une journée sur l'enseignement de l'informatique à l'ESPE de Paris¹², le 20 avril à l'Université de Laval (Québec)¹³, le 14 mai à

8. <http://www.societe-informatique-de-france.fr/2015/05/communique-suite-aux-annonces-du-president-de-la-republique/>

9. <http://www.societe-informatique-de-france.fr/wp-content/uploads/2015/04/2015-04-29-Communiqué-SIF-CSP.pdf>

10. <http://www.societe-informatique-de-france.fr/2015/05/cfem-math-info-csp/>

11. <http://binaire.blog.lemonde.fr/2015/05/30/>

12. <http://www.espe-paris.fr/actualite/enseigner-linformatique-sciences-numerique-enjeux-perspectives>

13. <https://www.itis.ulaval.ca/cms/lang/fr/pid/162347>

Lille au congrès de la PEEP¹⁴, le 20 mai aux Journées de l'Orme¹⁵, rencontres annuelles des professionnels du numérique pour l'éducation et la culture, le 3 juillet, à Taichung, au congrès *COMPSAC Symposium on Computer Education and Learning Technologies*, pour présenter les travaux de la SIF en direction de la formation à l'informatique (projet CLASS'CODE)¹⁶.

Enfin, le groupe de réflexion ITIC-EPI-SIF, composé¹⁷ d'enseignants du primaire, du secondaire, de chercheurs, d'enseignants-chercheurs et de diverses personnalités, poursuit ses travaux. Il s'est réuni pour la 25^e fois le 16 juin dernier¹⁸.

En recherche

La SIF est à l'origine d'un communiqué commun à propos du projet d'arrêté sur le doctorat, signé des sociétés savantes et professionnelles scientifiques françaises : Société informatique de France (SIF), Société de mathématiques appliquées et industrielles (SMAI), Société française de statistique (SFdS), Société mathématique de France (SMF), Société française de physique (SFP), Société française d'optique (SFO), Société chimique de France (SCF), ainsi que la Société professionnelle des enseignants et chercheurs en informatique de France (Specif-Campus).

Ce communiqué¹⁹ rappelle la spécificité du doctorat, expérience professionnelle de formation par et pour la recherche, fondamentalement différente des autres diplômes. Les signataires s'inquiètent de la possibilité d'un doctorat au rabais par le biais de la VAE. Ils soulignent qu'une limitation administrative stricte de la durée des thèses à trois ans pourrait entraîner une baisse de qualité dans plusieurs domaines scientifiques. Enfin, ils soutiennent la possibilité de stages doctorants (*internships*) comme césure pendant la thèse.

En médiation scientifique

Que diriez-vous d'un musée de l'informatique et du numérique en France ? Voici l'objectif du projet #MINF : créer non pas un simple musée mais un réseau de lieux d'exposition et de médiation scientifique répartis sur l'ensemble du territoire. Vous pouvez découvrir les grandes lignes de ce beau projet dans un rapport, rédigé par un groupe de travail, remis à l'administrateur général du Cnam et disponible en ligne²⁰.

Le Conseil d'administration de la SIF, réuni les 7 et 8 juillet à Bordeaux, a décidé de soutenir ce projet et de le promouvoir auprès des instances concernées.

14. <http://peep.asso.fr/thematiques/l-ecole-et-le-numerique/l-ecole-au-c-ur-du-numerique-enseigner-l-informatique-table-ronde-du-93eme-congres-peep/>

15. <http://www.orme-multimedia.org/r2015>

16. <http://www.computer.org/web/compsac/2015/celt>

17. http://www.epi.asso.fr/revue/editic/groupe_itic-epi-sif.htm

18. http://www.epi.asso.fr/blocnote/CR_25e_ITIC.pdf

19. <http://www.societe-informatique-de-france.fr/2015/07/communique-commun-doctorat/>

20. <http://www.musee-informatique-numerique.fr>

La SIF était présente à Science & You²¹, le rendez-vous de la médiation scientifique qui s'est tenu à Nancy du 1^{er} au 6 juin 2015.

Au sein d'un stand commun avec le CNRS et Inria, nous avons proposé une animation, « Informatique : halte aux préjugés ! » :

On peut rester anonyme sur Internet ; l'informatique est plus écologique que le papier ; l'ordinateur est une machine intelligente ; les robots dominent le monde ; je suis docteur en informatique donc je peux soigner ton ordi... Autant d'idées reçues qui ont la vie dure ! Et si on en parlait avec la Société informatique de France ?

Nous avons également participé à une session commune CNRS/Inria/SIF « Médiation en science du numérique : un levier pour comprendre notre quotidien » avec Gilles Dowek (Inria Paris Rocquencourt), Jean Mairesse (CNRS INS2I) et Sylvie Alayrangues (Univ. Poitiers, Société informatique de France).

La SIF était également représentée lors des journées « Programmer pour ne pas être programmé », organisées par l'Espace Mendès France à Poitiers du 26 au 30 mai 2015²², à la fois pour la table ronde autour du thème « Programmer pour ne pas être programmé ? Dans et hors l'école » et lors des ateliers organisés le mercredi après-midi.

Sur les enjeux de société

Le Conseil d'administration de la Société informatique de France a décidé de signer en avril la pétition #NiPigeonsNiEspions contre la surveillance généralisée d'Internet²³.

Dans la même lignée, elle soutient la volonté et l'initiative d'OpenClassrooms pour la formation des citoyens en matière de vie privée, à travers leur appel intitulé « Opération (R)enseignement »²⁴.

Mi-mai était également publiée une tribune sur Slate.fr : « Avec la loi renseignement, redécouvrons le mot "fortuit" »²⁵.

21. <http://www.science-and-you.com>

22. <http://emf.fr/21209/>

23. <http://www.societe-informatique-de-france.fr/2015/04/nipigeonsniespions/>

24. <http://www.societe-informatique-de-france.fr/2015/05/operation-renseignement/>

25. <http://www.slate.fr/story/101459/loi-renseignement-fortuit-hasard>



Il n'est pas impossible de résoudre le jeu d'échecs

Julien Lemoine¹ et Simon Viennot²

1. Introduction

Qu'entend-on par « résoudre le jeu d'échecs » ? Tout joueur d'échec s'est un jour confronté à un problème du type « les blancs jouent et matent en trois coups » (voir Figure 1). Pour un tel problème, quelles que soient les réponses du joueur noir, le joueur blanc arrive à mater en trois coups, ou moins. Le problème est correctement analysé dès lors que toutes les réponses du joueur noir sont prises en compte.

Au fur et à mesure que le joueur humain progresse, il peut étudier des positions de plus en plus complexes, mais la longueur des analyses requises fait que les problèmes dépassent rarement 5 ou 6 coups. L'ordinateur vient ensuite épauler le joueur humain dans son analyse, et pour des positions de fin de partie, on peut alors déterminer si, en supposant un jeu parfait, l'un des deux joueurs est en position de gagner, ou si la partie se terminera par un nul.

La même question peut se poser avec la position de départ. On sait qu'il est théoriquement possible de démontrer l'une de ces trois assertions :

- Blanc peut mater à partir de la position de départ.
- Noir peut mater à partir de la position de départ.
- Blanc et Noir peuvent obtenir le nul à partir de la position de départ.

1. Professeur en collège (LOL) à Mauriac (dans l'académie de Clermont-Ferrand), julien.lemoine@gmail.com.

2. Professeur assistant à JAIST (Japan Advanced Institute of Science and Technology), sviennot@jaist.ac.jp.

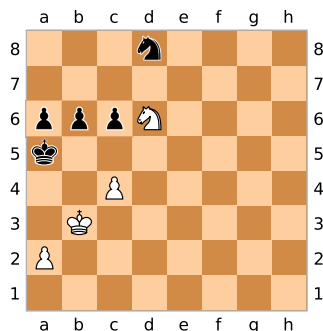


FIGURE 1. Les blancs jouent et matent en trois coups.

On aurait alors *résolu* le jeu d'échecs. Mais la quantité de calculs requise est telle qu'à l'heure actuelle, personne n'y est arrivé – pire, la plupart des spécialistes doute que ce soit un jour possible. Ainsi, dans l'article de 2002 *Games solved : now and in the future* [14], le jeu d'échecs est décrit comme « insoluble par quelque méthode que ce soit ». Interrogé sur le sujet suite à sa résolution du jeu de dames anglaises en 2007, Jonathan Schaeffer, plus prudent, déclare : « une avancée majeure, comme les ordinateurs quantiques, serait nécessaire pour ne serait-ce qu'essayer de résoudre les échecs » [11].

Prenant acte de ces funestes prédictions, nous allons, dans cet article, présenter un rapide tour d'horizon des principales techniques d'étude des jeux combinatoires, dont les échecs font partie. Ainsi, nous observerons qu'en effet, une analyse rapide du problème conduit à être sceptique concernant la possibilité de résoudre le jeu d'échecs ; mais ensuite qu'à l'inverse, une analyse plus poussée conduit à penser que les échecs seront résolus un jour, et même dans un avenir pas si lointain.

2. Les jeux combinatoires

Le jeu d'échecs appartient à un ensemble de jeux partageant des propriétés communes, appelés *jeux combinatoires*, dont voici les deux principales propriétés :

- Il n'y a pas d'intervention du hasard. On ne lance pas de dé comme au 421, on ne tire pas de cartes comme au Poker.
- Ce sont des jeux à *information complète*. Pour choisir son coup, chaque joueur dispose de toutes les informations concernant le jeu pour prendre sa décision. Ceci exclut par exemple le jeu de la bataille navale, où le plateau de l'adversaire est caché.

Ces deux propriétés permettent d'éliminer tout facteur chance, si bien qu'en présence de deux joueurs jouant parfaitement, le résultat du jeu est connu d'avance. Dans un jeu combinatoire, le perdant ne peut pas mettre en cause son manque de



FIGURE 2. Une position de Tic-tac-toe.

chance, il ne peut s'en prendre qu'à lui-même, sauf si, son opposant ayant joué parfaitement de bout en bout, il n'a jamais été en position de remporter la partie.

Parmi les jeux combinatoires les plus connus, outre les échecs, on peut citer les dames, le jeu de Go, ou le Connect-Four (« Puissance 4 »). Les techniques présentées dans cet article sont issues de l'étude informatique des jeux de ce type. Elles seront illustrées par le célèbre jeu de *Tic-tac-toe*³.

De par leur nature, les jeux combinatoires se prêtent particulièrement bien à la programmation. L'efficacité des programmes est corrélée à la complexité du jeu : pour un jeu simple, comme le Tic-tac-toe, on n'a aucune difficulté à déterminer informatiquement une stratégie optimale. Pour un jeu plus complexe comme le Connect-Four, il a fallu attendre 1988 pour qu'un programme calcule que le premier joueur dispose d'une stratégie gagnante [1].

En 2007, ce fut au tour des dames anglaises⁴ d'être résolues : si les deux joueurs jouent de manière optimale, la partie se termine par un nul [9]. Il s'agit du jeu combinatoire le plus complexe à avoir été résolu à l'heure actuelle.

Pour le jeu d'échecs, encore plus complexe, les ordinateurs battent désormais les meilleurs joueurs humains, mais sans disposer de la stratégie optimale. Enfin, pour le jeu de Go, considéré comme le jeu combinatoire classique le plus complexe, les ordinateurs sont encore d'un niveau inférieur à celui des meilleurs joueurs humains, mais ont atteint récemment le niveau de très bons joueurs [8].

3. Les échecs : état des lieux

Intéressons-nous plus particulièrement à la programmation du jeu d'échecs.

Les premiers développements théoriques sur la création d'un programme capable de jouer aux échecs datent de l'immédiat après-guerre. On peut par exemple citer la description détaillée, par Alan Turing, d'un programme d'un niveau de débutant, capable de jouer des coups pas trop ridicules, et dont il avait simulé le calcul à la main pour jouer contre une collègue [13].

Au début des années 1960 apparaît le premier programme à disposer d'un jeu crédible, *Kotok-McCarthy*, développé au MIT. En 1966 a lieu le premier match entre programmes, et *Kaissa*, développé à Moscou, bat *Kotok-McCarthy* lors d'un match de 9 mois dont les coups étaient échangés par télégraphie.

3. Jeu souvent désigné improprement comme *Morpion* : le Morpion consiste à aligner 5 symboles sur une grille illimitée.

4. *Checkers* ou *english draughts*, il s'agit d'une version des dames se déroulant sur un plateau 8 × 8.

Les progrès continuent, et les programmes commencent à se confronter à des joueurs humains. *Chess* de l'université Northwestern atteint un niveau Elo de 2000 en 1977, *Cray Blitz* atteint 2200 en 1981, et *HiTech* atteint 2500 en 1985. *HiTech* est également le premier programme à battre un grand maître en 1988.

Dans les années 1990, les programmes commencent à surpasser les meilleurs joueurs mondiaux, d'abord en Blitz, puis dans des conditions de temps normales, avec la victoire très médiatisée de *Deep Blue* contre Garry Kasparov en 1997. Si, pendant encore quelques années, les meilleurs joueurs humains ont pu rivaliser avec les ordinateurs, depuis la défaite de Kramnik contre Deep Fritz en 2006, il est acquis que les ordinateurs ont dépassé les humains.

Tous ces programmes ont un point commun : ils ont été développés pour fonctionner en temps limité, et recherchent un bon coup, plutôt que le meilleur coup théorique dont l'obtention est en général trop coûteuse en temps. Dans la suite, au contraire, nous discutons de la possibilité de développer un programme capable de jouer, à chaque coup d'une partie, le meilleur coup.

4. Arbre de jeu

Dans un article de 1950 resté célèbre [10], Claude Shannon donne une estimation du nombre de parties différentes possibles au jeu d'échecs : il obtient le nombre affolant de 10^{120} , et encore, son estimation est très optimiste, le nombre réel étant probablement bien plus important. Face à un nombre de cette ampleur, très supérieur au nombre estimé d'atomes dans l'univers (de l'ordre de 10^{80}), il apparaît bien présomptueux d'espérer résoudre le jeu d'échecs. Et de fait, le nombre de Shannon est régulièrement utilisé comme argument massue pour clouer le bec à tout impudent qui oserait évoquer cette possibilité ; or, ce nombre ne représente pas une bonne mesure de la difficulté du jeu d'échecs.

Pour comprendre à quoi correspond exactement ce nombre, il faut introduire la notion d'*arbre de jeu*, classiquement utilisée dans l'étude des jeux combinatoires. Un arbre de jeu est un *graphe*, c'est-à-dire un objet mathématique composé de *sommets* reliés par des *arêtes*. Dans le cas de l'arbre de jeu, les sommets sont appelés *nœuds*, et représentent les positions. Deux nœuds sont reliés par une arête si on obtient une position à partir de l'autre en un coup (on parle de *position fille*, ou de *nœud fils*). Les arbres sont représentés à l'envers de leur homologue végétal : la *racine*, la position de départ, est placée en haut. Puis, sur la ligne juste en-dessous, on place les positions filles de la racine, et les petites-filles encore en-dessous.

Prenons l'exemple du Tic-tac-toe. La racine est la position de départ, la grille vide. Le premier joueur peut alors jouer 9 coups différents, puis il reste 8 coups possibles au deuxième joueur, puis à nouveau 7 coups pour le premier joueur... La figure 3 présente le début de cet arbre, s'arrêtant après le premier coup pour des raisons évidentes de place.

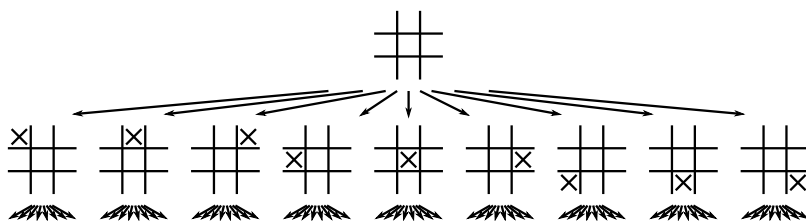


FIGURE 3. Arbre de jeu du Tic-tac-toe.

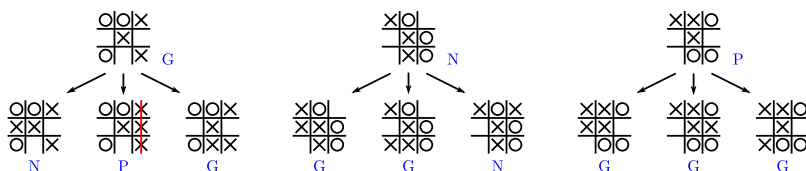


FIGURE 4. Un nœud gagnant, un nul, et un perdant.

Car si on développait cet arbre en entier, il y aurait *a priori* $9 \times 8 \times 7 \times \dots \times 1 = 9! = 362880$ feuilles : les feuilles de l'arbre représentent les positions terminales, lorsque plus aucun coup n'est possible. En réalité, il y en aurait moins, car le jeu s'arrête dès lors qu'un alignement de trois symboles identiques est réalisé. En tenant compte de cette correction, on dénombre 255 168 feuilles [3].

Le *résultat* d'une position est le résultat, pour le joueur dont c'est le tour, d'une partie jouée à partir de cette position, en supposant qu'aucun joueur ne commette d'erreur. Dans le Tic-tac-toe comme dans les échecs, il peut être *gagnant*, *nul* ou *perdant*. Il est facile de déterminer le résultat d'une feuille : c'est *nul* s'il n'y a aucun alignement après 9 coups, et *perdant* si un alignement vient de se produire (car s'il vient d'être tracé par un joueur, c'est à l'autre joueur de jouer... et il a perdu). Les trois règles suivantes permettent ensuite de calculer le résultat d'un nœud à partir des résultats de ses fils :

- Si un nœud a au moins un fils perdant, alors il est gagnant.
- Si un nœud n'a aucun fils perdant, mais au moins un fils nul, alors il est nul.
- Si un nœud n'a que des fils gagnants, alors il est perdant.

La figure 4 illustre chacune des trois possibilités :

- Le premier nœud étudié est gagnant, car il a un fils perdant, celui du milieu, obtenu en faisant une ligne (l'adversaire a alors perdu). Évidemment, le joueur à la croix va préférer ce coup aux deux autres, car c'est le seul qui assure sa victoire.

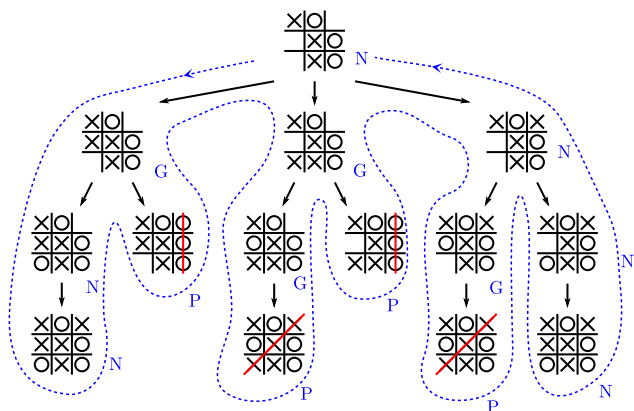


FIGURE 5. Déroulement de l'algorithme Minimax sur l'arbre de jeu d'une position de Tic-tac-toe.

- Le deuxième nœud étudié a deux fils gagnants (pour l'adversaire), et un nul. Le joueur à la croix joue donc le seul coup qui lui permet le nul, faute de mieux. Le nœud est donc nul.
- Le troisième nœud n'a que des fils gagnants : quel que soit le coup joué, le joueur à la croix va permettre à l'autre de faire une ligne (on reconnaît une situation de fourchette), il a donc perdu.

L'algorithme standard chargé d'utiliser ces règles pour calculer les résultats des nœuds autres que les feuilles, en remontant l'arbre, est appelé Minimax⁵. Concrètement, lorsqu'il est appelé sur un nœud, l'algorithme Minimax calcule le résultat de chacun de ses fils, puis utilise ces résultats pour calculer le résultat du nœud. La figure 5 permet de visualiser le déroulement du calcul sur une position de Tic-tac-toe.

L'algorithme explore les nœuds de l'arbre de jeu dans l'ordre indiqué par la ligne en pointillés, et le résultat de chaque nœud est indiqué à côté de la ligne dès lors que l'algorithme l'a calculé. Une fois l'arbre exploré en entier, on obtient le résultat de la racine. Ainsi, la connaissance de l'arbre de jeu dans sa totalité permet de déduire le résultat que peuvent escompter chacun des joueurs en début de partie.

Remarquons un avantage de cet algorithme : il n'utilise que très peu de mémoire. Par exemple, sur la figure 5, lorsqu'il est en train de calculer la branche du milieu, il n'a pas besoin de se rappeler tout le calcul de la branche de gauche ; il lui suffit de se rappeler que le premier fils est gagnant. En général, à un moment donné du

5. Les théoriciens remarqueront à la lecture des trois règles, d'une part qu'il s'agit d'une version épurée de l'algorithme Minimax, adaptée à la résolution de jeux plutôt qu'à la recherche d'un bon coup en temps limité ; d'autre part, qu'il s'agit plutôt de l'algorithme Negamax, version de l'algorithme adaptée aux jeux impartiaux.

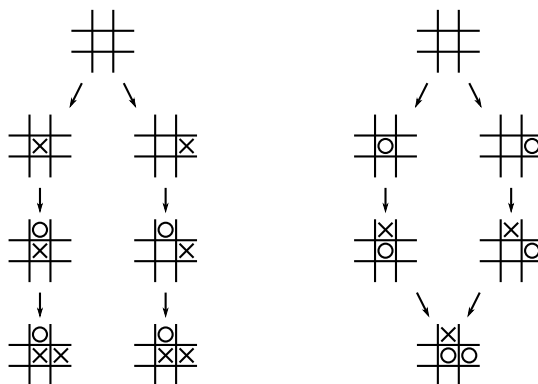


FIGURE 6. Une transposition, puis l'arbre de jeu représenté sous forme compacte..

calcul, il n'a besoin de se rappeler que d'une fratrie par étage. Même pour un jeu très complexe comme le jeu d'échecs, les parties ne peuvent dépasser quelques centaines de coups, et le stockage de quelques centaines ou même milliers de positions ne pose aucun problème. Le problème, car problème il y a, réside dans le temps de calcul.

En théorie des jeux combinatoires, la *game-tree complexity* est une mesure de la complexité d'un jeu : elle correspond au nombre de feuilles de l'arbre de jeu. Chaque feuille correspondant à une partie différente, il s'agit de manière équivalente du nombre de parties différentes de ce jeu. La game-tree complexity du jeu d'échecs est d'au moins 10^{120} , ce qui justifie l'impossibilité de résoudre ce jeu par le simple algorithme Minimax : même en disposant d'un ordinateur évaluant 3000 milliards de feuilles à la seconde, une année ne permettrait d'évaluer qu'à peine 10^{20} feuilles... on est très loin du compte !

Mais on ne peut pas pour autant en déduire l'impossibilité de résoudre le jeu d'échecs : la game-tree complexity du Tic-tac-toe a beau être de 255 168, nous allons pouvoir le résoudre par des moyens humains avant la fin de cet article, ce qui apparaît *a priori* comme une performance, vu la piètre puissance de calcul de notre cerveau. En effet, la game-tree complexity ne tient pas compte d'un certain nombre de techniques de simplification du calcul que nous allons maintenant décrire.

5. Transpositions

Il arrive que deux parties aux déroulements différents mènent à la même position. C'est le cas par exemple à gauche de la figure 6, où l'on a représenté une portion de l'arbre de jeu du Tic-tac-toe. À partir de la position initiale, deux parties différentes mènent en trois coups à la même position T : on dit qu'il s'agit d'une *transposition*.

Dans le cas du jeu d'échecs, on rencontre de nombreuses transpositions. Dans le cas le plus simple, elles se produisent après seulement deux coups joués par l'un des joueurs. Par exemple, les débuts de partie correspondant aux séquences de coups :

1.Cf3 e5 2.Cc3

et

1.Cc3 e5 2.Cf3

conduisent à la même position ; mais il est possible d'obtenir des transpositions après de plus nombreux coups⁶.

Du fait des transpositions, il est possible de représenter les arbres de jeu de manière plus compacte, en ne conservant qu'un seul nœud pour chaque transposition. Pour l'exemple de la figure 6, on obtient la portion d'arbre⁷ de jeu représentée à droite sur la figure.

Il est possible d'exploiter ce phénomène pour accélérer le calcul du résultat de la position initiale : une fois le résultat de la position T calculé, on le stocke. Lorsque plus tard dans le calcul, on rencontre à nouveau la position T , on peut utiliser le résultat stocké plutôt que de le calculer à nouveau. Ce faisant, on gagne du temps, mais on perd de la mémoire : là où l'algorithme de la section 4 ne nécessitait à chaque instant que le stockage de quelques positions, il est désormais nécessaire de stocker le résultat de toutes les positions rencontrées, dans ce que l'on nomme une *table de transpositions*.

On vient d'observer une des multiples manifestations d'une règle bien connue des programmeurs : le temps de calcul peut se transformer en mémoire, et vice-versa. Toute la difficulté consiste à effectuer ces transformations de façon à ce que le calcul effectué devienne possible par la machine dont on dispose.

Refermons cette parenthèse, et déterminons la taille maximale prise par la table de transpositions. À la fin du calcul, elle contient toutes les positions possibles du jeu étudié. Ce nombre de positions possibles se nomme *state-space complexity*.

La state-space complexity est une autre mesure de la complexité d'un jeu. Elle est généralement nettement moins grande que la game-tree complexity, et plus pertinente, en tenant compte de la possibilité d'utiliser une table de transpositions pour résoudre le jeu. Pour le Tic-tac-toe, la réduction est notable, car on passe de 255 168 parties à 5 478 positions – encore trop pour résoudre le jeu manuellement. Et de

6. Pour vérifier que deux positions sont bien équivalentes, il faut vérifier qu'elles le sont du point de vue des règles du roque, de la prise en passant, ou pire, de la règle des 50 coups sans prise, et de celle de la répétition trois fois d'une même position. Cette dernière règle en particulier exclut la quasi-totalité des transpositions, dont celle de l'exemple. Il est cependant raisonnable de ne pas trop se soucier de cette règle, comme expliqué dans [4] au sujet du *graph-history interaction problem*.

7. Il ne s'agit plus d'un *arbre* au sens de la théorie des graphes, mais nous conservons tout de même le terme d'*arbre de jeu*.

même, dans le cas des échecs, la réduction n'est pas suffisante pour laisser espérer résoudre le jeu un jour.

En effet, dans le même article de 1950 [10], Shannon estime la state-space complexity du jeu d'échecs à 10^{43} . Ce n'est qu'une estimation, basée sur un calcul simple, or le nombre exact est très complexe à déterminer. Car Shannon à la fois sous-estime (il oublie captures de pièces, promotions de pions) et sur-estime ce nombre (en conservant des positions illégales, comme deux fous blancs pour un même joueur sans promotion). Des évaluations récentes plus précises tendent à montrer qu'il avait tout de même visé assez juste, probablement en surestimant le résultat plutôt que le contraire (il est certain que le nombre correct est en-dessous de 10^{46} [12]).

Quoi qu'il en soit, 10^{43} est une valeur encore bien trop grande. L'espace de stockage de tous les ordinateurs existants à l'heure actuelle est de l'ordre de 10^{20} octets, aussi nous ne sommes pas près de voir apparaître un disque dur assez grand pour stocker 10^{43} positions. Encore une fois, nous semblons être dans une impasse.

Mais même si la state-space complexity est une notion plus pertinente que la game-tree complexity pour évaluer la complexité d'un jeu (ce dont les spécialistes sont conscients, c'est souvent 10^{43} qui est cité plutôt que 10^{120} dans les articles abordant la résolution du jeu d'échecs), elle a aussi ses faiblesses.

Car il est facile d'imaginer un jeu qui mette cette mesure en échec : sur un échiquier vide, Blanc et Noir placent, chacun leur tour, un de leurs 8 pions sur un emplacement libre. C'est le seul coup possible : il n'y a pas de prise, pas de déplacement. Blanc commence, et le premier à avoir placé tous ses pions a gagné. En dépit de sa state-space complexity monstrueuse⁸, il est aisé de voir que Blanc gagnera quoi qu'il arrive ce jeu idiot, plaçant son huitième pion juste après que Noir aura placé son septième...

L'algorithme Minimax ou la table de transpositions ne sont d'aucun secours ici. La résolution que l'on effectue naturellement dans ce cas est une résolution *par méthode*.

6. Résolution par méthode

La résolution par méthode consiste à trouver des simplifications diminuant le nombre de nœuds de l'arbre de jeu.

Dans le cas du jeu idiot, on sait que la partie se terminera en 15 coups quoi qu'il arrive. On peut fusionner tous les nœuds qui sont à la même distance de la racine, et l'arbre de jeu est réduit à une unique branche verticale de 16 nœuds. Cet exemple est exagérément simple ; en général, les jeux combinatoires ne se résolvent pas si facilement. Si certains se résolvent complètement par méthode (comme le jeu de

8. C'est un exercice plaisant de dénombrement que de montrer qu'elle vaut 1 216 708 506 251 460 977.



FIGURE 7. Position ne pouvant plus donner de ligne.

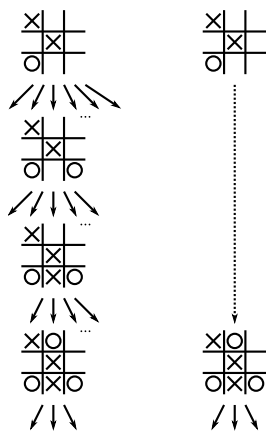


FIGURE 8. Coups forcés au Tic-tac-toe.

Nim), la plupart sont trop complexes pour cela, et les méthodes ne permettent que de simplifier l'arbre de jeu avant de l'explorer.

Ce sont souvent des considérations géométriques, comme la symétrie, qui permettent de fusionner des nœuds de l'arbre de jeu. Dans le cas du Tic-tac-toe, si l'on tient compte de la symétrie, on peut fusionner jusqu'à 8 nœuds à la fois, et la state-space complexity passe de 5478 à 765 positions. On peut aller plus loin⁹, et nous utiliserons deux méthodes supplémentaires dans la suite de l'article :

- Une position qui ne peut plus aboutir à la création d'une ligne, quels que soient les coups joués, peut être considérée directement comme nulle, comme sur la figure 7.
- Si l'adversaire menace de faire une ligne au prochain coup, il faut parer la menace, avec un *coup forcé*. Parfois, les menaces s'enchaînent, et ce sont plusieurs coups forcés qui sont joués à la suite. On peut alors supprimer les nœuds intermédiaires, comme illustré sur la figure 8.

9. En fait, le Tic-tac-toe pourrait se résoudre uniquement par méthode [6]. Nous ne le ferons pas dans le cadre de cet article car le jeu n'illustrerait plus les concepts développés dans les sections 7 et 8.

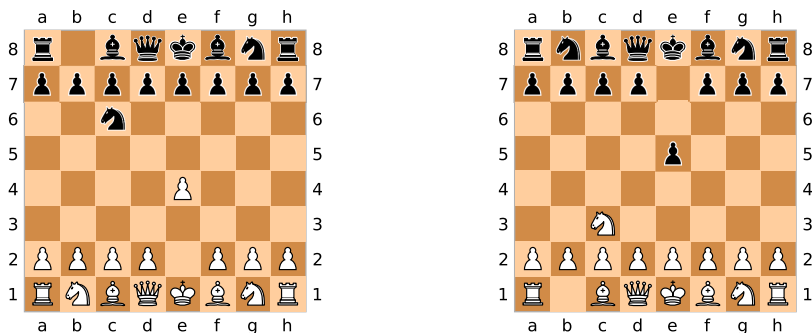


FIGURE 9. Une transposition pas si évidente.

La première méthode permet de déterminer directement le résultat de certains nœuds, ce qui aboutit à supprimer tous les nœuds situés en-dessous. La deuxième permet de supprimer des nœuds internes à l'arbre, car leur étude est inutile.

Des méthodes similaires peuvent être utilisées aux échecs. En début de partie, une seule symétrie est disponible : la symétrie d'axe horizontal, avec inversion des couleurs des joueurs. Donnons un exemple. Les deux positions obtenues après les séquences de coups :

1.e3 Cc6 2.e4

et

1.Cc3 e5

sont similaires (voir figure 9), si par exemple c'est Blanc qui dispose d'une stratégie gagnante à partir de la première position, alors c'est Noir qui dispose de cette même stratégie à partir de la deuxième position. Il s'agit en fait d'une transposition¹⁰.

Lorsque les possibilités de roque disparaissent, on peut utiliser la symétrie d'axe vertical ; puis lorsque les pions disparaissent, les symétries d'axes diagonaux et celle d'axe horizontal sans changement de couleur se rajoutent. Au maximum, ce sont donc 16 nœuds que l'on peut fusionner grâce à des considérations de symétrie, comme dans la figure 10.

Les deux autres méthodes vues à propos du Tic-tac-toe ont leur pendant aux échecs. Les coups forcés existent, et donnent, entre autres, quelques problèmes d'autant plus faciles à résoudre qu'il n'y a qu'une variante à explorer. Par exemple, sur la figure 11, les blancs jouent et matent en trois coups, et les deux réponses du joueur

10. En utilisant ces transpositions, on perd une caractéristique des arbres de jeu des échecs : l'alternance entre étages, Blanc ayant le trait pour les positions des étages pairs, et Noir ayant le trait pour les positions des étages impairs.

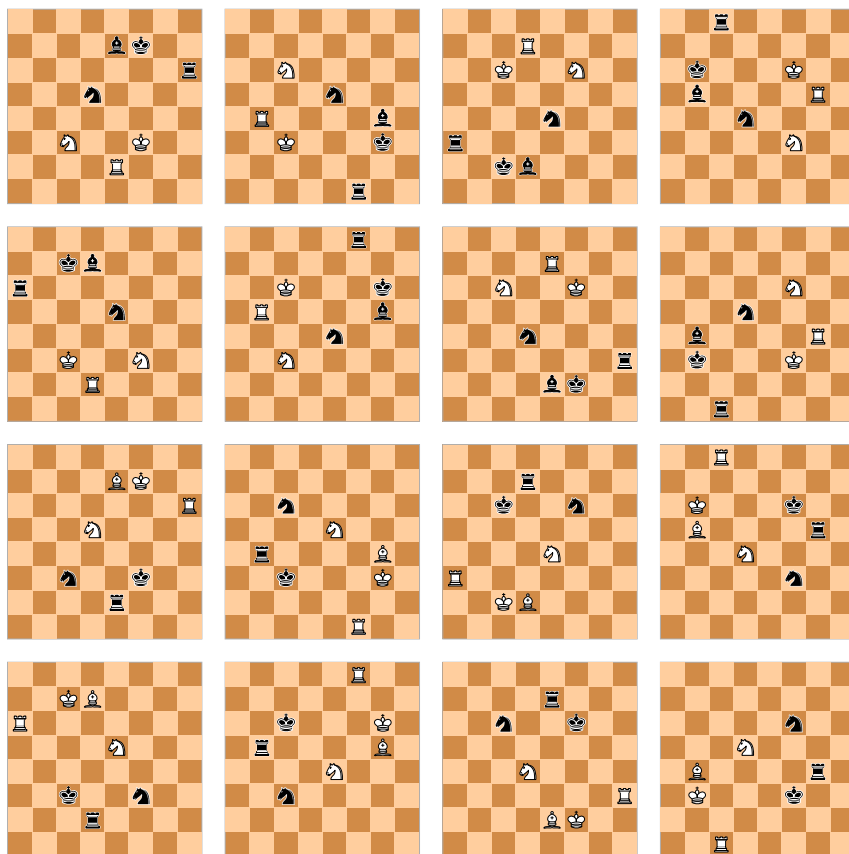


FIGURE 10. 16 positions équivalentes. Les 8 du haut sont avec trait aux blancs, les 8 du bas avec trait aux noirs.

noir sont des coups forcés, car toute autre réponse est sanctionnée d'un mat immédiat ¹¹.

Quant à la résolution directe de nœuds, c'est ce que font déjà les joueurs d'échecs lorsqu'ils apprennent le résultat des finales. Ils savent ainsi que toute position du type *Roi-Tour contre Roi* est gagnante pour le joueur qui a conservé sa tour, hormis les

11. Solution : 1. Dxf6 gxf6 (coup forcé, sinon mat au coup suivant avec la dame en g7)
2. Tg1+ Rh8 (coup forcé, car seul coup légal) 3. Fxf6# : noir

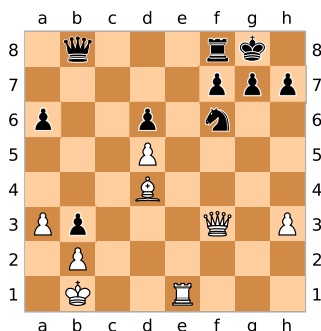


FIGURE 11. Les blancs jouent et matent en trois coups.

cas de pat¹². Du point de vue du programmeur, il est bien plus efficace de programmer une fonction traduisant le résultat de cette finale, que de stocker le résultat des dizaines de milliers de positions du type *Roi-Tour contre Roi*.

En poussant ce raisonnement, on peut chercher à déterminer le résultat de positions plus complexes que celles des finales classiques. C'est ce que font les *tables de finales*, cependant, il devient difficile de trouver des règles simples décrivant quelles sont les positions gagnantes ou perdantes lorsque la complexité des positions étudiées augmente. On se base alors sur des techniques de compression de données plus classiques, et à l'heure actuelle, les tables de finales de l'ensemble des positions à six pièces ou moins sont calculées¹³.

Toutes ces méthodes ne sont pas à négliger dans l'optique de la résolution des jeux d'échecs, mais leur effet ne sera probablement pas décisif. Le travail sur les finales ne concerne que les fins de partie ; or l'écrasante majorité des positions à étudier sont des positions de début ou de milieu de partie, car lorsqu'il y a plus de pièces, il y a plus de façons de les placer. L'effet des coups forcés est anecdotique : en partie réelle, les coups forcés sont assez rares, surtout en début de partie, et de plus ceux que l'on peut imposer à l'adversaire ne se soldent que rarement par un gain.

La méthode la plus efficace est probablement la symétrie. La quasi-totalité des positions légales ne permet pas de roquer, mais dispose aussi d'au moins un pion. La symétrie permet donc de réduire le nombre de positions à étudier d'un facteur 4 environ. Ce n'est pas négligeable, mais par rapport aux 10^{43} positions légales, cela ne

12. Roi A dans un coin (e.g. a1), tour B juste en diagonale (b2) et roi B protégeant la tour B, ou roi A dans un coin (e.g. a1), roi B en c1 ou a3, et tour B sur la ligne passant entre les deux rois, sans mettre le roi A en échec.

13. Pour un espace mémoire d'un peu plus de 1 To. Les tables de l'ensemble des positions à 7 pièces devraient bientôt être calculées, pour un espace mémoire de l'ordre de 100 To. Inutile donc d'espérer résoudre le jeu d'échecs par ce moyen, qui sature trop vite la mémoire.

fait que 1,4 % du chemin¹⁴. Il va donc falloir compter sur autre chose pour résoudre le jeu d'échecs.

7. Élagage alpha-bêta

La raison principale pour laquelle le jeu d'échecs pourrait être résolu un jour réside dans l'existence de l'*élagage alpha-bêta*. C'est un algorithme simple, qui apparaît naturellement dans l'étude des jeux combinatoires, et dont les premières formulations remontent aux années 1950. Il repose sur l'asymétrie entre les nœuds perdants et gagnants¹⁵ :

- Un nœud est perdant si tous ses fils sont gagnants.
- Un nœud est gagnant si au moins un de ses fils est perdant.

Ainsi, pour prouver qu'un nœud est perdant, il faudra de toute façon prouver que tous ses fils sont gagnants. Mais pour prouver qu'un nœud est gagnant, il suffit de prouver qu'un de ses fils est perdant. Une fois cette tâche effectuée, le calcul des autres fils devient inutile.

Reprenons le mat en trois coups de la figure 1. Pour prouver que cette position est gagnante pour le joueur blanc, il suffit de trouver un coup qui place le joueur noir dans une position perdante : le coup c5 fait l'affaire. Mais pour prouver que cette nouvelle position est bien perdante, il faut montrer que tous les coups jouables maintenant par Noir placent Blanc dans une position gagnante. On étudie donc les 5 coups Cb7, Ce6, Cf7, bxc5 et b5.

Étage après étage, on alterne entre positions gagnantes et perdantes, choix d'un coup et étude de tous les coups. La figure 12 résume la solution du problème, et l'arbre obtenu est un *arbre solution* : c'est une partie de l'arbre de jeu suffisante pour calculer le résultat de la position étudiée.

Sur cet arbre, il apparaît que lorsque Blanc a le trait, l'étude d'un coup suffit, et lorsque Noir a le trait, on étudie tous les coups. On peut remarquer que sur les quatre branches de gauche, le deuxième coup de Noir est forcé (représenté en pointillés), car il n'a alors qu'un seul coup légal.

Maintenant, il est possible de décrire le comportement de l'algorithme alpha-bêta. Il fonctionne de manière similaire à l'algorithme Minimax, hormis sur un point : lorsque l'on calcule les fils d'un nœud, dès qu'on a prouvé qu'un de ces fils est perdant, on en déduit immédiatement que le nœud père est gagnant, et on n'effectue pas le calcul des frères du fils perdant. Ce faisant, on tire parti du fait qu'il n'est pas nécessaire d'explorer tout l'arbre de jeu pour obtenir le résultat de la racine.

14. Car $\frac{\log(4)}{\log(10^{43})} \simeq 0,014$: les techniques employées pour réduire le nombre de nœuds étudiés tendent à les diviser par un certain facteur, ce qui justifie le recours à une échelle logarithmique.

15. Dans cette section, nous nous contenterons d'aborder le cas des nœuds gagnants et perdants pour simplifier le propos. Le cas du nul sera traité dans la section 8.

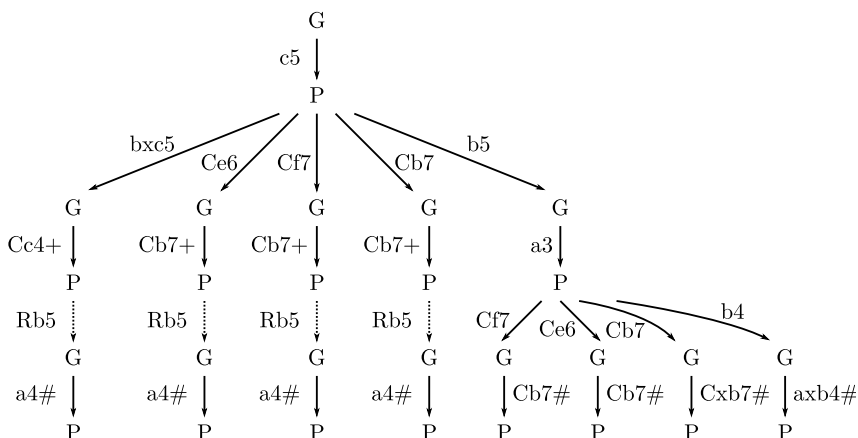


FIGURE 12. Arbre solution du problème de la figure 1.

La figure 13 présente un exemple d'utilisation de l'algorithme alpha-bêta appliqué à une position de Tic-tac-toe. Les flèches barrées représentent tous les nœuds qu'il n'est pas utile d'explorer, dès lors qu'un fils perdant a été trouvé dans leur fratrie. On voit qu'une réduction importante (en mémoire comme en temps de calcul) est opérée par rapport à l'algorithme Minimax : dans le cas de cette position, seules 18 positions sont étudiées, là où l'algorithme Minimax en aurait nécessité 66.

18 n'est cependant pas le nombre optimal. En effet, la racine de cet arbre est gagnante, or l'algorithme a d'abord calculé un premier fils, gagnant, puis un deuxième, perdant : le calcul du premier fils était donc inutile pour démontrer que la racine était gagnante, et ainsi toute la moitié gauche de l'arbre est inutile. En la supprimant, on constate qu'un arbre solution pour cette position contiendrait seulement 10 nœuds.

Ce cas n'est pas une anomalie. En général, l'algorithme alpha-bêta ne tombe pas miraculeusement sur l'arbre solution idéal ; si on explore l'arbre de jeu dans n'importe quel ordre, l'algorithme risque d'effectuer un certain nombre de calculs inutiles, si bien que la façon dont l'algorithme alpha-bêta est programmé influe fortement sur sa performance. Voici donc deux grands principes concernant l'ordre dans lequel on étudie les nœuds, et qui permettent de réduire le nombre de nœuds rencontrés :

- Il faut identifier les bons coups pour éviter les calculs inutiles : si un nœud est gagnant, il faut éviter de calculer ses fils gagnants, ce qui est du temps perdu, et ne calculer qu'un fils perdant. On ordonne donc les nœuds fils de façon à étudier en premier les nœuds qu'on suppose perdants (pour l'adversaire).

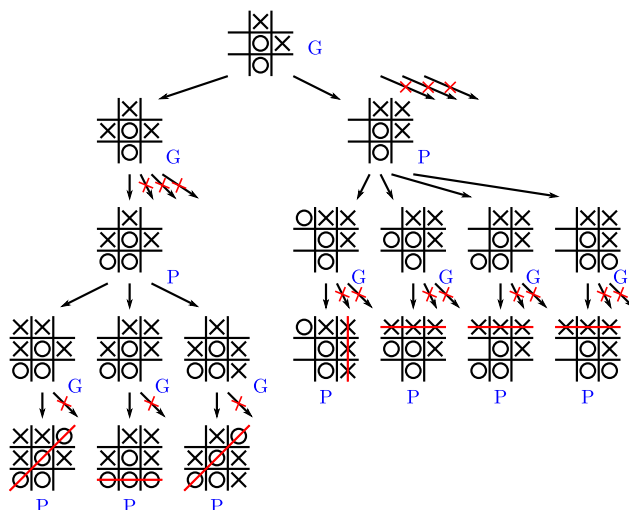


FIGURE 13. Algorithme alpha-bêta appliqué à une position de Tic-tac-toe.

— Il faut préférer un très bon coup à un bon coup : si un nœud gagnant a plusieurs fils perdants, il suffit d'en calculer un. Autant choisir celui dont l'étude sera la plus simple. On ordonne donc les nœuds fils de façon à étudier en premier les nœuds dont l'étude sera la plus rapide.

Si le premier principe est parfaitement appliqué, alors la table de transpositions ne contient qu'un arbre solution. Dans le cas contraire, en plus de cet arbre solution, on trouve certaines branches inutiles comme la branche de gauche de la figure 13.

Pour l'implémentation de ce principe, il serait utile de disposer d'un oracle qui indique quelles sont les positions gagnantes, et quelles autres sont perdantes. Ainsi, il devient utile d'être un bon joueur, ou tout du moins de connaître les stratégies des bons joueurs pour programmer l'alpha-bêta, car il faut que le programme étudie en priorité les bons coups ¹⁶.

Le deuxième principe commande de rechercher un arbre solution contenant le moins de nœuds possible. Car une même position possède, en général, de nombreux arbres solution très différents – de la même façon qu'à partir d'une position gagnante, des joueurs de niveaux différents peuvent s'assurer du mat de manière plus

16. Un programmeur qui serait un mauvais joueur d'échecs ne serait pas pour autant impuissant face à ce problème : des algorithmes de parcours de l'arbre de jeu, comme le PN-search [2], utilisent les informations obtenues durant le développement de l'arbre de jeu pour prédire le résultat des positions – et ce, même en l'absence de fonction d'évaluation cherchant à traduire les stratégies humaines.

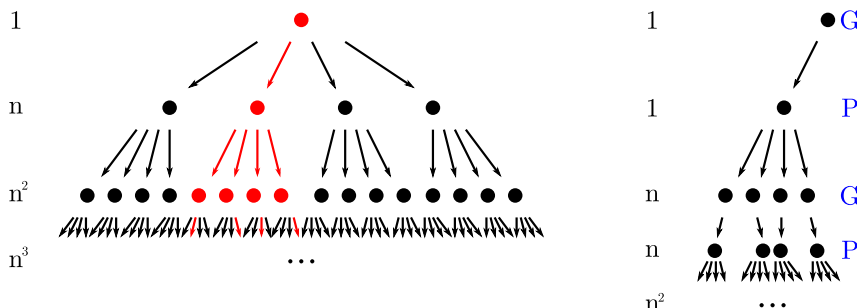


FIGURE 14. Arbre de jeu équilibré, et arbre solution extrait de cet arbre de jeu.

ou moins rapide. Plus la stratégie utilisée pour mater est rapide, plus l'arbre solution correspondant est simple.

Pour l'implémentation de ce principe, il faut tirer parti du fait que l'arbre de jeu des échecs est déséquilibré. Ainsi, quand on a le choix entre plusieurs bons coups, on doit sélectionner celui dont l'étude sera la plus courte. C'est ce que font instinctivement les joueurs qui, disposant d'un avantage matériel sur leur opposant, cherchent à forcer l'échange de pièces pour accélérer la fin de partie.

Évaluons maintenant l'intérêt de l'alpha-bêta dans la résolution du jeu d'échecs. Si on étudie un arbre parfaitement équilibré (chaque nœud interne comportant n fils), sans transposition, et de profondeur p (dont les parties se terminent en p coups), alors le nombre de nœuds de cet arbre est :

$$(1) \quad 1 + n + n^2 + \dots + n^p = \frac{n^{p+1} - 1}{n - 1} \sim n^p$$

Si on suppose de plus que p est impair, et que la racine de l'arbre est gagnante, alors le nombre de nœuds d'un arbre solution est :

$$(2) \quad 1 + 1 + n + n + n^2 + n^2 + \dots + n^{\frac{p-1}{2}} = 2 \cdot \frac{n^{\frac{p+1}{2}} - 1}{n - 1} \sim 2 \cdot n^{\frac{p-1}{2}} \simeq \sqrt{n^p}$$

La figure 14 illustre ce calcul avec les premiers étages d'un arbre de jeu, et d'un arbre solution correspondant, pour $n = 4$.

Le nombre de nœuds d'un arbre solution est donc, à peu de choses près, de l'ordre de la racine carrée du nombre de nœuds de l'arbre de jeu. Les autres cas (avec p pair ou une racine perdante) donnent la même estimation. Le gain est énorme ! Pour le jeu d'échecs, comportant de l'ordre de 10^{40} nœuds, cela représenterait seulement 10^{20} nœuds pour un arbre solution – et donc pour la table de transpositions. Ce n'est

déjà plus la même difficulté... L'algorithme alpha-bêta est ainsi la raison principale pour laquelle le recours au nombre de Shannon n'est pas un argument correct pour affirmer que le jeu d'échecs ne peut être résolu.

10²⁰ n'est cependant toujours pas une estimation fiable de la taille d'une table de transpositions à même de résoudre le jeu d'échecs. Voici plusieurs phénomènes qui peuvent influencer sur la taille de cette table :

- L'algorithme alpha-bêta réalise des calculs inutiles, ce qui conduit à réviser cette estimation à la hausse (le premier principe ne pouvant généralement pas être parfaitement respecté).
- L'effet des transpositions sur cette estimation est ambigu. Par rapport à un arbre sans transpositions, les transpositions réduisent à la fois le nombre nœuds de l'arbre de jeu, et celui de l'arbre solution. Comme on s'intéresse au rapport entre ces deux nombres, les deux effets sont antagonistes. On verra ce qu'il en est au final dans la section 9.
- En revanche, le calcul précédent repose sur un arbre parfaitement équilibré, alors que l'arbre de jeu des échecs est fortement déséquilibré. Concentrer l'étude sur les parties de l'arbre les plus simples (deuxième principe) permet ainsi de réduire fortement le nombre de nœuds étudiés.

On peut prendre conscience de l'intérêt de cette dernière remarque en revenant au problème de la figure 1 : l'arbre solution de la figure 12 ne contient que 28 nœuds, alors que le recours à l'algorithme Minimax nécessiterait de développer tout l'arbre de jeu, arbre qui contient des milliers de positions. Ce sont toutes les positions que l'on peut obtenir à partir de la position de départ de ce problème, avec tous les déplacements des rois, des cavaliers, les promotions de pions, y compris les plus improbables, en fou ou en tour... Ce phénomène est lié au fait que les échecs est un jeu avec une propriété de *mort subite* : le mat. Ainsi, certaines parties se terminent en quelques coups¹⁷, quand d'autres en nécessitent des centaines.

8. Prise en compte du Nul

L'algorithme alpha-bêta expliqué à la section 7 convient à un jeu comportant des positions gagnantes et perdantes, mais ne prend pas en compte les positions nulles. Or, il est facile de l'adapter pour un jeu qui, comme le Tic-tac-toe ou les échecs, contient de telles positions.

Il suffit pour cela de faire deux calculs séparés¹⁸. Dans un premier calcul, on considère que le premier joueur veut gagner ou faire match nul, et que le deuxième

17. La partie d'échecs la plus courte ne nécessite que deux coups de chaque joueur (et une grande naïveté du joueur blanc) :

1.f3 e6 2.g4 Dh4#.

18. C'est tout du moins la méthode présentée dans cet article. D'autres méthodes existent, par exemple l'algorithme alpha-bêta peut être étendu à trois valeurs.

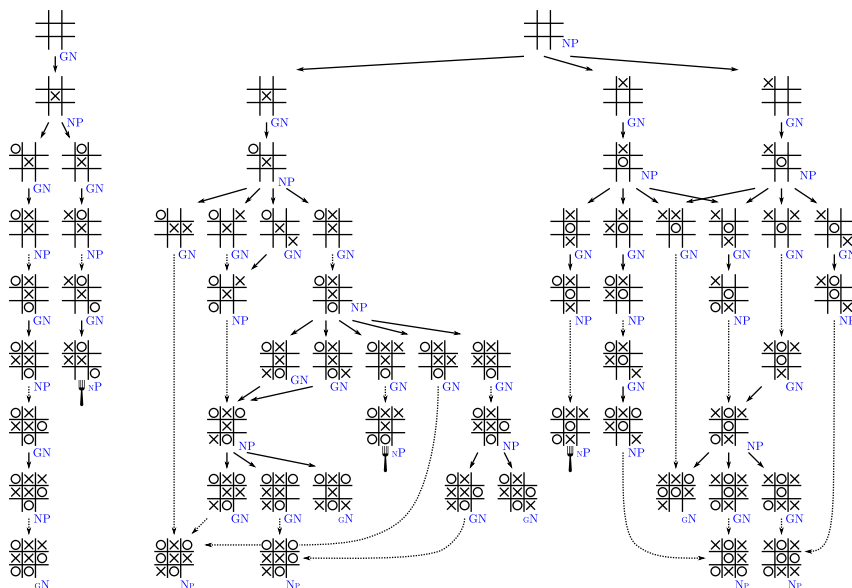


FIGURE 15. Résolution du Tic-tac-toe.

veut seulement gagner. Ainsi, un seul des deux joueurs sera satisfait à la fois, ce qui correspond à un calcul classique du type gagnant/perdant.

Si le résultat du calcul est « perdant », ce qui signifie que le deuxième joueur gagne, alors le travail est terminé. Sinon, on effectue un deuxième calcul, dans lequel on considère que le premier joueur ne veut que gagner, tandis que le deuxième veut gagner ou faire match nul. Une fois le résultat de ce calcul connu, on peut en déduire le résultat de la position étudiée.

En guise d'exemple, nous allons résoudre une bonne fois pour toutes le jeu du Tic-tac-toe, dans la figure 15. Il y a deux calculs séparés : le premier, l'arbre de gauche, montre que la position de départ est gagnante ou nulle (notée « GN »). Le second, l'arbre de droite, montre que la position de départ est nulle ou perdante (notée « NP »). On en déduit que la position de départ est un nul : joué parfaitement, le Tic-tac-toe se finit par un match nul.

On retrouve la dichotomie à l'œuvre dans les calculs du type Gagnant/Perdant : pour montrer qu'un nœud est GN, il suffit d'un fils NP, tandis que pour montrer qu'un nœud est NP, il faut montrer que tous les fils sont GN.

Explicitons les notations. Les pointillés traduisent les coups forcés. Les trois nœuds marqués par une fourchette indiquent des positions où interviennent des... fourchettes, elles sont donc perdantes (et notées « NP » : une position perdante est a

fortiori « nulle ou perdante »). Les autres positions terminales sont nulles, et donc notées « gN » ou « Np » suivant leur utilisation.

Les deux arbres ont respectivement 13 et 48 nœuds : la différence de taille traduit qu'il est plus difficile d'obtenir le nul pour le deuxième joueur, ce qui se conçoit car il a un temps de retard. Enfin, remarquons que 61 nœuds suffisent à résoudre le Tic-tac-toe. La comparaison avec la game-tree complexity de 255 168, ou la state-space complexity de 5 478 montre bien que ces deux mesures n'évaluent pas bien la difficulté d'un jeu.

9. Estimation des ressources requises pour la résolution

L'estimation donnée, à la fin de la section 7, de la taille d'une table de transpositions assez grande pour résoudre le jeu d'échecs, est assez grossière ; nous allons maintenant l'affiner, en évaluant l'effet de trois phénomènes différents sur cette taille.

Calculs inutiles

L'algorithme alpha-bêta a pour but de déterminer un arbre solution. Lorsqu'il cherche à démontrer qu'un nœud est gagnant, il choisit soigneusement l'ordre dans lequel les fils de ce nœud sont étudiés, en essayant d'explorer en priorité un fils perdant. Or, quels que soient les raffinements mis en œuvre (par exemple, tri en utilisant la valeur des pièces, les menaces potentielles...), l'ordre choisi se trouve mis en échec sur certaines positions, et l'algorithme se retrouve à explorer des branches inutiles de l'arbre de jeu. En effet, un programme qui trouverait un fils perdant systématiquement sans erreur jouerait parfaitement aux échecs, et le jeu serait alors résolu par méthode.

Aussi, en pratique, pour certains nœuds gagnants, l'algorithme rate son coup et étudie un ou plusieurs fils gagnants avant de trouver un fils perdant. Estimons l'impact de ces calculs inutiles sur la taille de l'arbre solution. On suppose toujours étudier un arbre de jeu équilibré dont chaque nœud interne a n fils, et dont les parties se terminent en p coups. Nous avons vu avec la relation (1) que le nombre de nœuds de cet arbre de jeu est de l'ordre de n^p .

On suppose alors qu'en moyenne, au lieu d'un unique fils, l'algorithme explore α fils avec $\alpha \in [1; n]$. On démontre en annexe que le nombre de nœuds étudiés est de l'ordre de $\alpha^{\frac{p}{2}} \sqrt{n^p}$: on retrouve $\sqrt{n^p}$, l'estimation (2) du nombre de nœuds de l'arbre solution, corrigée d'un facteur $\alpha^{\frac{p}{2}}$ qui traduit l'augmentation de la longueur du calcul. Ce facteur traduit directement la qualité de la programmation, il est possible de le faire diminuer en programmant des algorithmes qui jouent de meilleurs coups.

Transpositions

Étudions maintenant l'effet des transpositions. Nous avons vu que l'effet était difficile à prévoir *a priori*, c'est donc le calcul qui va nous indiquer dans quel sens penche la balance.

On suppose donc que l'arbre de jeu comporte des transpositions, uniformément réparties dans l'arbre ; si bien que lorsqu'on passe d'un étage au suivant, le nombre de nœuds de l'étage n'est pas multiplié par n , mais par tn , où $t \in]0; 1]$ (t représente le taux moyen de nœuds restant après transpositions ; $t = 1$ lorsqu'il n'y a pas de transposition).

On démontre cette fois (les calculs sont en annexe) que le nombre de nœuds de l'arbre de jeu est de l'ordre de $t^p n^p$, tandis que le nombre de nœuds de l'arbre solution est de l'ordre de $t^{\frac{p}{2}} \sqrt{t^p n^p}$. On obtient à nouveau quelque chose de l'ordre de la racine carrée du nombre de nœuds de l'arbre de jeu ; la seule différence est le facteur $t^{\frac{p}{2}}$, plus petit que 1, qui laisse penser que l'existence de transpositions tend à faire diminuer la taille des arbres solution.

Mort subite

Pour estimer l'effet de la mort subite, étant donné un étage de l'arbre de jeu, on appelle $m \in [0; 1]$ le taux de mort subite parmi les nœuds fils. Notons que m dépend de l'étage : en début de partie, il y a moins de positions de mat que plus bas dans l'arbre. Une fois m déterminé pour chaque étage (par exemple, par une formule), il est possible de calculer le nombre de nœuds de chaque étage de l'arbre de jeu, car si un étage père contient S nœuds, l'étage fils contient $S \times n$ nœuds fils, dont $S \times n \times m$ sont terminaux – et ainsi, l'étage encore en-dessous contient $S \times n^2 \times (1 - m)$ nœuds fils.

On peut mener un calcul similaire pour l'arbre solution, dont les étages alternent entre nœuds perdants et gagnants (voir figure 14). Ainsi, si un étage père de l'arbre solution contient S nœuds perdants, il y a de même $S \times n$ nœuds fils gagnants, dont $S \times n \times m$ sont terminaux. Il y a cependant une différence : si l'étage père contient S nœuds gagnants, alors l'étage fils contient S nœuds perdants, dont $S \times m$ sont terminaux.

On peut même être plus efficace dans ce dernier cas, car lorsqu'un nœud gagnant a plusieurs fils perdants, on peut choisir en priorité un fils terminal, ce qui va permettre de réduire la taille de l'arbre solution. Pour traduire cela par le calcul, il faut commencer par estimer les proportions de nœuds gagnants et perdants dans l'arbre de jeu. Appelons g la probabilité qu'un nœud soit gagnant. Alors, la probabilité qu'un nœud soit perdant est $1 - g$, mais aussi g^n car un nœud est perdant lorsque ses n fils sont gagnants. On obtient donc l'équation :

$$(3) \quad g^n = 1 - g$$

Cette équation a une seule solution entre 0 et 1, qui donne donc la probabilité qu'un nœud soit gagnant. Cette probabilité ne dépend que de n ; par exemple, pour $n = 18$ (valeur retenue dans notre estimation ci-après), on obtient 88,6 % de nœuds gagnants¹⁹.

Un calcul explicité en annexe permet alors de déterminer que l'on peut choisir les S nœuds fils perdants de sorte que $S \times \frac{1}{g} (1 - [1 - m(1 - g)]^n)$ de ces nœuds soient terminaux, ce qui est meilleur que la valeur initiale de $S \times m$.

Estimation

Nous allons maintenant utiliser ces trois modélisations pour obtenir l'estimation recherchée. Pour cela, il faut commencer par estimer les divers paramètres nécessaires à ces modélisations.

En premier lieu, des observations expérimentales permettent d'obtenir un ordre de grandeur du nombre moyen de coups jouables à partir d'une position. Par exemple, dans l'article [8], on parle de 35 ; dans le livre [5], il est dit être compris entre 30 et 40.

Cependant, François Labelle a calculé les 10 premiers étages de l'arbre de jeu des échecs [7], et le rapport du nombre de positions d'un étage sur l'autre semble être plutôt de l'ordre de 9 que de 35. Il y a deux raisons à cet écart : d'une part, les transpositions, et d'autre part, dans les 35 coups jouables à partir d'une position, certains ramènent à une position qu'il est possible d'obtenir en moins de coups. Ceci étant considéré, nous choisissons $n = 18$, et $t = 0,5$ pour le taux de transpositions, ce qui donne bien 9 pour le rapport.

Quant à la mort subite, m évolue en fonction de la hauteur des étages : au début de l'arbre de jeu, le taux est nul, tandis qu'au dernier étage de l'arbre de jeu, il vaut 1. Entre les deux, l'évolution est difficile à estimer. La page [7] permet encore d'obtenir des valeurs sur les premiers étages, dont nous avons tiré la formule suivante :

$$m(h) = \frac{1 + \tanh(h/8, 5 - 5)}{2}$$

où h représente la distance de l'étage à la racine de l'arbre.

Reste à évaluer α , qui traduit les calculs inutiles effectués par l'algorithme alphabêta. Nous prendrons $\alpha = 2$, soit en moyenne, un nœud inutilement évalué avant que le nœud perdant soit trouvé, hypothèse nous semblant prudente.

L'utilisation conjointe de ces paramètres et des modélisations décrites précédemment donnent, après un calcul réalisable avec un tableur :

- $2,5 \times 10^{43}$ pour le nombre de positions de l'arbre de jeu.
- $3,1 \times 10^{12}$ pour le nombre de positions de l'arbre solution.

19. Il est logique que les nœuds gagnants soient plus nombreux : il suffit d'un fils perdant pour qu'un nœud soit gagnant, tandis qu'il faut que tous ses fils soient gagnants pour qu'il soit perdant, ce qui paraît plus difficile à obtenir.

— $7,1 \times 10^{18}$ pour le nombre de positions de la table de transpositions, en prenant $\alpha = 2$.

Cette estimation est bien sûr critiquable. Le nombre de positions de l'arbre solution paraît bien faible, la responsabilité en incombe aux transpositions dont l'effet paraît surestimé. Concernant les calculs inutiles effectués par l'algorithme alpha-bêta, la valeur $\alpha = 2$ est quelque peu artificielle, et là encore, son effet est probablement surestimé (ce qui compense l'influence des transpositions).

À l'inverse, l'effet de la mort subite est lui probablement sous-estimé : il ne compte que pour un facteur 5 dans la taille de l'arbre solution, et 11 dans la table de transpositions. En réalité, les nœuds terminaux ne sont pas répartis de manière homogène comme dans notre modélisation, si bien que concentrer l'étude sur les branches de l'arbre de jeu comprenant le plus de nœuds terminaux est plus efficace que ce que notre calcul laisse entendre.

Tout ceci étant considéré, et constatant que par rapport à l'estimation initiale de 10^{20} donnée à la section 7, seuls les calculs inutiles tirent ce nombre dans le mauvais sens, il nous paraît très vraisemblable qu'une table de transpositions capable de résoudre les échecs contienne entre 10^{15} et 10^{20} positions.

Voyons ce que cette estimation implique en terme de délai avant que le jeu d'échecs soit résolu. Le meilleur point de repère à notre disposition est la résolution du jeu de dames anglaises en 2007, pour lequel la quantité de calculs effectués était de l'ordre de 10^{14} [9]. Si l'on suppose que la loi de Moore, qui implique le gain d'un facteur 10 tous les 5 ans, continuera à être suivie, 10^{20} correspond à 2037 comme date limite pour la résolution du jeu d'échecs.

Conclusion

Cet article a permis de mettre à jour certaines idées reçues quant à la possibilité de résoudre le jeu d'échecs. Trop souvent, quand on imagine le fonctionnement des programmes spécialisés dans les jeux combinatoires, on voit le calcul comme un problème de force brute, et donc on surestime la puissance de calcul nécessaire ; et à l'inverse, on sous-estime l'intérêt et l'efficacité des techniques développées pour simplifier les calculs. Nous avons ainsi montré que des techniques pourtant très basiques dans l'étude des jeux combinatoires permettent de faire redescendre la complexité du jeu d'échecs de 10^{120} à un nombre aux alentours de 10^{20} .

Et de fait, les progrès qui ont permis aux ordinateurs de battre les joueurs humains aux échecs sont à créditer au moins autant à l'amélioration des méthodes de calcul qu'à l'augmentation de la puissance des ordinateurs. Notre estimation de la date de résolution du jeu d'échecs est imprécise, avec des faiblesses que nous avons déjà pointées, mais de plus, nous n'avons pas tenu compte d'éventuels progrès des méthodes de calcul : la date pourrait ainsi être encore plus proche qu'escomptée.

Pour en avoir le cœur net, il est nécessaire de commencer la recherche sur le sujet, se confronter au problème étant le meilleur moyen d'en prendre la mesure.

Références

- [1] Louis Victor Allis. A knowledge based approach to connect-four. the game is solved. Master's thesis, University of Vrije, Amsterdam, 1988.
- [2] Louis Victor Allis. *Searching for Solutions in Games and Artificial Intelligence*. PhD thesis, University of Limburg, Maastricht, 1994.
- [3] Henry Bottomley. How many tic-tac-toe (noughts and crosses) games are possible ? <http://www.sel6.info/hgb/tictactoe.htm>, 2001.
- [4] Dennis M. Breuker. *Memory versus Search in Games*. PhD thesis, Universiteit Maastricht, 1998.
- [5] Tristan Cazenave, Mark H.M. Winands, and Hiroyuki Iida. Computer games. *Communications in Computer and Information Science*, 408, 2014.
- [6] Kevin Crowley and Robert S. Siegler. Flexible strategy use in young children's tic-tac-toe. *Cognitive Science*, 17(4) :531–561, 1993.
- [7] François Labelle. Statistics on chess positions. <http://wismuth.com/chess/statistics-positions.html>, 2012.
- [8] Alan Levinovitz. The mystery of go, the ancient game that computers still can't win. <http://www.wired.com/2014/05/the-world-of-computer-go>, 2014.
- [9] Jonathan Schaeffer and al. Checkers is solved. *Science*, 317 :1518–1522, 2007.
- [10] Claude E. Shannon. Programming a computer for playing chess. *Philosophical magazine*, 41, 1950.
- [11] Suhas Sreedhar. Checkers, solved ! <http://spectrum.ieee.org/computing/software/checkers-solved>, 2007.
- [12] John Tromp. John's chess playground. <http://tromp.github.io/chess/chess.html>, 2010.
- [13] Alan Turing. Digital computers applied to games. 1953.
- [14] H. Jaap van den Herik, Jos W. H. M. Uiterwijk, and Jack van Rijswijck. Games solved : Now and in the future. *Artificial Intelligence*, 134(1-2) :277–311, January 2002.

Annexe : Détail des calculs pour les estimations

Calculs inutiles

Soit un arbre de jeu équilibré dont chaque nœud interne a n fils, et dont les parties se terminent en p coups. En moyenne, au lieu d'un unique fils, l'algorithme explore α fils avec $\alpha \in [1; n]$ (ce qui traduit les calculs inutiles). Alors le nombre de nœuds étudiés est :

$$\begin{aligned}
 & 1 + \alpha + \alpha n + \alpha^2 n + \alpha^2 n^2 + \alpha^3 n^2 + \dots + \alpha^{\frac{p+1}{2}} n^{\frac{p-1}{2}} \\
 &= (1 + \alpha) + (1 + \alpha)n + (1 + \alpha)n^2 + \dots + (1 + \alpha)n^{\frac{p-1}{2}} \\
 (4) \quad &= (1 + \alpha) \frac{(\alpha n)^{\frac{p+1}{2}} - 1}{\alpha n - 1} \sim (1 + \alpha)(\alpha n)^{\frac{p-1}{2}} \simeq \alpha^{\frac{p}{2}} \sqrt{n^p}
 \end{aligned}$$

Transpositions

Soit un arbre de jeu équilibré dont chaque nœud interne a n fils, dont les parties se terminent en p coups, et comportant des transpositions uniformément réparties : quand on passe d'un étage au suivant, le nombre de nœuds de l'étage est multiplié par tn avec $t \in]0; 1]$. Le nombre de nœuds de l'arbre de jeu est alors :

$$(5) \quad 1 + tn + t^2 n^2 + \dots + t^p n^p = \frac{(tn)^{p+1} - 1}{tn - 1} \sim t^p n^p$$

Quant au nombre de nœuds de l'arbre solution, il vaut :

$$\begin{aligned}
 & 1 + t + t^2 n + t^3 n + t^4 n^2 + t^5 n^2 + \dots + t^p n^{\frac{p-1}{2}} \\
 &= (1 + t) + (t^2 n)(1 + t) + (t^2 n)^2 (1 + t) + \dots + (t^2 n)^{\frac{p-1}{2}} (1 + t) \\
 (6) \quad &= (1 + t) \frac{(t^2 n)^{\frac{p+1}{2}} - 1}{(t^2 n) - 1} \sim (1 + t)t^{p-1} n^{\frac{p-1}{2}} \simeq t^p n^{\frac{p}{2}} = t^{\frac{p}{2}} \sqrt{t^p n^p}
 \end{aligned}$$

Mort subite

Nous étudions ici l'effet du choix prioritaire d'un fils terminal parmi les fils perdants d'un nœud gagnant. Pour commencer, nous allons utiliser la relation suivante :

$$(7) \quad \sum_{i=0}^n \binom{n}{i} g^i (1-g)^{n-i} = 1$$

Sa démonstration est triviale, il suffit d'appliquer la formule du binôme à $(g + (1 - g))^n$. Remarquons que le terme numéro i de cette somme traduit la probabilité qu'un nœud ait i fils gagnants, et donc $n - i$ fils perdants. En particulier, le terme numéro n est la probabilité que le nœud soit perdant (car tous ses fils sont gagnants), et vaut g^n , soit $1 - g$ par la relation (3). La somme des $n - 1$ autres termes vaut donc g .

Maintenant, nous pouvons calculer la proportion de nœuds gagnants possédant au moins un fils perdant terminal :

$$(8) \quad \frac{1}{g} \sum_{i=0}^{n-1} \binom{n}{i} g^i (1-g)^{n-i} [1 - (1-m)^{n-i}]$$

En effet, le nœud étant gagnant, on enlève le terme numéro n de la relation (7), et on divise par la somme des autres termes, d'où le facteur $\frac{1}{g}$. Pour justifier le crochet, on remarque que s'il y a $n-i$ fils perdants, la probabilité qu'aucun ne soit terminal est $(1-m)^{n-i}$, donc la probabilité qu'on puisse en choisir un terminal est $[1 - (1-m)^{n-i}]$.

Remarquant que le crochet s'annule si $i = n$, la somme (8) devient :

$$\begin{aligned} & \frac{1}{g} \sum_{i=0}^n \binom{n}{i} g^i (1-g)^{n-i} [1 - (1-m)^{n-i}] \\ &= \frac{1}{g} \sum_{i=0}^n \binom{n}{i} g^i (1-g)^{n-i} - \frac{1}{g} \sum_{i=0}^n \binom{n}{i} g^i [(1-g)(1-m)]^{n-i} \\ &= \frac{1}{g} [g + (1-g)]^n - \frac{1}{g} [g + (1-g)(1-m)]^n \\ (9) \quad &= \frac{1}{g} (1 - [1 - m(1-g)]^n) \end{aligned}$$

Ainsi, comme annoncé, on peut choisir les S nœuds fils perdants de sorte que $S \times \frac{1}{g} (1 - [1 - m(1-g)]^n)$ de ces nœuds soient terminaux.

Remarquons que l'inégalité de Jensen permet logiquement de constater que l'expression (9) est supérieure à m , et donc que ce choix réalise une amélioration par rapport à la méthode initiale :

$$\begin{aligned} (1-m).1^n + m.g^n &\geq [(1-m).1 + m.g]^n \\ 1-m + mg^n &\geq [1-m + mg]^n \\ 1-m(1-g^n) &\geq [g + 1-m-g + mg]^n \\ 1-mg &\geq [1-m(1-g)]^n \\ (10) \quad m &\leq \frac{1}{g} (1 - [1-m(1-g)]^n) \end{aligned}$$



Humanités numériques

Serge Abiteboul¹ et Florence Hachez-Leroy²

Cet article est paru dans l'Encyclopédie de l'humanisme méditerranéen : Serge Abiteboul et Florence Hachez-Leroy, « Humanités numériques », in Houari Touati (éd.), Encyclopédie de l'humanisme méditerranéen, printemps 2014, <http://www.encyclopediae-humanisme.com/?Humanites-numeriques>.

Nous le publions dans ce numéro de 1024 avec leur aimable autorisation.

« The digital humanities, also known as humanities computing^{3,4}, is a field of study, research, teaching, and invention concerned with the intersection of computing and the disciplines of the humanities. It is methodological by nature and interdisciplinary in scope. It involves investigation, analysis, synthesis and presentation of information in electronic form. It studies how these media affect the disciplines in which they are used, and what these disciplines have to contribute to our knowledge of computing. »

Wikipedia, <http://en.wikipedia.org>, 2014.

1. INRIA & ENS Cachan.

2. Université d'Artois & CRH (CNRS / EHESS).

3. John Unsworth, « What is Humanities Computing and What is Not ? », in Melissa Terras, Julianne Nyhan and Edward Vanhoutte (ed.), *Defining Digital Humanities : A Reader*, Londres, Ashgate, 2013.

4. Willard McCarty, *Humanities Computing*, New York, Palgrave MacMillan, 2014.

Introduction

La transformation des humanités par le numérique. L'expression « humanité numérique⁵ » est aujourd'hui moquée au motif que l'association entre une technologie et les sciences humaines et sociales tiendrait de l'oxymore. Or, si l'on compare avec l'imprimerie, celle-ci n'a-t-elle pas été aussi une technologie dont l'apport transforma en profondeur les humanités d'alors ? Une autre critique plus incisive considère que l'adjonction de l'adjectif « numérique » relèverait de la tautologie. Dans un monde numérique, les humanités pourraient-elles éviter de suivre le mouvement et devenir elles-aussi numériques ? Peut-on par exemple imaginer aujourd'hui la géographie sans les systèmes d'information géographique ou la linguistique sans le traitement automatique de la langue ? Cela n'est pas plus censé que d'imaginer, par exemple, l'astronomie sans ses pipelines de calculs informatiques, ou la génomique sans ses algorithmes d'analyse de séquences ADN. Nous vivons donc bien une transition des humanités. En même temps, il existe un risque qu'à être partout, le numérique ne soit singulièrement nulle part, qu'il manque un lieu de *coordination* des efforts autour du numérique. Alors pendant cette transition au moins, il nous semble intéressant d'examiner spécifiquement les aspects numériques de cette transition.

Un chercheur en humanités aujourd'hui consulte des documents sur Internet, classe ses informations dans des bases de données, extrait des informations dans des bases de données de plus en plus riches et nombreuses, tweete, blog, chatte à distance avec ses collègues, ses étudiants, etc. Est-ce que les Humanités numériques se résument à cela ? Certainement pas, mais il n'est pas simple d'en donner une définition. Nous allons plutôt essayer de répondre à la question : en quoi est-ce que l'informatique et le numérique transforment les humanités ? Walter Benjamin a expliqué que la photographie a transformé l'art. De la même façon, nous allons chercher à mettre en évidence dans cet article comment le numérique et fondamentalement la science et la technique informatique ont transformé en profondeur les sciences humaines et sociales.

Une convergence entre sciences et humanités. Les humanités numériques sont un lieu privilégié de convergence entre sciences et humanités. La dissociation entre sciences et humanités s'est développée progressivement aux 19^e et 20^e siècles en particulier du fait de la spécialisation des recherches et du changement de regard sur les disciplines littéraires jugées peu compatible avec la « modernité ». L'enseignement a suivi au point que, par exemple, en France, on demande aux lycéens de première de choisir entre sciences et lettres. Un tel choix n'a pas véritablement de sens et il s'avère tout particulièrement absurde pour un élève qui serait intéressé par les humanités numériques. Un ordinateur est une machine à tout faire (« *general purpose* ») : le même système, par exemple un système de gestion de bases de données,

5. Marin Dacos et Pierre Mounier, *Humanités numériques : État des lieux et positionnement de la recherche française dans le contexte international*, Institut Français, 2014.

peut être utilisé que la science soit « humaine » ou pas, et le même algorithme, par exemple un algorithme de recommandation, peut être utilisé dans les deux cas. Un historien ou un sociologue est aussi à même de saisir la poésie d'un code informatique qu'un physicien ou un biologiste. Les méthodes, les concepts, les techniques, les outils de l'informatique rapprochent les chercheurs de toutes ces disciplines. Les humanités sont donc en train de se réinventer par l'informatique et de se rapprocher des autres sciences, nous semble-t-il. Ces convergences sont si fortes que plutôt que de parler des humanités numériques, peut-être aurait-il fallu discourir de « sciences numériques » en général.

Organisation. Dans un premier temps, il est utile de délimiter le territoire et de préciser en particulier le sens d'« humanités » et de « numériques ». Nous examinerons ensuite divers aspects des humanités numériques. Celles-ci sont déjà, malgré leur jeunesse, d'une telle richesse que chaque choix d'une illustration en écartera de nombreuses autres tout aussi intéressantes. Évidemment, ces choix seront biaisés, discutables. Notre propos n'est pas d'établir un classement mais uniquement d'illustrer nos propos. Organiser les facettes du monde des humanités numériques n'est pas chose facile. D'une certaine façon, nous allons passer en revue des classes d'outils, mais ces outils existent par les techniques, les méthodes, les usages qu'ils impliquent. Mais considérons d'abord le territoire.

Le territoire

Les humanités. Le terme est imprécis. En anglais, par exemple, le terme « humanities » ne couvre pas les sciences sociales. Pour nous le domaine des humanités numériques est à prendre dans un sens très général, incluant l'histoire, la linguistique, la littérature, les arts et le design, mais aussi la géographie, l'économie, la sociologie, le droit, la théologie et les sciences des religions, etc. D'un côté, les humanités, de l'autre, l'informatique et le numérique.

Le numérique et l'informatique

L'informatique est la science et la technique de la représentation de l'information d'origine artificielle ou naturelle, ainsi que des processus algorithmiques de collecte, stockage, analyse, transformation, communication et exploitation de cette information, exprimés dans des langages formels ou des langues naturelles et effectués par des machines ou des êtres humains, seuls ou collectivement.

L'informatique : la science au cœur du numérique⁶,
Société Informatique de France, 2014.

6. Binaire. <http://binaire.blog.lemonde.fr/informatique-quesaco/>

Il nous faut ici considérer l'articulation entre le monde numérique et la science qui en est au cœur, l'informatique ^{6, 7}. Dans un premier temps, on a parlé de « *humanities computing* », en mettant l'accent sur les aspects tenant de l'informatique. Puis la focale a été élargie en préférant l'adjectif « *digital* » (en français, numérique) qui qualifie les activités s'appuyant sur la numérisation de l'information. Si l'informatique a rendu possible les humanités numériques, se limiter à cette science, à cette technique, serait réducteur. Par exemple, le Web qui a une place si essentielle dans les humanités numériques, au delà de ses aspects purement techniques, tient d'une philosophie de la mise à disposition pour tous, du partage. Cet aspect comme bien d'autres de la culture numérique font partie intégrante du cadre des humanités numériques. Au-delà de l'informatique, nous considérerons le numérique, ses usages, le monde qu'il construit, ses cultures.

Les humanités numériques se situent donc quelque part aux points de rencontre d'une part, des humanités et de l'autre, de l'informatique et du numérique. On y inclut typiquement de nombreux aspects comme, l'enseignement, la création artistique ou la recherche. Nous concentrerons notre propos sur cette dernière et considérerons comment

la recherche dans les sciences humaines et sociales évolue dans sa rencontre avec la science et la technique informatique, le monde et la culture numérique.

Notre propos (et c'est un choix arbitraire) porte sur les humanités numériques uniquement lorsqu'un processus de recherche scientifique est impliqué. Par exemple, la conception d'un cours de grec ancien ⁸ par un Flot (Mooc en anglais) n'entre pas dans notre panel, pas plus que la conception d'une visite de musée à partir d'un logiciel spécialisé. En revanche, un chercheur intéressé par les flots ou la visite numérisée de musées et un ingénieur concevant un logiciel pour ce faire entrent dans notre champ.

Nous ne traiterons pas non plus de l'enseignement des humanités à l'heure du numérique. Évidemment, l'éducation change. Et il ne s'agit pas seulement d'utiliser des outils numériques pour enseigner, pas seulement d'enseigner autrement, mais également de faire évoluer les contenus aussi bien d'ailleurs dans les matières scientifiques que dans les humanités. Cet aspect est intimement lié au sujet de cet article, mais c'est une autre thématique. Et tout aussi arbitrairement encore, nous excluons d'autres aspects, comme la création d'art numérique ou les interactions entre les machines et les personnes, bien que certains les considèrent comme partie des humanités numériques.

En dépit de ces restrictions, le territoire reste vaste. Depuis les tout débuts de l'informatique, des chercheurs et des ingénieurs de laboratoires SHS et informatique se

7. Milad Doueïhi, *La grande conversion numérique*, Paris, Seuil, 2011.

8. Les flots Sillages. <http://flot.sillages.info/?portfolio=grec-ancien-debutants>

sont intéressés aux humanités numériques. Ils ont considéré des modèles quantitatifs, par exemple pour des analyses statistiques de texte s'appuyant sur la linguistique computationnelle. Ils sont passés au qualitatif avec des initiatives comme le « *Text encoding initiative*⁹ » en développant des outils informatiques qui faisaient véritablement évoluer la représentation et l'échange de connaissances. Ils ont développé de nouveaux outils, de nouvelles techniques, conçu de nouvelles façons de faire la recherche. Ils ont bâti une discipline avec ses textes de référence (comme *Une introduction aux humanités numériques*¹⁰), ses revues (comme le *Journal of Digital Humanities*), ses sociétés savantes (comme l'*Alliance of Digital Humanities Organizations*), ses conférences (comme *Digital Humanities*), ses instituts (comme l'*Institut des Humanités Digitales de Bordeaux*), ses séminaires (comme « *Digital Humanities. Les transformations numériques du rapport aux savoirs* » à l'EHESS), ses financements (comme l'Axe 6 du Défi 8 du dernier appel ANR « *Révolution numérique et mutations sociales* ») et des chercheurs et enseignants-chercheurs qui se réclament spécialistes des humanités numériques. C'est de ce champ que nous allons essayer de dresser un rapide panorama.

La transformation des sciences par la pensée informatique

Au fil des recherches en humanités numériques, ce qui nous paraît le plus passionnant, c'est une remise en question radicale des SHS, de l'essence de leurs pratiques, une entreprise de transformation fondamentale de ces sciences par la pensée informatique. C'est ce que nous aurions aimé placer au cœur de cet article. Mais nous manquons sans doute encore de recul. Ces transformations sont encore pour une grande part en devenir. Si ces considérations sous-tendent tout notre article et plus spécifiquement les sections sur les connaissances, ou la simulation, l'accent est aussi mis sur les nouveaux outils et les nouvelles pratiques.

Les bases : numérisation, bases de données, hypertexte et Internet

Le point de départ des humanités numériques est la représentation de connaissances sous forme numérique. La numérisation des textes a ainsi énormément bénéficié des outils d'OCR (« optical character recognition »). Mais la numérisation touche bien plus que le texte : les images fixes et animées, la musique, l'architecture, etc. Si quelques bastions résistent un peu comme l'odorat, le champ du numérique ne cesse de s'étendre.

L'objet numérisé apporte son lot de problèmes. Comment garantir des propriétés comme l'authenticité, l'intégrité, l'identité, la pérennité ? Ces problèmes ne sont pas nouveaux (les faussaires, les manipulateurs de documents, les tricheurs en tout genre

9. Text Encoding Initiative. <http://www.tei-c.org/>

10. Pierre Mounier (dir.), *Read/Write Book 2 : une introduction aux humanités numériques*, Marseille, OpenEdition Press, 2012 (généré le 26 octobre 2014). <http://books.openedition.org/oep/226>. ISBN : 9782821813250.

n'ont pas attendu le numérique pour sévir) mais se posent avec plus d'acuité du fait que le numérique permet la reproduction du même objet quasi sans coût (on parle de biens non rivaux ¹¹). Pourtant, les techniques apportent aussi des solutions qu'il reste à développer, à déployer. Pour ne prendre qu'un exemple l'authenticité. En utilisant des algorithmes cryptographiques comme RSA ¹², il est possible d'accompagner un texte d'une signature qui l'authentifie bien mieux que l'on ne pourrait authentifier quelque objet physique que ce soit. Reste que de telles signatures sont de nos jours encore trop peu utilisées. Reste aussi que le numérique accentue l'affaiblissement de la notion d'auteur, et la rigidité de la notion d'œuvre.

Le texte fut l'application liminaire de la numérisation. Une fois numérisé, il pouvait être indexé automatiquement. Une des premières utilisations massives de l'informatique dans les humanités numériques a été la gestion informatisée des catalogues de bibliothèques avec des index réalisés totalement manuellement dans les premiers temps. Les bibliothécaires voyaient leur monde évoluer sans peut-être imaginer qu'après les catalogues, les livres et les bibliothèques pouvaient un jour devenir numériques.

Au-delà du simple texte linéaire et des index, l'informatique offre le moyen de gérer en général des données et des informations. Les structures naturelles pour organiser l'information en mathématiques et en informatique sont les tableaux, les arbres et les graphes. Nous allons naturellement rencontrer ces trois types de structures : le tableau est la structure la plus utilisée dans les bases de données, l'arbre est la structure interne des documents HTML ou XML, et nous retrouvons les graphes dans les documents hypertextes.

Un système de gestion de bases de données sert de médiateur entre des individus et des machines. Dans le modèle relationnel proposé par Ted Codd dans les années 1970, les données sont organisées en tableaux à deux dimensions qu'on appelle des relations. Les données sont interrogées en utilisant des « requêtes » dans le langage SQL (inspiré de la Logique du premier ordre). Ces requêtes sont évaluées par le système en s'appuyant sur l'« algèbre relationnelle ».

Serge Abiteboul, Leçon inaugurale ¹³ du Collège de France, 2012.

Les chercheurs de SHS ont vite compris l'intérêt de réunir des données, de les organiser dans des bases de données notamment relationnelles. La production d'une base de données peut d'ailleurs être aujourd'hui l'objet d'un sujet de recherche au même titre que l'écriture d'un article dans la meilleure revue d'un domaine. La

11. Lawrence Lessig, *L'avenir des idées*, Lyon, Presses Universitaires de Lyon, 2005.

12. Ron Rivest, Adi Shamir, Len Adleman, « Method for Obtaining Digital Signatures and Public-Key Cryptosystems », *Communications of the ACM* 21 (2), 1978, pp. 120–126.

13. Serge Abiteboul, *Sciences des données : de la logique du premier ordre à la Toile*, Paris, Fayard, 2012.

même situation existe dans d'autres domaines scientifiques notamment la biologie. Par exemple, le point de départ de la base de données de séquences protéiques Swiss-Prot est le doctorat d'Amos Bairoch en 1986. A partir de données entrées manuellement, Swiss-Prot donne des informations extrêmement détaillées sur des séquences de protéines. Cette base de données est une contribution scientifique essentielle. De même en SHS, l'Internet Movie Database (IMDb) est une gigantesque base de données en ligne sur le cinéma mondial (films, acteurs, réalisateurs, etc.). IMDb a même été en 2011 parmi les 40 sites les plus visités du Web.

Après la base de données, un autre concept informatique essentiel pour les SHS est l'« hypertexte ».

Un système hypertexte est un système contenant des nœuds liés entre eux par des hyperliens permettant de passer automatiquement d'un nœud à un autre. Un document hypertexte est donc un document qui contient des hyperliens et des nœuds. Un nœud est une « unité minimale d'informations », notion assez floue qui signifie simplement que l'information d'un nœud sera toujours présentée entière. Les liens entre les parties du texte sont gérés par ordinateur et permettent d'accéder à l'information d'une manière associative ou, tout au moins, d'une façon de naviguer personnalisée, de manière non linéaire, au gré de l'utilisateur.

Wikipédia, français, 2014.

Il n'est sans doute pas besoin d'expliquer plus loin la notion d'hypertexte, le lecteur ayant déjà l'expérience de navigation sur le Web, qui illustre la réussite planétaire des hypertextes. Les ingrédients du Web sont des plus simples : un réseau de machines, Internet, et un protocole d'hypertextes pour accéder à un réseau de textes disponibles sur ce réseau de machines, le World Wide Web.

You affect the world by what you browse.

Tim Berners-Lee

Bases de données et hypertextes se combinent à une troisième grande réalisation de l'informatique, Internet, pour former un des plus beaux succès des humanités numériques : les « bibliothèques numériques ». Par exemple, le Projet Perseus¹⁴ de l'université Tufts s'est attaqué à la construction d'une bibliothèque numérique qui rassemble des textes du monde méditerranéen en particulier en grec, latin et arabe. Les textes numérisés, indexés, disponibles sur la Toile, sont facilement accessibles à tous. Son ambition affichée est : « *to help make the full record for humanity as intellectually accessible as possible to every human being, providing information adapted to as many linguistic and cultural backgrounds as possible.* »

Voilà un cas exemplaire de contribution aux humanités.

14. Perseus Digital Library Project, <http://www.perseus.tufts.edu/hopper/>

Cette diffusion s'accompagne du développement d'interfaces qui vont au-delà de l'accès classique aux connaissances par le texte. C'est le cas par exemple pour ce qui est de l'architecture¹⁵ avec Usine 3D. Le « visiteur » peut découvrir la reconstitution du premier atelier automobile au début du 20^e siècle, avec des chaînes de montage. Il s'agit d'une véritable expérience qui va plus loin que la consultation de nombreux ouvrages et plans d'architectes. Photographies, images figurées, textes, plans, commentaires des historiens spécialistes, etc. sont analysés, les informations recoupées pour former un corpus hétérogène qui concourt à cette reconstitution scientifiquement encadrée.

La transformation radicale de l'accès à l'information fait qu'à l'heure du Web, des étudiants, mais aussi des amateurs, des journalistes, tous les citoyens ont accès à des sources considérables. Chacun peut s'improviser chercheur ou journaliste et contribuer à l'enrichissement des connaissances en rédigeant par exemple les articles d'une encyclopédie, domaine réservé jusque là aux chercheurs ou aux érudits. Observer les oiseaux et alimenter ensuite une base de données scientifique par ses informations est désormais à la portée de tous.

Le partage : interaction et communication

Depuis des siècles, le travail des chercheurs s'est appuyé sur la notion de réseau. On échangeait des lettres. On voyageait pour consulter une bibliothèque ; on en profitait pour rencontrer ses homologues locaux. Ces échanges, ces rencontres physiques participaient à élaborer et à enrichir les connaissances, à construire des réseaux.

Après Internet et le réseau des machines, après le Web des débuts et le réseau de contenus, le Web 2.0 s'est proposé pour faciliter les communications entre individus, enrichir les interactions entre eux, réinventer les réseaux sociaux. Pour les scientifiques, cela permettait de redéfinir le travailler ensemble. On pouvait partager des textes, les annoter ensemble, les commenter, voire co-rédiger des contenus très riches en s'éloignant du texte linéaire bien défini aux auteurs bien précisés.

Les exemples fourmillent. Le réseau peut être par exemple massif et à vocation généraliste comme ResearchGate avec plus d'un million de membres ou dédié à un partage entre les chercheurs et le grand public comme Nature Network de la revue Nature. Pour citer un exemple riche en symbole, le projet¹⁶ « *Mapping the republic of letters* » lancé par Stanford a permis de mettre en commun des recherches pour étudier comment, depuis la Renaissance, les lettrés européens partageaient leurs connaissances à travers des textes et des rencontres. Un réseau social numérique pour expliquer un réseau social « classique » !

15. Usine 3D, <http://www.usines3d.fr/>

16. Mapping the Republic of letters, <http://republicofletters.stanford.edu/>

Un scientifique comme Claude Bernard notait ses expériences dans des carnets¹⁷ dont la consultation en ligne est une expérience chargée d'émotions. Que sont de tels carnets aujourd'hui ? Des notes partagées avec des collaborateurs sur Dropbox ? Des bases de données en ligne ? Des articles dans ResearchBlogging ? Des workflows d'expériences sur des sites MyExperiment ? Des carnets de recherche en ligne comme dans Hypotheses.org ?

En utilisant un jeu vidéo, Foldit, des internautes sont arrivés à décoder la structure d'une enzyme proche de celle du virus du sida¹⁸. Ils ont compris ce qui bloquait experts et ordinateurs, comment cette enzyme se repliait dans un espace en trois dimensions pour construire sa structure. Le jeu se marie ici au réseau, dans le plus pur esprit des réseaux sociaux. Si un individu a exposé le pliage, les scientifiques qui ont imaginé le jeu, les programmeurs qui l'ont implémenté, tous les joueurs qui ont essayé les pliages qui ne marchaient pas, tous ont contribué.

Les chercheurs en humanités n'ont pas attendu le Web pour se rencontrer, communiquer entre eux, collaborer. Il n'empêche que c'est seul devant la page blanche que s'élaborait le plus souvent une œuvre. Si cela reste en partie vrai, le travail est devenu typiquement plus collaboratif, les œuvres plus collectives. Le passage au travail en réseau s'accompagne de changements fondamentaux dans nos rapports aux connaissances. Un univers des fragments se substitue aux contributions monolithes. Les outils de recherche, les sites de co-rédaction encouragent cet effet, qui s'accompagne aussi de l'affaiblissement de la contribution de l'auteur individuel devant les contributions du groupe. Dans le cadre des revues académiques, s'étend l'« *open access* », c'est-à-dire la mise à disposition en ligne de contenus numériques. Et au-delà, avec le « *creative commons* » se développe une approche globale pour libérer les « œuvres » des droits de propriété.

La constitution de réseaux devient à son tour objet d'études. La théorie des réseaux s'applique en sociologie¹⁹ aussi bien que pour les réseaux électriques.

La connaissance et le Web sémantique

L'informatique permet de traiter des données, de l'information, des *connaissances*. Elle nous permet de transformer des données en information, en connaissances¹³. Dans sa forme la plus simple, cette connaissance permet d'expliquer le sens de documents textuels publiés²⁰, d'éléments qui les composent, de services Web proposés.

17. La Salamandre, <https://salamandre.college-de-france.fr>

18. Seth Cooper, Firas Khatib, Adrien Treuille et al., « Predicting protein structures with a multiplayer online game », *Nature*, 2010, vol. 466, no. 7307, pp. 756-760.

19. Duncan Watts J., *Six degrees : The science of a connected age*, New York, W. W. Norton & Company, 2004.

20. Davis Randal, Howard Shrobe, Peter Szolovits, « What is a Knowledge Representation ? », *AI Magazine*, 14(1), 1993, pp. 17-33.

C'est la base du *Web sémantique*²¹. Des balisages sont utilisés pour préciser le sens des mots d'un texte, pour faire des ponts entre des ressources distinctes avec le *linked data*. Avec les ontologies, un pas supplémentaire est franchi pour atteindre le monde des connaissances structurées, classifiées, organisées.

Quels sont les buts recherchés ? D'abord d'améliorer l'information publiée, de la préciser, en lever les ambiguïtés pour permettre par exemple de proposer de meilleures réponses aux questions des internautes. Ensuite, il s'agit de faciliter le fusionnement²² de connaissances proposées par plusieurs bases de connaissances (de plusieurs chercheurs, plusieurs labos, plusieurs communautés) en une base de connaissances unique, pour aller pourquoi pas vers une base universelle.

Un des premiers exemples très populaire de balisage de texte est le « *Text encoding initiative* » initié en 1987. Le but du balisage était de permettre de trouver plus facilement de l'information dans de larges collections de textes de bibliothèques. (Les balises étaient en SGML ; elles sont passées depuis en XML.) TEI était défini comme « un système pour faciliter la création, l'échange, l'intégration de données textuelles informatisées²³ ». Nous sommes au cœur du sujet.

Plus récemment, le *Conceptual Reference Model*²⁴ est une ontologie développée pour décrire le patrimoine culturel. C'est une norme très utilisée notamment pour enrichir l'information et les connaissances autour des collections de musées.

Ces deux exemples sont dans des domaines larges mais ciblés. La base de connaissances Yago a une vocation plus encyclopédique. Elle a d'ailleurs été développée à partir de la version anglaise de l'encyclopédie textuelle Wikipédia, un bon point de départ pour développer une grande base de connaissances. L'ontologie Yago a été construite à l'aide d'un logiciel développé à l'Institut Max Planck²⁵. En 2011, Yago avait déjà 2 millions d'entités et plus de 20 millions de relations entre ces entités. Un autre projet, Wikidata, se propose de réunir dans une base de données éditée de manière collaborative des données objectives, telles que les dates de naissance ou bien le PIB des pays, pour être utilisées en complément de Wikipédia. Pour les connaissances proposées typiquement à droite de l'écran quand une question comme « Paul Verlaine » est posée, le moteur de recherche Google va utiliser Wikidata. (Voir Figure 1)

21. Tim Berners-Lee, James Hendler, Ora Lassila, « The semantic web », *Scientific american*, 2001, 284.5, pp. 28-37.

22. Marie-Christine Rousset et al., « Construction de médiateurs pour intégrer des sources d'information multiples et hétérogènes : le projet PICSEL » *Revue I3 (Information-Interaction-Intelligence)*, 2002, Vol. 2, n° 1.

23. Lou Burnard, *Le schéma XML TEI pour l'édition*, Cours donné à l'Université d'été de l'édition électronique ouverte, 2009.

24. The CIDOC Conceptual Reference Model, <http://www.cidoc-crm.org/>

25. Yago2, <http://www.mpi-inf.mpg.de/yago-naga/yago/>

Paul Verlaine

Poète

Paul Marie Verlaine est un poète français, né à Metz le 30 mars 1844 et mort à Paris le 8 janvier 1896. Après que le père, Nicolas-Auguste Verlaine, a démissionné de l'armée, la famille Verlaine s'installe à Paris en 1851. Wikipédia

Naissance : 30 mars 1844, Metz

Décès : 8 janvier 1896, Paris

Inhumation : 10 janvier 1896, Cimetière des Batignolles, Paris

Épouse : Mathilde Mauté de Fleurville (m. 1870–1885)

Parents : Élisabeth Stéphanie Dehée, Nicolas-Auguste Verlaine

FIGURE 1. Connaissances présentées dans le moteur de recherche Google.

Les ontologies servent aussi à formaliser les connaissances que nous pouvons acquérir (automatiquement ou pas) dans un domaine particulier. Par exemple, le projet ANR Hyperprince²⁶ s'est focalisé sur la mise en correspondance de plusieurs traductions du texte de Machiavel. Il a permis d'étudier notamment l'évolution du vocabulaire historique.

Pour pouvoir échanger des connaissances, pour pouvoir être aidés par des systèmes informatique pour stocker, sélectionner, rechercher, inférer des connaissances, etc., nous formalisons ces connaissances dans des ontologies qui ne capturent que des visions finalement assez limitées de ces connaissances. Pour représenter des connaissances dans un langage formel, il nous faut les structurer, les « normaliser », d'une certaine façon les affadir. Évidemment, nous pouvons repousser sans cesse les limites. Par exemple, les informations subjectives contenues dans les textes sont le plus souvent ignorées. Cette connaissance est donc le plus souvent absente des représentations numériques. Des techniques récentes d'« analyse de sentiment » basées sur la linguistique computationnelle (traitement du langage naturel) permettent d'enrichir la représentation de connaissances avec des informations subjectives. Un rôle des humanités numériques est de repousser toujours nos limites, dans la quête de toujours plus de connaissances.

Le passage des données aux connaissances est au cœur du développement des humanités numériques parce que ce passage permet de mieux faire collaborer les chercheurs et les systèmes au développement de la recherche. Il exige des chercheurs d'aller plus loin dans la formalisation des connaissances dont ils disposent. C'est donc dans ce passage que peut le mieux se situer la nécessité de la « pensée informatique » : pour que cette collaboration puisse s'installer, les chercheurs sont amenés à penser autrement. Ils sont amenés à dépasser la question de comment utiliser des moyens numériques pour aller jusqu'à repenser leur façon de mener leurs recherches.

26. Hyperprince, <http://hyperprince.ens-lyon.fr/>

Toujours plus de connaissances

Dans une quête permanente de connaissances, des systèmes informatiques permettent d'obtenir toujours plus de connaissances.

Collecte et analyse de données. Il est intéressant de remarquer que c'est peut-être une des dimensions les plus anciennes des humanités numériques avec Roberto Busan, un jésuite italien, qui a imaginé dans les années 1940 et réalisé ensuite l'analyse linguistique basée sur l'informatique des œuvres complètes de Thomas d'Aquin. On retrouve les techniques d'analyse de texte qu'il a utilisées (indexation, cotexte, concordance, cooccurrence, etc.) dans de nombreuses disciplines notamment en histoire ou en littérature.

Les plus paradigmatiques exemples de cette analyse de données (notamment de par leurs masses) viennent peut-être de Google trends. Google trends permet d'avoir accès à la fréquence d'un mot dans les requêtes au moteur de recherche Google (près de 10 milliards de requêtes par jour en 2014). Il a donné lieu à de nombreuses études. Par exemple :

- La détection d'épidémies. Une corrélation entre la présence de certains mots dans les requêtes et le nombre de cas de gripes détectés a été mise en évidence.
- L'index d'orientation future. Il a été constaté que les internautes des pays dont le PIB brut est plus élevé sont plus susceptibles de rechercher des informations sur l'avenir plutôt que sur le passé.

Le système Hyperbase²⁷ d'Étienne Brunet est un exemple plus académique. Ce logiciel permet l'exploration et les analyses statistiques de corpus textuels. Il a été utilisé en linguistique, littérature, histoire, sociologie et sciences politiques. On peut aussi utiliser des outils numériques pour étudier l'intertextualité sous ses formes multiples de la citation au plagiat, en passant par l'allusion.

La production d'information atteint des niveaux que l'on pouvait difficilement prévoir. Par exemple, une personne « un peu connectée » construit une image digitale dans presque toutes ses activités : déplacements, communications, achats, travail, vie sociale, etc. Ces traces forment un matériel extraordinaire pour les chercheurs en sciences humaines. Encore récemment, pour étudier les déplacements estivaux d'une population, la collecte d'information était fastidieuse et coûteuse. Aujourd'hui on peut disposer de masses d'information considérables par les opérateurs téléphoniques (GPS de portables), ou des plateformes d'achat comme booking.com. Pour prendre un autre exemple, nous disposons de volumes de texte de plus en plus gigantesques. Cela a ouvert un champ d'étude considérable, celui des approches statistiques en linguistique computationnelle. Grâce à ces énormes corpus, l'extraction de connaissances à partir de textes a fait d'énormes progrès ; et en utilisant en particulier des textes multilingues alignés, la traduction automatique a progressé.

27. Hyperbase, <http://fr.wikipedia.org/wiki/Hyperbase>

Évidemment, ces collectes massives d'information posent aux chercheurs de nouvelles questions comme les intrusions sur la vie privée d'un individu qui peut en résulter, ou le déséquilibre que les effets réseaux peuvent introduire : les plus grosses plateformes collectent plus d'information et introduisent des biais dans les concurrences commerciales.

L'inférence. Un avantage d'une représentation numérisée est que des programmes peuvent automatiquement aller chercher de l'information présente implicitement dans les données. Dans le plus simple des cas, c'est un système de gestion de bases de données qui stocke des données sur les joueurs de football et les matchs. Il peut facilement répondre à des requêtes comme de comparer les statistiques des joueurs sortis de différents centres de formation au cours du temps. Nous avons vu aussi comment des connaissances pouvaient être dérivées à partir d'analyses statistiques des données. Les systèmes autour des ontologies permettent également de réaliser des inférences. Pour être capable à partir de données brutes comme la consommation d'aluminium d'un pays, sa production, ses exportations, de dériver la place du recyclage dans ce pays, il faut disposer d'ontologies qui expliquent les concepts du domaine et relient les données disponibles à ces concepts. Dans un monde où nous sommes entourés de plus en plus de systèmes informatiques qui produisent, stockent et échangent de l'information, la place de l'inférence de connaissances ne peut qu'être appelée à grandir et ce notamment dans les humanités numériques.

La modélisation et la simulation

Dans les sciences physiques et les sciences de la vie, la modélisation numérique tient une place considérable. En simplifiant, le chercheur propose un modèle du phénomène complexe étudié, et le simule ensuite numériquement pour voir si les comportements résultants correspondent à ceux observés dans la réalité. Parmi les plus grands challenges actuels, on notera par exemple le « *Blue brain project* » à l'initiative de l'École Polytechnique Fédérale de Lausanne qui vise ni plus ni moins que de simuler numériquement le cerveau humain.

La modélisation et la simulation tiennent une place de plus en plus importante en SHS.

Un exemple intéressant est celui du droit. Dans un ouvrage²⁸ maintenant célèbre Lawrence Lessig écrivait en 1999 : « *Code is law* ». Bien sûr, nous ne pouvons plus ignorer l'importance sur notre société de codes informatiques comme celui qui classe les pages des résultats d'un moteur de recherche (*PageRank*) ou celui du DRM qui nous empêche de prêter à un ami un film numérique que nous venons d'acheter. Mais

28. Lawrence Lessig, *Code and other laws of cyberspace*, New York, Basic books, 1999.

avant même ces codes informatiques, on peut voir le droit comme un ensemble d'instructions qui définissent le fonctionnement d'un système. Si pour des raisons historiques et involontaires, les lois sont imprécises, incomplètes, ambiguës, la pensée informatique peut permettre d'aller vers une plus grande formalisation des lois pour les rendre plus compréhensibles notamment de programmes informatiques. Dans cette direction, des outils informatiques sont mis au service de l'analyse des lois comme par exemple *TheLawFactory.fr*²⁹.

En SHS, les phénomènes sont souvent complexes et ouvrent des challenges pour la modélisation mathématique/informatique, voire la simulation numérique. La sociologie est en particulier un candidat évident. Il est possible de s'appuyer sur la modélisation (extrêmement simplifiée) des comportements d'un très grand nombre d'acteurs (agent dans une terminologie informatique populaire) et de leurs interactions avec leur environnement. La puissance de calcul de clusters d'ordinateurs permet ensuite de réaliser des simulations. Avec un millier de machines simulant chacune des milliers d'individus, il devient par exemple possible de simuler le comportement de population de millions de personnes. La comparaison des résultats avec la réalité permet de « paramétrer » le modèle, voire de le modifier, pour mieux coller à la réalité observée. Une fois le modèle mis au point (« tuné »), rien n'interdit les prédictions. Nous sommes dans des processus très semblables à ce qui se fait dans des domaines scientifiques comme la météo.

Nous retrouvons dans ce contexte par exemple l'étude³⁰ de Paola Tubaro et Antonio Casilli sur les émeutes de Londres. Ils ont cherché à savoir si la censure des médias sociaux, solution proposée par David Cameron pour calmer la situation, avait un effet sur le développement d'émeutes. A l'aide d'une simulation numérique, ils ont montré que la censure participait à augmenter le niveau général de violence.

La tâche paraît plus ardue dans d'autres disciplines des sciences humaines ne serait-ce que par la difficulté de développer des modèles assez formels pour se prêter à la simulation. On notera pourtant un ouvrage récent comme *Modeling complexity in the humanities and social sciences*³¹ avec des sujets comme le développement et la transmission du langage ou la propagation des croyances, idées et idéologies.

29. TheLawFactory.fr est une application Web en logiciel libre qui permet de visualiser des informations disponibles sur 290 projets de loi promulgués entre 2010 et 2014.

30. Antonio A. Casilli and Paola Tubaro, « Social Media Censorship in Times of Political Unrest-A Social Simulation Experiment with the UK Riots », *Bulletin of Sociological Methodology / Bulletin de Méthodologie Sociologique*, 2012, 115, pp. 5-20.

31. Paul A. Youngman and Mirsad Hadzikadic, *Complexity and the Human Experience : Modeling Complexity in the Humanities and Social Sciences*, New York, Taylor and Francis, 2014.

L'archivage

L'archivage est devenu essentiellement numérique. On peut mentionner par exemple Europeana³², une bibliothèque numérique européenne lancée en novembre 2008 par la Commission européenne qui compte déjà plus de 26 millions d'objets numériques, textes, images, vidéos, fin 2013. Les États européens (à travers leurs bibliothèques nationales, leurs services d'archivages, leurs musées, etc.) numérisent leurs contenus pour assurer leur conservation, et les mettent en commun. En partant de telles initiatives, nous pouvons imaginer dans moins de 50 ans des historiens qui trouveraient numériquement toutes les informations dont ils ont besoin, passant d'une archive à une autre simplement en changeant de fenêtre sur leur écran.

L'immatérialité de l'information et des données pose aux archivistes et aux bibliothécaires la question de leur conservation et de leur archivage à long terme.

Préservation dans un monde numérique. L'information sur un disque magnétique dure bien moins longtemps que, par exemple, dans un livre imprimé sur papier. Les solutions techniques existent : il faut par exemple répliquer l'information pour se prémunir contre les défaillances matérielles, et il faut la régénérer régulièrement pour lutter contre l'obsolescence rapide des supports d'information numériques. Et puis, il faut se protéger contre l'obsolescence également très rapide des formats informatiques, des logiciels informatiques ainsi que des matériels. L'obsolescence des contenus est souvent accentuée par la disparition des logiciels qui leurs donnaient vie, voire des matériels. C'est le cas notamment pour ce qui est des jeux électroniques. Le projet *Preserving Virtual Worlds* s'est par exemple attaché à archiver des jeux informatiques, des fictions interactives et des communautés virtuelles. Il ne s'agit pas de préserver seulement des formats qui décrivent les contenus, mais aussi des logiciels et les matériels qui permettent de les faire vivre.

La préservation implique de préserver aussi la « qualité » de l'information. Les données numériques comme d'autres données, plus peut-être que d'autres données, sont susceptibles d'être manipulées. L'archivage doit pouvoir garantir leur authenticité.

Culture d'abondance et l'échelle. Avec le numérique, nous sommes passés pour l'information disponible d'une culture de rareté, à une culture d'abondance. Avec la baisse du coût des machines et surtout du stockage, on a cru à la possibilité de tout conserver. Il a fallu déchanter. La préservation de l'information numérique (pour les raisons citées précédemment) revient chère. Pourtant, dans un contexte où nous nous sommes habitués à l'abondance, il faut conserver massivement l'information. Devant le déluge informationnel, choisir ce qu'il faut archiver, voilà un vrai défi pour les humanités numériques. Et il s'agit vraiment de l'automatisation de ce choix parce

32. Europeana, <http://www.europeana.eu/>

que l'archiviste est tout petit devant la masse des données. Sa seule chance de réaliser la tâche qui lui est confiée est d'utiliser massivement des outils numériques.

L'archivage du Web. Il a fallu faire évoluer des centres classiques d'archivage comme les Archives nationales, la BNF et l'INA, et des outils anciens comme le dépôt légal. Un exemple donnera toute la dimension de l'archivage dans le monde numérique : l'archivage du Web. Nous trouvons de plus en plus d'informations sur le Web. Que seront devenues ces informations dans 50 ans quand des chercheurs en auront besoin ? La plus importante initiative pour archiver le Web est venue de la fondation Internet Archive³³ qui archive des pages du Web depuis 1996. A titre d'exemple, la *Library of Congress* a récupéré 170 milliards de tweets juste entre 2006 et 2010 représentant 133.2 téraoctets ! Cela donne une idée des dimensions du problème.

Il faut aussi avoir conscience que, même si l'archivage du Web a fait énormément de progrès, une part importante du Web n'est pas aujourd'hui archivée pour de nombreuses raisons comme la taille du Web, les pages qui changent rapidement de contenu, les pages protégées par mot de passe (comme Facebook) ou du fait des « exclusions³⁴ de robot », et surtout des pages générées par des requêtes (comme eBay).

Conclusion : limites, difficultés et ambitions

Les humanités numériques modifient les modes de travail et de pensée dans les sciences humaines et sociales, non sans difficultés.

Limites de la technique. Si les opportunités sont nombreuses, tout n'est pas possible. Certains problèmes demandent des puissances de calcul dont nous ne disposons pas ou que nous ne sommes pas prêts à mobiliser pour un problème particulier. Les simulations peuvent être trop coûteuses (trop d'acteurs en sociologie), les problèmes trop complexes dans le cadre des connaissances actuelles (analyse sémantique de texte). Les machines gagnent sans cesse en puissance, vitesse des processeurs et des réseaux, taille des mémoires et des disques, parallélisme massif. La qualité et les performances des algorithmes ne cessent de progresser. Reste que les plus grandes avancées en humanités (numériques ou pas) reposent sur l'imagination d'humains qui trouvent la bonne question, énoncent la bonne hypothèse, proposent l'approche révolutionnaire. Les machines et les algorithmes ne sont pas prêts à remplacer ces personnes ; ils sont à leur service, évoluant sans cesse pour repousser le domaine du possible.

33. Internet Archive, <https://archive.org/>

34. Le propriétaire de la page interdit son archivage.

Limites de l'objectivité. Dans le cadre des SHS, il faut aussi savoir accepter les limites de l'objectivité. Les humanités numériques ne peuvent se réduire à des équations ou des algorithmes (les plus beaux soient-ils) et des nombres. Le sujet principal est l'être humain, souvent trop complexe pour être mis en équation ou même en algorithme.

Mutabilité du monde actuel. Une limite aussi est la rapide mutabilité du monde actuel. La culture est devenue numérique et évolue sans cesse ; l'homme évolue également avec le numérique dans sa manière de lire, d'apprendre, etc. Le numérique tient une place essentielle *dans le sujet même* des humanités numériques, comme l'explique Milad Douhei³⁵, en conduisant à la définition de nouvelles humanités. Une difficulté est donc cette fluidité du contexte où à la fois le sujet même et les sciences qui nous permettent de l'étudier changent à l'unisson.

Mutualiser les savoirs. Une conclusion à tirer de cette étude serait peut-être la similarité des idées communes à un spectre très large de disciplines scientifiques (humaine ou pas) et trouvées avec le numérique, par de nouveaux outils, de nouvelles approches, de nouveaux savoirs. C'est au niveau des outils que cette évidence s'impose. On rappellera qu'un ordinateur est « *general purpose* », et que le même système de gestion de bases de données peut servir pour des données en histoire, en astronomie, etc. Une ambition du chercheur en humanités numériques et plus généralement en sciences numériques doit donc être de mutualiser ses réalisations (logicielles, conceptuelles, ou autres) au delà de son propre domaine. Les principes même de la « pensée informatique » (*computational thinking*³⁶) sont généraux et se doivent d'être également « mutualisés ».

Inventer un nouvel humanisme. La culture numérique est fortement imprégnée de valeurs humanistes. Si Internet et le Web sont au départ des techniques et des outils, ils sont inséparables de modèles sociaux et économiques basés sur le partage et l'échange. Nous avons mentionné par exemple Wikipédia et les *Creative commons* dont le caractère humaniste doit être souligné. Il nous semble donc que, par essence, les humanités numériques ne peuvent se résumer au seul développement des connaissances scientifiques. Elles se doivent d'adopter les ambitions humanistes déjà présentes dans la culture numérique. Et par exemple, quelle plus grande ambition humaniste que de mieux diffuser les connaissances et la culture à tous. L'ambition des humanités numériques doit donc bien être de participer à l'invention d'un nouvel humanisme.

Remerciements. Merci à Gilles Dowek, Mathieu Latapy, Pierre-Michel Menger et Marc Tommasi pour leurs critiques de versions préliminaires de ce texte.

35. Milad Doueïhi, *Pour un humanisme numérique*, Paris, Seuil, 2011.

36. Jeannette M. Wing, « Computational thinking », *Communications of the ACM*, 2006, 49.3.



Du datagramme à la gouvernance de l'Internet

Entretien avec Louis Pouzin¹
(réalisé par Claudia Marinica et Marc Shapiro)

Nous avons interviewé [Louis Pouzin](#), célèbre informaticien français, connu entre autres pour son rôle dans la création du réseau [Cyclades](#) et dans l'invention du concept de [datagramme](#), qui est à la base des protocoles de l'Internet d'aujourd'hui.

Louis Pouzin nous a livré, avec franc-parler, une information très riche sur les débuts de la recherche en réseau en France, son évolution, les interférences politiques, la gouvernance de l'Internet, la sécurité et la vulnérabilité des données personnelles.

C'étaient les années 60...

Marc Shapiro : *Bonjour, Louis Pouzin. Nous allons faire une interview pour la revue 1024 de la SIF. Pour les jeunes générations, qui ne les connaissent pas du tout, peut-être pouvez-vous nous raconter, pour commencer, les débuts de l'informatique. Être chercheur en informatique, c'était comment ? Comment vous est venue l'idée de faire des réseaux, de donner des cours d'informatique, alors que rien n'existait avant ?*

Louis Pouzin : Même pas le mot [informatique](#) ! Ma carrière numérique a commencé dans les années 1960 chez la [Compagnie des Machines Bull](#), qui vendait essentiellement à l'époque du matériel à cartes perforées. C'était, à l'époque, le seul concurrent en France d'IBM. C'était le tout début de ce qui n'était pas encore appelé les ordinateurs — on appelait ça des calculateurs électroniques — cela avait consisté à rajouter un tambour à des appareils à cartes perforées.

1. Texte mis en forme par Marc Shapiro. Se référer à la version électronique pour utiliser des [liens cliquables](#) sur quelques termes techniques, sigles ou noms de l'époque.

Ça servait à faire des calculs mais ça ne servait pas à écrire vraiment. On rentrait des paquets de cartes ; le traitement se faisait en partie dans le calculateur électronique, qui allait chercher ses informations et ses programmes sur le tambour, et après il recrachait les résultats sur une imprimante qui était du genre vraiment très ASCII, parce que, par exemple, le 1 et le I, le O et le 0 c'était presque pareil (cela permettait d'économiser des positions d'impression).

Assez rapidement, on a vu arriver sur le marché des IBM 650, qui avaient aussi un tambour.

Claudia Marinica : *Les gens aujourd'hui ne savent peut-être pas ce qu'est un tambour.*

L.P. : Un tambour, c'est un objet qui a une forme de tambour cylindrique (qui peut être vertical ou horizontal). C'est magnétisé, et on écrit dessus exactement comme sur une bande magnétique, seulement le temps d'accès est plus rapide. En fait, il n'y a plus de tambour maintenant ; les derniers, c'étaient des énormes bazars, qui étaient au moins de la largeur peut-être de cette salle². C'était métallique mais magnétisé, et ça tournait très vite — très joli à voir. C'était Remington Rand qui faisait ça. La SNCF était un client, Air France également.

M.S. : *C'est l'équivalent d'un disque pour les gens d'aujourd'hui.*

L.P. : Oui, mais plus rapide et de plus grande capacité que les disques de l'époque. Évidemment, la technique de stockage a évolué, parce que la densité a augmenté beaucoup. Après, on est passé aux bandes magnétiques ; elles n'étaient pas très fiables, il fallait toujours écrire en double.

Les grosses IBM 700-et-quelques (703, 705) étaient destinées au calcul scientifique, mais elles ont poussé la Compagnie des Machines Bull à réaliser un matériel qui pouvait concurrencer, peut-être pas en vitesse, mais en capacité de traitement non numérique, plutôt de type administratif. C'était le Gamma-60, qui est apparu en 1960-61.

Il a eu pas mal de succès en France. Beaucoup de grandes sociétés en ont acheté, notamment la SNCF, l'Union (assurance), EDF, les grosses sociétés administratives qui traitaient beaucoup de papier, les assureurs, les banques, etc.

M.S. : *Votre rôle là-dedans ?*

L.P. : Eh bien, j'étais chez Bull, et j'avais une bande d'ingénieurs. Au début c'étaient surtout des gens d'origine technique, qui avaient pour ainsi dire appris les cartes perforées, simplement en les faisant marcher. Mais, à partir du moment où Bull a commencé à faire du calcul scientifique, et puis ensuite a commencé à développer le Gamma-60, on a embauché beaucoup d'ingénieurs, des ingénieurs de grandes

2. Peut-être dix mètres sur quatre, [N.D.L.R.].

écoles. Alors, j'avais 30 ou 40 ingénieurs de grandes écoles, qui programmaient le Gamma-60, alors c'est là où j'ai découvert que...

J'étais dans un service qui faisait l'interface entre le commercial et les Études — ce qu'on appelait les Études, c'est-à-dire les services techniques — qui faisait du développement de machines, essentiellement du matériel, parce qu'à l'époque la programmation était encore une activité en soi, assimilée au support technique.

La programmation, au début, c'était piquer des fils sur des tableaux de connexions de machines à cartes perforées. Après, c'est devenu un langage, des langages assez simplistes pour écrire des procédures de traitement ; essentiellement, ce qu'on appelait à l'époque des *control cards*, des cartes de contrôle. Du côté scientifique on programait dans des langages pré-Fortran, des trucs « faits maison ».

On commençait donc à rentrer dans la période ingénieurs, pour la programmation. C'est à ce moment-là, je crois, que le terme ordinateur a été inventé, de même que le terme informatique. Le terme informatique a été introduit en France par [Philippe Dreyfus](#), qui s'était inspiré d'une société en Californie, qui s'appelait Informatics. Quant à ordinateur, c'est IBM qui a lancé un petit concours pour savoir quel terme utiliser. Ordinateur c'est pas mal, ça a marché tout de suite. Ce n'est pas aussi ronflant qu'en espagnol ; *computador* ça fait un peu toréador ! [Rires] Et c'était mieux que *computer* finalement.

La Compagnie des Machines Bull avait ses grandes ambitions de Gamma-60. Mais elle a eu beaucoup de difficultés techniques, parce qu'une fois installés, il y avait beaucoup de pannes, la programmation n'était pas très simple. C'était une machine très intéressante, mais un peu trop théorique. La machine prenait deux salles, avec la réfrigération et tout ça.

M.S. : *Pourquoi la programmation était-elle plus difficile qu'avec l'IBM ?*

L.P. : D'abord parce que c'était une machine faisant du parallélisme. Ce n'était pas encore courant à l'époque. Il y avait des systèmes de synchronisation, des balises, ce qu'on pourrait appeler maintenant des sémaphores, qui permettaient de lancer des instances de programmes, qui pouvaient se courir après, et s'arrêter si ce n'était pas prêt. C'était très sophistiqué.

M.S. : *Tout ça en hardware ?*

L.P. : Tout ça en *hardware*, oui. Et donc, ce n'était pas simple à utiliser, mais c'était très excitant intellectuellement.

L'inconvénient, c'est qu'il n'y avait pas de contrôle d'erreurs *hardware*. Quand un composant ne marchait pas, il fallait beaucoup de temps de dépanneur pour trouver. Il y avait une pénurie considérable de temps de mise au point : le programmeur prenait toute la machine pour lui, donc il y avait une pénurie constante de temps machine pour mettre au point les programmes.

Ce qui fait qu'il y a eu des difficultés avec les clients, qui n'étaient pas contents. Parallèlement, IBM avait sorti une machine de type comptable, le RAMAC je crois. C'était essentiellement un énorme disque avec plein de têtes, qui permettait donc de stocker beaucoup d'informations.

Elle n'était pas très rapide, mais l'idée a beaucoup plu à la clientèle. Mais ils se sont aperçus, après, que ce n'était pas si bon que ça, parce qu'il n'y avait pas d'entrées-sorties rapides. Donc, quand il fallait charger ou décharger le RAMAC, cela prenait du temps. Il n'y avait pas de contrôle d'erreurs non plus ; donc, la machine n'était pas suffisamment fiable.

C'est après qu'IBM a amené je ne sais plus quelle machine, un précurseur de la 360. Ça, ça a un petit peu poussé Bull dans ses retranchements. Ils avaient préparé une gamme de matériels à cartes perforées, qui s'appelaient la Série 300. C'était certainement aussi intelligent, et également totalement en *hardware*, des moyens de faire des séquences de programmes qui pouvaient se bloquer en attendant la suite. C'était très intelligent mais c'était du matériel à cartes perforées avec des relais, des ficelles, etc. En face d'IBM, qui avait du matériel électronique, ça ne pouvait pas tenir. Donc, Bull est allé chercher un comparse aux États-Unis, pour avoir une machine plus moderne. Il s'agissait du RCA 301, qui a été rebaptisé en France le Gamma-30.

Ça marchait pas mal, finalement. Mais les accords avec RCA ont introduit en France une nouvelle culture. Il a fallu envoyer les ingénieurs aux États-Unis pour se former, étudier la documentation, passer par les contrôles américains, etc. Comme RCA travaillait aussi pour la défense, il y avait des docs, ou bien des parties des bâtiments, qui n'étaient pas accessibles, enfin, c'était la bureaucratie américaine. Donc, ils ont ramené un peu de cette culture aussi. Et puis, ils parlaient anglais, et il n'y avait pas beaucoup de gens à l'époque qui parlaient anglais.

Y compris moi, je ne parlais pas anglais ! Et je ne connaissais pas la programmation ! Je faisais la programmation des machines à cartes perforées, bien sûr, mais je ne connaissais pas la programmation sur machine électronique. Je me suis aperçu que si je n'apprenais pas à programmer et à parler anglais, je n'avais aucune carrière possible, aucun avenir dans ce métier.

M.S. : *Cela voulait dire quoi concrètement à l'époque, apprendre à programmer ?*

L.P. : Cela voulait dire prendre son crayon et regarder dans un manuel, en général pas très pédagogique, quels sont les jeux d'instructions qu'on peut utiliser, s'en servir, chercher pourquoi cela ne marche pas...

M.S. : *C'était de l'assembleur ?*

L.P. : C'était de l'assembleur, c'était même du code machine.

M.S. : *C'est de l'assembleur, pas de structures, il n'y a pas de théorie.*

L.P. : Absolument pas, il n'y avait aucune formation à l'époque pour apprendre à programmer. C'était à l'usage.

M.S. : *Donc, langage machine, ça veut dire aux clés au tableau de bord³ ?*

L.P. : Non, c'était en général des feuilles de programme, avec des petites cases où on mettait des lettres et des chiffres. Ensuite on perforait sur des cartes. Cela constituait le programme, directement tel qu'il était exploité par le calculateur.

C'est à cette époque qu'il y a eu des premiers langages de programmation de type scientifique. La programmation de type administratif c'était déjà un peu plus malin, parce qu'il faut faire des recherches de caractères, de *strings*, etc. Alors qu'eux [les scientifiques], ils travaillaient essentiellement sur des données numériques. C'était un peu du genre Basic, en plus simpliste.

Le MIT et Multics

M.S. : *Là, on est en quelle année ?*

L.P. : C'était au début des années 60, 62, 63, par là. Alors, c'est là que j'ai demandé à mon chef de voir s'il n'y avait pas moyen d'aller aux États-Unis.

M.S. : *Vous avez quel âge à cette époque ?*

L.P. : À cette époque, en 61, j'avais trente ans. J'avais un congé de longue durée avec Bull, qui simplement continuait à payer mes charges sociales, ce qui me donnait un filet de sécurité en cas de problème de santé. Sinon, j'étais embauché comme programmeur système au MIT, dans le premier grand système de temps partagé qui s'appelait CTSS, sur IBM 704, qui est devenu après le 7090. En fait, c'est grâce aux relations de Philippe Dreyfus, qui m'avait présenté à [Fernando Corbató](#), qui était à l'époque directeur-adjoint, c'est-à-dire en pratique directeur opérationnel, du centre de calcul du MIT. C'était l'époque où on commençait à se dire que les ordinateurs utilisés pour faire de la mise au point un par un, c'était du temps perdu, qu'il fallait trouver un moyen de partager la ressource chère, qui à l'époque était le calculateur, de manière à pouvoir faire de la mise au point en parallèle.

On n'avait pas encore l'idée de faire du parallélisme de traitement, on ne faisait que du *batch*. On se faisait un paquet de cartes de cartes contrôle et d'assembleur. On travaillait en assembleur au MIT, c'était un peu plus évolué, et même en macro-assembleur.

Je suis resté là deux ans et demi. On m'a fait faire des petits programmes pour commencer, pour voir ce que je pouvais faire. Et puis, il n'y a rien de très sorcier là-dedans, tout ce qui est la programmation, on apprend très vite.

3. On pouvait programmer en binaire directement par des interrupteurs situés sur le tableau de contrôle de l'ordinateur, [N.D.L.R.].

Il fallait aussi apprendre l'anglais, et puis les mœurs américaines, parce que j'avais une femme et deux enfants.

Alors donc, c'est là que j'ai connu la vie américaine, qui est sympathique, c'est sûr, mais qui montre que ce n'est pas le même genre de culture que nous, ce n'est pas le même genre d'esprit, ce n'est pas le même humour. On peut dire que c'est quand même très conventionnel. Ils sont très ouverts, il n'y a pas de problème, beaucoup plus ouverts que les français dans une certaine mesure ; c'est que, dès qu'on les connaît, ils reviennent prendre un pot, etc. Ça marche très bien, mais finalement au bout d'un an ou deux, c'est la même chose, ça n'évolue pas beaucoup. C'est très conformiste.

Donc, je n'avais jamais eu l'intention de rester aux États-Unis. J'avais juste l'intention de me mettre au niveau, professionnel, informatique et linguistique. Au bout de deux ans et demi, je suis rentré.

M.S. : À l'époque, au MIT, il y avait déjà un département d'informatique ? Il y avait déjà de l'enseignement en informatique, de la recherche en informatique ?

L.P. : Il y avait des chercheurs en informatique. Ça n'existait pas, en France, la recherche informatique. Si, les mathéux : programmer des algorithmes, puis trouver éventuellement les moyens de les simplifier. Il commençait à y avoir des cours de recherche opérationnelle, essentiellement sur les méthodes mathématiques pour programmer des séries, des convergences, accélérer le calcul. C'était essentiellement de la traduction, en Fortran, d'algorithmes qui étaient de type mathématique. Les traitements administratifs, eux, ne représentaient que la traduction mécanique de ce que l'on faisait à la main.

Bull et le Gamma-60

L.P. : Bull a commencé par les cartes perforées, les machines de comptabilité mécaniques. IBM avait déjà pris une certaine ampleur, c'était vraiment le niveau de la tabulatrice, avec les poinçonneuses à cartes. Tout le monde était à peu près du même niveau, mais il n'y avait pas beaucoup de fabricants à l'époque. Il y avait essentiellement Bull, IBM, et un troisième concurrent qui avait des trous ronds ; je crois que c'était SAMAS ou un truc comme ça — il n'a pas tenu la route, mais il existait à l'époque.

Après, l'envolée est venue essentiellement des États-Unis. Il y avait IBM, mais il y avait aussi Burroughs. et puis Remington.

Avec les calculateurs, Remington a abandonné, Burroughs a continué. Il y avait le Burroughs 5000, qui est une machine très intéressante. Dans les années 70, c'était encore une machine assez respectable, très rapide, avec aussi pas mal d'instructions de programmation assez habiles. Je crois qu'il faisait déjà de la mémoire virtuelle à l'époque.

Bull, de son côté, a essayé de rattraper un petit peu le train avec du matériel importé, le Gamma-30 qui était un RCA 301. Ils ont passé des accords avec Olivetti, et ont développé une gamme de machines en commun. Les Italiens faisaient le bas de gamme, et Bull le haut de gamme. Mais évidemment les Italiens sont assez malins, et ils sont arrivés à ce que le bas de gamme devienne assez haut de gamme. Il y avait un peu de gué-guerre de concurrence.

Il y avait aussi les accords avec les États-Unis qui duraient. Les accords avec RCA sont tombés à l'eau, parce qu'ils n'avaient pas suffisamment accès aux données techniques, à cause des autorisations de la Défense. Donc, ils sont passés à General Electric. Il y a eu d'abord un GE 300, puis un GE 600 qui était une machine, disons de milieu de gamme, qui n'a pas été extrêmement répandue en Europe, sauf comme machine de temps partagé.

Au Dartmouth College, aux États-Unis, ils avaient développé le Basic sur une machine GE. En fait, c'était une machine qui n'avait pas beaucoup d'avenir, ni en calcul ni en traitement administratif, mais qui, comme machine d'entraînement à la programmation en Basic, marchait pas mal.

Finalement, c'est IBM qui a gagné le marché.

M.S. : *Donc, les gens qui voulaient apprendre l'informatique, à l'époque, c'était le Basic ?*

L.P. : Alors, à l'époque, les premières formations informatiques, en France, c'était des ingénieurs de Bull qui allaient faire des cours à l'université. Ils faisaient aussi des cours internes chez les clients. Mais une formation de type pédagogique, c'était les ingénieurs qui avaient appris à programmer par la pratique, et qui faisaient des cours à l'université.

Ça a été relayé ensuite par les mathématiciens, qui ont commencé à faire des cours de langage. Il y avait Fortran. C'était aussi le début d'Algol, qui excitait pas mal les esprits, parce qu'intellectuellement c'était nettement plus évolué. Par exemple, Henri Leroy, mort en janvier 2014, qui était ingénieur chez Bull à l'époque, avait développé un compilateur Algol pour le Gamma-60.

M.S. : *Là, on est en quelle année ?*

L.P. : Le Gamma-60, il a dû faire ça dans la fin des années 60, probablement 63, 64, 65.

Toutes ces informations, on les retrouve sans trop de mal sur le site des [anciens de Bull](#). Là, ils donnent tous les détails sur l'évolution, il était tenu par [Jean Bellec](#), mort aussi en mai 2012. Il se donnait beaucoup de mal pour maintenir à jour toute l'évolution de la compagnie, aussi bien en matériel, que vis-à-vis de la concurrence, les produits, etc. Il est très riche ce site. Là, on pourra voir toute la filiation des machines.

À mon retour des États-Unis, c'était le début des accords avec General Electric. Il y avait chez Bull un directeur américain, qui avait pris le contrôle, et pour lui, ce qu'il fallait, c'était vendre. Ils avaient la culture du retour sur investissement. C'était l'époque où les gens découvraient le temps partagé ; alors, comme je parlais anglais et que je connaissais le temps partagé, on me promenait comme un chien savant dans les différents pays, pour aller faire des exposés aux grandes sociétés susceptibles de prendre du GE-600. Ça a duré à peu près un an et demi.

La Météorologie Nationale et l'informatique à l'université

L.P. : Là c'était en 1965 ou 67 à peu près. Alors, au bout d'un an et demi, je me suis dit, je connais tous les aéroports, maintenant je peux peut-être essayer de faire quelque chose de plus intéressant. C'est bien tombé, parce qu'il y a eu justement la météorologie française, avenue Rapp, qui avait commandé un GE-600 pour faire de la prévision.

Ça, c'était tout à fait spécial, parce qu'il fallait recevoir des données qui venaient par télétype, faire des traitements de type mathématique, basés sur des équations différentielles qui font évoluer l'atmosphère, la température, la pression, l'humidité, etc., la direction du vent, et en tirer des cartes de prévision. Les premières tentatives des Américains, il fallait trois jours pour faire la prévision du lendemain, donc c'était pas très utile ! Mais après avoir amélioré la vitesse des ordinateurs, et les algorithmes de traitement, ils sont arrivés à tenir à peu près dans la demi-journée. Donc la Météorologie voulait faire quelque chose comme ça.

Avant de commander un GE-600, ils s'étaient amusés pendant quelques années sur une machine qui s'appelait KL. Ces machines n'avaient strictement aucun support technique, aucun langage de programmation, aucun périphérique, donc ils ont perdu leur temps, mais ils ont pris goût à l'informatique. Quand ils ont commandé un GE-600, ils se sont dit : « on va voir ce que savent faire les gens de Bull ». Alors ils [Bull] m'ont collé sur ce projet, en me disant, voilà c'est un projet très intéressant, c'était vrai d'ailleurs. Alors je me suis installé à la Météo. J'ai recruté quelques personnes.

J'ai recruté des gens plus jeunes et les ai formés. J'ai pris quelques personnes de la Météo, et un « coopérant »⁴.

C'était un français, Michel Rocher, que j'avais connu au MIT en 1964 ; je me suis arrangé pour qu'il fasse son service [militaire] à la Météo.

M.S. : *Alors là, c'est pareil, la même question, la formation : comment vous les formiez, ils étaient formés sur le tas ?*

4. Le Service Militaire civil, qu'on pouvait faire à l'étranger au titre de la coopération (d'où le nom de coopérants), ou parfois dans des organismes d'état, [N.D.L.R.].

L.P. : Mais ils avaient déjà un *background* ! Ce garçon, il avait été au MIT, il avait appris à programmer là-bas. Après, il avait développé un système temps réel pour une raffinerie. Donc, il avait déjà un bon *background* de programmation, mais sans jamais avoir suivi de formation, c'était uniquement à l'apprentissage. Les gens de la Météo, c'était pareil, on leur donnait une feuille de programmation et un crayon, et puis le manuel de programmation, c'est tout.

Évidemment, on discutait ; je leur ai fait des petits cours de programmation sur l'assembleur, le macro-assembleur, et quand ils avaient des problèmes de mise au point, je leur montrais comment on arrivait à trouver les bugs. En fait, c'est une formation interne, mais au fil du gaz, au fur et à mesure qu'on vivait avec.

C.M. : *Et est-ce que c'était facile d'apprendre ça ? Avaient-ils des difficultés à comprendre l'assembleur ?*

L.P. : Je crois que tous les gens qui ont un esprit logique apprennent vite. Et il y a des gens qui n'ont pas l'esprit logique, qui sont toujours incapables de comprendre quelle peut être la cause de ceci par exemple ; il faut un peu d'imagination. On les trie, hein, c'est-à-dire qu'au bout de quelques mois, on a tout de suite compris qui est capable de programmer, et qui n'est pas capable.

Donc, j'ai commencé le projet de développer le système sur la météo. J'ai écrit les spécifications, j'ai convaincu la Météo qu'avoir des disques, ce n'était pas suffisant pour la rapidité de calcul, qu'il valait mieux avoir un tambour ; ils ont commandé un tambour, très bien...

M.S. : *Et les spécifications, c'est en langage naturel ?*

L.P. : C'était du baratin, les spécifications. Les modules, qu'est-ce qu'ils font, les interfaces, etc.

M.S. : *Il y a quand même une structure ?*

L.P. : Oui bien sûr. Il y avait essentiellement tous les éléments, et d'ailleurs, ça m'a permis de convaincre la Météo que vous pouvez faire du design très ambitieux, parce que ça ne coûte pas cher le design. Ça coûte du temps d'ingénieur, mais après on n'est pas obligé de tout implémenter. On peut implémenter uniquement la partie dont on a besoin immédiatement, mais au moins, on a tout le détail de ce qu'on pourrait ensuite faire, si on veut agrandir le système. Je les ai convaincus qu'agrandir un système, quand on a un plan de bataille, c'est beaucoup moins cher que de bricoler un système qui n'était pas conçu pour être plus grand. Ça, ils ont très bien compris. Donc c'est un petit peu ce qu'ils disaient : « oui, c'est un peu du design à la Pouzin », et puis après on en implémente un petit bout, et puis après... ça va plus vite au total.

Alors c'était aussi l'époque 1967, c'était l'époque de la révolution en France, 67, 68. Alors, c'est là que je me suis trouvé impliqué dans différents comités, l'Institut

de Programmation⁵, c'était [Jacques] [Arsac](#) qui en était le directeur à l'époque. C'est là que j'ai connu un petit peu des élèves en programmation, parce qu'Arsac, c'était l'un des premiers qui enseignait la programmation en France.

M.S. : *Ça consistait en quoi, enseigner la programmation ?*

L.P. : Un peu de théorie des langages, la compilation, les bibliothèques, les modules, appels-retours, tous les trucs de base, de constitution de bibliothèque, et puis, éventuellement, un peu de moyens de mise au point, encore que ce n'était pas très développé à l'époque, et puis beaucoup d'exercices.

Ils m'ont demandé aussi de faire des cours à Paris-VI. Alors, je leur ai fait des cours système. Personne ne leur avait fait des cours système avant. Je leur expliquais les principes de fonctionnement, les contextes, vous récupérez les contextes, vous faites travailler en parallèle, des trucs comme ça, et puis les moyens de synchronisation, le principe des piles.

C'était aussi apprendre le vocabulaire. Je leur donnais comme exemple des machines existantes. À l'époque, c'était le Control Data, et puis aussi le Burroughs. Cela leur permettait de comprendre un petit peu comment, dans la pratique, on retrouvait ces fonctions.

M.S. : *Il y avait déjà un mode privilégié, un mode utilisateur ?*

L.P. : Oui, bien sûr, le mode maître, le mode esclave, il y avait de la mémoire virtuelle. C'était bien connu dans le milieu scientifique, mais peu répandu, mal connu dans le milieu des programmeurs, et ce n'était pas disponible sur les petites machines, donc il fallait une certaine imagination.

Le fait, par exemple, qu'un espace d'adresses peut changer de place, qu'il peut y avoir deux espaces d'adresses qui se recouvrent, pour eux, était tout à fait inconnu. Et puis, aussi, des méthodes pratiques, avoir des noms de symboles, des *goto* dont on peut retrouver ultérieurement la signification : il faut que le gars qui va passer derrière pour modifier le programme, il ait néanmoins des moyens, pour voir comment ça marche. Donc être assez abondant en commentaires, mettre des noms qui ont un sens et non pas *a, b, c, x, y, z*. Si ce n'est pas prononçable, ce n'est pas un bon identifiant.

C'était un mélange de pratique et de petite théorie. La théorie n'était pas très évoluée. Les amphithéâtres étaient pleins, à Paris-VI, à l'époque, parce qu'ils n'avaient pas d'autre endroit pour apprendre ces choses-là.

M.S. : *Combien d'étudiants dans un amphithéâtre ?*

L.P. : Je ne sais pas, 250, les amphithéâtres étaient pleins. Ils n'étaient peut-être pas tous réellement étudiants, mais ils venaient assister aux cours.

5. À Jussieu, universités Paris-VI et Paris-VII, devenues depuis Université Pierre-et-Marie-Curie et Denis-Diderot respectivement, [N.D.L.R.].

Ils passaient le DEA ⁶. Je corrigeais cinq ou six copies et, à partir de là, ils faisaient une sélection de copies entre les bonnes et les mauvaises, et je faisais une correction d'un échantillon, et puis après les assistants corrigeaient le reste. Comme ça, on arrivait à alimenter une formation à peu près correcte. Ça marchait bien, les gens découvraient ça. Surtout qu'à l'époque il y avait la révolution de 68. Il y avait une assez bonne disposition des gens au changement, donc apprendre des choses nouvelles, qui n'étaient pas trop classiques. Ils n'avaient pas encore usé les formateurs.

M.S. : *Et ces 250 étudiants, avaient-ils des ordinateurs ?*

L.P. : Non pas du tout. Ils travaillaient en *batch* sur les machines du centre de calcul de Paris-VI, ou sur celles de leurs employeurs. Le PC n'existait pas à l'époque. Il y avait les premiers ordinateurs personnels qui existaient. Il y avait très peu de gens qui en avaient à l'époque.

Donc ça c'est l'époque 1970, etc., et c'est l'époque où je faisais le système météo. La Météo me louait à une société de service, la Sacs, dirigée par Jean Carteron, un ancien ingénieur qui dirigeait tout ce qui était calcul à l'EDF. Il n'avait pas eu de chance, parce qu'il avait commandé un Gamma-60, et comme ça n'avait pas très bien marché, il avait un peu perdu de sa réputation. Alors il avait créé la Sacs, qui marchait bien. Comme je le connaissais bien, il a obtenu avec la Météo le contrat de pouvoir me louer à la Météo. La Météo était prête à me louer n'importe où ; toutes les boîtes de service voulaient m'embaucher pour avoir le contrat de la météo !

À ce moment-là un autre changement s'est produit : General Electric a abandonné le marché des calculateurs, des ordinateurs. Alors ils ont refilé leurs billes, ils ont vendu leur département informatique à Honeywell. Alors Bull-General Electric est devenu Honeywell-Bull, et la Météo qui avait commandé un GE-600 s'est retrouvée le bec dans l'eau.

Qu'est-ce qu'on fait ? Ils ont passé tout en revue. Ils étaient fanatiques d'avoir un matériel qui calcule très vite, et comme ils s'étaient habitués à un système qui n'avait pratiquement pas de système de programmation, ça ne les gênait pas du tout de prendre un Control Data. Celui-ci n'avait également pratiquement pas le système de programmation : uniquement l'assembleur, rien pour faire la mise au point, très peu de périphériques. Mais ça marchait très vite, et avec des mots de 60 bits, ça faisait beaucoup de décimales. Donc ils ont commandé un Control Data 6400. C'était une machine assez remarquable, avec un processeur central qui allait très vite, et puis huit ou dix processeurs périphériques, qui étaient en fait des processeurs virtuels ; chacun prenait des instructions, et faisait les entrées-sorties. Le système était

6. Le Diplôme d'Études Approfondies, un an d'études à Bac+5, était l'équivalent du Master actuel. Le DEA faisait suite à la licence (Bac+2) et la maîtrise (Bac+4), [N.D.L.R.].

essentiellement réalisé en programmation des calculateurs périphériques. Ils avaient aussi leur assembleur, qui était différent de celui du calculateur central [Rires].

Alors, qu'est-ce qu'on faisait à la Météo ? Ils voulaient me garder bien sûr, parce qu'ils avaient déjà pris goût au système que je leur proposais ; j'ai quitté Bull pour passer à la Sacs et à la Météo. Il leur fallait un *operating system* solide pour faire des tâches en parallèle, des trucs comme ça, qui n'existaient absolument pas dans le Control Data. Le Control Data c'était un *batch* extrêmement primitif. Il n'y avait aucun fichier permanent, ils s'effaçaient toujours à la fin du travail !

Il fallait pratiquement tout faire. Alors là, j'ai récupéré un gars de Control Data, qui lui aussi avait envie de s'émoustiller un peu les méninges. Il a développé une partie du système de fichiers. J'ai repris le coopérant, que j'ai mis sur le système. Deux gars de l'Institut de Programmation avaient été embauchés à la Météo, mais dans un département qui ne leur convenait pas ; alors on les a transférés dans la partie informatique. Et puis un autre gars, de chez Bull aussi. Alors ça, ça faisait une assez bonne équipe, on était six ou huit.

Moi, j'ai repris la partie centrale, parce qu'il fallait quand même quelque chose pour diriger, organiser tout ça. Les calculateurs périphériques n'étaient pas très rapides, programmation assez primitive ; et puis aussi il fallait les synchroniser, sinon il n'y en a qu'un qui fait le boulot, et puis s'il a des ennuis physiques, etc.

Tout ce qui était *dispatching*, gestion des tâches, c'était en central, et les [processeurs] périphériques ne servaient qu'à faire des synchronisations, à traquer les entrées-sorties, les bandes magnétiques, les cartes, etc., les imprimantes.

En deux ans on a fait le système, la Météo était satisfaite. On avait des queues de données qui pouvaient passer d'une tâche à l'autre. Et par ailleurs la synchronisation en central permettait de gérer parallèlement différentes tâches de fond. Avec ça, ils pouvaient faire passer du *batch*, du temps partagé, et recevoir des données de télétype : parce que les données météorologiques elles voyagent d'ouest en est, des États-Unis, du Canada ; et après traitement elles sont refilees à l'est vers les autres. Tout cela marche en parallèle. Et puis l'impression des cartes de prévisions.

Chez Simca : comment faire la nique à IBM

L.P. : Tout ça marchait, alors donc ils n'avaient plus de raisons de me payer. [Rires] Ils ont exploité ce système pendant 15 ans. Alors j'ai regardé un peu les petites annonces. Les techniques de programmation de base, c'était un peu passé de mode. On disait qu'il fallait maintenant s'occuper des applications, ça devenait important ; c'était vrai d'ailleurs. J'ai vu une petite annonce à laquelle j'ai répondu. C'était Simca qui embauchait. Simca avait déjà été saisie financièrement par Chrysler.

C'est un américain de Chrysler qui m'a embauché. Je crois que c'est uniquement parce que j'avais été au MIT.

C.M. : *C'était en quelle année ?*

L.P. : Simca c'était en 1969, parce qu'en 68 j'étais à la Météo, et en 71 je suis passé chez Cyclades.

Ils avaient des ordinateurs IBM. IBM étaient les maîtres là-dedans. Ce sont eux qui décidaient quand il fallait acheter quelque chose. Tout le monde suivait les oracles IBM.

La plupart des gens là-dedans, c'était des vieux de la vieille. Il y avait des clans, parce qu'autrefois cela avait été Ford, ensuite c'était devenu italien, Fiat, ensuite c'est devenu Simca. D'ailleurs le patron initial de Simca c'était un Italien qui était parti de chez Fiat.

À chaque fois il y avait des traditions qui suivaient des groupes de gens : les anciens de Ford, les anciens de Fiat... Pendant que j'y étais, c'est devenu Chrysler France. Alors comme c'est l'Américain qui m'embauchait, c'était la culture qu'il introduisait. La culture de Chrysler, c'était le moins de gens possible. S'il y avait quelqu'un qui n'était pas occupé à 100%, il fallait couper des têtes.

Je gérais essentiellement le département calculateurs. Il y avait deux parties dans ce département : le développement d'applications, les processus qui gèrent la construction des voitures, la gestion des fournisseurs, la gestion des représentants, la gestion des pièces détachées, etc. C'était du *batch*, il n'y avait absolument rien d'automatisé. Il faut dire qu'ils avaient été traumatisés par un système qu'IBM leur avait fait installer, terminaux spécialisés, des transmissions spécialisées, tout un tas de trucs spécialisés, qui n'a jamais marché.

Une des premières choses que j'ai faites, c'est de regarder le taux d'occupation des ordinateurs, parce qu'IBM les facturait au temps, à l'horloge. C'était du matériel loué. Pas exactement le vrai temps : il y avait des conventions, selon lesquelles, si le processeur se mettait à turbiner, alors ça comptait, clic clic, un peu comme un compteur de taxi.

Je me suis dit, là il doit y avoir de l'argent à gagner. J'ai obtenu un logiciel américain, qui permettait de faire des histogrammes de temps passé. On voyait qu'il y avait des gros trous. En faisant des modifications raisonnables, pas trop importantes, dans les programmes d'application, partout où il y avait des boucles, etc., on a accéléré l'exécution, et on a fait baisser d'au moins 30 à 40% la facture. Ça a un peu emmerdé les commerciaux d'IBM évidemment, par contre ça a donné des idées aux gens de Simca. Ils se sont dits « Tiens, on peut faire des choses qu'IBM ne sait pas faire. »

Ma deuxième étape, c'est qu'ils rêvaient d'un système qui leur permettrait de suivre la production presque en temps réel, et non pas à partir des sorties papier, qui arrivent avec un ou deux jours de retard. Alors j'ai proposé un petit système, pas très malin, de consultation de fichiers : ça permet de voir à tout instant le fichier de toute la production, qu'est-ce qui est en cours, ce qui en est à tel stade, quelle quantité, etc.

Alors IBM s'amène, « on a des matériels, comme ci, comme ça ». Moi, je leur dis, mais je ne veux pas de ça, c'est trop cher. La location des modems, au bout de six mois, ça représentait le prix du modem, des trucs aberrants. J'ai dit : on va prendre des télétypes. Le commercial dit : moi, télétype, je ne connais pas. Je savais que le service Bureaux d'IBM avait des télétypes ; je leur ai téléphoné, et j'ai obtenu un numéro de commande. J'ai dit au commercial IBM « Vous me commandez ça ». Il dit « Ah bon, mais il n'y a pas de support technique ». J'ai dit « Mais ça ne fait rien ». [Rires] Après, il fallait des modems, il fallait les commander à la SAT, une société française. Alors là, il y a les gens qui faisaient le téléphone dans Simca, qui me disent « Là il faut faire une étude », etc. Alors bon, combien ça va coûter ? Là, ils facturaient un prix aberrant, qui leur permettait d'améliorer leur budget, c'est tout. Je dis, ben non, pas besoin. Ils disent : « Mais alors, vous en prenez la responsabilité ! ». Oui, oui, je la prends... Ah bon.

Avec IBM il fallait du câblage spécial. Alors on a dit, on va installer des modems SAT, des lignes téléphoniques ordinaires, pas de câblages spéciaux, on a commandé des télétypes, on a mis tout ça en place...

M.S. : Excusez-moi, vous n'êtes pas dans l'usine ? Pourquoi y avait-il besoin de télétypes, de modems et de lignes à distance ?

L.P. : Mais parce qu'il faut bien communiquer avec les terminaux. Les télétypes sont dans les bureaux, mais il faut accéder à l'ordinateur. Il y avait aussi quelques usines décentralisées en France, à La Rochelle.

M.S. : Là vous êtes en train de développer les premiers réseaux.

L.P. : Bien entendu, il n'y avait aucun logiciel IBM pour gérer tout ça. Mais ce qu'on avait découvert, c'est que le logiciel qui permettait de suivre l'activité du processeur, qui était un logiciel fait par une boîte de service, non IBM, eh bien, il passait en mode maître pour accéder à l'horloge.

On s'est dit, ça peut être intéressant. Pour faire le système à temps partagé, on avait aussi besoin de passer en mode maître, de passer le contrôle. En principe c'était parfaitement impossible, mais en utilisant des paramètres qui n'étaient pas dans les bouquins, on pouvait passer en mode maître à partir du mode esclave ; scandaleux n'est-ce pas ? On l'a expliqué aux commerciaux IBM, ils n'étaient pas rassurés du tout. J'avais publié un article pour expliquer comment ça se passait, alors ils ont dit qu'ils allaient me poursuivre... Je savais très bien que c'était du bluff, parce qu'ils ne pouvaient rien faire contre un article où on explique comment ça se passe.

Donc on a mis le système en place. Ça a été marrant, parce que sociologiquement il y a les petits chefs qui voulaient avoir des terminaux, s'ils avaient un terminal ça les montait en grade. Alors, on n'a jamais dit non ; on a simplement fait les statistiques sur l'usage du terminal. Il y avait des terminaux qui n'étaient pas utilisés, alors par

économie on les a enlevés. On leur montrait les statistiques, vous ne vous en servez pas, on l'enlève.

M.S. : *Alors là en fait, vous avez carrément refait un bout du système d'exploitation.*

L.P. : Oui. Un dernier problème c'était les disques. Il y avait déjà sur le marché français des compatibles de la société Amdahl, créée par un ancien d'IBM. On a regardé les performances, les prix, etc., on s'est renseigné auprès des gens qui en avaient déjà commandé ; tout ça paraissait assez viable. Sur nos trois armoires de disques IBM, j'ai dit, on va en garder une pour comparaison, et en remplacer deux par des disques Amdhal. Levée de boucliers bien sûr ! En commençant par les techniciens de la maison ! « Qu'est-ce qui va se passer ? ». En plus, c'était le début de l'été, les vacances, etc.

Je n'ai pas hésité quand même, je l'ai fait, et il y a eu des petites difficultés techniques. Il s'est trouvé qu'un des disques Amdhal n'arrivait pas à obtenir les signaux *hardware*. Alors le gars d'Amdhal, qui connaissait un peu le système, il a interverti un disque Amdhal avec un disque IBM, ce qui faisait que ça marchait. On n'a rien dit à IBM. Ils s'en sont aperçus évidemment : « Vous avez touché à notre matériel ! ». On a dit « Ben oui », mais on n'a pas bougé pour autant. Ce qui fait que les disques ont marché, et heureusement parce que le gars d'Amdhal devait partir en vacances ; il était très inquiet, et nous aussi d'ailleurs.

M.S. : *Je vous interromps, excusez-moi. Au début, vous avez dit que chez Simca vous étiez dans le département des calculateurs, qu'ils faisaient deux choses : développer des applications...*

L.P. : Et l'autre, c'était la gestion du matériel, des ordinateurs, et des programmeurs qui faisaient du *batch*. Mais la programmation d'application, c'est dans une autre section de la compagnie, le département dit d'organisation. On avait le même patron. Ils faisaient des procédures, c'est-à-dire, les modes standards de fonctionnement de la compagnie, et par ailleurs développement, applications, et gestion du parc informatique.

Après tout ça, IBM avait été un petit peu écorné, parce qu'on avait fait largement la preuve qu'on était plus forts qu'eux. Les commerciaux ont perdu beaucoup de leur auréole. Ils avaient un beau bureau qui trônait sur l'usine ; on l'a récupéré pour en faire une bibliothèque, et on les a mis au sous-sol. Évidemment, ils n'étaient pas très heureux.

Ça a changé l'esprit, les techniciens informatiques ont pris l'habitude d'en faire selon leur propre idée, et non pas selon l'idée d'IBM. Ils ont été ramenés au rôle de fournisseurs, et non pas au rôle de maîtres à penser.

L'aventure Cyclades

L.P. : Alors mi-70, je reçois un coup de fil de [François Sallé](#), à l'époque directeur technique, ou quelque chose dans ce goût-là, chez la nouvelle CII. Je reçois un coup de fil le même jour de [Louis Bolliet](#). Ils avaient participé à une réunion d'une commission mise en place par la Délégalation à l'Informatique, c'est-à-dire [Maurice Al-lègre](#). 1968, c'est l'époque où l'on a créé la Délégalation à l'Informatique, on a créé la CII, on a créé l'IRIA⁷. Il y avait une nouvelle configuration française de l'informatique. Ce n'était plus simplement les Américains, mais ils gardaient des accords avec les Américains.

Alors donc on m'a téléphoné pour me dire, la commission en question, elle a fait un tour aux États-Unis, elle a fait un rapport — je n'étais pas au courant parce que j'étais chez Simca à l'époque, j'étais un peu coupé de ce milieu — et finalement on a conclu qu'il fallait faire un réseau en France. Est-ce que ça t'intéresse ? J'ai dit, oui, pourquoi pas !

M.S. : *C'est quand même une décision politique finalement, c'est venu du haut. On a dit aux informaticiens : vous devez faire un réseau.*

L.P. : Mais on n'a pas dit ça à tous les informaticiens. Ils ne l'ont dit à pratiquement personne à l'époque. C'est une décision qui s'est prise au niveau, en effet, politique, mais avec une commission dans laquelle il y avait des universitaires, il y avait le CEA, il y avait la CII donc qui commençait, il y avait des gens genre [\[Henri\] Boucher](#), qui était un ingénieur de l'armement. Il y avait une vingtaine de personnes dans cette commission, ils apportaient chacun un point de vue.

Il y avait la SIA aussi, qui était l'opérateur, pas vraiment clandestin, de calcul de la bombe, parce que les États-Unis avaient refusé de livrer un Control Data au CEA, parce qu'ils ne voulaient pas que ça accélère la bombe française. Donc on a créé la SIA, Société d'Informatique Appliquée. Théoriquement, c'était une société de service, mais qui en fait travaillait avec les calculateurs de Control Data pour le compte du CEA. C'était des secrets de polichinelle, mais ça sauvait la face.

Donc, cette décision s'est prise, on pourrait dire d'une manière élitiste, entre un certain nombre de gens qui étaient censés avoir des idées ou du pouvoir, et après, il n'y avait plus qu'à faire. Il y avait de l'argent, parce que la Délégalation n'en manquait pas.

Les objectifs n'étaient pas toujours très clairs. On m'avait dit, il faut faire un réseau comme l'[Arpanet](#), mais évidemment, pour quoi faire ?

Pour la CII, c'était pour avoir une techno réseau. La CII, qui était toute débutante, était déjà impliquée dans le futur consortium Unidata, avec Plessey, Siemens, Philips

7. Institut de Recherche en Informatique et Automatique ; voir plus loin sur les transformations menant à l'Inria, [N.D.L.R.].

et Olivetti. C'était des poids lourds, et la CII ne pesait pas grand-chose, ni au plan politique, ni au plan technique. Elle avait de bons ingénieurs, mais il n'y avait pas de gamme de produits très impressionnante. Le *time-sharing* étant devenu à la mode, notamment dans les milieux universitaires, il fallait avoir de quoi contrer IBM.

C'était un objectif qu'avaient tous les constructeurs européens. On a dit, eh bien, ça nous permettra de mettre en réseau les différents systèmes de temps partagé d'une part. C'était l'objectif technique industriel.

Et puis il y avait l'objectif pour ainsi dire bidon, administratif. Tout un tas de services de l'administration française voulaient absolument avoir des bases de données. C'était le nouvel Eldorado : on voulait tout mettre dans des bases de données, sans aucune étude d'application ; c'était pour faire comme tout le monde. Évidemment, ils demandaient de l'argent ensuite à la Délégation pour acheter leur truc.

Allègre leur disait : attention, les bases de données, ça coûte cher, et puis il faut quand même que ce soit rentable. L'idée de l'époque, c'était de permettre le partage des données entre les différentes administrations. Chose qui, même aujourd'hui, ne se passe pas encore !

C.M. : *C'est ce que je voulais dire !*

L.P. : Ils sont extrêmement jaloux de leurs données. Mais enfin, c'était l'argument financier, logique, qui permettait de justifier le prix de développement du réseau. Tout le monde n'était pas dupe, mais il n'y avait pas beaucoup de gens qui croyaient vraiment à l'opération. Les administrations elles-mêmes n'y croyaient pas du tout, mais du moment que les Finances étaient d'accord, ça suffisait.

Donc, j'ai eu un budget qui au total devait représenter vingt et quelques millions de francs de l'époque.

M.S. : *Cela veut dire quoi, combien d'ingénieurs... ? Nous on ne se rend pas compte. Les millions de l'époque, qu'est-ce que cela veut dire ?*

L.P. : Disons que ça m'a permis de démarrer. J'ai à nouveau commencé par impliquer des gens que je connaissais, un ingénieur de chez Simca, deux ingénieurs que me prêtait la CII gratuitement, un ingénieur que j'ai embauché, c'était [Hubert Zimmermann](#).

Un autre ingénieur, Jean le Bihan, d'une boîte de services, et [Gérard Le Lann](#), que j'ai récupéré au CERN.

M.S. : *[Michel Gien](#) c'est plus tard ?*

L.P. : Michel Gien était au centre de calcul de l'IRIA. Un an plus tard à peu près, on l'a incorporé dans nos équipes.

Avec ça, je pouvais démarrer, j'avais des mètres carrés à l'IRIA. Je ne pouvais pas dépenser l'argent comme je voulais, bien sûr. Les crédits étaient gérés par le service

financier de l'IRIA. J'ai aussi fait appel à des sociétés de service, pour la programmation du réseau lui-même, des [Mitra 15](#). J'ai dit aux patrons de ces sociétés, la Sacs, la SESA, Cap Sogeti, etc., vous les louez cinq fois leur salaire. Moi je vous les prends à deux fois et demi, mais vous n'avez pas de problème de planning, ils sont occupés tous les jours, on les loge, vous n'avez aucun frais supplémentaire. Après cela, vous les récupérez. Ils étaient tous d'accord.

Au bout d'un an, un an et demi, ils les ont immédiatement recasés sur des projets de réseaux en France. Le transfert de technologie était beaucoup plus rapide que les publications, etc. Je suis sûr que maintenant ça marcherait aussi bien, parce que la formation des ingénieurs aux techniques nouvelles, ce n'est pas facile par l'université. Il faut pratiquer, et puis voir ce qui se passe ailleurs qu'en France.

Le problème c'est que ces ingénieurs n'avaient pas la moindre idée sur les protocoles de communication. Il fallait leur apprendre tout, y compris les procédures de base, synchrone, asynchrone, ils ne savaient pas faire.

M.S. : *Ça faisait quoi... dix personnes ?*

L.P. : Ça a dû faire une douzaine assez vite. La première année, on n'a pratiquement pas programmé. On a d'abord écrit un rapport, parce qu'on nous a dit aux Finances que, s'il n'y avait pas un rapport, le projet ne pouvait pas être présenté. On a écrit deux bouquins, qui décrivaient l'état des lieux, ce que c'était que des réseaux, ce qui existait aux États-Unis, ce qu'on pouvait faire avec. C'est là qu'on a appelé le réseau [Cyclades](#), et le bouquin a été diffusé un peu partout.

On a fait ça pendant les trois premiers mois à peu près. J'ai embauché Jean-Louis Grangé et [Hubert Zimmermann](#), et puis les gars de CII, mais ils n'étaient pas très doués pour écrire, on a fait surtout ça à trois ou quatre. Ça nous a pris en gros jusque vers mars-avril de l'année 1972. Je suis rentré là-dedans fin 71, en octobre ou novembre.

La deuxième étape, ça a été de faire les protocoles. C'était essentiellement Jean-Louis Grangé et Hubert Zimmermann, et puis moi, parce que j'avais quand même déjà pas mal de connaissances sur la question. Je les ai pilotés là-dessus, et ils ont mis au point le protocole de transport, c'est-à-dire le bout en bout.

J'avais déjà pris l'option datagramme, parce que j'avais étudié à fond les expériences menées au National Physical Lab, et je connaissais assez bien le réseau de paquets de l'ARPA. C'était un service à circuit virtuel, mais son fonctionnement à l'intérieur, c'était du datagramme.

Fin 72, on a fait une grande réunion de toutes les universités françaises, et on leur a sorti tout le paquet de specs, en leur disant « Est-ce que vous voulez coopérer ? ».

Ils ne savaient pas quoi dire ! Ils étaient estomaqués ! [Rires] Après ils se sont ressaisis, et petit à petit on a vu qu'ils pouvaient commencer à mettre des chercheurs sur le sujet ; pas refaire les protocoles, mais déjà les assimiler, et voir le genre d'usage qu'ils pourraient en faire, commencer à penser à des applications.

Puisque j'avais de l'argent, je leur passais des contrats de recherche, pour adapter leurs systèmes au protocole Cyclades, pour former des gens, pour des expériences diverses. Il y avait des trucs qui étaient un peu exotiques et pas forcément réalistes, mais de toutes les façons ça les forme, alors allons-y !

Au début, c'était surtout Rennes, Grenoble, Toulouse, et un peu Lyon, qui étaient les premières à coopérer avec nous. Plus tard on a eu Lille, Strasbourg, ainsi de suite.

M.S. : *Vous pouvez nous faire un tableau de ce qu'était la recherche en informatique à l'époque ? Les universités, qu'est-ce qu'elles faisaient ?*

L.P. : Elles faisaient du langage. Arsar commençait à faire un peu de système, mais ce n'était pas encore très convaincant. Beaucoup de labos se faisaient à la main sur des mini-systèmes, ils développaient des petits bidules de *time-sharing* ou de messagerie. C'était soit à base d'un système de temps partagé comme l'IBM 360-65 à Grenoble ; puisqu'il était multitâche, il pouvait faire du temps partagé ; ou bien c'était des mini-ordinateurs, qu'ils utilisaient comme système.

M.S. : *Des Mitra ?*

L.P. : Plutôt des **PDP**. Je sais pas s'il était vraiment disponible, à l'époque, le Mitra. Il l'était pour nous, mais il était vraiment très débutant dans l'industrie. Il était très peu équipé en système de programmation, il n'était pas très connu. Et puis il y avait aussi des machines exotiques, il y avait un truc de Xerox, l'Alto, qui a fini par donner le Macintosh.

C'était un mélange chaotique de machines de toutes sortes à l'époque.

M.S. : *Chez Xerox, si je me souviens bien, il y avait déjà un réseau, un ethernet.*

L.P. : Chez Xerox oui, mais en France l'Ethernet était mal vu, c'était américain, c'était du datagramme, ce n'était pas l'idée des PTT. Il n'a pas pris facilement, l'Ethernet.

M.S. : *Mais ça c'est plus tard.*

L.P. : C'est plus tard, oui, c'est plutôt dans les années 73–74.

Donc, l'intérêt pour les réseaux a démarré peu à peu, l'idée de dire « on va pouvoir interconnecter tout ça ». Ce n'était pas toujours bienvenu, parce que ça apparaissait comme une entreprise étatique, *top-down*. Il y a ceux qui voulaient faire leur réseau régional, un réseau breton, un réseau Aquitaine (pas à Grenoble : à Grenoble, ils ont toujours bien marché dans la combine) ; dire « mais d'abord on va faire notre réseau, on va les interconnecter » ; en fait ça n'a marché dans aucun cas.

À partir du moment où on a eu les specs, on a réalisé les protocoles, on a programmé les Mitra 15 pour faire du datagramme. Fin 73, on a fait une première démonstration d'interconnexion entre l'IRIA et Grenoble, et des protocoles en fonctionnement. On envoyait un job *batch* à Grenoble, qui s'exécutait et qui renvoyait les résultats à l'IRIA.

M.S. : *Physiquement, le réseau c'était des lignes téléphoniques ?*

L.P. : Oui. On n'avait même pas le 9600 [bits/s] à l'époque, ça devait être du 2400, 4800, un truc comme ça ; des modems absolument standard.

M.S. : *Des lignes louées ?*

L.P. : Oui. Les lignes étaient louées gratuitement par les PTT. Heureusement, parce que sinon, le budget lignes aurait pris près de la moitié du budget du projet. Il a fallu pas mal négocier avec les PTT, parce qu'ils voyaient d'un assez mauvais œil cette histoire de réseau lancé par la Délégation à l'Informatique. « Ce n'est quand même pas aux informaticiens de faire des réseaux. » Ça devait être aux PTT de faire les réseaux !

En faisant le tour des popotes, j'ai découvert qu'ils avaient commencé à programmer un PDP-11 comme nœud de réseau pour connecter des terminaux aux ordinateurs. C'était un système de communication par paquets, mais circuit virtuel. C'était l'idée propagée par [Rémi Després](#) qui, lui aussi, avait fait un PhD à Berkeley. Là, il avait acquis un certain *background* en programmation et en gestion de temps partagé, mais aux PTT il est clair que l'idée de datagramme ce n'était pas très acceptable, parce qu'on ne sait pas ce qui se passe, on envoie des trucs, on ne sait pas si cela arrive. Ce n'est pas très propre, il fallait des circuits.

Ils avaient fait ça dans leur coin au [CNET](#). Ils n'en avaient pas parlé parce qu'ils avaient un matériel américain, alors qu'à l'époque la Délégation à l'Informatique imposait presque aux administrations d'avoir du matériel français. Alors, ils ne s'en vantaient pas.

Je connaissais pas mal [Alain Profit](#), qui était responsable au CNET. Il m'a raconté tout ce qu'il faisait sans problèmes ; mais quand je l'ai raconté à Allègre, il n'était pas content du tout ! [Rires]

On avait un bon négociateur, c'était Michel Monpetit, ingénieur de l'Armement à la Délégation à l'Informatique. À l'Armement, on connaît bien les procédures, et les sources de fric, etc. ; ils sont en connexion avec l'industrie, parce que les chars d'assaut, les missiles, il faut quand même l'industrie pour les faire. Donc, il était bien placé, et très bon négociateur. Il a négocié avec les PTT une convention par laquelle on avait les lignes gratuites, et on faisait le réseau en commun.

Au début, j'avais appelé le réseau Mitranet, mais il y a quelqu'un aux Finances qui a dit « Ce n'est pas possible, Mitranet ce n'est pas un nom français ». Bon, on

l'a appelé Cigale ! Pourquoi Cigale ? C'est parce que quand on avait fait la démonstration, fin 73, à tout le ban et l'arrière-ban, les ministres et cetera, on avait mis un haut-parleur sur la ligne, et quand il passait un paquet, ça faisait « creuh creuh ».

Donc, la démonstration étant faite, politiquement on était assurés d'avoir un minimum de tranquillité. C'est là qu'il a fallu donc faire participer les différentes composantes de la recherche française, avec qui on passait des contrats, et puis on voyait bien ce qui se passait après.

Par ailleurs, les industriels, non seulement on leur avait loué des ingénieurs, mais on leur avait passé des contrats, pour faire des petits gadgets intéressants. Par exemple, j'ai commandé à la Steria, une autre société de service, un multiplexeur numérique. À l'époque, quand on avait plusieurs lignes qui arrivaient sur un ordinateur, il fallait, ou bien autant de portes (très mal utilisées, parce que le trafic n'était pas assez important), ou bien un multiplexeur de type téléphonique, très cher. Et il n'y avait pas de logiciel pour les gérer à partir de l'ordinateur, parce que c'était en mode téléphonique, avec les fréquences, etc.

Très vite, on est arrivés à avoir une petite armada de dispositifs qui permettaient, plus facilement, d'entrer-sortir des ordinateurs, de faire des conversions. À peu près à la même époque, [Najah Naffah](#) a développé chez nous le Terminal-Paquet. C'était un terminal IBM 3270 trafiqué, qu'il a reprogrammé au niveau du micro-programme, pour gérer tous les protocoles de réseau jusqu'au niveau transport. Cela servait en même temps de système de mise au point, parce que, comme on pouvait voir les séquences de caractères qui passaient sur la ligne, on pouvait beaucoup plus facilement détecter les problèmes de protocole.

Il fallait des gens qui sachent bien bricoler pour faire ça, avec une bonne connaissance technique, et puis pas mal d'imagination. Ça, on le trouvait à l'époque dans les sociétés de service ; seulement il fallait choisir les bonnes personnes.

Vers l'année 74, on devait être une vingtaine de personnes à l'IRIA. Beaucoup de gens travaillaient avec nous, mais ce n'est pas nous qui les payions directement. C'était, soit des gens des sociétés de service, soit des gens prêtés par différentes sociétés qui travaillaient dans notre équipe.

La rivalité avec X.25 et la fin de Cyclades

L.P. : Mais [Pompidou](#)⁸ est mort, donc, d'où changement de politique : on a coupé tous les crédits de Cyclades. Le problème c'était que CGE était rivale de Thomson. Thomson, traditionnellement, c'était la défense et l'électroménager, mais ils ont voulu se lancer dans le téléphone, chasse gardée de la CGE. La CGE avait fortement financé l'élection de [Giscard](#)⁹, et il y a un truc qui ne plaisait pas du tout à

8. [Georges Pompidou](#), président de la République, [N.D.L.R.].

9. [Valéry Giscard d'Estaing](#) succéda à [Georges Pompidou](#) comme président, [N.D.L.R.].

la CGE, c'était [Unidata](#), le consortium dans lequel il y avait Siemens. Siemens était le concurrent majeur de la CGE, non seulement en France, mais plutôt à l'étranger. Alors, Siemens dans le groupe Unidata, cela voulait sûrement dire qu'ils auraient une part de marché plus importante dans le téléphone.

Ce n'était pas bon pour la CGE, tout ça. Ils se sont arrangés pour que, premièrement, on arrête le téléphone de la Thomson, et toute activité dans les réseaux. Par ailleurs, les PTT sont aussi venus dire « Nous aussi cela nous gêne, ce truc-là, nous on prépare [X.25](#) ». Donc c'était le moment de se débarrasser de Cyclades, parce qu'on changeait pas mal de choses.

M.S. : *Thomson avait une grosse participation dans Cyclades ?*

L.P. : C'était l'actionnaire majeur de la CII.

Giscard était assez ignorant de tout ça, c'étaient ses conseillers qui le poussaient. Comme d'habitude dans les administrations, il y a toujours les lobbies derrière, qui font en sorte de faire du marchandage de projets, du marchandage de crédit, etc.

Toujours est-il qu'on a modifié la politique. Unidata a été cassé. Les Allemands étaient furax, car ils avaient cru dur comme fer à ce développement. Ils avaient même fermé leurs filiales informatiques en France, et ils n'étaient pas du tout contents qu'on leur joue ce tour de pendu.

Nous, on nous a mis un petit peu au rencart : « vous ne faites plus de réseau, vous faites autre chose. » Il y avait d'autres projets pilotes ; il y avait un projet de mettre Pascal sur je ne sais quelle machine à l'époque, peut-être Mitra ; on avait un projet de sûreté de fonctionnement. À l'époque, j'étais responsable des projets pilotes à l'IRIA, donc j'avais d'autres sujets d'intérêt, mais évidemment, celui qui prenait la plus grande partie de mon temps, c'était Cyclades.

M.S. : *On est en quelle année, là ?*

L.P. : Fin 1974, début 75. Pompidou est mort au printemps ; il y a eu les élections et différents changements ; le directeur du CNET a changé, le chambardement, quoi. C'est un peu plus tard qu'on a créé l'INRIA.

M.S. : *Giscard avait pourtant une image de moderniste ?*

L.P. : Bah, sur les photos, oui !

La Délégation étant supprimée, c'était le ministère de l'Industrie qui a repris le manche, avec [Hugues de L'Estoile](#), un gars de Dassault, qui avait dit que l'informatique c'est un métier comme les autres ; qu'il y avait la phase recherche et puis après l'industrie. On nous a dit qu'on avait bien travaillé, mais que ce n'était plus notre rôle de développer les réseaux, mais celui de l'industrie. J'ai même reçu une lettre d'[André Danzin](#), à l'époque directeur de l'IRIA, qui me disait qu'il ne fallait plus que je m'occupe de réseaux. Comme il n'était pas idiot, il savait bien que je n'en ferais rien, mais il avait ordre de m'écrire cela de la part du ministère de l'Industrie.

On a quand même continué à faire fonctionner Cyclades sans aucun support, ni budget, jusque vers 1978, parce qu'il y avait des bénévoles, différentes équipes de recherche qui travaillaient là-dessus, qui se servaient du réseau. Je n'ai pas tenu de chronologie du démantèlement de Cyclades ; les nœuds (une douzaine de Mitra), je ne sais pas ce qu'ils sont devenus. De toute façon ils avaient vieilli ; ils ont disparu petit à petit, sans fanfare.

M.S. : *Mais les lignes louées sont restées ?*

L.P. : Elles sont restées jusqu'au bout ; je ne sais pas à quel moment elles ont été supprimées. Les PTT, de ce côté-là, ont eu un certain laxisme bienveillant. Ils avaient gagné, ils voulaient X.25, ils l'ont eu, alors ils n'avaient plus vraiment de raison d'être méchants.

Bon après ça... alors pas grand-chose. Je suis passé au CNET quand on a créé l'INRIA. On a créé aussi l'Agence de l'Informatique en même temps. Elle avait été créée sur l'idée que ce n'était plus la peine de faire de la recherche en réseau en France, c'était aux industriels de s'en occuper ; que, donc, les équipes réseau qui étaient à l'IRIA, on pouvait en faire autre chose ; que ce qui était important c'était les usages et les applications, etc. Donc, il fallait créer une Agence de l'Informatique qui fasse la promotion des applications.

M.S. : *Cela me rappelle quelque chose. C'est toujours un peu comme cela.*

L.P. : Ils ont demandé de l'argent au ministère des Finances. Celui-ci leur a dit, écoutez, vous pouvez créer une Agence de l'Informatique, mais vous vous démerdez avec zéro budget. Ils ont eu des locaux à la Défense, mais pour trouver des gens, ils ont commencé par dépecer les équipes Cyclades. Pas que Cyclades d'ailleurs ; également d'autres équipes de recherche : une équipe de l'IRIA qui travaillait sur le Mitra 15 ; une équipe qui travaillait sur des modems à haute vitesse, etc.

[Jacques-Louis] Lyons avait été nommé directeur adjoint, pour assurer la transition. Il a gardé les matheux, et il a viré les autres. Ce qui fait que, pratiquement, la plupart des ingénieurs qui n'étaient pas dans les domaines mathématiques se sont retrouvés dispersés. Certains, comme moi, Michel Gien et Hubert Zimmermann, on est passés au CNET. On est passés chez l'ennemi ! Ils nous ont embauchés parce qu'ils voulaient, eux aussi, faire des projets pilotes ; le CNET c'était un peu trop cloisonné, il y avait trop de particularismes locaux, régionaux. Mais celui qui m'a embauché a été remplacé par un directeur du genre apparatchik 150% ; pour lui, hiérarchie-hiérarchie, s'il y a un projet qui se passe, à droite, à gauche, alors ça ne marche pas. On n'essaye pas de mélanger les projets entre les centres ! Ça permettait d'arrêter toute idée de projet pilote.

Finalement, je me suis concentré sur la normalisation.

M.S. : *Toujours en restant au CNET ?*

L.P. : Toujours en restant au CNET. À l'époque, la Commission Européenne trouvait que l'Europe était très mal organisée en matière de normalisation. Il y avait dans chaque pays une agence de normalisation, comme l'AFNOR par exemple. Ils allaient en ordre dispersé défendre leur point de vue dans les réunions ISO. Il fallait un organisme européen, qui prenne des positions, et qui soit un peu plus blindé contre les Américains ou contre les Japonais.

C'est à ce moment-là qu'ils ont créé l'ETSI côté télécom, et ils ont ressuscité un truc qui était en catalepsie, qui s'appelait le CEN-CENELEC. CEN, ça devait vouloir dire Confédération Européenne de Normalisation, et l'autre c'était normalisation électronique, un truc comme ça. Ils ont fait de ce truc-là le fondé de pouvoir de la normalisation européenne en matière informatique.

C'est à ce moment-là, en 1976, alors que Cyclades s'était déjà arrêté en principe depuis deux ans, qu'a commencé l'OSI. C'était lancé par les Européens, pour essayer d'harmoniser un peu la normalisation de l'ensemble des systèmes informatiques. Avant, il n'y avait que des normes d'échange entre les systèmes, pas de norme d'échange au niveau architecture. Les protocoles de plus haut niveau, transport, présentation, etc., tout ça n'existait pas.

Au début, les constructeurs, même Bull, disaient « Nous, on voudrait que tous les terminaux soient compatibles avec nos matériels, mais on n'a aucune raison de communiquer entre ordinateurs ». Mais ils avaient un peu changé d'avis, et ils admettaient que les ordinateurs devaient communiquer entre eux, même quand ils n'étaient pas de même marque.

Il y avait bien SNA d'IBM, mais c'était du matériel IBM. IBM a relativement bien accepté de coopérer dans l'opération OSI.

Hubert Zimmermann était disponible pour meubler l'OSI. Comme on était au CNET, cela donnait pas mal de poids politique. Avec un Américain qui s'appelait [Charles] Bachman, lui aussi fana OSI, ils ont bien managé la mise en place de l'OSI en Europe et aux USA. De ce fait, on s'est trouvés impliqués dans le développement de protocoles, fortement inspirés de Cyclades.

Ils se sont répandus, et cela a assez bien fonctionné pendant quelques années, jusque vers 1982–83, l'époque où les poids lourds de l'informatique se sont trouvés pris dans une crise financière. Les grosses machines ne se vendaient plus. Les sociétés commandaient des matériels de taille moyenne, et faisaient des réseaux départementaux à base d'Ethernet (ou *Token Ring* quand c'était du matériel IBM). Les grosses machines, c'était en déficit ; les grands constructeurs, soit ont fermé, soit se sont mis à faire du compatible IBM. Leurs ingénieurs de top niveau, ceux qui faisaient de la normalisation OSI, ils les ont envoyés dans des divisions commerciales comme support technique.

L'action OSI s'est arrêtée faute de vapeur. Pratiquement tous les constructeurs importants ont laissé tomber OSI, qui n'arrivait pas à sortir, parce qu'il y avait une

inflation de choses à faire. Tous les métiers se disaient « il faut s'OSI-fier maintenant » ; il aurait fallu faire des protocoles particuliers à chaque métier ; ils n'ont pas réussi à maintenir le développement OSI dans un périmètre gérable.

De toutes façons, il y avait la concurrence des petites machines, qui se satisfaisaient très bien des réseaux locaux. Aux États-Unis, le schisme a commencé, car les liaisons à haut débit, c'est-à-dire à 1500 kbits, c'était pas cher du tout. Donc, ils faisaient des réseaux locaux interconnectés par des liaisons louées. Ils n'avaient pas besoin de réseau public ; ils n'avaient pas besoin de réseau à un niveau de protocole plus important.

C'est à ce moment-là qu'est introduit quelque chose qu'on appelle le *Frame-Relay* (relais de trames). Le protocole *Frame-Relay* permettait de faire de l'*HDLC* sans passer par des systèmes de commutation de paquets.

On a vu apparaître aussi les premiers routeurs, les Cisco et compagnie. Au début, ils ont fait des lignes louées entre deux réseaux locaux. Rapidement, on passe à plusieurs lignes, et donc on met un routeur. Comme il n'y avait pas vraiment de norme inter-matériels, il fallait que les procédures de ligne soient les mêmes aux deux bouts ; le routeur faisait la conversion si nécessaire. L'interconnexion utilisait une procédure de relativement bas niveau, qui était un *HDLC* un peu amélioré, le *Frame-Relay*.

C'est donc une technologie relativement bas niveau qui a réussi. Après, ils sont passés aux datagrammes, puis, beaucoup plus tard, aux techniques Internet.

Le *Frame-Relay*, développé par les Américains, a contaminé l'Europe par les nor-diques et par les Anglais, qui n'avaient jamais été très fanatiques de X.25.

Transpac avait bien réussi en France, parce que c'était pratiquement imposé. Cela a réussi aussi dans différents pays, parce qu'à l'époque il y avait très peu de connaissances techniques sur la commutation de paquets, et les sociétés qui avaient réussi à avoir un système confortablement opérationnel, il n'y en avait pas des tonnes. Il y avait la France ; les Allemands avaient raté le coche, parce qu'ils voulaient faire X21, c'est-à-dire de la commutation de circuits, mais sans paquets ; eh bien, ils n'y ont pas résisté. Les Espagnols ont réussi à percer, surtout en Amérique du Sud. La SESA a installé du paquet « français » au Brésil, du X.25, et également en Nouvelle-Zélande.

Tout ça, ça marchait, sauf que petit à petit la technologie TCP-IP s'est répandue un peu partout, et donc X.25 est devenu inutile. Et l'an dernier on a arrêté Transpac et le Minitel.

M.S. : *C'est l'époque où j'ai été embauché à l'INRIA, pour travailler sur la SM90 et le réseau. J'étais payé par le CNET. Le CNET a su, je ne sais pas comment, que je travaillais sur TCP-IP, et j'ai reçu une lettre comminatoire du directeur du CNET, me disant qu'il était interdit que je travaille sur TCP-IP.*

L.P. : Ils voyaient rouge à l'époque !

Ils ont réussi à faire X.25, c'est vrai, mais il n'a pas duré très longtemps. Ils n'ont toujours pas complètement digéré la question.

L'aventure Theseus

M.S. : *Et donc, de cette époque-là à maintenant, qu'est-ce que vous avez fait ? Vous ne vous êtes pas arrêté je suppose ?*

L.P. : Ah bien non, j'en suis à ma troisième retraite. Ma première retraite n'a pas été bien longue.

M.S. : *En dernier, vous étiez au CNET ?*

L.P. : En dernier j'étais au CNET. Il y avait à l'époque un directeur de la formation aux PTT (je devrais dire à France Télécom, puisque cela a déjà changé de nom), qui adorait ouvrir des formations à droite à gauche.

Alors, une des dernières idées était d'ouvrir une école de management pour ingénieurs. Les grands industriels ou les grands utilisateurs se plaignaient qu'ils pouvaient difficilement faire des réseaux chez eux, parce que leurs ingénieurs administratifs, commerciaux, qui faisaient des affaires, ne connaissaient rien à la technique réseau, et les ingénieurs réseau ne connaissaient rien aux applications. L'idée, c'était de former des gens qui avaient déjà une culture d'ingénieur, en leur mettant en plus une culture de management.

Cela a démarré aux États-Unis, du côté technique, après à l'ENST, et puis après à Fontainebleau.

M.S. : *À l'INSEAD ?*

L.P. : C'est la première école de management qu'il y a eu en France.

C'est devenu la mode de dire « maintenant, il faut donner une culture de managers à nos ingénieurs réseau, de manière à ce qu'ils puissent vraiment parler aux utilisateurs, en terme d'application, et non pas en termes de protocole ». Par ailleurs cela leur donnait plus de pouvoir dans la société, parce qu'avant, l'informatique c'était relégué dans les services techniques.

Donc on a créé une société de participation d'état, où il y avait les Télécom et l'EDF, l'Union, la BNP, enfin, tout un tas de gens bien, pour constituer une école de management à Sophia-Antipolis. Ça s'appelait Theseus, et ça a disparu.

Le premier directeur était un type assez jeune, qui avait probablement 20 ans de moins que moi. Ils ont embauché un Italien comme professeur de management, un Américain comme professeur de management, et moi comme professeur de technologie.

M.S. : *En quelle année ?*

L.P. : Cela doit être en 1989 ou 90.

M.S. : *Et votre titre officiel là-dedans ?*

L.P. : J'étais doyen de Technologie. Il y avait un doyen du Management, et un doyen de l'Économie. On était trois. Je ne dirais pas qu'on était les Trois Mousquetaires !

J'ai été chargé de mettre en place une formation technique, technologique, à base de réseaux, parce que c'est ce qui manquait précisément dans le paysage. J'avais des rapports assez mauvais avec l'Américain, parce qu'il voulait contrôler tout. Comme le jeune directeur était plutôt du genre à ne pas prendre position, on se bagarrait un peu.

Il y avait trente-quarante élèves. La moitié avait une formation technique, et l'autre non, ce qui créait une certaine tension dans le milieu des étudiants. Certains voulaient faire plus de technique, et d'autres en voulaient moins. On n'arrivait pas à se mettre d'accord, ce qui fait que ce n'a pas été charmant cette histoire.

Sauf que j'ai compris comment ils faisaient la formation business. C'était largement du baratin : on étudie des cas, des tas de machins, il faut toujours prendre position très très vite. Il faut lire des tas de bouquins, se référer à des trucs déjà faits, mais, au bout du compte, c'est principalement du bon sens, plus des tas de trucs qui font illusion. Bon, c'était toujours utile à savoir.

J'y suis rentré en 91-92, ou à la fin de l'été 1990 peut-être. Je suis parti début 1993, donc, j'ai dû rester là-dedans pas tout à fait trois ans.

Le Sommet Mondial de la Société de l'Information

L.P. : C'était une époque où l'on parlait beaucoup de la faillite des régimes de retraite complémentaire. Comme j'avais collecté tous les trimestres nécessaires, j'ai décidé de me mettre en retraite au début 93. J'avais tous les éléments, c'était le début de ma première retraite. J'ai quitté Sophia-Antipolis.

Il y avait aussi un autre mythe, c'est que, comme j'aime bien le sud, bien après je pourrais prendre ma retraite sur place. Non. D'abord, le sud c'est extrêmement peu réceptif aux étrangers. Sophia-Antipolis ce ne sont pratiquement que des étrangers venus d'ailleurs, ce ne sont pas des gens du coin. En plus de ça, la région est très pourrie politiquement (c'était encore l'époque [\[Jacques\] Médecin](#)). Le béton pousse très vite ; sur l'autoroute de la Côte, un camion sur trois trimballe du gravier, du ciment ou de quoi les mélanger. C'est l'emballage constructions, Bouygues n'avait aucun contrat, tous les contrats sont passés avec les sociétés du coin. On s'aperçoit à l'usage que ce n'est pas forcément un endroit où il fait bon vivre. Bon, pour les vacances, mais pas forcément y travailler.

Comme j'étais à Paris presque tous les week-ends, pas la peine de continuer ça. Donc je suis parti à la retraite. Pendant six ou sept ans j'ai continué à donner des cours à Sophia-Antipolis ; uniquement les cours qui m'intéressaient, ceux qui

n'avaient pas de conflit avec l'Américain. D'ailleurs, l'Américain a été viré aussi, parce qu'entre-temps, Theseus a été vendue à la CCI de Nice. France Télécom s'est retirée de la formation, parce qu'on a créé la compétition entre les opérateurs. France Télécom, qui supportait à cent pour cent les écoles techniques, a dit « je ne vois pas de raison de continuer de payer pour la formation des concurrents ». Donc Theseus a disparu en fait.

Et alors, pendant ce temps-là, la première étape de ma retraite, c'était d'abord de m'initier à tous les trucs, avec un Mac, de me remettre à programmer, parce que j'avais décroché depuis la période où j'avais fait des projets pilotes à l'IRIA. Je n'avais pas eu de contact avec les PC, et comme j'avais autre chose à faire, j'avais perdu la main là-dessus. Après, j'ai fait ce que j'appelle de la formation alimentaire, c'est-à-dire des sociétés de service qui font du séminaire de recyclage de cadres. Je faisais trois ou quatre séminaires par an en plus des formations à Sophia-Antipolis.

À Sophia je m'étais arrangé pour que ça tienne sur une ou deux interventions par an. J'y allais pour, trois, quatre jours et puis c'était réglé. Les autres formations, c'était généralement à Paris, des séminaires de deux à trois jours. Cela payait bien ; c'était la deuxième retraite.

La troisième retraite est arrivée vers l'an 2000. Je commençais à être un peu en bout de course en matière de formation. De plus en plus, les gens ne voulaient pas des idées mais des pense-bête. « Quelle est la recette pour..., quelle est la recette pour... ? » Cela devient moins intéressant. Ça correspondait aussi à l'évolution technologique : il y avait de moins en moins de contacts avec les systèmes, et de plus en plus de contacts avec les appareils, les applications ; c'était pas mon truc.

C'est à ce moment-là qu'a démarré le SMSI, Sommet Mondial de la Société de l'Information, un sommet ONUien, qui était en principe destiné à analyser et remédier à la fracture numérique, qui se produisait entre les pays. Cela aurait pu me passer sous le nez sans que je m'en aperçoive, mais il se trouvait que la France avait un ambassadeur délégué à ce sommet. C'est un gars que je connaissais bien, parce qu'il avait été secrétaire général de l'IRIA, [Michel Peissik](#). Il a été secrétaire général de l'IRIA, dans les dernières années où j'y étais ; il est à la retraite maintenant. C'était un diplomate qui avait été dans pas mal de pays, Russie, Yémen. Il avait roulé sa bosse, et on était restés en contact. Quand il a été nommé ambassadeur SMSI, il m'a appelé pour me dire, on a des positions à prendre techniquement, est-ce que tu ne pourrais pas venir ? Je me suis trouvé, comme ça, impliqué dans les réunions à Genève préparatoires au sommet.

Le sommet, c'est une grande réunion mondiale, préparé par une douzaine de réunions préparatoires, qui durent deux à trois ans, dans lesquelles on retrouve des diplomates de tous les pays. Je me suis donc retrouvé à Genève à rencontrer des diplomates, qui étaient des gens très intelligents, mais qui ne connaissaient rien à l'Internet. Les Américains racontait partout que l'Internet c'est très compliqué, qu'il ne faut surtout pas laisser les gouvernements s'en occuper, qu'ils ne font que des

bêtises. Moi, j'ai commencé à parler aux Arabes, aux Chinois, aux Indiens, en leur disant qu'Internet c'est bien moins compliqué qu'un système de transmission par satellite. C'est bien moins compliqué qu'un système de gestion de missiles. Ce sont des choses qu'on peut faire partout, seulement il faut être formé.

Ils ont très vite compris, comme ils n'étaient pas bêtes du tout, que dans n'importe quel truc comme l'Internet, il y a un problème de gestion politique — qu'est-ce qu'on veut en faire dans le pays — et puis ensuite la technique, des Télécom, il faut des ingénieurs. Ils ont très vite compris qu'ils avaient intérêt à regarder de plus près ce qu'était Internet. Au début, ils avaient du mal.

On les a assez bien convaincus en leur expliquant que l'Internet était en ASCII, que si leur pays voulait se développer avec l'Internet, il fallait que les gens puissent s'en servir dans leur langue. Ils ont très bien compris ce truc-là. Le multilinguisme est devenu le fer de lance pour attaquer la forteresse américaine. Ils ont donc introduit les [IDN](#) après, sous la pression des Asiatiques.

On a bien accroché là-dessus ; le Brésil avait même pris l'initiative de créer un groupe, qui s'appelait les [Like-Minded Countries](#), des pays qui pensent de la même façon.

M.S. : *On est en quelle année ?*

L.P. : On est fin 2001, courant 2002, en partie 2003. Le sommet a été organisé en deux temps : un premier sommet à Genève, en 2003, et un deuxième en 2005 à Tunis. Avant le premier sommet, on a commencé à endoctriner les gens, on leur a expliqué que l'Internet était important pour leurs pays. Les *Like-Minded Countries* sont devenus le groupe des 77. La Chine, fallait pas mélanger la Chine avec les 77 quand même, c'était pas sérieux !

Pendant un an et demi, on a été invités à leurs réunions. On était les seuls occidentaux invités là-dedans, parce qu'on pouvait discuter avec eux, de comment ils voyaient la technique Internet, les développements, la politique. Au bout d'un an et demi, ils étaient parfaitement au courant, ils n'avaient plus besoin de nous après. [Rires]

Le sommet a eu lieu. Il y a eu une première vague de négociations pour arriver à un texte qui expliquait ce qu'était l'Internet, les développements à faire, un tas d'objectifs à atteindre.

Ça n'a pas été complètement concluant. Il y avait un deuxième sommet prévu, ce qui n'était pas réglé au premier pouvait être réglé au second. Alors, on a continué à participer à tout ça. L'ambassadeur ayant changé, je n'étais plus dans la délégation française. Au début, j'assistais en tant que représentant du gouvernement, puis comme représentant d'[Eurolinc](#), l'association loi 1901 qu'on avait créée en 2001 pour promouvoir les langues natives dans Internet. On s'en servait comme base de départ pour contrer la domination américaine.

Le premier texte de 2003 était déjà très explicite sur pas mal de choses : il fallait que chaque pays prenne en charge les politiques publiques, la formation, tout un tas de trucs. Il y avait là-dedans la « religion » américaine, il fallait que les développements soient essentiellement dirigés par les sociétés privées ; sachant que l'Internet, comme de bien entendu, était développé par le gouvernement. Ce qu'ils voulaient surtout, c'était permettre aux sociétés américaines d'avoir de belles parts de marché.

On est arrivés au deuxième sommet. On est allés régulièrement aux réunions, on faisait des communications, on expliquait quand il y avait des conneries techniques dans les présentations. On pouvait éventuellement faire des contre-propositions pour dire que ce n'était pas bon. C'était quand même assez marginal, ce qu'on pouvait faire là-dedans, parce que le pouvoir c'était d'abord les Américains et leurs satellites, qui votaient toujours avec les Américains, le Canada, l'Angleterre, l'Australie, la Nouvelle-Zélande, le Japon et Israël, les inconditionnels du point de vue américain. Ensuite, il y avait les Européens : pas vraiment de point de vue, ils voulaient un peu de libéralisation, un peu plus de pouvoir dans l'Internet, mais ils ne voulaient pas faire de tort aux Américains. Donc, l'Europe, pratiquement aucune position sérieuse. Et puis tous les autres, qui étaient contre les Américains.

Les textes sortis de ce deuxième sommet étaient pas mal explicites, sur certains points qui n'ont pas été du tout pratiqués. Par exemple, on disait dans ce texte que le choix et la gestion des noms de domaine dans les pays, par exemple .fr, étaient strictement de la responsabilité des pays, et qu'aucun autre gouvernement n'était autorisé à intervenir et à se mêler de ça ; pourtant c'est l'ICANN qui continue à mettre les veto, et qui fait payer les gens pour ça. Cela a été largement ignoré, mais ça a permis à pas mal de pays de se braquer contre les Américains, et c'est encore le cas aujourd'hui.

Donc, ça c'était ma troisième retraite, dans laquelle je suis encore.

La gouvernance et l'avenir d'Internet

L.P. : Aujourd'hui, les trucs de Snowden et les révélations sur la NSA, cela ne fait qu'exacerber cette opposition. On le savait d'ailleurs depuis 2005, pas parce que j'étais avec les services spéciaux, mais parce que je lisais les journaux. Les services spéciaux qui coopèrent entre eux, je le savais depuis le début, c'est-à-dire au début des années 2000. Le gouvernement était forcément au courant. Ils n'ont rien fait du tout, c'était parfaitement prévisible, la situation d'aujourd'hui. Tous les gens informés le savaient. Maintenant ils découvrent qu'un état, un état policier, les surveillait ; mais ils les ont laissé faire.

Fin 2013, je suis allé à Dublin, parce que c'est là que l'on discute de RINA, que je considère comme le futur *Operating System* et protocole Internet. Il est développé par John Day de l'université de Boston ; les Irlandais et les Catalans sont très bons là-dessus (voir www.irati.eu).

M.S. : *Donc vous êtes toujours impliqué dans la gouvernance d'Internet.*

L.P. : Oui, c'est ça. Comment contrer l'hégémonie et la dominance complète des Américains sur l'Internet ? Il n'y a évidemment pas de solution rapide. J'ai justement écrit un papier, un pavé dans la mare, sur la façon de démanteler l'ICANN. Il part d'une constatation simple : on ne peut rien faire qui ne nécessite l'accord des États-Unis. Ils sont toujours contre, sauf si c'est dans leur intérêt. La seule chose que l'on peut faire, ce sont des trucs pour lesquels il n'y a pas besoin de leur accord. Mais là, on peut faire des tas de choses : les taxer ! Décider que les noms de domaine ne sont plus un monopole de l'ICANN, décider que l'on peut avoir différentes racines, et à la longue, chiffrer évidemment. Monter des systèmes de courrier électronique, par exemple : ça pourrait être l'Union Postale, ou bien des groupements de sociétés de courriers qui mettent en place des systèmes sécurisés, indépendamment des Américains. Il y a des tas de choses qu'on peut faire, la liste est loin d'être close, indépendamment des Américains.

Bien sûr, ça ne se fera pas du jour au lendemain.

M.S. : *Est-ce que vous proposez de créer et de faire des nouvelles normes d'Internet ?*

L.P. : RINA, ce sont effectivement des nouvelles normes, au lieu de TCP-IP qui est une passoire, qui est très bien pour assurer la communication de bout en bout, mais qui ne protège absolument pas contre les pirates. C'est donc de faire un système dans lequel tout ce qui est communication est sécurisé.

M.S. : *Toujours des datagrammes ?*

L.P. : Toujours des datagrammes, mais ce n'est même pas nécessaire. Ce qui est différent, c'est l'architecture. Le réseau est un ensemble de réseaux virtuels. Chaque réseau est complètement indépendant, et ne connaît pas les autres réseaux. Ils ont leur système de nommage à eux, ils ont leurs ressources à eux, ce qui fait qu'ils peuvent être complètement différents à la fois en technologie, en principe d'architecture et en protocoles.

Ce qui n'empêche pas de faire des normes, parce que cela évite à l'industrie de faire 36 fois les mêmes choses. Les réseaux s'ignorent, mais ils peuvent communiquer entre eux, à condition de se mettre d'accord, de s'identifier, de s'assurer qu'ils ont les ressources nécessaires. Les réseaux virtuels constituent un super-réseau.

M.S. : *Mais il n'y a plus de communication de bout en bout dans votre nouveau... ?*

L.P. : C'est ce qu'on veut. On peut utiliser TCP-IP à l'intérieur, si on veut ; il faut bien un protocole plus développé. En fait, la sécurité est introduite en fonction du besoin. Si on a besoin de sécurité de bout en bout, on prend un protocole de bout en bout. Si c'est simplement une transmission de messages en *tweet*, ce n'est pas nécessaire. La

sécurité, à la fois en tant que secret, et en tant que fiabilité, cela peut être défini en fonction des besoins. Il suffit d'avoir un catalogue de procédés, à utiliser en fonction des besoins.

Ça permet aussi de récupérer l'Internet tel qu'il est aujourd'hui. RINA peut utiliser Internet comme un simple tuyau qui n'est ni fiable, ni sécurisé, donc en faisant passer dessus du chiffré. Maintenant, on peut utiliser aussi les services de l'Internet d'aujourd'hui ; les gens continuent à s'en servir. S'ils veulent introduire de nouveaux services plus sécurisés, ils prennent un système RINA qu'ils branchent sur l'Internet. À ce moment-là, les gens peuvent accéder à l'Internet tel qu'il est, avec ses limites et ses risques, c'est tout. Cela n'oblige pas les gens à changer du jour au lendemain.

M.S. : *Et là-dedans, IPv6 cela sert à quelque chose ?*

L.P. : Non, pas plus que maintenant. Parce qu'aujourd'hui, IP et les noms de domaine, tout ça c'est basé sur l'idée qu'il y a un monopole qui fixe les rapports. À partir du moment où il n'y a plus de monopole, on peut admettre qu'il y a autant de systèmes de nommage que l'on veut. On a le choix entre plusieurs, et on n'est pas obligés de dépendre d'une autorité centrale pour ça.

M.S. : *Le protocole unique, ça a quand même l'avantage que tout le monde peut parler avec tout le monde.*

L.P. : Oui, s'ils admettent d'avoir un système en commun. Ce n'est pas tout le monde.

On peut téléphoner à tout le monde par exemple : c'est parce qu'il existe une autorité centrale qui distribue les numéros. L'autorité centrale, qui la prend ? Aujourd'hui l'UIT existe depuis 150 ans environ, donc elle n'est pas contestable, mais il n'empêche que les Américains ont créé la leur. Les numéros IP, d'un côté, et les numéros de téléphone de l'autre ; il y a déjà une dualité. Si on crée un truc qui doit être unique, il y aura la bagarre pour savoir qui va l'avoir : l'UIT ou les Américains ; ce n'est pas soluble, ça. Le mieux, c'est d'avoir autant de trucs qu'on veut ; ça veut dire qu'on peut introduire le niveau de cloisonnement et de sécurité que l'on souhaite.

M.S. : *Mais qu'est-ce qui empêche à ce moment-là la Chine ou l'Iran de dire « je veux mon réseau à moi » ?*

L.P. : Mais qu'ils le fassent !

M.S. : *Mais les gens ne pourront plus communiquer avec le reste du monde !*

L.P. : Si les Chinois ne veulent pas, c'est normal. C'est bien ce qui se passe aujourd'hui. Ils ont créé leur réseau chinois en chinois. On n'a pas accès aux IP chinois.

Vouloir être indépendant et séparé, c'est un choix, qui est parfaitement légitime. Après tout, il y a beaucoup d'organisations où il y a des tas de choses secrètes qui ne doivent pas en sortir. C'est comme ça, c'est la loi de la société.

Donc, cela permet de communiquer avec les gens qui sont d'accord pour communiquer ; s'ils ne sont pas d'accord, alors, on ne peut pas communiquer, c'est tout.

M.S. : *C'était quand même un des apports de l'Internet : tout d'un coup, il y a un espace de liberté qui s'est ouvert, parce qu'il y avait une technologie unique, et que tout le monde pouvait communiquer avec tout le monde. Là vous êtes en train de le refermer.*

L.P. : En fait, c'est le cas du téléphone par exemple. Le téléphone, on savait que ça pouvait être épié, on savait parfaitement qu'il y avait des écoutes, mais ce n'était pas développé au point d'être un véritable système social. C'était des activités de type exceptionnel. Internet, maintenant, c'est la passoire. Comme tout est accessible à toute société qui a les moyens de payer, cela devient totalitaire. À partir du moment où il y a la NSA, par exemple, qui a des informations sur tout le monde. Il ne faut pas se faire d'illusions, des Snowden il y en aura bien d'autres, et ils ne seront pas tous gentils.

Il peut y avoir des gars qui vont utiliser l'information qu'ils ont de la NSA pour la revendre, éventuellement triturée. On vient d'un système qui est parfaitement incontrôlable, et qui ouvre la porte à toutes les mafias possibles. On a lâché le diable, on a laissé le diable sortir de la bouteille. Le faire rentrer, cela ne va pas être facile !

Il y aura toujours une part qui restera non contrôlable. On peut admettre que, si le chiffrement coûte cher à décoder, ils s'aperçoivent à l'usage que finalement cela ne sert à rien, ou à pas grand-chose.

Après, il y aura probablement des pays qui vont dire, écoutez, c'est comme la bombe atomique, on peut tout casser, mais il vaut mieux se mettre d'accord pour limiter les dégâts, arrêter les frais.

M.S. : *Ne pensez vous pas que ce sera perçu comme un retour en arrière ? Remplacer un système ouvert par une fédération de systèmes fermés, si j'ai bien compris ce que vous dites ?*

L.P. : Fermables ! Cela ne veut pas dire qu'ils le sont en permanence. Ça veut dire que ce sont des systèmes entièrement évolutifs ; si on crée un réseau, il peut être créé comme fermé, ou comme ouvert. Il ne dure pas jusqu'à l'éternité, et on peut l'arrêter ou le supprimer.

Aujourd'hui, dans le système postal, n'importe qui, le facteur, distribue des lettres. Théoriquement, on n'ouvre pas le courrier, c'est admis. Aujourd'hui, il est admis que, quand on écrit un message sur gmail par exemple, ils sont en train de le lire. Il y a quand même une différence considérable de niveau de *privacy*. C'est ça qu'il faudrait arriver à rétablir, c'est un niveau de sécurité et d'indépendance qui soit généralement respecté, sachant qu'il y aura toujours des nécessités importantes, politiques ou militaires ou autres, pour le transgresser de temps à autre. Ça, ça me

paraît possible. Comme la bombe atomique ; si on n'avait rien fait, chaque pays pourrait avoir aujourd'hui une bombe.

M.S. : *Techniquement, il sera toujours possible d'ouvrir le courrier. Vous dites bien que ce sont les conventions sociales qui font que le facteur n'ouvre pas le courrier. Est-ce qu'on a donc besoin d'une solution technique ?*

L.P. : Les solutions techniques ne résolvent pas les problèmes, ce sont justement les conventions sociales qui les résolvent. Aujourd'hui, il n'y a pas de convention sociale bien établie là-dessus. Aujourd'hui, il y a encore des gens qui pensent que c'est très bien d'écouter tout ce qui se passe, et d'autres qui ne le supportent pas. Cela veut dire qu'entre différents groupes d'une société, entre différents pays, suivant leur culture, il y a des tas de choses qui ne sont pas compatibles. Alors, je ne sais pas si cela peut être compatible au niveau mondial, j'en doute ; mais cela peut au moins être compatible dans certaines catégories de pays, qu'ils soient ou non techniquement avancés.

Le chiffrement n'a jamais empêché de casser. Le niveau de connaissances se propage assez vite, et pratiquement au bout de très peu de temps, tout le monde est à peu près au même niveau. Alors la technique permet de faire ce qu'on veut, c'est-à-dire de faciliter ou d'interdire. Aujourd'hui, on se trouve dans une situation où les sociétés n'ont pas été préparées à réfléchir, à décider de ce qu'il fallait interdire ou pas. C'est trop récent, cela s'est introduit trop vite dans le milieu sociétal.

M.S. : *Et donc, RINA, c'est une solution technique au service de choix sociaux ?*

L.P. : C'est une solution technique. Ce sont les pouvoirs en place, quels qu'ils soient, qui vont décider de la manière de s'en servir. Si ce sont des pouvoirs totalitaires, ils s'en serviront essentiellement pour brider leur population, pour se défendre contre leurs voisins un peu trop ambitieux. Si ce sont des gouvernements plus libéraux, ils laisseront les gens dire ce qu'ils veulent, sans essayer de pister tout le monde. Il vont suivre, comme d'habitude, les opposants et les détraqués, c'est à peu près évident. Au moins, ils ne le feront pas systématiquement.

M.S. : *Donc, pour vous, cela va où l'Internet ?*

L.P. : Je pense qu'il y a une période, qui va être de l'ordre de deux, trois ans, pour arriver à contrer les géants américains, Facebook, Google, eBay, etc. Faire en sorte, tout d'abord, qu'ils payent leurs impôts, ce qui n'est pas le cas aujourd'hui. Ensuite, qu'ils ne fassent pas n'importe quoi de toute l'information qu'ils collectent ; si possible, les empêcher de collecter, sans autorisation des individus et des sociétés, les réduire petit à petit dans ce domaine. Cela c'est encore relativement non agressif.

Simultanément, voir si on ne peut pas aussi mettre en place des noms de domaine indépendants. Moi, je le fais aujourd'hui : je vends des noms de domaine indépendants de l'ICANN (www.open-root.eu). Ce n'est pas pour être un ICANN-bis, mais

pour prouver que c'est faisable. Ça, c'est le genre de choses qui est très faisable techniquement.

La question est de savoir si les gens ont vraiment envie de s'en servir. Comme, aujourd'hui, les noms de domaine servent surtout à collecter des informations pour les Américains, peut-être qu'ils vont finir par comprendre que l'on peut s'en passer des Américains.

Il y a des pays comme l'Inde, comme l'Allemagne, qui commencent à dire « il faut que toutes les données personnelles restent dans le pays ». Tous les marchands de données commencent à pousser des hauts cris, « c'est absurde ce truc, ce n'est pas faisable ». N'importe quoi : chaque fois qu'on touche au territoire d'un monopole, il pousse des hauts cris, c'est inévitable.

M.S. : *Facebook, par exemple, ne vole pas vos données, les gens les donnent volontairement.*

L.P. : C'est parce qu'ils sont inconscients.

M.S. : *Qu'est-ce qu'on peut faire ?*

L.P. : Moi, je ne suis pas sur Facebook, je ne suis sur aucun réseau social. J'utilise gmail, parce que j'ai été obligé un petit peu de le faire, à une époque où j'avais un compte à l'ENST, il me l'ont fermé brutalement. J'ai tout rerouté sur gmail, mais c'est épouvantable gmail, c'est la porte ouverte à tout.

M.S. : *Vous auriez pu utiliser laposte.net, pourquoi gmail ?*

L.P. : Mais j'y suis aussi sur laposte ! Sauf que ce n'est pas aussi utilisé que les autres. Gmail, on ne peut y échapper. Aujourd'hui, avec une messagerie laposte.net, j'écris à quelqu'un qui est sur gmail : tout ce que j'écris sera sur gmail, c'est fini ! Aujourd'hui on est contaminés, indépendamment de notre volonté, par tous les gens qui se servent de gmail.

Je crois que dans trois à cinq ans on pourra utiliser les systèmes RINA. Aujourd'hui, il y a des protos, mais ce sont encore des bricolages de labo. Ce qu'il faut, ce que j'essaie de trouver en France, ce sont des industriels qui s'y intéressent un peu, mettre une personne pour suivre l'évolution, de manière à pouvoir commencer et imaginer les produits qui peuvent en sortir. Déjà, s'assurer de faire fonctionner ce qui existe avec un minimum d'adaptation. Après, mettre des produits entièrement RINA, c'est-à-dire qui utilisent toutes les possibilités de sécurisation, de séparation, sachant que c'est à la carte. Ce n'est pas un système *one size fits all*, c'est au contraire extrêmement adaptable.

Après, il faut commencer à bâtir et mettre sérieusement en place des systèmes anti-totalitaires. On peut espérer que dans ce laps de temps, il y aura différents pays qui auront tourné casaque, et qui penseront que la disposition des données personnelles abusivement ce n'est pas tolérable.

Je ne sais pas si les Américains changeront d'avis, mais pour l'instant ils ne sont pas intéressés. Si on dit que c'est pour défendre l'Amérique, alors ils sont d'accord. Il y a quand même des gens aux USA qui sont tout à fait contre, mais ils n'ont pas le pouvoir.

C.M. : Je me demande, quel que soit le système qui sera mis en place, même très sécurisé, si les Américains ne vont pas trouver une faille, pour quand même aller au-delà de la sécurité du système et chercher les données...

L.P. : Mais tout le monde va le faire ! Tout le monde va essayer de casser les sécurités. Dans tous les pays, il y aura des gens capables de casser la sécurité, s'ils ont les moyens techniques, financiers. Ça ne coûte pas très cher de casser la sécurité, ce qui coûte cher c'est la capacité de former des gens pour cela. Les Chinois sont parfaitement capables ; les Russes aussi, ils sont très doués en trucs compliqués. Les Français aussi, sauf qu'ils n'ont pas toujours la volonté politique de le faire. Les USA auront les moyens de casser aussi.

À partir du moment où presque tous les grands pays ont des moyens de casser les trucs des autres, c'est comme tous les systèmes de brouillage-débrouillage militaires, tous ces systèmes ont une durée de vie. Après, ils sont cassés et on passe à un autre. Cela sera le même cache-cache. Cela ne me paraît pas un problème majeur, dans la mesure où il n'est pas nouveau. Ce qui est nouveau, c'est qu'il s'est développé très vite, et que les USA ont pris une avance considérable, mais il la perdront. Seulement ce qu'il faut éviter c'est qu'ils la gardent trop longtemps.

L'avenir de l'informatique et ses utilisateurs

M.S. : Qu'est-ce que vous pensez de l'informatique aujourd'hui, du futur de l'informatique ? On parle de l'informatique dans les écoles, qu'est-ce que ça vous inspire ?

L.P. : Dans les écoles, on apprend à tout le monde à compter, à écrire, etc. Fatalement, tout le monde devra apprendre à se servir d'outils informatiques.

Les appareils deviendront de plus en plus bon marché, parce que ce n'est pas ça qui coûte. Ce qui coûte, c'est inventer et produire les applications intéressantes, et les promouvoir. Si une application ne dépasse pas l'année, cela veut dire qu'elle ne sera jamais bonne, elle ne sera jamais bien au point. Il faut arriver à ce que les applications durent quelques années pour qu'elles soient rentables, que les gens aient eu le temps de s'habituer.

Il ne faut pas que cela change trop vite pour les gens. Il faut qu'il y ait un problème de régulation pour l'introduction de la nouveauté.

M.S. : Cela va trop vite ?

L.P. : Pour beaucoup de gens, oui. Donc ce n'est pas efficace. Qu'on le fasse en labo, c'est très bien, mais après il faut décider sur quoi on veut mettre la priorité. Ce n'est pas parce qu'on a dix manières de faire les choses qu'il faut toutes les promouvoir en même temps.

M.S. : *Mais aujourd'hui, il n'y a pas d'autorité pour pouvoir forcer quelque chose. C'est le marché.*

L.P. : C'est le marché, le marché actuel. Mais on sait bien que le marché est loin d'être aussi transparent qu'on le prétend. On dit « la loi du marché », mais il n'y a pas de loi du marché, parce qu'il est toujours biaisable.

M.S. : *Vous aimeriez les influencer dans quelle direction ?*

L.P. : D'une part, réduire les coûts d'usage, et d'autre part, faciliter la compréhension de ce que cela implique. Par exemple, aujourd'hui, quand on se connecte à une banque, pour son compte en banque, il faut taper des trucs... Quel est le degré de protection qui va avec ? On n'en a pas la moindre idée.

Il faudrait que les gens comprennent exactement ce que ça implique de taper son nom ou son code : qu'est-ce qui se passe derrière ? Que les gens se rendent vraiment compte. De même, si aujourd'hui vous signez un bout de papier vis-à-vis des impôts, il faut savoir que cela voudra dire tel et tel impôt. Même si on est négligent, l'explication existe, on sait à quoi on s'engage. Alors qu'avec les applications informatiques, on ne sait pas à quoi on s'engage.

Il faudrait que les gens soient beaucoup plus conscients d'où on leur fait mettre les pieds. Cela demande certainement un certain Bureau Veritas. Cela veut dire qu'il faut des moyens de validation de la qualité de certains produits ou services. Ce n'est jamais complètement honnête ou neutre, mais il faudrait que cela soit au moins indépendant des groupes qui vendent ces choses-là. On ne peut pas être à la fois vendeur et arbitre.

M.S. : *Bien, nous vous avons gardé très longtemps.*

L.P. : C'est bien, cela évitera de le refaire !

On verra dans quelques années si mes prédictions étaient bien fondées.

C.M. : *Oui, on va se retrouver d'ici trois ans pour voir où on en est.*

M.S. : *Merci beaucoup.*



Le plan Informatique pour tous dans l'académie de Créteil : une enquête d'évaluation

Jean-Pierre Archambault ¹

Cet article sur le « Plan Informatique pour tous dans l'académie de Créteil » fait suite à celui de Jacques Baudé « Le plan Informatique pour tous », paru dans le numéro 5 de 1024². Il porte sur l'enseignement scolaire. Concernant l'académie de Créteil, on pourra également se référer aux textes « Au terme de deux enquêtes³ », de Michel Narcy, et « Les stages Informatique pour tous – Expérience de formateurs⁴ », d'Astrid Baljoux.

Dans l'académie de Créteil comme dans l'ensemble de la France, le plan Informatique pour tous (IPT) a comporté trois axes essentiels : les matériels, les logiciels et la formation des enseignants. Il a consisté en une dotation généralisée des lycées, des collèges et des écoles en nanoréseaux, micro-ordinateurs personnels et logiciels pédagogiques, et dans la formation des enseignants pendant les vacances scolaires⁵. Le plan IPT était également ouvert au supérieur et au grand public et a eu un volet télématique dont on se souvient moins. Pourtant, lorsqu'Internet est arrivé, dans une continuité des usages, on a pu s'appuyer sur une culture de communication électronique que le Minitel avait permis que l'on crée dans le système éducatif.

1. Président de l'EPI, <http://www.epi.asso.fr>, ex-responsable des stages IPT sur l'Académie de Créteil.

2. <http://www.societe-informatique-de-france.fr/wp-content/uploads/2015/04/1024-5-baude.pdf>

3. <http://www.epi.asso.fr/revue/histo/h85-gredip.htm>

4. <http://www.epi.asso.fr/revue/histo/h85-stageIPT.htm>

5. <http://www.epi.asso.fr/revue/articles/a0509a.htm>

Semaine de stage	Nombre de cibles remplies	Nombre de réponses de stagiaires	Pourcentages obtenus : 3 choix possibles dans l'ordre croissant de satisfaction																	
			Acquisition de connaissances en informatique			Qualité des logiciels proposés			Aséquration entre les attentes et les contenus du stage			Méthodes pédagogiques			L'objectif du stage est-il atteint ?					
			1	2	3	1	2	3	1	2	3	1	2	3	1	2	3			
juillet 1	42	750	12	66	21	28	66	6	12	49	39	8	34	57	18	61	21			
juillet 2	45	780	9	67	24	27	63	10	7	45	48	6	21	73	18	49	33			
juillet 3	28	490	8	70	21	28	62	10	10	40	49	6	23	72	15	52	33			
juillet 4	17	290	8	66	26	26	64	10	9	41	50	8	21	71	13	64	24			
pré-rentree	38	640	5	63	32	27	60	12	5	42	53	2	16	82	10	51	39			
Toussaint	24	430	3	46	51	16	67	16	3	35	61	0	17	83	6	47	48			
Ensemble des stages	194	3380	8	64	28	26	64	10	8	43	49	5	23	72	14	54	33			

TABLEAU 1. Résultats de l'enquête (valeurs arrondies à l'entier le plus proche).

Une enquête d'évaluation – une vue d'ensemble

Pour l'académie de Créteil, il y eut, en juillet, septembre et à la Toussaint, de l'ordre de 6000 stagiaires dans 303 stages (6500 en intégrant les formations des vacances de printemps). Comment cela s'est-il passé ? Quelle en fut la perception des enseignants ?

Pour y répondre, on peut s'appuyer sur une enquête à laquelle ont participé 3380 stagiaires dans 194 stages s'étant déroulés en juillet, septembre et à la Toussaint (voir Tableau 1). Elle a pris la forme de « cibles » d'évaluation fournissant des éléments mesurables, à savoir des appréciations chiffrées (de 1 à 3 dans l'ordre croissant de satisfaction) concernant les domaines suivants :

- accueil,
- acquisition de connaissances en informatique,
- qualité des logiciels proposés,
- adéquation entre les attentes et les contenus du stage,
- méthodes pédagogiques,
- « l'objectif du stage est-il atteint ? ».

D'autres informations sont remontées, à l'occasion de visites dans les stages et de contacts divers avec des personnes impliquées dans l'opération. En tout, en complétant les cibles, des informations issues de 208 stages. L'échantillon est représentatif : 3380 répondants sur environ 6000 stagiaires. Nous n'avons pas cherché à cerner les raisons des non-réponses qui peuvent être fortuites, par exemple des formateurs n'ayant pas disposé des documents à faire remplir. Des bilans négatifs nous

étant parvenus, nous n'avons pas pensé qu'il y avait nécessairement une relation entre opinion défavorable et absence d'information écrite.

La prudence s'impose toujours en matière d'interprétation des chiffres. Il s'en dégage cependant une satisfaction générale, confirmée à l'époque par des échos divers et multiples. En la circonstance, l'Éducation nationale avait su montrer sa capacité à réussir ce qui semblait être, au départ, un pari impossible. Le plan IPT avait su répondre à un besoin réel de formation. Les enseignants avaient apprécié et ils demandaient un suivi.

Les pourcentages obtenus dans la rubrique « Méthodes pédagogiques » sont les meilleurs (23 % en 2 et 72 % en 3). L'intitulé de la rubrique permet d'avoir quelques idées sur la perception des formateurs par les stagiaires. De fait, l'académie de Créteil disposait d'un vivier de formateurs en informatique qui avait permis la réussite des stages. La politique de formation des années précédentes avait porté ses fruits⁶.

On avait pu constater, l'expérience acquise par les formateurs aidant, une évolution positive des appréciations au cours de l'été, un saut assez net entre la première et la deuxième semaine de juillet, un contentement évident lors des stages de la Toussaint. La brièveté de certains délais, la livraison tardive des matériels (fin juin, début juillet) et des documentations du nanoréseau dans leur nouvelle version avaient sans aucun doute pesé négativement sur les stages de la première semaine de juillet. Ainsi n'avait-il pas été possible d'intégrer la présentation de cette nouvelle version dans les regroupements organisés pour les formateurs dans le courant du mois de juin.

Par ailleurs, l'attribution de crédits spécifiques pour l'organisation des stages aux plans académique et des établissements aurait facilité les opérations et rentabilisé davantage l'investissement financier consenti. Les services rendus par les constructeurs avaient été, pour le moins, assez inégaux.

Les objectifs des stages

Ils étaient ambitieux. Si l'on se référait à ceux que l'on pouvait raisonnablement fixer à une formation très intensive d'une semaine, on avait pu dire qu'ils avaient été atteints. Ce constat pouvait évoluer car il dépendait aussi de la nécessaire consolidation apportée par un suivi, suivi qui fut inégalement réalisé.

Il y a eu initiation, démythification et démystification, éveil d'un goût pour des usages ultérieurs. Quand les formateurs y ont veillé, l'objectif visant la maîtrise du matériel a été mieux atteint que celui se proposant l'insertion dans une démarche pédagogique (les logiciels n'étaient pas ce qu'ils sont aujourd'hui). Quand le matériel fonctionnait convenablement, les stagiaires montraient une relative autonomie. Mais, à la sortie du stage, ils semblaient ne pas être immédiatement prêts à une utilisation dans leurs classes. D'où une demande quasi générale d'un suivi pour consolider les

6. C'était le cas à Créteil comme dans les autres académies. Notons que, dans l'ensemble, beaucoup de formateurs étaient membres de l'EPI.

acquis sous forme, entre autres, de la présence d'animateurs ou de « conseillers pédagogiques en informatique ». Dans son bilan, un formateur écrivait « l'attente est grande et la déception le sera si... ».

Demande de suivi et Plan académique de formation : la programmation plébiscitée !

C'était la vocation du plan académique de formation que de répondre à cette demande de suivi, très forte en ce qui concernait la programmation⁷. Pour les raisons suivantes :

- intérêt pour la programmation en elle-même,
- la programmation en tant que moyen de compréhension de l'ordinateur et de l'informatique (les enseignants « voulaient savoir comment ça fonctionne »),
- la programmation comme moyen d'approche, de motivation pour l'outil informatique, par exemple pour les professeurs des disciplines scientifiques, alors que pour d'autres, le service pédagogique rendu jouait ce rôle,
- des connaissances indispensables pour utiliser le langage Logo dans le primaire,
- aide à la formulation des besoins dans l'élaboration d'une maquette pédagogique,
- pour les enseignants et les élèves, moyen de démystification de la machine.

La demande de suivi portait aussi sur l'insertion des logiciels dans une démarche pédagogique et l'étude de logiciels professionnels (pour les enseignements techniques et professionnels) ainsi que sur la nécessité de faire le point quelques mois après l'arrivée des logiciels dans les établissements.

Qualité des logiciels et pédagogie

L'opinion d'ensemble était réservée, notamment concernant les produits des différentes disciplines tournant sur le nanoréseau. En revanche, les logiciels-outils (par exemple *Scriptor* ou *Texte*, des traitements de texte), les logiciels de simulation, les logiciels professionnels avaient été, eux, appréciés.

Cela étant cette opinion réservée avait été nuancée. Pour différentes raisons. Elle était formulée par des débutants en informatique dont l'avis, certes intéressant et à prendre en compte, pouvait manquer de pertinence. D'ailleurs, des stagiaires écrivaient « Nous n'avons pas de moyens de comparaison ». Le même logiciel, étudié en début et en fin d'un même stage, suscitait des réactions différentes, plus favorables le dernier jour. Les stagiaires n'avaient pas l'expérience de situations vécues avec

7. Les langages utilisés étaient le LSE (Langage Symbolique d'Enseignement, syntaxe française), Basic et Logo.

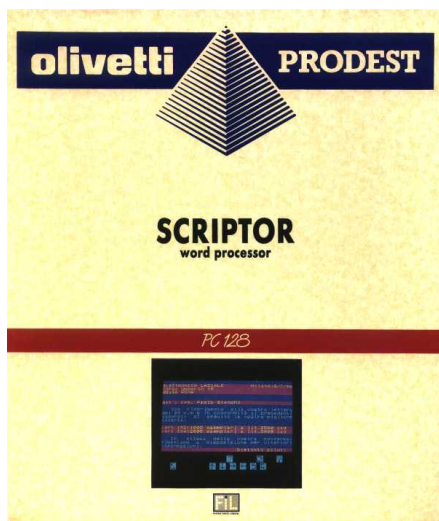
les élèves. En 48 heures de formation on n'abordait que rarement les « gros » logiciels qui, par leur ampleur et leur puissance, impressionnent favorablement. L'image de l'éventail disponible était ainsi tronquée, surtout si une typologie d'ensemble n'était pas donnée. L'idée que l'usage fait d'un logiciel dans une démarche pédagogique peut être plus importante que sa qualité intrinsèque n'était pas répandue. Des bogues, des fautes d'orthographe et de grammaire grossières dans des logiciels de français contribuaient à déformer les premières impressions des débutants, l'absence de connaissances informatiques préalables ne permettant pas de relativiser le jugement porté.

Cela étant, les résultats chiffrés d'une dizaine de cibles tranchaient avec le reste : une satisfaction certaine s'en dégageait, la présentation faite par le formateur avait plu. Ce fut le cas avec des logiciels professionnels bien maîtrisés, à la finalité bien définie, sans interrogations sur l'insertion possible dans un cours.

Il existait déjà des logiciels de qualité reconnus, ceux de l'INRP par exemple, mais on en était encore aux débuts de l'informatique pédagogique⁸. On a pu se demander si l'opinion sur les logiciels ne renvoyait pas pour une part aux formateurs, par ailleurs appréciés. Un certain nombre d'entre eux avaient bénéficié d'un stage « lourd » d'un an à temps plein les années précédentes⁶.

Ces stages donnaient pleinement satisfaction sur le plan informatique mais provoquaient parfois chez les stagiaires, pour des raisons et des responsabilités diverses mais communes, une déception sur le plan des usages pédagogiques. Répétons-le, l'offre logicielle n'était pas ce qu'elle est devenue. On connaissait à peine le traitement de texte. Et il n'y avait pas le recul que donnent le temps et l'expérience.

Dans son rapport de stage, un formateur IPT s'étonnait du manque fréquent d'explications concernant les utilisations prévues par le concepteur et se demandait, non sans humour, si c'était voulu ! Quelques scénarios pédagogiques, ne prétendant pas à l'exhaustivité, étaient souhaités. Étaient posées les questions de l'insertion d'un



© Daniel Coulom

<http://dcmoto.free.fr/>

8. Voir L'expérience des « 58 lycées », de Jacques Baudé : <http://www.societe-informatique-de-france.fr/wp-content/uploads/2014/10/1024-4-baude.pdf>, http://www.epi.asso.fr/revue/histo/h70-58lycees_jb.htm

logiciel dans un cours, de la pédagogie avec l'informatique (utilisation raisonnée de l'outil, faire autrement, faire mieux, faire ce que l'on ne peut pas faire autrement), de la pédagogie de l'informatique.

Des « conseils » étaient donnés pour les futurs logiciels. En vrac : ne plus abuser du crayon optique (de plus pas très fiable), de la couleur, de la musique, la forme pouvant servir à vouloir masquer les insuffisances de fond ; soigner la clarté des messages ; éviter les logiciels trop fermés, permettre les retours en arrière ; soigner la documentation... des recommandations qui peuvent ça et là avoir encore une actualité.

Le matériel et les constructeurs

Le concept de nanoréseau avait séduit formateurs et stagiaires. Une nouveauté à une époque où le poste de travail isolé était souvent la règle. Le réseau n'était pas encore le « mode d'existence quasi unique de l'informatique » mais n'allait pas tarder à le devenir.

Globalement, le matériel était arrivé à temps et avait fonctionné, ce qui n'était pas forcément évident quand l'opération avait été lancée (annonce faite en janvier pour juillet si l'on fait abstraction des stages des vacances de printemps peu nombreux). Il y eut le volant « normal » de pannes. Mais des perturbations non négligeables lors de la première semaine de juillet, là où les livraisons furent tardives (fin juin, ou ladite semaine) et où donc les formateurs n'avaient pas pu se familiariser avec la nouvelle version du nanoréseau.

Les constructeurs n'avaient pas sérieusement pensé ni donc mis en place un véritable service après-vente pendant l'été et à la rentrée. La situation fut contrastée. Certains constructeurs avaient, dans les conditions de l'opération, rempli leur contrat, dont l'un qui avait beaucoup d'appareils dans l'académie. Parmi les autres, l'un n'avait mis en place un dispositif sérieux que dans le courant de novembre, on en eut un qui fut insaisissable pendant de longs mois. Et un autre livrait encore fin octobre la version et la documentation de Pâques du nanoréseau.

Et il y eut des problèmes avec l'impression et des matériels jugés un peu fragiles, les installations auraient gagné à être gainées et l'utilisation des lecteurs de cassette fut dissuasive. Les temps ont changé depuis. Le transport des matériels ne s'est pas fait sans difficultés ni pertes, au détriment des établissements destinataires ultimes. On signalera le travail accompli par les personnels et les responsables (proveurs, principaux et directeurs) ainsi que le bon accueil réservé aux stagiaires.

Les stagiaires

6500 enseignants de l'académie ont été initiés, des centaines de candidatures n'ont pas pu être satisfaites, surtout dans le primaire. De tels chiffres, non évidents au moment de l'appel à candidatures, devaient être pris en compte, révélateurs qu'ils

étaient d'une attente et d'un besoin considérables, de la place que l'informatique allait prendre dans le système éducatif, et ailleurs. D'une manière générale, l'accent était mis sur une très forte motivation des stagiaires, les réfractaires étant à la fois peu nombreux et manifestement à contre-courant.

Les groupes avaient été soit hétérogènes, de par les catégories d'enseignants ou les aptitudes liées en partie au passé informatique, soit homogènes. Les appréciations sur le bienfait de telles situations étaient contradictoires.

Les bilans mentionnaient l'ouverture d'esprit des stagiaires, une bonne assimilation relative si l'on tient compte du rythme imposé, une bonne ambiance de groupe.

Les stagiaires se déclaraient dans l'ensemble satisfaits (voir les cibles et des remarques du type « un stage où l'on fait quelque chose ! »).

Une semaine de 48 h

Bien que la demande d'une formation sur le temps de travail soit restée forte, la formule d'une semaine rémunérée pendant les vacances, en dehors de préoccupations et des contraintes de l'année scolaire, avait été appréciée.

Néanmoins, 6 jours consécutifs de 8 heures, c'était très, trop intense. Des groupes de 20 stagiaires c'était beaucoup. Des problèmes de fatigue et d'assimilation s'étaient posés malgré la grande motivation d'ensemble. Les formateurs s'étaient efforcés de varier les situations, d'alterner le cours magistral et l'étude libre. Ils avaient privilégié une certaine directivité, une présentation structurée et organisée des connaissances plutôt qu'une pédagogie de la découverte qui nécessite le temps.

Quelques mots pour conclure

On pouvait donc affirmer que ces stages IPT organisés pendant les vacances scolaires 1985 avaient été globalement positifs, voire même très positifs. En témoignait le haut degré de satisfaction exprimé par les stagiaires (et les responsables académiques).

Dans ce cadre, certaines limites et difficultés avaient été constatées. L'opération avait connu une réelle fébrilité, surtout à Pâques et au début de l'été. Cela avait été sans doute bénéfique du point de vue psychologique et avait eu un impact certain sur l'opinion publique et enseignante. Mais aussi quelques conséquences fâcheuses sur la préparation des contenus des stages et l'implantation des matériels. On aurait peut-être pu se hâter lentement. C'est le lot des opérations « coup de poing » que de désorganiser les affaires courantes. Il ne faut donc pas en abuser. Mais elles créent une dynamique et un enthousiasme gages d'efficacité. Le succès fut réel, dépassant les espérances initiales. En la circonstance, on peut dire que l'Éducation nationale avait réussi un « joli coup ».

Le plan IPT a suscité, et suscite encore, des débats vifs. Rappelons les chiffres. Le plan IPT ce fut 110 000 enseignants formés, rémunérés pour leur participation à des

stages de 48 heures pendant les vacances de printemps, d'été ou de la Toussaint : une initiation, une sensibilisation de masse à l'informatique⁹, une entrée remarquée et significative dans le futur. 5 500 000 heures de formation. Certains parlent d'échec. Des échecs comme cela on aurait tendance à en redemander !

9. Le mot informatique est très présent dans cet article ; il était fort usité à l'époque...



Michael Stonebraker, ACM Turing Award pour ses contributions au domaine de la gestion de données

Patrick Valduriez¹

Michael Stonebraker, chercheur au Massachusetts Institute of Technology (USA), vient de remporter le prestigieux ACM Turing Award, souvent considéré comme « le prix Nobel de l'informatique ». Dans son annonce du 25 mars 2015, l'ACM précise que Stonebraker « a inventé de nombreux concepts qui sont utilisés dans presque tous les systèmes de bases de données modernes... ». Cette reconnaissance au plus haut niveau international me donne l'occasion de donner un éclairage sur la place singulière du domaine de la gestion de données dans la recherche en informatique.

La gestion de données se préoccupe du stockage, de l'organisation, de la recherche et de la manipulation de données de toutes sortes, souvent très grandes et complexes. C'est aujourd'hui un domaine majeur de l'informatique, avec à la fois une grande communauté de recherche et une industrie forte. En particulier, le transfert continu de la recherche vers l'industrie a conduit au développement des systèmes de gestion de bases de données (SGBD), au cœur de tout système d'information moderne.

Le principe fondateur de la gestion de données est l'*indépendance des données*, qui permet de travailler avec les données à un niveau conceptuel, tout en ignorant les détails d'implémentation. Le modèle relationnel, en s'appuyant sur une théorie

1. Directeur de Recherche Inria, ACM Fellow 2012.

simple et solide (ensembles, logique du 1^{er} ordre) a révolutionné la façon de concevoir les SGBD. Il a aussi favorisé le nécessaire rapprochement entre théorie et pratique des bases de données, qui s'est concrétisé par la fusion de grandes conférences internationales comme ACM SIGMOD/PODS et EDBT/ICDT.

L'innovation majeure des SGBD relationnels a été de permettre la manipulation de données avec des langages de requêtes déclaratifs, intégrant des concepts puissants comme les transactions ou les règles d'intégrité, et facilitant l'optimisation automatique. Arrivés sur le marché dans les années 1980, les SGBD relationnels ont remarquablement réussi le test du temps, par l'ajout régulier de nouvelles fonctionnalités (p. ex. sécurité, auto-réglage), de nouveaux types de données (p. ex. objet, XML ou JSON) et en s'adaptant à toutes sortes de plateformes, depuis les appareils mobiles (p. ex. smartphones) jusqu'aux très grands clusters dans des environnements distribués.

Aujourd'hui, avec les nouveaux défis du *big data* et du *cloud*, la gestion de données doit être réinventée, tant les besoins des utilisateurs et des applications sont divers et ne peuvent plus s'accommoder de l'aspect « taille unique » des SGBD relationnels. La recherche en gestion de données devient alors pluridisciplinaire, associant notamment chercheurs et grands producteurs de données pour mieux étudier leurs données (*big data analytics*). En France, ces défis sont au cœur d'initiatives pluridisciplinaires récentes comme le défi CNRS Mastodons (grandes masses de données scientifiques) et le GdR MaDICS (Masses de Données, Informations et Connaissances en Sciences).

En 40 ans de carrière, Stonebraker a profondément marqué le domaine des SGBD, depuis le relationnel jusqu'au *big data*. Sa récompense s'ajoute à celles de trois autres *ACM Turing Awards* du domaine : Charles Bachman (1973) pour ses contributions aux SGBD navigationnels, Edgar Frank Codd (1981) pour l'invention du modèle relationnel et James Gray (1998) pour ses contributions aux SGBD et au transactionnel.

Stonebraker a d'abord été un pionnier dans la conception de SGBD relationnels, en développant deux des projets de recherche les plus influents au monde, Ingres et Postgres (relationnel et objet), à l'université de Berkeley. Puis il a conçu de nouveaux SGBD comme Mariposa (SGBD distribué), Streambase (SGBD pour flux de données), Vertica (SGBD orienté-colonne pour le décisionnel) et SciDB (SGBD orienté-tableau pour les applications scientifiques). Il a été aussi un entrepreneur exceptionnel, en créant neuf startups autour de ses projets dont Relational Technology Inc. (Ingres), devenue aujourd'hui Ingres Corp., Illustra (à partir de Postgres) acquise par Informix, Cohera (à partir de Mariposa) acquise par PeopleSoft, Streambase acquise par Tibco et Vertica acquise par HP. Enfin, il a créé deux grands projets en logiciel libre : PostgreSQL et SciDB. Aujourd'hui, Stonebraker poursuit ses travaux au MIT autour des systèmes NoSQL.



Bilan du prix de thèse Gilles Kahn 2014

Michel Riveill

Le prix de thèse Gilles Kahn 2014, décerné par la SIF et patronné par l'Académie des sciences, a été attribué à :

TANCRÈDE LEPOINT

pour sa thèse « Design and Implementation of Lattice-Based Cryptography ». Cette thèse, soutenue à l'École Normale Supérieure et l'Université du Luxembourg, a été préparée au LIENS (École Normale Supérieure, France) et à la Faculty of Sciences, Technology and Communication (Université du Luxembourg) sous la direction de David Pointcheval et Jean-Sébastien Coron et a été financée dans le cadre d'un contrat CIFRE par la société CryptoExperts.

Les deux deuxièmes prix ont été attribués à (par ordre alphabétique) :

YVONNE JANSEN

pour sa thèse « Physical and Tangible Information Visualization », soutenue à l'Université Paris-Sud, et préparée au sein de l'Équipe-projet Aviz (Inria Saclay Île-de-France) sous la direction de Jean-Daniel Fekete et Pierre Dragicevic.

MICHAËL THOMAZO

pour sa thèse « Conjunctive query answering under existential rules – Decidability, complexity and algorithms » soutenue à l'Université de Montpellier, et préparée au sein de l'équipe GraphIK, Laboratoire d'Informatique, Robotique et Micro-Électronique de Montpellier (LIRMM) et Inria Sophia-Antipolis Méditerranée, sous la direction de Marie-Laure Mugnier et Jean-François Baget.

Les thèses des lauréats sont accessibles sur le site de la SIF¹.

Le jury a reçu cette année 47 dossiers couvrant un très large spectre de travaux de recherche et en provenance de la plupart des laboratoires de recherche en informatique.

Merci à tous pour la qualité de vos travaux !

1. <http://www.societe-informatique-de-france.fr/recherche/prix-de-these-gilles-kahn/prix-de-these-2014-2/>



Conception et implémentation de cryptographie à base de réseaux

Tancrède Lepoint¹

Prix de thèse Gilles Kahn 2014

Tancrède Lepoint a soutenu sa thèse² en juin 2014. Celle-ci a été préparée au LIENS (École Normale Supérieure, France) et à la Faculty of Sciences, Technology and Communication (Université du Luxembourg), sous la direction de David Pointcheval et Jean-Sébastien Coron, et a été financée dans le cadre d'un contrat CIFRE par la société CryptoExperts.



La cryptographie à base de réseaux Euclidiens est aujourd'hui un domaine scientifique en pleine expansion. Son attractivité est plurielle : les opérations sont élémentaires, sa complexité asymptotique quasi-optimale, elle résiste aux ordinateurs quantiques (contrairement aux algorithmes actuels qui seront obsolètes dès l'existence de calculateurs quantiques suffisamment performants), mais surtout rend possible de nouvelles applications qui pourraient permettre à terme d'obtenir un *Cloud* complètement sécurisé et respectueux de la vie privée. Exemple frappant : vous pourriez effectuer une requête sur un moteur de recherche *sans que celui-ci ait connaissance* de ce que vous avez recherché (voir Figure 1).

1. <https://www.cryptoexperts.com/tlepoint>

2. Consultable à l'adresse <https://tel.archives-ouvertes.fr/tel-01069864>

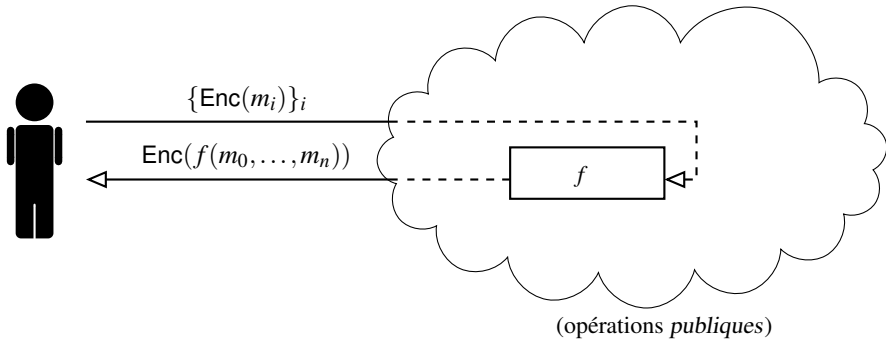


FIGURE 1. Avec un schéma de chiffrement complètement homomorphe Enc , une personne peut envoyer ses données $\{m_i\}_i$ au Cloud sous forme chiffrée. Celui-ci peut alors les manipuler publiquement (sans les connaître) et renvoyer l'évaluation de *n'importe quel algorithme* f sur ces dernières.

Considéré comme le Saint Graal de la cryptographie [14], l'existence d'un tel chiffrement n'a été prouvée possible qu'en 2009 [12] et utilise les réseaux Euclidiens. Il s'agit du *chiffrement complètement homomorphe*. Plus généralement, la richesse mathématique des réseaux Euclidiens apparaît être un composant clé pour permettre la manipulation publique de données secrètes, mais les constructions cryptographiques à visée pratique pour un niveau de sécurité fixé se révèlent être particulièrement inefficaces. L'objectif de la thèse est donc de réduire l'écart entre la théorie et la pratique de la cryptographie à base de réseaux.

Dans [10], nous proposons un nouveau schéma de signature compact (avec des signatures de l'ordre de 5000 bits), performant, sûr et adapté aux environnements contraints. Signer numériquement des données permet d'en garantir l'intégrité et l'authenticité, et est nécessaire à toute communication numérique sécurisée. Des améliorations théoriques et optimisations pratiques ont permis d'obtenir une signature numérique basée sur les réseaux aussi efficace (voire plus efficace) que celles actuellement utilisées. Moins de 18 mois après publication, notre signature digitale a été intégrée à *strongSwan* (une solution VPN open source) [15] et permet de s'authentifier de manière sécurisée et pérenne, indépendamment des avancées en physique quantique.

Nos contributions principales au chiffrement homomorphe consistent à en améliorer l'efficacité au travers du parallélisme (SIMD), d'optimisations algorithmiques

théoriques et pratiques [4, 8] et à la réduction de problèmes d'optimisation au problème SAT [13]. Il s'agit de notre sujet de recherche principal : nous sommes impliqués dans plusieurs projets européens visant à déployer la cryptographie homomorphe dans les produits d'ici l'horizon 2020 [1], et travaillons tant sur des aspects théoriques (complexité APX [3]) que pratiques (bibliothèque efficace en C++ [2]).

Finalement, nous construisons et implémentons des applications multilinéaires cryptographiques dans [7]. Cette primitive très récente (la première construction datant également de 2013) a de nombreuses conséquences inattendues et à très fort potentiel, comme par exemple l'existence de réelle *obfuscation*³ logicielle [11]. Cette nouvelle primitive vit en ce moment même un véritable naufrage à cause de nombreuses attaques (auxquelles nous avons aussi participé) [5, 6]. Récemment, nous avons amélioré notre schéma dans [9] : ce dernier se révèle être l'*unique* construction rendant possible de nombreuses constructions cryptographiques théoriques.

Références

- [1] Homomorphic encryption applications and technology, 2015. <https://heat-project.eu/>.
- [2] Carlos Aguilar-Melchor, Joris Barrier, Serge Guelton, Adrien Guinet, Marc-Olivier Killijian, and Tancrède Lepoint. NTLlib : NTT-based fast lattice library, 2015.
- [3] Fabrice Benhamouda, Tancrède Lepoint, and Hang Zhou. Optimization of bootstrapping in circuits. 2015.
- [4] Jung Hee Cheon, Jean-Sébastien Coron, Jinsu Kim, Moon Sung Lee, Tancrède Lepoint, Mehdi Tibouchi, and Aaram Yun. Batch fully homomorphic encryption over the integers. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 315–335. Springer, 2013.
- [5] Jung Hee Cheon, Kyoohyung Han, Changmin Lee, Hansol Ryu, and Damien Stehlé. Cryptanalysis of the multilinear map over the integers. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 3–12. Springer, 2015.
- [6] Jean-Sébastien Coron, Craig Gentry, Shai Halevi, Tancrède Lepoint, Hemanta K. Maji, Eric Miles, Mariana Raykova, Amit Sahai, and Mehdi Tibouchi. Zeroizing without low-level zeroes : New attacks on multilinear maps and their limitations. 2015. To appear at CRYPTO 2015.
- [7] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Practical multilinear maps over the integers. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 476–493. Springer, 2013.
- [8] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Scale-invariant fully homomorphic encryption over the integers. In Hugo Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 311–328. Springer, 2014.
- [9] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. New multilinear maps over the integers. 2015. To appear at CRYPTO 2015.
- [10] Léo Ducas, Alain Durmus, Tancrède Lepoint, and Vadim Lyubashevsky. Lattice signatures and bimodal Gaussians. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 40–56. Springer, 2013.

3. Technique d'obscurcissement de code, rendant celui-ci particulièrement difficile à comprendre.

- [11] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *FOCS 2013*, pages 40–49. IEEE Computer Society, 2013.
- [12] Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, *STOC 2009*, pages 169–178. ACM, 2009.
- [13] Tancrede Lepoint and Pascal Paillier. On the minimal number of bootstrappings in homomorphic circuits. In Andrew A. Adams, Michael Brenner, and Matthew Smith, editors, *FC 2013 Workshops, USEC and WAHC 2013*, volume 7862 of *LNCS*, pages 189–200. Springer, 2013.
- [14] Daniele Micciancio. A first glimpse of cryptography’s Holy Grail. *Commun. ACM*, 53(3):96, 2010.
- [15] Andreas Steffen et al. *strongSwan (Version 5.2.2)*, 2015. <https://www.strongswan.org/>.



Visualisation physique et tangible de l'information

Yvonne Jansen¹

Accessit du prix de thèse Gilles Kahn 2014

Yvonne Jansen a soutenu sa thèse en mars 2014 à l'université Paris-Sud XI, sous la direction de Jean-Daniel Fekete et Pierre Dragicevic. Elle effectue actuellement un stage post-doctoral à l'université de Copenhague (Danemark).



La représentation visuelle de l'information pour l'augmentation de l'intellect humain a une longue histoire. Les représentations externes permettent de raisonner sur l'information abstraite à l'aide de la vision pour détecter les tendances, les corrélations et les anomalies. Le besoin de visualiser des données remonte aux premières civilisations sous une forme physique. Un aspect remarquable de ces anciennes représentations de données physiques, telles que les jetons d'argile mésopotamiens, est qu'elles peuvent être librement manipulées et réorganisées. L'invention des tablettes d'argiles, puis du papier, a grandement facilité la complexification, la production et la diffusion de représentations visuelles de données, mais celles-ci sont devenues figées : on peut lire, mais non plus manipuler. Par la suite les ordinateurs graphiques ont réintroduit la manipulabilité à travers l'interactivité qui, couplée au traitement automatique, offre des possibilités infinies pour l'exploration de données. Cependant les visualisations interactives demeurent virtuelles et leur manipulation nécessite un apprentissage préalable.

1. <http://yvonnejansen.me>

Avec le développement des outils de fabrication numérique et la démocratisation des « *fab labs* » (lieux permettant à tout un chacun de produire des objets complexes), les représentations physiques de données sont de retour. On les trouve aujourd'hui dans de nombreux musées de sciences et d'art sous forme de « sculptures de données » (voir Figure 1), ainsi que dans des vidéos éducatives en ligne (voir <http://dataphys.org/list/> pour en savoir plus). Or, la fabrication numérique est loin d'être la seule opportunité offerte par les nouvelles technologies. Les technologies se diversifient et fournissent maintenant un large éventail d'environnements interactifs dont la visualisation d'information peut tirer parti. Cette thèse a pour but de poser les bases d'un nouveau domaine de recherche scientifique, la « physicalisation de données », dont l'objectif est d'examiner les avantages uniques apportés par les représentations physiques de données, et de comprendre comment exploiter au mieux leur potentiel.



FIGURE 1. Sculpture de données « Global Cities » au Tate Modern, Londres, 2007.

Cette thèse fournit donc une première exploration systématique des bénéfices possibles et des limitations actuelles des visualisations physiques et tangibles (voir Figure 2), présente des modèles pour décrire, comparer et critiquer de tels systèmes [1], et propose également un outil pour faciliter la création de tels systèmes [3]. Les résultats montrent qu'un avantage pertinent est la possibilité de transférer au monde physique une partie des efforts cognitifs nécessaires à l'analyse de données [5]. Les capacités humaines à percevoir et manipuler des objets physiques facilitent l'exploration des visualisations d'information physiques, tandis que l'exploration des



FIGURE 2. Histogramme physique modulaire [1] et télécommandes tangibles [2].

visualisations d'information virtuelles exige un apprentissage explicite. Avec une représentation de données physique, l'utilisateur s'aide de ses doigts par exemple pour faciliter la navigation dans les données ainsi que la mémorisation.

Un nouveau domaine de recherche scientifique se dessine [4], la « physicalisation de données », un champ de recherche à la croisée de l'informatique tangible et de la visualisation d'information qui a pour but de faciliter l'exploration, la compréhension et la communication de données par le biais de représentations physiques et numériques de données. Ce domaine est encore jeune mais offre des perspectives considérables, notamment pour engager le grand public et promouvoir la littératie des données.

Références

- [1] Yvonne Jansen, Pierre Dragicevic, *An Interaction Model for Visualizations Beyond The Desktop*, IEEE Transactions on Visualization and Computer Graphics 19 (12), pp. 2396 - 2405, 2013.
- [2] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete, *Tangible remote controllers for wall-size displays*, Proceedings of the 2012 annual conference on Human Factors in Computing Systems (CHI'12), pp. 2865-2874, 2012.
- [3] Saiganesh Swaminathan, Conglei Shi, Yvonne Jansen, Pierre Dragicevic, Lora Oehlberg, and Jean-Daniel Fekete, *Supporting the Design and Fabrication of Physical Visualizations*, Proceedings of the 2014 annual conference on Human Factors in Computing Systems (CHI'14), pp. 3845-3854, 2014.
- [4] Yvonne Jansen, Pierre Dragicevic, Petra Isenberg, Jason Alexander, Abhijit Karnik, Johan Kildal, Sriram Subramanian, Kasper Hornbæk, *Opportunities and Challenges for Data Physicalization*, Proceedings of the 2015 annual conference on Human factors in Computing Systems (CHI'15), pp. 3227-3236, 2015.
- [5] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete, *Evaluating the Efficiency of Physical Visualizations* Proceedings of the 2013 annual conference on Human factors in Computing Systems (CHI'13), pp. 2593-2602, 2013.



Réponse à des requêtes conjonctives en présence de règles existentielles – décidabilité, complexité et algorithmes

Michaël Thomazo¹

Accessit du prix de thèse Gilles Kahn 2014

Michaël Thomazo a soutenu sa thèse² en octobre 2013 à l'Université Montpellier 2, sous la direction de Jean-François Baget et Marie-Laure Mugnier, au sein de l'équipe Inria GraphIK. Il est actuellement boursier Alexander von Humboldt à la Technische Universität Dresden, en Allemagne.



Les données sont omniprésentes dans notre vie : catalogues de produits en ligne, statistiques de criminalité, données médicales... Les exploiter de manière satisfaisante présente de multiples défis : garantie de la confidentialité, passage à l'échelle, interrogation sémantique... Ma thèse a porté sur ce dernier point : comment enrichir l'interrogation des données à l'aide de connaissances générales d'un domaine ? Le cadre est le suivant : des données sont stockées, par exemple dans une base de données relationnelle, en utilisant un certain vocabulaire. Les liens sémantiques entre les différents termes de ce vocabulaire sont précisés à l'aide d'une *ontologie*. Le but est de répondre à une requête exprimée grâce aux termes de l'ontologie.

1. TU Dresden, Germany.

2. Consultable à l'adresse <https://tel.archives-ouvertes.fr/tel-00925722>

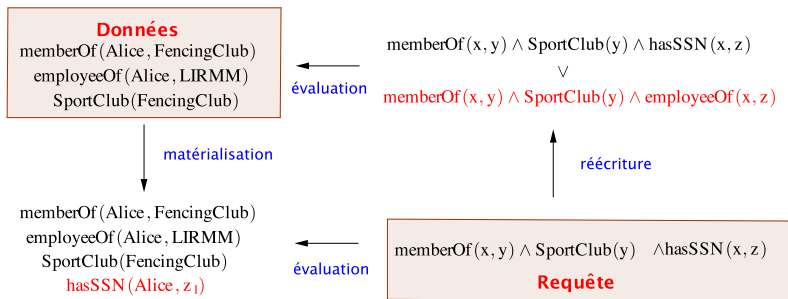


FIGURE 1. Matérialisation et réécriture avec la règle $\forall x \forall y (\text{employeeOf}(x, y) \rightarrow \exists z \text{hasSSN}(x, z))$

Le langage de représentation d'ontologie que j'ai étudié durant ma thèse est celui des règles existentielles. Celles-ci possèdent une syntaxe flexible et permettent de décrire des individus qui ne sont pas initialement présents dans la base de données, ce qui est crucial d'un point de vue modélisation. Étant donné des données D , un ensemble de règles existentielles \mathcal{R} et une requête q , le problème est de savoir si q est une conséquence logique de D et de \mathcal{R} . Deux grandes approches ont été proposées dans la littérature. La première est une approche de matérialisation, où les règles sont appliquées jusqu'à saturation – on dit que l'on construit le modèle canonique. Une fois ce modèle canonique construit, il suffit d'effectuer une interrogation classique. La deuxième approche est la réécriture de requêtes, où la requête initiale est reformulée en une requête qui est évaluée sur les données initiales. La dualité entre ces deux approches est représentée figure 1, où soit les données sont modifiées (à gauche), soit la requête est modifiée (à droite).

Les procédures résultant de ces deux approches ne sont que des procédures de semi-décidabilité. Dans le cas de la matérialisation par exemple, le modèle canonique peut-être infini. Un important travail de recherche a donc été mené par la communauté pour définir des restrictions sur les ontologies qui assurent la décidabilité du problème. Trois principaux critères sont connus : le modèle canonique est fini, ou il est infini mais de largeur arborescente bornée, ou il existe une réécriture finie en union de requêtes conjonctives.

Dans ma thèse, je me suis intéressé aux deux approches. Premièrement, j'ai étudié les cas où le modèle canonique n'est pas fini, mais où il est possible d'en construire une décomposition arborescente de largeur bornée selon une procédure donnée. J'ai alors montré comment construire une représentation finie du modèle canonique et l'interroger à l'aide d'une opération *ad hoc*, mais de même complexité qu'une simple recherche d'homomorphisme. Cela a fourni le premier algorithme optimal (dans le

pire des cas) pour de nombreuses classes de règles existentielles. Des travaux ultérieurs permettent de réutiliser un moteur Datalog, ouvrant la voie à des implémentations efficaces. De plus, la représentation finie proposée peut servir de point de départ à des développements dans d'autres cadres, en particulier celui du raisonnement en présence d'inconsistances.

Dans un deuxième temps, je me suis intéressé aux approches de réécritures, et en particulier à l'explosion combinatoire qui prend place lorsque de grandes hiérarchies de classes (tout chat est un mammifère, tout mammifère est un animal...) sont présentes dans les ontologies. Des premières expérimentations ont montré que l'approche utilisée, à base de factorisation, permet un gain d'un à deux ordres de grandeur par rapport à l'état de l'art. Je m'intéresse actuellement à des implémentations efficaces des algorithmes présentés dans ma thèse.



Prix de thèse Gilles Kahn 2015

Michel Riveill

PRIX DE THÈSE GILLES KAHN 2015

patronné par l'Académie des sciences et décerné par la SIF

[http://www.societe-informatique-de-france.fr/recherche/
prix-de-these-gilles-kahn/](http://www.societe-informatique-de-france.fr/recherche/prix-de-these-gilles-kahn/)

Date limite de candidature : 10 septembre 2015.

Le prix de la Société informatique de France (SIF) a été créé en 1998 pour récompenser chaque année une excellente thèse en Informatique. Gilles Kahn, qui a présidé les trois premiers jurys du prix, était convaincu de l'intérêt de promouvoir les jeunes talents les plus prometteurs de notre discipline. En son honneur, le prix a pris depuis 2007 le nom de Prix de thèse Gilles Kahn et est patronné par l'Académie des sciences qui rend ainsi hommage à un de ses membres éminents.

La SIF souhaite par ce prix promouvoir toutes les facettes de l'informatique, des travaux fondamentaux aux travaux appliqués ayant donné lieu à transfert industriel, de ceux réalisés dans les grands centres à ceux réalisés dans des centres plus modestes. L'objectif de ce prix est de dynamiser et de motiver de jeunes chercheurs en les récompensant, et de faire connaître à l'ensemble de la communauté informatique d'excellents travaux de recherche. Un jury d'universitaires et de chercheurs, présidé par Michel Riveill, sélectionnera parmi les thèses soutenues au cours de l'année universitaire celle qui recevra ce prix. En outre, le jury pourra également distinguer, s'il le souhaite, au plus deux accessits.

La remise officielle du prix se fera début 2016 au cours d'une cérémonie associant la SIF et l'Académie des sciences. À cette occasion, le récipiendaire se verra

remettre un chèque de 1500 euros et chacun des autres lauréats éventuels un chèque de 500 euros. Tous seront également invités à présenter leurs travaux à l'ensemble de la communauté scientifique. Sous réserve de remplir les conditions de candidature, les lauréats au prix de thèse Gilles Kahn seront considérés comme candidats à la nomination par Inria pour le prix Cor Baayen de l'ERCIM.

Les critères pris en compte par le jury pour sélectionner les lauréats sont notamment l'originalité des résultats, l'originalité du domaine et des méthodes utilisées, l'importance et l'impact des résultats obtenus, et bien évidemment la qualité de la rédaction puisqu'il s'agit de récompenser non seulement un travail mais surtout une thèse.

En 2014, sous la présidence de Michel Riveill et avec l'assistance de Nathalie Bertrand, secrétaire du prix, le jury était constitué de Patricia Bouyer-Decitre, André Chailloux (prix 2011), Jérémie Chalopin, Delphine Demange (prix 2013), Mathieu Feuillet (prix 2012), Patrick Girard, Jean-Marc Jézéquel, Delia Kesner, Jacques-Olivier Lachaud, Jean Ponce, Robert Cori, Nicolas Trotignon, Claudia Runcancio, Hélène Touzet, Christine Solnon, Olivier H. Roux, Pierre Sens, Ye-Qiong Song et Florence Sèdes. Comme c'est l'usage, le jury sera partiellement renouvelé pour le prix 2015, le jury étant totalement renouvelé tous les trois ans.

Calendrier

- 15 juillet 2015 : ouverture de l'interface web de soumission
- 10 septembre 2015 : date limite de dépôt des candidatures
- 1^{er} décembre 2015 : notification des résultats aux candidats
- début 2016 : remise officielle du prix lors de l'Assemblée générale de la SIF

Interface web de candidature

<http://prix-sif.inria.fr/>

Dossier de candidature

Peut candidater tout étudiant ayant soutenu son doctorat d'informatique dans une école ou université française entre le 1^{er} septembre 2014 et le 31 août 2015. Toute candidature devra être explicitement soutenue par le directeur de thèse ou un des co-directeurs. Il n'est pas permis à un même encadrant de soutenir deux candidats.

Tous les documents doivent être déposés, sous forme de fichiers PDF exclusivement, par le biais de l'interface web. En cas de problèmes à utiliser l'interface, ou pour toute autre question concernant le prix, les candidats sont invités à contacter par courrier électronique la secrétaire du prix, Nathalie Bertrand (nathalie.bertrand@inria.fr).

Chaque dossier doit notamment comprendre :

- la thèse (en PDF),
- les rapports de pré-soutenance des rapporteurs (scannés, en PDF),

- le rapport de soutenance (scanné, en PDF),
- une lettre appuyant la candidature au prix de thèse, directement envoyée par le(s) directeur(s) de thèse,
- des rapports complémentaires que le candidat jugerait utile de fournir au jury, envoyés par les personnes concernées.

Le formulaire en ligne demande également de saisir certaines informations : un résumé de deux pages de la thèse, un CV d'une page maximum ainsi qu'une liste de publications.



Moins de 20 cartes

Jean-Paul Delahaye¹

La rubrique « Récréation informatique » propose une petite énigme algorithmique ou à propos d'un thème de mathématiques discrètes susceptible d'intéresser un lecteur de 1024. La solution est donnée dans le numéro suivant.

Rappel et solution du problème précédent

S'APPROCHER DU BUT

Ce petit problème m'a été proposé par Aurélien Géron. Vous êtes placé en un point O , et à une distance d'une unité, il y a un point I . Vous devez vous rendre le plus près possible d'un point X sur le segment de droite OI (par exemple à moins d'un millièmme d'unité de X), mais vous ne pouvez vous déplacer que par sauts successifs selon la règle simple suivante :

Vous choisissez le point O ou le point I , et votre saut vous conduit alors au point situé au milieu entre le point P ou vous êtes, et le point choisi.

Si, par exemple, vous partez de O et choisissez la séquence I, O, I, I , cela vous mène respectivement aux positions $0,5 - 0,25 - 0,625 - 0,8125$.

Fixons maintenant un objectif X à atteindre à moins d'un millièmme (ou un millièmme, etc.). Par exemple X situé à $1/\pi$ de O . Quelle est la plus courte séquence de choix O ou I qui permet de réussir ? Vous noterez qu'appliquer le principe naturel « si je suis entre O et X , je vais vers I , si je suis entre X et I , je vais vers O » ne marche pas du tout.

1. Université de Lille 1, Sciences et Technologies, Centre de recherche en informatique signal et automatique de Lille (CRISTAL), UMR 9189 CNRS, Bât M3-ext, 59655 Villeneuve d'Ascq Cedex. E-mail : jean-paul.delahaye@univ-lille1.fr.

SOLUTION. Merci aux lecteurs qui m'ont fait parvenir la bonne réponse. Ce sont dans l'ordre d'arrivée des messages : Éric Wegrzynowski, Raymond Namyst, Julien Bernard, Patrick Kamnang, Dimitri Gallois.

Pour aller vers le point situé à X de O (X un nombre entre 0 et 1), voici comment procéder :

- (a) on écrit X en base 2, $X=0,110101101$ par exemple ;
- (b) on lit les chiffres à l'envers ce qui donne 1 0 1 1 0 1 0 1 1 ;
- (c) on interprète les '0' par « aller vers O », et les '1' par « aller vers I ».

Ici, cela donne : vers I, vers O, vers I, vers I, vers O, vers I, vers O, vers I, vers I. Le nombre $1/\pi$ s'écrit $0,0101000101$ en base 2 au millième près car :

$$1/\pi - (1/4 + 1/16 + 1/256 + 1/1024) = 0,000927073684.$$

On en déduit que pour s'en approcher à moins d'un millième, il faut choisir les mouvements suivants : vers I, vers O, vers I, vers O, vers O, vers O, vers I, vers O, vers I, vers O.

Démontrons que ce procédé fonctionne.

(A) Si je suis à X de O et que je vais vers O, je passe en $X/2$, ce qui revient à placer un 0 entre la virgule et les chiffres de X écrit en base 2 ($0,11110101$ devient par exemple $0,011110101$).

(B) Si je suis en X que je vais vers I, la distance à I qui était $(1-X)$ devient $(1-X)/2$, et donc le point où j'arrive est à $1 - (1-X)/2 = 1/2 + X/2$ de O. Ce nombre s'écrit en base 2 en plaçant un '1' entre la virgule et les chiffres de X écrits en base 2 ($0,1011001$ devient par exemple $0,11011001$).

En résumé, si on calcule en base 2 :

- quand on va vers O, on insère un '0' après la virgule ;
- quand on va vers I, on insère un '1' après la virgule.

Examinons maintenant un exemple avec plusieurs déplacements vers O ou vers I. Si on part de O et qu'on va successivement vers I, vers I, vers O, vers O, vers I, vers O, vers I (ce qu'on peut noter IIOOIOI), on obtient successivement

0,
 0,1 (insertion d'un 1 après la virgule)
 0,11 (insertion d'un 1 après la virgule)
 0,011 (insertion d'un 0 après la virgule)
 0,0011 (insertion d'un 0 après la virgule)
 0,10011 (insertion d'un 1 après la virgule)
 0,010011 (insertion d'un 0 après la virgule)
 0,1010011 (insertion d'un 1 après la virgule)

Comme on le voit, le nombre obtenu a pour développement binaire $0,1010011$, ce qu'on obtient en inversant l'ordre des opérations « vers O » et « vers I » (et en assimilant les 'O' aux '0', et les 'I' aux '1').

Cela signifie que pour aller vers un nombre X et s'en approcher avec une précision de $1/2^n$, il faut écrire X en base 2 jusqu'au n -ième chiffre après la virgule puis lire à l'envers les chiffres obtenus pour connaître les opérations de saut à exécuter.

Assez étrangement, pour s'approcher d'un point donné il faut donc commencer par les derniers chiffres de la valeur approchée qu'on souhaite obtenir à la fin. C'est un peu étonnant, car le plus souvent – et en particulier pour se déplacer sur une carte vers un point précis – on procédera progressivement. On s'approche en gros du point visé (on va, par exemple, dans la capitale du pays du lieu qu'on cherche à approcher), puis on s'occupe plus finement du but (on va vers la grande ville la plus proche du but), puis encore plus finement (on va vers le village le plus proche du but), etc. Ici, paradoxalement, il faut procéder par une méthode inverse, et s'occuper en premier des plus petits détails !

Nouveau Problème

MOINS DE 20 CARTES

Voici un petit problème dont on dit qu'il a été utilisé par les recruteurs de Google lors de tests d'embauche.

Cent nombre différents sont écrits sur cent cartes différentes, qu'on dispose à plat sur une table en 10 lignes de 10 cartes. Dans chaque ligne et dans chaque colonne les nombres sont classés par ordre croissant (de gauche à droite et de haut en bas). Les cartes ont été tournées et vous ne voyez donc pas les nombres qui y sont inscrits. Trouvez un algorithme qui, pour tout nombre donné, vous indique si oui ou non il est présent sur une des cent cartes, et cela en retournant moins de 20 cartes.

Envoyez vos réponses à jean-paul.delahaye@univ-lille1.fr. Le nom des premiers lecteurs à me donner la bonne réponse (et à la justifier) seront mentionnés dans le prochain numéro de 1024.



de la société informatique
de France



Institut Henri Poincaré,
11 rue Pierre et Marie Curie,
75231 Paris Cedex