



HAL
open science

Rôle et enjeux du design graphique

Nicolas Thély, David-Olivier Lartigaud, Gilles Rouffineau

► **To cite this version:**

Nicolas Thély, David-Olivier Lartigaud, Gilles Rouffineau. Rôle et enjeux du design graphique. THAT-Camp Paris 2012, Non-actes des non-conférences des humanités numériques, Éditions de la Maison des sciences de l'homme, pp.170-176, 2012. hal-01801146

HAL Id: hal-01801146

<https://univ-rennes2.hal.science/hal-01801146>

Submitted on 28 May 2018

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

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

Rôle et enjeux du design graphique

Proposé par : Nicolas Thély, David-Olivier Lartigaud et Gilles Rouffineau

Collectif

Les enjeux du design graphique

- 1 L'idée de cet atelier est d'ouvrir un nouveau chantier. Un atelier précédent intitulé *l'historien programmeur*, a posé le couple chercheur-programmeur. On se propose d'ajouter un troisième personnage troisième, le designer.
- 2 Si l'on veut définir le design, on peut dire que c'est de la conception d'interfaces. C'est d'ailleurs la définition donnée ici depuis deux jours. Or, le design n'est-il pas plutôt la conception d'une structure et la définition d'une fonction ? La définition admise est la conception d'outils, de corpus, et la restitution des résultats, (design par wikipedia).
- 3 Nous devons réfléchir à de nouveaux outils et méthodes de travail, être inventifs et innovants. Nous utilisons des programmes conçus ou des petites machines de calculs dont nous devons discuter le statut. Ces programmes sont standards ou non, et d'ailleurs on peut se poser la question de tous les standardiser. Quelles visualisations données aux nouveaux objets produits par ces programmes ?
- 4 Quelques questions :
- 5 Les artefacts qui sont produits : quelle visualisation, quel schéma ?
- 6 Les nouvelles « formes de sensibilité » : identifier, authentifier, commenter ?
- 7 Quels médiums de véracité ?

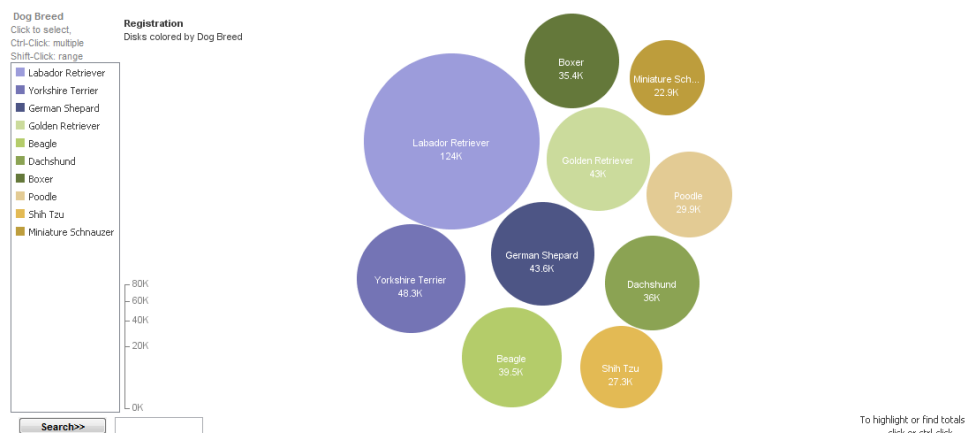
Quelques références et éléments de réflexion

- 8 What is visualization de Lev Manovich (2009) : Lev Manovich distingue le « design » d'information c'est-à-dire le questionnement sur la représentation de la donnée et la visualisation de l'information (extraction des données et *processing*) comme deux éléments différents.

- 9 DIRT (Digital Research Tool) est une boîte à outils très riche inventariant des outils des humanités numériques pour des utilisations universitaires.

- 10 DIRT propose ainsi des programmes libres ou payants. Parmi eux, on peut trouver le logiciel GEPHI (libre) qui est une plate-forme interactive de visualisation et d'exploration de toutes sortes de réseaux et de systèmes complexes et/ou dynamiques. Un autre exemple d'outil proposé par la plateforme DIRT mais développé cette fois par IBM est ManyEyes. Manyeyes est un outil de visualisation de données, sans caractère privatif, dont l'accès est public.

- 11 Manyeyes permet notamment d'expérimenter les représentations graphiques de données (voir la copie d'écran ci-dessous).



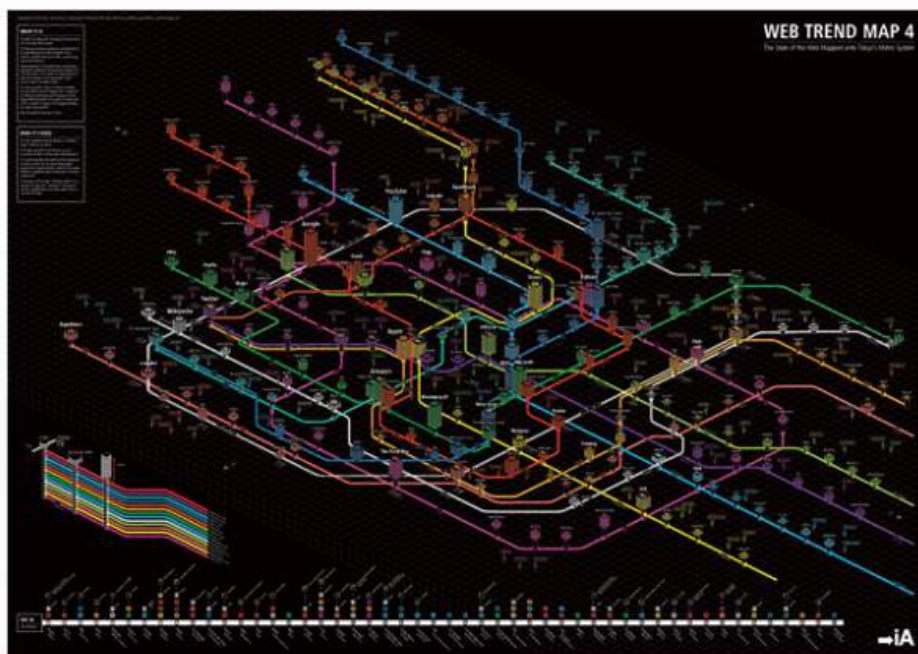
Autres références intéressantes :

- 12 John Maeda au MIT : Design By Numbers (1999)
- 13 processing.org et David-Olivier Lartigaud
- 14 Aaron Koblin qui a travaillé au CreativeLab Google
- 15 visualcomplexity.com et Manuel Lima
- 16 Fernanda B. Viégas et Martin Wattenberg, qui depuis la fin des années 1990, ont essayé des exemples de visualisation : cas de 2003 et Historyflow
- 17 Trois livres circulent dans la salle :
- 18 Form+Codes édité par Casey Reas et Chandler McWilliams
- 19 Software Studies. A Lexicon édité par Matthew Fuller
- 20 Data Flow 2. Visualizing Information in Graphic Design édité par R. Klanten, S. Ehmann, T. Tissot, N. Bourquin

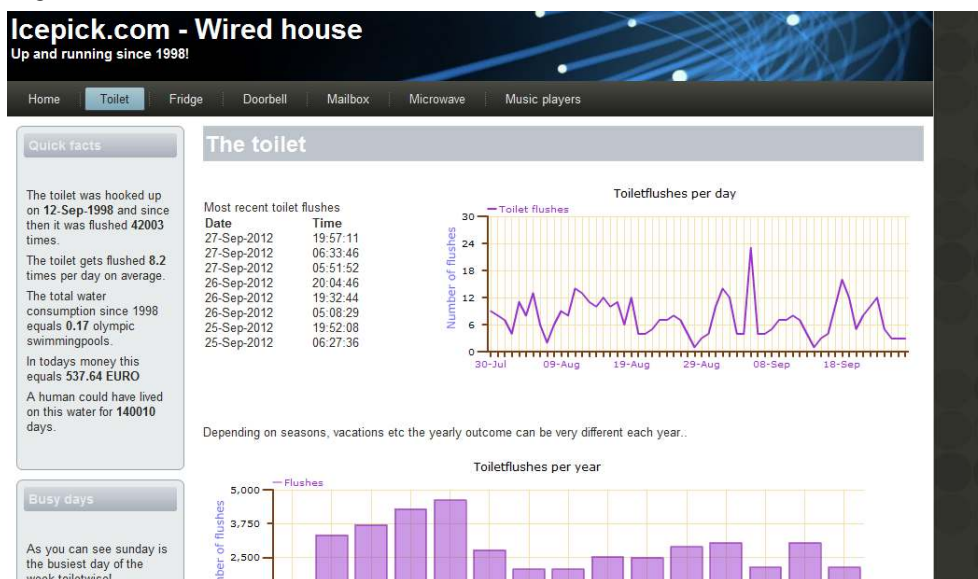
Démonstration à partir d'un outil de visualisation

- 21 Des données vont être visualisées de différentes façons sur un logiciel plutôt orienté graphistes 2D/3D et lié au prototypage d'interface (Arduino).
- 22 Cette démonstration est réalisée à partir d'une masse de données issues de la captation du déplacement du doigt de l'utilisateur sur l'écran d'une tablette. Il est possible de visualiser ce déplacement de plusieurs manières différentes : avec des points, des lignes, des couleurs différentes, des nuages de zone de contact qui s'amplifient en fonction du nombre de fois où le doigt passe sur une partie de la tablette. La représentation est aussi possible par points en fonction de la chronologie des événements. Une information temporelle peut être transmise en plus d'une information spatiale. En comparant avec l'action d'un autre utilisateur de cette même tablette, on voit que le toucher n'est pas le même grâce à une visualisation appropriée. Cette démonstration pose déjà une question de design : quelle est l'option de visualisation la plus appropriée pour ce que l'on souhaite montrer ?

- 23 Il existe donc différents types de visualisation pour les mêmes données. Ce n'est pas seulement un effet décoratif car ce que dit le graphisme est signifiant pour celui qui reçoit l'information.
- 24 La question du design peut se faire aussi en amont dans la conception de la base de donnée, de l'interface des logiciels utilisés, etc.
- 25 Exemple : Carte de métro du trafic internet ou des éléments de la vie quotidienne
- 26 <http://store.informationarchitects.net/product/web-trend-map-4>



- 27 Icepick.com : visualisation liées à la vie quotidienne, fréquence d'ouverture de la porte du frigo, chasse d'eau, éléments culinaires, etc.



- 28 La mise en forme des données pose la question des outils à utiliser (Python, Processing ?). Afin de choisir l'outil adéquat, une étude sur les logiciels « Software Studies » rédigé par M. Fuller est disponible à l'url <http://mitpress.mit.edu/catalog/item/default.asp?tttype=2&tid=11476>. Ce *software studies* permet la critique de logiciels en 40 courtes critiques.
- 29 Un autre exemple intéressant de mise en forme de données est l'application Pirate Pad, utilisé pour prendre des notes collaboratives lors de ce THATCamp Paris 2012. Pirate Pad modifie la pratique d'écriture de par les fonctionnalités qu'il propose et son ergonomie.
- 30 La question de la critique logicielle, au sens de M. Fuller, est donc un domaine qui doit entrer dans le cadre des « bonnes pratiques » des humanités numériques.
- 31 Pour Norman Potter (*What is a designer*), « Chaque être humain est un designer » (p. 1). Les trois notions clefs du design sont la spéculation, le rapport à la technique et la responsabilité.
- 32 Le bon design oppose une solution élégante à une solution productive (p. 50) c'est à-dire une économie de moyen (l'élégance) contre une ambition plus grande (lié à la production) où l'on propose une nouvelle intégration du sens (par déplacement).
- 33 Un autre exemple d'interface est le site UbuWeb. UbuWeb est une ressource indépendante dédiée à tous les courants avant-gardistes d'art... La composition du site est très structurée et laisse à penser qu'on est dans une base de données. L'approche graphique ressemble à un programme, les pages sont combinées comme un tableau à double entrée.
- 34 Au-delà de la séduction qu'offre le design, chaque position des auteurs et titres-thèmes dans les pages respecte une grille-code graphique homogène. Il s'agit d'une rigueur programmatique, une pensée visuelle et plastique.

Le public visé par le design : le cas des chercheurs

- 35 À la suite de la présentation des différentes interfaces, la question que l'on pose est celle de la réception des informations proposées, c'est-à-dire le public visé. Quel est le meilleur design et pour quel public et quels objectifs ? Faut-il former ce public à lire des designs particuliers ?
- 36 En effet, l'efficacité diffère d'un mode de visualisation à un autre suivant l'objectif visé. La frontière entre l'art et le design, entre la visée expressive et un moyen plus « pragmatique » de transmettre, de communiquer un certain sens précis, est ténue. Il est, de ce fait, important de faire une distinction entre l'artiste et le designer : le designer n'est pas l'artiste.
- 37 La question de la réception se pose en plusieurs temps. Elle se pose d'abord au moment de la production, en fonction des potentiels destinataires et des formes produites. Dans le cadre de cet atelier, la cible c'est d'abord le chercheur. L'habitus du chercheur peut jouer contre une nouvelle ergonomie proposée. La transmission d'information implicite se réalise également dans un design achevé. Une des limites cependant est le design qui va très loin et qui empêche parfois l'interprétation correcte des données.
- 38 Dans la question du design, la conception même de l'outil est importante. Quelles sont les frontières réelles entre les compétences du programmeur, du chercheur et du designer ? C'est une triade de savoir faire, triade mouvante où les compétences circulent. Certaines

personnes sont les trois à la fois mais ce sont des personnes rares ! La conception débute dans la base de donnée, on est déjà dans le design, c'est-à-dire une pensée structurante.

- 39 La question de la formation en design se pose. Les chercheurs ont une appréhension du rôle du graphiste. Le travail sur la visualisation de la donnée est souvent sous-évalué par les chercheurs et les doctorants. On peut se demander ce qui fait obstacle à cette prise de conscience de l'importance du design. Les temps d'apprentissage du design sont compliqués. La difficulté réside également dans l'accoutumance à certains codes préexistants. Cependant, au moment de la production des données, une des dimensions importantes pour le chercheur est de prendre en compte la visualisation des données et d'anticiper la conception.
- 40 Existe-t-il des outils préférables à d'autres pour les SHS ? Et pourquoi ?
- 41 Il est difficile de généraliser, de dire quel outil est préférable pour quoi. Il est tout à fait possible de construire son propre outil mais on ne devient pas programmeur « comme ça ». L'étape de programmation d'un outil ne peut pas se cumuler avec la fonction de chercheur. Un des risques est la perte des données : il y a un enjeu de qualité et d'exigence quand on programme. Le programmeur du dimanche le fait à ses risques et périls. Faire appel à un développeur pour programmer un logiciel est une garantie.
- 42 Cependant, un des avantages à ce que le chercheur se frotte à la programmation est que cela lui permet de prendre un certain recul et d'avoir un esprit critique sur « la boîte noire de l'algorithme qu'il utilise pour ses données ». S'interroger systématiquement sur les outils utilisés, c'est aussi une partie du travail du chercheur. Il faut un dialogue chercheur-designer mais il est important également qu'un contact se crée entre le chercheur, la plasticité de l'information et l'algorithme. Un exemple concret du lien design/chercheur est le carnet de recherche d'Hypothèses : comment cet outil détermine-t-il un type d'expression lié à son format ?
- 43 Est-ce que c'est le déficit de formation qui fait que l'historien doit faire appel au développeur ?
- 44 Il est important d'organiser du dialogue entre les compétences. Dans le cas du design, cela se passe en dehors du temps identifié de la recherche. Concernant la question de la pensée de l'objet final tel qu'il se présente dans sa forme visuelle, il faut que dans les formations, les chercheurs soient sensibilisés à ces traces de conception aussi.
- 45 Aux Éditions de la Maison des sciences de l'homme, l'équipe a travaillé avec des graphistes qui sont aussi développeurs et à qui il a été imposé d'entendre la parole de l'auteur-chercheur. L'objectif était de pouvoir fabriquer quelque chose qui rendait compte (permettait de visualiser sa démonstration argumentée et longue) mais qui faisait un peu perdre le fil aux autres. Cette initiative a été bien reçue par tous et le chercheur a en plus aperçu (à la visualisation) de nouveaux éléments en regardant le schéma travaillé (animé) par le graphiste. Le médium : l'éditeur qui veut un propos, le graphiste qui rend visible et le chercheur qui rajoute finalement un chapitre car il a « vu » de nouvelles choses ! On peut ainsi imaginer ouvrir le triangle chercheur/designer/programmeur avec l'éditeur. D'autres acteurs sont, bien entendu, à ajouter.
- 46 Dans les *software studies*, on est obligé d'avoir un rapport critique au logiciel utilisé.
- 47 Marie-Haude Caraës et Nicole Marchand-Zanartu, *Images de pensée*, RMN, 2011.
- 48 <http://strabic.fr/La-pensee-a-l-oeuvre.html>

RÉSUMÉS

À partir de cas de figures et de situations concrètes, cet atelier propose de discuter la manière dont le design graphique est convoqué dans l'agencement et la production de savoir.

INDEX

Mots-clés : graphisme, programmation, design