

# Questions

comment articulez-vous constructivisme et déterminisme?

avec le concept de script de madeleine akrich  
de manière symétrique, mais aussi avec des gradations. la question est aussi celle de la généralité des résultats: quels aspects tendent au déterminismes? quels aspects tendent au constructivisme?

---

quelles différences avec le séminaires codes sources?

Resource précieuse, que je voudrais pousser à l'international, mais mon travail est aussi différent de Baptiste Mèlès.

---

quel rapport avec l'histoire du code?

- beaucoup de travaux déjà établis, histoire de communauté culturellement restreinte, moi je suis plutôt intéressé par des alternatives actuelles
-

## rapport entre code et environnement?

très compliqué d'établir l'analyse de cycle de vie  
différence entre impact environnemental et impact énergétique. le  
premier a des ramifications complexes, tandis que le second tend plutôt à des  
implémentations pratiques d'expériences utilisateurs (durée de vie de la  
batterie).

moins une question d'implémentation que de *features*.

---

## question des grammaires

le code ce sont des grammaires de chomsky, mais c'est justement pas la  
seule manière d'envisager le langage. dans ce cas, ces grammaires montrent  
une certaine limite de ce que peut être un langage de programmation.

ici, les LLMs peuvent justement être productifs, parce qu'ils proposent  
une nouvelle sorte de grammaire.

---

## quelle est votre définition des humanités numériques

trois aspects:

- outils numériques pour des problèmes humanistes
- outils humanistes pour des problèmes numériques

typiquement le problème d'interprétation des LLMs, c'est un problème  
que la littérature a au moins depuis St. Augustin

---

## définition sciences sociales et sciences humaines?

c'est une distinction complexe, et je pense qu'il n'y a pas de réponse réellement satisfaisante, mais la distinction qui est productive dans mon contexte, c'est en différenciant les objets d'étude: en sciences humaines, cela va plutôt se pencher d'abord sur le document, tandis que les sciences sociales vont se pencher plutôt sur le terrain.

---

## comment s'adopte ma méthode à l'évolution des logiciels?

les logiciels évoluent, mais les structures pas tellement. la branche conditionnelle est là depuis le langage assembleur, dans les années 1950. les innovations de code même ne sont que très ponctuelles.

une grande différence serait le code généré par les LLMs, mais là aussi, ce qui change sont les autorialités, et pas le code en lui-même.

---

## vous aussi vous êtes un homme blanc, et alors?

effectivement, donc ce que je propose ce sont notamment des différences de discipline, notamment comme la recherche création, notamment avec des artistes qui ne font pas partie de cette démographie.  
Ramsey Nasser, winnie Soon

---

quelle est la définition de la fonction?

rapport d'une action avec une spécification, mais aussi action résultant d'une intention

différence avec fonctionnalité

---

quelle est votre chronologie? à 4 ans? à 8 ans?

4 ans: Yandex et les méthodes

8 ans: Moodle et la généricité du numérique

**PIEGE:** entre l'ambitieux et le raisonnable

---

avez-vous des exemples d'action sur le monde?

- visibilité par défaut dans Moodle (roles)
  - représentation des frontières dans Yandex (pré-2016)
  - petite marque de synchronisation des messages (lu/non-lu/entraîn de taper)
- 

comment transposer ces méthodes à des LLMs?

le code a une multi-auctorialité de toutes facons

---

que pouvez-vous dire des différences culturelles des logiciels?

individuel en europe, collectif en Asie, notamment en termes de protection de données. ptet en termes de social credit system?

---

quel exemple avez vous des alternatives?

solar protocol

---

Pourquoi Yandex?

unique exemple de plateforme boite noire, google en filigrane

---

Cadre méthodologique en sciences sociales? Concepts ou auteurs? (toujours auteurs fr)

- imaginaires socio-techniques
  - tactique et stratégie
  - théorie de la réception
-

## Question de l'accès aux codes sources

grande question, limitée actuellement, mais il s'agit aussi d'établir des modes de lecture **avant** que des codes sources deviennent plus accessibles, afin de doublement argumenter pour cette accessibilité

---

## Avez vous des éléments de grammaire

les tests, les classes, les enums

---

## Quelle est la tradition de la recherche en software studies en France?

Codes sources avec Baptiste Mèlès, l'ANR Programme avec Liesbeth de Mol, aussi les travaux à Costech en philo de la technique (mais c'est plus large), ainsi que ce qui se passe au CNAM, avec le séminaire sur la socio-histoire de l'informatique

---

## On lit quoi quand on lit des codes sources? L'approche textuelle n'est-elle pas un peu limitée?

On lit des architectures

---

Avec tant d'objets de recherche différents, comment avoir  
une approche unifiée?

Les structures computationnelles elles-mêmes sont unifiées  
(embranchements, classes, fonctions, initialisation de variables), donc il y a  
une sorte de langage sous-jacent, de **rhétorique**.

---

Si la technologie facilite le progrès, comment vous  
définissez le progrès?

défini de manière quantitatives par des indicateurs et des modélisations  
(cf. alain derosières)

---

comment allez-vous influencer les normes juridiques?

luc pellissier, établir la jurisprudence, mais aussi programme de la  
commission européenne, research fellow at the European Center for  
Algorithmic Transparency

---

question du genre?

véritablement cruciale, notamment sur la diversité des personnes qui  
créent les langages de prog

---

rapports entre singulier et pluriels? que faire des cultures universitaires spécifiques?

une université? des contextes? en tout cas actuellement l'informatique est singulier au sein de la fac (il n'y a qu'une seule DSI). donc la question est de savoir si c'est la spécificité des contextes qui l'emporte, ou la transversalité du numérique

---

quid de la généricité des langages de programmation?

voir l'usage des langages de programmation dans différents contextes (Rust dans les systèmes embarqués que ce soit dans le lowtech ou chez thalès, OCaml entre INRIA et Jane Street) et voir si il y a des parallèles

---

est-ce que vous imaginez que cela ait un retour sur la conception des langages de programmation?

oui, notamment exemple turkle et papert, par exemple compilateur de Rust, par exemple felienne hermanns

---

## Pourquoi FEMTO-ST?

Parce que les UT sont des endroits uniques pour effectuer de la recherche prenant autant au sérieux les sciences informatiques que les sciences sociales

les compétence que j'ai encore le plus à développer sont des compétences en informatiques

---

## différence entre code sources et artefacts

### computationnels?

différence entre le plan et le bâtiment, entre la vision idéale et l'usage actuel, à ceci près que, comme le fantasmait le corbusier pour le plan vis a vis du bâtiment, c'est véritablement le code qui génère l'artefact computationnel

---

## qu'est-ce qu'une grammaire d'action

concept de phil agre, précisions des ensembles de pratiques et de configurations matérielles (y compris les inscriptions) qui font sens dans un système plus large

---

## compétences russophones?

pas de compétence actuelles, à part le fait de pouvoir décrypter certaines lettres de l'alphabet cyrillique, en revanche compétence en polonais, et donc déjà familier avec une langue slave, même si elles sont assez distinctes. dans tous les cas, motivé pour apprendre.

---

## contacts chez moodle?

développeurs à l'Open University (moins loin que Perth), mais Moodle Meet pour établir des premiers contacts (**ou se passe le prochain?**)