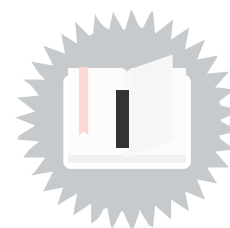


L'écriture numérique, entre signe et calcul

Table des matières

I - L'écriture numérique, entre signe et calcul	3
II - Des documents et des données	5
III - L'approche logique : formaliser l'information	7
IV - Exemples de formalisation par programmation	10
V - Les limites de l'approche logique : certaines voitures ont trois roues...	12
VI - L'approche sémiotique : what you see is what you get	14
VII - Les limites de l'approche sémiotique	15
VIII - L'approche documentaire : le balisage	16
IX - Les balises : entre code et texte	17
X - Langages et méta-langages à balises	18
XI - Exercice : Questions de synthèse	22
Abréviations	23
Bibliographie	24
Crédits des ressources	25

L'écriture numérique, entre signe et calcul



L'écriture numérique consiste à :

1. utiliser un ordinateur
2. pour produire des signes
3. qui sont représentés sous la forme d'inscriptions codées en binaire
4. qui sont destinées à être décodées par un ordinateur
5. pour produire des signes
6. destinés à être interprétés par un humain



Fondamental

La question du codage est donc centrale car elle détermine comment l'information peut être produite, stockée et lu.

C'est la condition de l'interprétation de l'information.

Approche logique



Méthode

On peut coder l'information sous forme logique, par exemple dans une base de données sous forme tabulaire, sous formes de propositions logico-mathématiques.



Exemple

- 1 Est-un (Voiture, Véhicule)
- 2 Possède (Voiture, Roue, 4)
- 3 Est-un (Ferrari, Voiture)

Avantages et inconvénients de l'approche logique



Remarque

- L'avantage principal est que cela permet d'exploiter les possibilités de calcul de l'ordinateur (exemple de la recherche dans une base de données).
- Un premier inconvénient est qu'il existe de (nombreuses) informations que l'on sait mal (voire pas du tout) représenter sous une forme purement logique.
- Un second inconvénient est que l'humain n'accède pas (facilement) à l'information telles qu'elle est enregistrée, il est nécessaire de mobiliser un programme informatique qui la transforme en information lisible.

Approche sémiotique



Méthode

On peut coder la forme sémiotique de l'information, c'est à dire la façon dont elle se manifeste aux humains qui l'écrivent et la lisent, c'est ce que permettent de faire les logiciels de bureautique (traitement de texte, manipulation audiovisuelles...).



Une **voiture** est un véhicule à quatre roues.

- Exemple : Une Ferrari est une voiture

Avantages et inconvénients de l'approche sémiotique



- L'avantage principal est que cela ressemble à la façon de faire du monde analogique, donc une partie des compétences existantes est facilement transposable (principe du *WYSIWYG* *).
- Un premier inconvénient est que ce type de codage rend les manipulations calculatoires par la machine limitées (on peut faire moins de traitements automatiques).
- Un autre inconvénient (en partie lié au précédent) est que le rédacteur est obligé de fournir un travail spécifique de mise en forme à chaque fois qu'il écrit pour faire en sorte que le logiciel produise la forme qu'il souhaite.
- Un dernier point à noter est que tout enregistrement numérique étant soumis au calcul, l'accès au contenu reste dépendant d'un programme qui doit fonctionner correctement et auquel il faut faire confiance.

Approche documentaire



L'approche documentaire consiste à chercher un compromis entre approche logique et sémiotique :

1. les contenus sont représentés sémiotiquement,
2. et ils sont associés à des balises qui permettent leur manipulation par le calcul.



```

1 <definition>
2   Une <notion>voiture</notion> est un véhicule à quatre roues.
3 </definition>
4 <exemple>
5   Une Ferrari est une voiture.
6 </exemple>

```

Avantages et inconvénients de l'approche documentaire



- L'avantage principal réside dans le compromis : il est plus facile de produire de l'information qu'avec l'approche logique, on peut faire plus de manipulations qu'avec l'approche sémiotique.
- L'inconvénient principal réside dans le compromis : il est plus difficile de faire des traitements qu'avec l'approche logique, il est plus difficile de produire de l'information qu'avec le paradigme sémiotique.

Des documents et des données



Écrire avec une machine à calculer

« Un ordinateur est fondamentalement une machine à calculer, c'est-à-dire à manipuler des nombres par des opérations arithmétiques.

Les signes qui fondent la communication entre humains - textes, paroles, images - sont « codables », donc calculables. Les ordinateurs peuvent manipuler des signes et ainsi devenir des outils sémiotiques, des machines à lire et à écrire.

L'écriture numérique est une inscription codée avec des logiciels via des ordinateurs

« Le concept d'écriture numérique renvoie en fait à trois composantes du numérique :

1. l'écriture numérique est ce qui est produit avec un dispositif numérique de production des signes (un ordinateur et un logiciel d'écriture) ;
2. l'écriture numérique est ce qui produit des inscriptions qui sont numériques (un code binaire) ;
3. l'écriture numérique est ce qui produit des signes destinés à être interprétés via un dispositif numérique de lecture (un ordinateur et un logiciel de lecture et de d'interaction).

La dualité écriture-programmation de l'écriture numérique

« L'écriture numérique et la programmation informatique semblent relever de deux pratiques différentes : l'écriture numérique consisterait à coder des données, les signes que l'on code et que l'on souhaite voir restitués "tels quels" ; tandis que la programmation consisterait à coder les calculs qui manipulent ces signes, ce qui serait de l'ordre du traitement.

Qu'est ce qu'un document, qu'est ce qu'une données ?

« Le document est ce qui est accessible et interprétable par l'homme et la donnée est ce qui est accessible et calculable par la machine.

Fragilité théorique de la séparation entre données et instructions

« La rupture fondamentale apportée par l'ordinateur, qui le distingue des automates qui le précèdent est l'indifférenciation du programme et de la donnée, ou dit autrement, le fait de considérer un programme comme une donnée.

Documents et données : repères historiques, frontières technologiques

Documents	Données
1969 : GML (Charles Goldfarb, IBM)	1970 : Relationnel (Edgar Codd, IBM)
1984 : LaTeX (Leslie Lamport, SRI)	1980 : Premiers SGBD (Oracle, Ingres)
1986 : SGML	1999 : RDF
1992 : HTML	2001 : XML Schema
1998 : XML	2002 : JSON
2000 : Premiers Web CMS (Typo3, SPIP, Drupal, Plone)	

Perméabilité technologique et terminologique



Fondamental

« Il y a fusion des formats, fusion des domaines, et finalement fusion des gestes d'écrire et de calculer. »

Réflexions sur l'écriture numérique

<https://aswemay.fr/co/010020.html>

L'approche logique : formaliser l'information



Le positivisme logique

Le positivisme logique est un courant de pensée fondé par le Cercle de Vienne (un groupe de philosophes et logiciens animé par Moritz Schlick entre 1929 et 1936).

Il pose que toute connaissance est formelle et donc se rapporte à une expression logique (au sens de la logique mathématique).

Les autres énoncés (musique, poésie, métaphysique...) relèvent du sentiment et non de la connaissance.

(Soulez, 1986)*.

La calculabilité

La machine de Turing, inventé en 1936 par Allan Turing (à partir des travaux de Hilbert et Gödel), est un modèle abstrait d'une machine universelle permettant d'automatiser tout calcul symbolique (alors que l'ordinateur n'existe pas encore).

Elle montre que toute formalisation logique est calculable par une machine.

<https://aswemay.fr/co/machine-de-turing.html>

Etat	Lit	Ecrit	Déplace	Suivant
e1	VIDE	VIDE	gauche	e2
e2	0	0	gauche	e2
	1	1	gauche	e2
e3	VIDE	VIDE	droite	e3
	0	1	droite	fin
	1	0	droite	e3
	VIDE	1	droite	fin

Exemple de machine de Turing permettant d'ajouter 1 à un nombre binaire

L'intelligence artificielle (symbolique)

L'IA* est fondée en août 1956, au Dartmouth College, lors d'un séminaire organisé par McCarthy où sont présents des figures telles que Simon, Newell, Minsky ou Shannon.

Elle se construit sur un concept de modélisation (représentations logiques des problèmes à traiter) et d'effectivité (les représentations sont calculables).

Le mouvement de l'IA (dite forte) a pour hypothèse qu'un ordinateur peut penser puisque la connaissance est formalisable et que le formalisé est calculable par une machine de Turing.

Exemples de réalisations historiques de l'IA**• Systèmes experts**

Dendral en 1965 par Edward Feigenbaum : résolution de problèmes en sciences (le programme permet l'identification de structures chimiques compatibles avec des données expérimentales fournies en entrée)

• Le traitement automatique de la langue naturelle

ELIZA en 1966 par Joseph Weizenbaum : donner l'illusion de la compréhension par la machine (le programme simule un dialogue avec un psychologue par des mécanismes langagiers de reformulation très simples, mais assez efficaces)

• Robotique et langage naturel

SHRDLU en 1970 par Terry Winograd : déplacer des cubes dans un micromonde



```

ELIZA: How do you do. Please tell me your problem.
YOU: I need to write a course, but I'm not inspired.
ELIZA: Why do you want to write a course ?
YOU: I think it makes me sound more serious...
ELIZA: You say you think it makes you sound more serious ?
YOU: Yes. Shouldn't I ?
ELIZA: You seem to be quite positive.
YOU: Ha ? I thought I was not...
ELIZA: Were you really ?
YOU: I don't know...
ELIZA: Do you wish to be able to know ?
YOU: Yes, it would help.
ELIZA: I understand.
YOU: Do you have knowledge in artificial intelligence ?
ELIZA: We were discussing you -- not me.
YOU: Sorry...
ELIZA: Please don't apologise.
YOU: Sor... ok.
ELIZA: What does that suggest to you ?

```

Real dialogue with Eliza (2020)



[cf. Demonstration of SHRDLU]

Hypothèse du paradigme logique

L'approche logique considère les inscriptions numériques en tant que formalisations de la connaissance.

L'ordinateur calcule sur la connaissance formalisée.

Représentation logique de l'information

? *Exemple*

- 1 Est-un (Voiture, Véhicule)
- 2 Possède (Voiture, Roue, 4)
- 3 Est-un (Ferrari, Voiture)

Un ordinateur pourra calculer (déduire ?) qu'une Ferrari est un véhicule et qu'il possède quatre roues.

Les robots aussi veulent jouer au poker

 *Complément*

<https://aswemay.fr/co/040006.html>

Exemples de formalisation par programmation



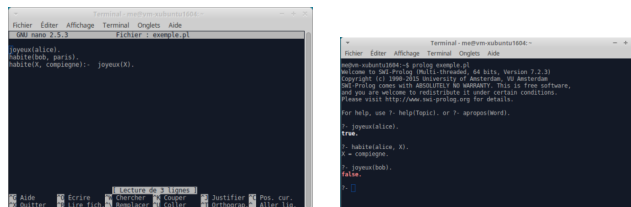
Hypothèse du paradigme logique



Le paradigme logique considère les inscriptions numériques en tant que formalisations de la connaissance.

L'ordinateur calcule sur la connaissance formalisée.

Calcul des prédicats (Prolog)



Exemple de déduction en Prolog (logique du premier ordre)

Modèle relationnel (SQL)



```
1 CREATE TABLE vehicule_type (  
2 type TEXT PRIMARY KEY,  
3 roue INTEGER);  
4  
5 CREATE TABLE vehicule (  
6 immat TEXT PRIMARY KEY,  
7 type TEXT REFERENCES vehicule_type (type));  
8  
9 INSERT INTO vehicule_type (type, roue)  
10 VALUES ('voiture', 4);  
11  
12 INSERT INTO vehicule (immat, type)  
13 VALUES ('AA-123-AA', 'voiture');  
14  
15  
16 SELECT immat, roue  
17 FROM vehicule v  
18 JOIN vehicule_type t  
19 ON v.type=t.type;  
20
```

1	immat		roue
2	-----+-----		
3	AA-123-AA		4

Programmation Objet (TypeScript)



```
1 class Animal {
2   type: string
3   legs: number
4   public getLegs(): number {
5     return this.legs
6   }
7 }
8
9 class Insect extends Animal {
10  name: string
11  constructor(name: string) {
12    super()
13    this.name = name
14    this.type = 'Insect'
15    this.legs = 6
16  }
17 }
18
19 const f = new Insect('Fly')
20
21 console.log (`A fly has ${f.getLegs()} legs`)
22
```

```
1 A fly has 6 legs
```

Les limites de l'approche logique : certaines voitures ont trois roues...



L'aporie de l'IA (symbolique)

L'hypothèse du positivisme logique qui fonde l'IA, à savoir que la connaissance est formelle et n'est que formelle, se révèle impossible à tenir. Il n'est pas possible de formaliser logiquement l'ensemble des connaissances humaines et l'intelligence humaine ne peut être réduite à une manipulation de symboles sans signification.

L'IA comme projet de création de machines pensantes est remis en cause (*Winograd et Flores, 1989*)*.

Traduction automatique



Parmi les programmes les plus ambitieux et les plus stratégiques de l'IA, celui de la traduction automatique, largement financé par la défense américaine en période de guerre froide, avait pour terrain d'application la traduction anglais-russe. Une illustration mythique d'échec de ce programme était le test de traduction de la phrase : « *The spirit is strong, but the flesh is weak* » (« *l'esprit est fort mais la chair est faible* ») qui donna après traduction automatique en russe, puis de nouveau en anglais : « *The vodka is good, but the meat is rotten* » (« *la vodka est forte, mais la viande est avariée* »).

Au delà de son caractère amusant cet exemple illustre très bien l'aporie d'une informatique symbolique qui bute sur des concepts de base tels que le contexte, la polysémie, etc.

Des machines qui pensent aux machines qui donnent à penser

Les années 1980 voient alors la naissance d'une IA modeste (ou « *faible* »), dont l'objet n'est plus la construction de machines qui pensent, mais de « *machines qui donnent à penser* » (*Bachimont, 1996*)*.

Théorème d'incomplétude de Gödel



Gödel montre en 1931 que les mathématiques ne peuvent être réduites à la logique. Un corollaire direct est qu'il existe des connaissances qui ne sont pas représentables par la logique.

Notons que Turing a également démontré que certains problèmes sont indécidables par une machine de Turing (sans pour autant que cela remette en cause sa vision positiviste).

La chambre chinoise de Searle



L'ordinateur ne pense pas car il n'accède pas au sens, ainsi que l'illustre Searle avec la métaphore de la "chambre chinoise" : Un opérateur qui recevrait des idéogrammes chinois et disposerait de procédures de traitement adapté à ces signes pourrait exécuter des opérations correctes, par application stricte des procédures (algorithmes). Pour autant cet opérateur ne comprendrait pas le chinois.

Ainsi, réaliser automatiquement des actions au sens d'une machine de Turing n'est pas penser.

Représentations locales versus représentation globale



La représentation formelle des connaissances a donné des résultats opérationnels tout à fait satisfaisant (par exemple certains systèmes experts), tant que cette représentation et que les raisonnements qui la concerne restent local à un domaine, en particulier technique (médecine par exemple).

L'approche sémiotique : what you see is what you get



Numérisation

La numérisation d'un contenu correspond à son codage (donc un calcul) de telle façon que l'on soit capable par un décodage (un autre calcul) de le restituer **tel qu'il a été codé**.

Un tel traitement correspond à une **numérisation de la forme**, car le contenu est inscrit mais non formalisé au sens logique.

WYSIWYG

Un enregistrement analogique représente une information par une autre information à laquelle elle peut se rapporter : quelque chose qui lui est similaire (analogue). Ainsi des points de couleur sur une feuille de papier ou un sillon sur un disque vinyle.

Le numérique par essence n'est pas analogique, il n'y a aucun rapport physique entre l'enregistrement et ce qui est enregistré, seul un calcul fait le lien entre les deux.

Le paradigme sémiotique vise à simuler l'analogie, c'est le principe du *WYSIWYG**.



- Scanner
- Traitement de texte
- etc.

Hypothèse du paradigme sémiotique



L'approche sémiotique considère les inscriptions numériques en tant que simple numérisation de formes sémiotiques.

L'humain interprète ce qu'il perçoit.

Présentation sémiotique de contenu



Une **voiture** est un véhicule à quatre roues.

- *Exemple* : Une Ferrari est une voiture

Un humain pourra déduire qu'une Ferrari est un véhicule et qu'il possède quatre roues. Un ordinateur ne peut que présenter cette information, il n'accède pas au contenu.

IA statistique et formalisation a priori



Un algorithme d'analyse de données pourra conclure qu'une Ferrari est une voiture et qu'elle possède 4 roues par l'analyse de grandes quantités de textes et par proximité statistique des termes.

Les limites de l'approche sémiotique



C'est théoriquement impossible

Parce que le contenu numérique est toujours calculé (« *il a été manipulé* »), ce que l'on lit n'est jamais ce qui a été écrit.

L'approche consistant à simuler un enregistrement analogique masque le fonctionnement réel de la machine.

C'est une sous-utilisation de l'informatique

Si l'usage du support numérique reste limité au codage et décodage de l'information, les potentialités du calcul, le propre du numérique, sont sous-utilisés.

Photocopieuse numérique



Une approche exclusivement sémiotique équivaudrait au champ fonctionnel d'une photocopieuse, sans la certitude de la conformité de la copie à l'original fournie par la photocopieuse.

De la numérisation à l'informatisation



La question se pose donc de l'opérationnalisation, au sens du calcul, de connaissances en langue naturelle : Comment manipuler par des algorithmes des informations codées numériquement ?

« [...] le formalisme comme principe de modélisation n'est plus tenable. En revanche, le formalisme comme principe d'effectivité reste indispensable (Bachimont 1996) * »

L'enjeu est donc d'**enrichir** le codage sémiotique pour lui ajouter des propriétés qui seront exploitables par le calcul.

Bachimont parle d'informatisation, comme état supérieur de la numérisation, du point de vue de la manipulation calculatoire (Bachimont, 2004) *.

La désorientation



L'impossibilité de l'enregistrement analogique et le calcul comme principe de manipulation conduisent à une perte de **référence** (in-vérifiabilité théorique de ce que suis je en train de lire) avec comme conséquence potentielle la désorientation du lecteur. Une informatique documentaire se doit donc de prendre acte de cet état de fait pour l'intégrer dans sa réflexion et ses instrumentations.

L'approche documentaire : le balisage



Coupler enregistrement sémiotique et modélisation logique

Le support numérique est un **outil sémiotique**, permettant de mémoriser et de restituer de l'information moyennant un **calcul**.

Une approche uniquement calculatoire butte sur les limites de la formalisation en tant que principe de représentation de la connaissance et une approche uniquement sémiotique butte sur la nature même du support qui est le calcul. D'où la nécessité de coupler les deux.

Une informatique documentaire consiste à associer une dimension spatio-temporelle, l'inscription sémiotique de contenus, et une dimension calculatoire, le modèle de manipulation de ces inscriptions. Le support numérique n'a plus vocation à modéliser la connaissance, il a vocation à la véhiculer, et l'objectif d'une informatique documentaire est de concevoir des modèles qui optimisent ce véhicule.

Hypothèse de l'approche documentaire



Fondamental

L'approche documentaire propose une description calculable d'inscriptions sémiotiques. Il s'agit d'un compromis entre la formalisation logique du sens et l'enregistrement sémiotique de la forme.

L'ordinateur calcule sur les descriptions formelles pour décider des modalités de présentation des inscriptions, et l'humain interprète le résultat présenté qu'il perçoit.



Exemple

```
1 <definition>
2   Une <notion>voiture</notion> est un véhicule à quatre roues.
3 </definition>
4 <exemple>
5   Une Ferrari est une voiture.
6 </exemple>
```

L'ordinateur peut répondre à une requête telle que "Je voudrais une définition de la notion voiture" en présentant l'information :

Une **voiture** est un véhicule à quatre roues.

Il n'a aucune idée de ce qu'est une voiture, mais il permet à l'humain de le savoir en calculant une présentation adéquate du contenu qui véhicule cette connaissance.

Les balises : entre code et texte



L'ingénierie documentaire met à profit deux thèses complémentaires :

- le contenu est numérisé dans sa forme signifiante : il est manipulable par la machine mais indépendamment de sa signification qui lui reste inaccessible ;
- le contenu est enrichi par des balises qui sont connues syntaxiquement et sémantiquement par la machine ; elle sait quoi en faire.



Le principe du balisage consiste à enrichir un contenu numérisé (dans sa forme sémiotique), sans l'altérer, pour lui ajouter des poignées qui vont être manipulables par l'ordinateur (logiquement).



Le contenu est donc interprétable par l'homme et la machine, chacun via ce qui lui est destiné :

- l'humain interprète le contenu signifiant numérisé ;
- la machine manipule les balises.

XML



```
1 <?xml version='1.0' encoding='iso-8859-1'?>
2 <mail>
3   <de>stephane.crozat@utc.fr</de>
4   <a>fabrice.issac@utc.fr</a>
5   <objet>Demande d'information</objet>
6   <date>01-03-2001</date>
7   <corps>
8     <paragraphe>Bonjour Fabrice,</paragraphe>
9     <paragraphe>Peux tu m'expliquer ce qu'est le langage XML ?</paragraphe>
10    <paragraphe>Il me faudrait en effet ton avis éclairé sur le sujet.
11    </paragraphe>
12    <paragraphe>J'attends ta réponse, à bientôt</paragraphe>
13  </corps>
14 </mail>
```

Langages et méta-langages à balises



GML, 1969, langage

```
1:h1.Présentation de GML
2:p.GML est le premier langage à balises.
3:p.Les balises fermantes peuvent être omises.
4:ol.
5:li.Il existe des listes ordonnées.
6:li.Il existe des listes non-ordonnées.
7:eol.
```

LaTeX, 1984, langage

```
1 \documentclass{article}
2 \title{Le W3C}
3 \begin{document}
4 \maketitle
5 The World Wide Web Consortium (W3C)is an international community where Member
  organizations, a full-time staff, and the public work together to develop Web
  standards. Led by Web inventor and Director Tim Berners-Lee and CEO Jeffrey Jaffe,
  W3C's mission is to lead the Web to its full potential.
6 \paragraph{Vision} W3C's vision for the Web involves participation, sharing
  knowledge, and thereby building trust on a global scale.
7 \paragraph{Web for All} The social value of the Web is that it enables human
  communication, commerce, and opportunities to share knowledge.
8 \paragraph{Web on Everything} The number of different kinds of devices that can
  access the Web has grown immensely.
9 \paragraph{} \textit{Source : w3c.org}
10 \end{document}
```

Le W3C

September 10, 2020

The World Wide Web Consortium (W3C)is an international community where Member organizations, a full-time staff, and the public work together to develop Web standards. Led by Web inventor and Director Tim Berners-Lee and CEO Jeffrey Jaffe, W3C's mission is to lead the Web to its full potential.

Vision W3C's vision for the Web involves participation, sharing knowledge, and thereby building trust on a global scale.

Web for All The social value of the Web is that it enables human communication, commerce, and opportunities to share knowledge.

Web on Everything The number of different kinds of devices that can access the Web has grown immensely.

Source : w3c.org

SGML, 1986, méta-langage

```

1 <TASK>
2 <TITLE>Taking out the garbage
3 <STEPS>
4 <STEP>Find out from the town what day of the week garbage is collected in your
  neighborhood.
5 <STEP>The night before collection, place the garbage cans on the curb.
6 <STEP>After the garbage has been collected, move the cans back into your garage.

```

HTML, 1992, langage SGML

```

1 <HTML>
2 <TITLE>Costech</TITLE>
3 <P>Bernard Stiegler a fondé l'unité de recherche <I>Connaissances, organisations et
  systèmes techniques</I> (Costech) en 1993 à l'Université de technologie de
  Compiègne (UTC).
4 <P>L'unité de recherche Costech est une équipe d'accueil pluridisciplinaire
  regroupant des chercheurs travaillant sur les relations entre humain, technique et
  société.
5 <P><B>Ceci est un texte écrit en HTML 1.0

```

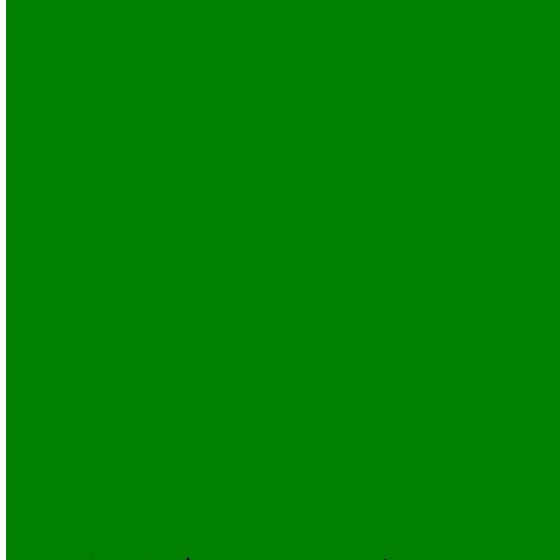
XML, 1997, méta-langage

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <grammar
3   xmlns="http://relaxng.org/ns/structure/1.0"
4   ns="http://www.w3.org/2000/svg">
5   <start>
6     <element name="svg">
7       <attribute name="version"/>
8       <attribute name="width"/>
9       <attribute name="height"/>
10      <element name="title"><text/></element>
11      <interleave>
12        <element name="rect">
13          <attribute name="width"/>
14          <attribute name="height"/>
15          <attribute name="fill"/>
16          <empty/>
17        </element>
18        <element name="text">
19          <attribute name="x"/>
20          <attribute name="y"/>
21          <text/>
22        </element>
23      </interleave>
24    </element>
25  </start>
26 </grammar>

1 <?xml version="1.0" encoding="utf-8"?>
2 <svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="300" height="300">
3   <title>Exemple de SVG</title>
4   <rect width="280" height="280" fill="green" />
5   <text x="10" y="290">Je suis coincé sous ce gros cube vert.</text>
6 </svg>
7

```



Je suis coincé sous ce gros cube vert.

XHTML, 2000, langage XML

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4 <html xmlns="http://www.w3.org/1999/xhtml">
5   <head>
6     <title>Costech</title>
7   </head>
8   <body>
9     <p>Bernard Stiegler a fondé l'unité de recherche <em>Connaissances,
10    organisations et systèmes techniques</em> (Costech) en 1993 à l'Université de
11    technologie de Compiègne (UTC).</p>
12    <p>L'unité de recherche Costech est une équipe d'accueil pluridisciplinaire
13    regroupant des chercheurs travaillant sur les relations entre humain, technique et
14    société.</p>
15    <p><strong>Ceci est un texte écrit en XHTML1.0</strong></p>
16   </body>
17 </html>

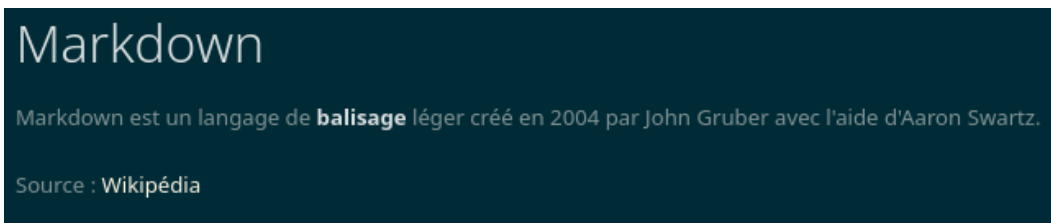
```

Markdown, 2004, langage texte

```

1 # Markdown
2
3 Markdown est un langage de balisage léger créé en 2004 par John Gruber avec
4 l'aide d'Aaron Swartz.
5 Source : [Wikipédia] (https://fr.wikipedia.org/wiki/Markdown)

```



HTML5, 2014, langage SGML

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Costech</title>
5     <meta charset="utf-8">
6   </head>
7   <body>

```

```
8      <p>Bernard Stiegler a fondé l'unité de recherche <em>Connaissances,  
organisations et systèmes techniques</em> (Costech) en 1993 à l'Université de  
technologie de Compiègne (UTC).</p>  
9      <p>L'unité de recherche Costech est une équipe d'accueil pluridisciplinaire  
regroupant des chercheurs travaillant sur les relations entre humain, technique et  
société.</p>  
10     <p><strong>Ceci est un texte écrit en HTML5</strong></p>  
11     </body>  
12 </html>  
13
```

Exercice : Questions de synthèse



Question

Quelles différences peut-on relever entre une base de documents et une base de données ? Contiennent-ils le même type d'information ? Peut-on les exploiter de la même manière ?

Indice :

<https://stph.scenari-community.org/contribs/doc/fdl/buc1>

<https://stph.scenari-community.org/contribs/doc/fdl/buc2>

Abréviations



IA : Intelligence Artificielle

WYSIWYG : What You See Is What You Get

Bibliographie



Bachimont Bruno, *Arts et sciences du numérique : ingénierie des connaissances et critique de la raison computationnelle*, Mémoire de HDR, Université de Technologie de Compiègne, 2004.

Bachimont Bruno, *Herméneutique matérielle et artéfacture*, Thèse en épistémologie, 1996.

Soulez Antonia, *Manifeste du Cercle de Vienne et autres récits*, Presses Universitaires de France, 1985.

Winograd Terry, Flores Fernando, *L'intelligence artificielle en question*, Presses Universitaires de France, 1989.

Crédits des ressources



Exemple de machine de Turing permettant d'ajouter 1 à un nombre binaire p. 7

Hamdi Ben Abdallah, 2013, https://interstices.info/jcms/nn_72391/comment-fonctionne-une-machine-de-turing

Real dialogue with Eliza (2020) p. 8

<https://www.masswerk.at/elizabot>

Demonstration of SHRDLU p. 8

<https://projects.csail.mit.edu/video/history/aifilms/26-robot.mp4>