



**HAL**  
open science

# Techniques d'interaction avec un espace de données temporelles

Chaouki Daassi

► **To cite this version:**

Chaouki Daassi. Techniques d'interaction avec un espace de données temporelles. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 2003. Français. NNT : . tel-00005156

**HAL Id: tel-00005156**

**<https://theses.hal.science/tel-00005156>**

Submitted on 27 Feb 2004

**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.

**Université Joseph Fourier Grenoble I**  
**U.F.R. d'Informatique et Mathématiques Appliquées**

Thèse

pour obtenir le grade de :

Docteur de l'Université Joseph Fourier Grenoble I

Discipline : Informatique

Présentée et soutenue publiquement par

Chaouki Daassi

***Techniques d'interaction avec un espace de données  
temporelles***

Directeurs de thèse : Laurence Nigay & Marie-Christine Fauvet

Date de soutenance : 16 Juillet 2003

**Jury**

Président : Mr Yves Chiaramella

Rapporteur : Mme Jocelyne Nanard

Rapporteur : Mr Jacques Le Maitre

Examineur : Mr Eric Lecolinet

Examineur : Mr Pierre Dumolard

Thèse préparée au sein des laboratoires :

**CLIPS : Communication Langagière et Interaction Personne-Système**

**LSR : Logiciels, Systèmes, Réseaux**

**Fédération IMAG-Université Joseph Fourier Grenoble I**



# *Techniques d'interaction avec un espace de données temporelles*

## **Chapitre 1. Introduction**

<u>1</u> <a href="#">Sujet</a> .....	9
<u>2</u> <a href="#">Objectifs</a> .....	11
<u>3</u> <a href="#">Structure du mémoire</a> .....	12

## **Chapitre 2. Temps : perception humaine et représentation informatique**

<u>1</u> <a href="#">Introduction</a> .....	17
<u>2</u> <a href="#">Perception humaine du temps</a> .....	18
<u>2.1</u> <a href="#">Processus de perception du temps</a> .....	18
<u>2.1.1</u> <a href="#">Perception basée sur les événements externes</a> .....	18
<u>2.1.2</u> <a href="#">Perception basée sur les événements internes</a> .....	19
<u>2.2</u> <a href="#">Représentation mentale du temps</a> .....	20
<u>2.2.1</u> <a href="#">Structure de l'espace temps</a> .....	20
<u>2.2.2</u> <a href="#">Éléments de l'espace temps</a> .....	21
<u>2.3</u> <a href="#">Synthèse</a> .....	24
<u>3</u> <a href="#">Représentation informatique du temps : point de vue base de données</a> .....	25
<u>3.1</u> <a href="#">Information temporelle</a> .....	26
<u>3.1.1</u> <a href="#">Définition d'une donnée temporelle</a> .....	26
<u>3.1.2</u> <a href="#">Dimension temporelle</a> .....	27
<u>3.2</u> <a href="#">Modèle du temps</a> .....	28
<u>3.2.1</u> <a href="#">Partition du temps</a> .....	28
<u>3.2.2</u> <a href="#">Unité d'observation</a> .....	29
<u>3.2.3</u> <a href="#">Calendriers</a> .....	31
<u>3.2.4</u> <a href="#">Types temporels</a> .....	32
<u>3.3</u> <a href="#">Modèle d'historique</a> .....	34
<u>3.3.1</u> <a href="#">Domaine temporel</a> .....	34
<u>3.3.2</u> <a href="#">Historisation des valeurs des données</a> .....	35
<u>3.4</u> <a href="#">Synthèse</a> .....	36
<u>4</u> <a href="#">Conclusion</a> .....	37

## **Chapitre 3. Techniques de visualisation de données temporelles**

<u>1</u> <a href="#">Introduction</a> .....	41
<u>2</u> <a href="#">Processus de visualisation de données</a> .....	41
<u>3</u> <a href="#">Processus de visualisation de la dimension structurelle</a> .....	45
<u>3.1</u> <a href="#">Techniques orientées pixels</a> .....	46
<u>3.2</u> <a href="#">Techniques à base d'éléments graphiques</a> .....	50
<u>3.3</u> <a href="#">Synthèse</a> .....	52
<u>4</u> <a href="#">Processus de visualisation de la dimension temporelle</a> .....	52
<u>4.1</u> <a href="#">Espace temps</a> .....	53
<u>4.2</u> <a href="#">Point de vue sur l'espace temps</a> .....	55
<u>4.2.1</u> <a href="#">Formes cycliques du temps</a> .....	56
<u>4.2.2</u> <a href="#">Formes linéaires du temps</a> .....	59
<u>4.2.3</u> <a href="#">Forme logarithmique du temps</a> .....	63
<u>4.2.4</u> <a href="#">Forme multigranulaire du temps</a> .....	63
<u>4.2.5</u> <a href="#">Relations entre formes de représentation du temps</a> .....	64
<u>4.3</u> <a href="#">Synthèse</a> .....	65
<u>5</u> <a href="#">Techniques de visualisation d'une donnée temporelle</a> .....	66
<u>5.1</u> <a href="#">Fusion complète ou fusion Point de vue sur la donnée</a> .....	68
<u>5.2</u> <a href="#">Fusion Espace de visualisation</a> .....	71

5.3	<a href="#">Fusion Point de vue sur l'espace de visualisation</a>	74
5.4	<a href="#">Temps puis Espace de visualisation de la dimension structurelle</a>	75
5.5	<a href="#">Temps puis Point de vue sur l'espace de visualisation de la dimension structurelle</a>	77
5.6	<a href="#">Processus complètement séparés</a>	79
5.7	<a href="#">Synthèse</a>	79
6	<a href="#">Techniques de visualisation de plusieurs données temporelles</a>	81
6.1	<a href="#">Utilisation d'un seul espace temps</a>	82
6.2	<a href="#">Utilisation d'un espace temps par donnée</a>	86
6.3	<a href="#">Synthèse</a>	87
7	<a href="#">Conclusion</a>	88

## Chapitre 4. Conception de techniques d'interaction avec des données temporelles

1	<a href="#">Introduction</a>	91
2	<a href="#">Approche centrée tâches utilisateur</a>	92
2.1	<a href="#">Tâches d'interaction avec la source de données</a>	94
2.2	<a href="#">Tâches d'analyse de données</a>	96
2.2.1	<a href="#">Tâches de navigation</a>	96
2.2.2	<a href="#">Tâches d'observation</a>	97
2.2.3	<a href="#">Tâches de comparaison</a>	102
2.2.4	<a href="#">Tâches de manipulation</a>	105
2.3	<a href="#">Synthèse</a>	106
3	<a href="#">Processus de conception de l'espace temps</a>	106
3.1	<a href="#">Lignes temps proposées dans la littérature</a>	107
3.2	<a href="#">Vue en mur en perspective pour représenter le temps</a>	110
3.2.1	<a href="#">Gestion d'un grand espace d'informations temporelles</a>	110
3.2.2	<a href="#">Caractéristiques du temps</a>	111
3.2.3	<a href="#">Tâches utilisateur se rapportant à la dimension temporelle</a>	112
3.3	<a href="#">Dispositif physique pour l'interaction avec l'espace temps</a>	113
4	<a href="#">Techniques de visualisation de données temporelles</a>	115
4.1	<a href="#">Technique cercles concentriques</a>	115
4.1.1	<a href="#">Représentation graphique</a>	115
4.1.2	<a href="#">Processus de visualisation</a>	118
4.1.3	<a href="#">Tâches utilisateur</a>	119
4.2	<a href="#">Technique de représentation en étoiles</a>	121
4.2.1	<a href="#">Représentation graphique</a>	121
4.2.2	<a href="#">Processus de visualisation</a>	123
4.2.3	<a href="#">Tâches utilisateur</a>	123
4.3	<a href="#">Représentation en diagrammes superposés</a>	125
4.3.1	<a href="#">Représentation graphique</a>	125
4.3.2	<a href="#">Processus de visualisation</a>	127
4.3.3	<a href="#">Tâches utilisateur</a>	128
4.4	<a href="#">Technique représentation tabulaire</a>	129
4.4.1	<a href="#">Représentation graphique</a>	129
4.4.2	<a href="#">Processus de visualisation</a>	129
4.4.3	<a href="#">Tâches utilisateur</a>	130
4.5	<a href="#">Technique représentation en pixels</a>	132
4.5.1	<a href="#">Représentation graphique</a>	132
4.5.2	<a href="#">Processus de visualisation</a>	134
4.5.3	<a href="#">Tâches utilisateur</a>	134
4.6	<a href="#">Synthèse</a>	135
5	<a href="#">Conclusion</a>	136

## Chapitre 5. INVEST : une plateforme pour l'analyse visuelle de données temporelles

1	<a href="#">Introduction</a>	139
2	<a href="#">Expérimentation</a>	140
2.1	<a href="#">Expérimentation 1</a>	141

2.2	<a href="#">Expérimentation 2</a>	142
2.2.1	<a href="#">Actualiser les valeurs de la fonction d'utilisabilité</a>	143
2.2.2	<a href="#">Tester les hypothèses de travail</a>	146
2.3	<a href="#">Synthèse</a>	148
3	<a href="#">INVEST</a>	149
4	<a href="#">Conclusion</a>	157

## Chapitre 6. Conclusion

1	<a href="#">Résumé des contributions</a>	161
2	<a href="#">Limites et extensions</a>	163
2.1	<a href="#">Réalisation logicielle</a>	164
2.2	<a href="#">Evaluation</a>	164
3	<a href="#">Prolongements</a>	165
3.1	<a href="#">Prolongements en Interaction Homme-Machine</a>	165
3.2	<a href="#">Prolongement en Base de Données</a>	167

## Bibliographie

1	<a href="#">Références papiers</a>	171
2	<a href="#">Sites Web</a>	185

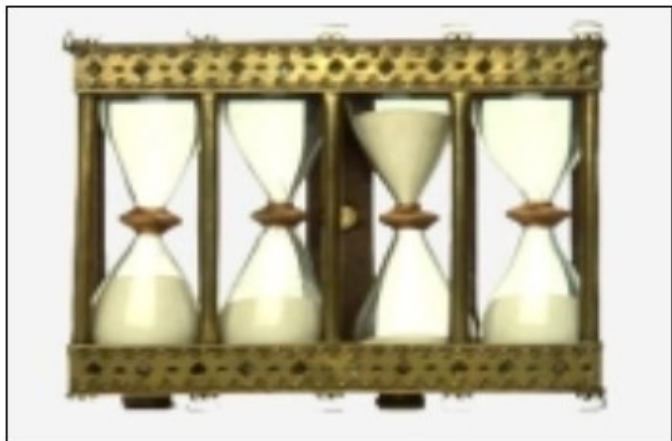
## Annexe A

1	<a href="#">Propriétés de souplesse</a>	189
2	<a href="#">Propriétés de robustesse</a>	193



# *Chapitre 1*

## *Introduction*



**Sablier, 12<sup>e</sup> siècle**



Table des matières

**Chapitre 1.** Introduction.

<u>1</u>	<u>Sujet</u> .....	9
<u>2</u>	<u>Objectifs</u> .....	11
<u>3</u>	<u>Structure du mémoire</u> .....	12

Nos travaux de recherche ont trait à la conception et à la réalisation de techniques d'interaction adaptées à la manipulation de données temporelles. Nos travaux sont pluridisciplinaires et s'inscrivent dans deux domaines de recherche : l'Interaction Homme-Machine (IHM) et les Bases de Données Temporelles (BDT).

Menées en collaboration entre les équipes de deux laboratoires de l'institut IMAG (Informatique, Mathématiques Appliquées de Grenoble) l'équipe IIHM (Ingénierie de l'Interaction Homme-Machine) du laboratoire CLIPS (Communication Langagière et Interaction Personne-Système) et l'équipe STORM (Structure et Temporalité des Objets Répartis et Multimédia) du laboratoire LSR (Logiciels, Systèmes et Réseaux), cette thèse repose sur des expériences complémentaires et fructueuses.

### 1 Sujet

Les travaux relatifs aux Bases de Données Temporelles [Canavaggio 97, Scholl 98, Snodgrass 95, Dumas 00] visent à intégrer toutes les fonctionnalités nécessaires à la prise en compte du temps dans les Systèmes de Gestion de Base de Données (SGBD). Dans ce contexte, une donnée temporelle désigne l'historique des valeurs d'une donnée, c'est-à-dire l'ensemble des états observés de cette donnée au cours du temps. Une donnée temporelle est donc définie comme une séquence de couples  $\langle v, t \rangle$ , où  $v$  dénote la valeur structurelle de la donnée et  $t$  la valeur temporelle estampillant  $v$ . Le couple  $\langle v, t \rangle$  désigne quant à lui un instantané. La valeur  $v$  est caractérisée par un type, tel que quantitative, image, objet simple et objet complexe. La Figure 1 présente une représentation alphanumérique d'une donnée temporelle quantitative.

1990/06/19/13/0	14.500
1990/06/19/13/1	14.700
1990/06/19/13/2	14.900
1990/06/19/13/3	15.100
1990/06/19/13/4	15.300
1990/06/19/13/5	15.500
1990/06/19/14/0	15.700
1990/06/19/14/1	15.800
1990/06/19/14/2	16.000
1990/06/19/14/3	16.300
1990/06/19/14/4	16.500
1990/06/19/14/5	16.700
1990/06/19/15/0	16.700
1990/06/19/15/1	16.800
1990/06/19/15/2	16.900
1990/06/19/15/3	17.000

Figure 1 : Représentation alphanumérique d'une donnée temporelle quantitative.

Les données temporelles sont présentes dans un grand nombre d'applications utilisant des bases de données, depuis les plus classiques (finance, comptabilité, etc.) jusqu'aux plus récentes (entrepôts de données, systèmes d'information géographique, etc.). Différentes d'un domaine d'application à l'autre, les tâches utilisateur de manipulation de ces données temporelles sont variées. Des tâches d'analyse concernent par exemple la corrélation de deux données, l'identification des irrégularités ou encore l'identification de motifs de comportement dans le temps. Si nous considérons l'exemple de la Figure 1, nous constatons immédiatement que la représentation alphanumérique de la donnée temporelle ne favorise pas les tâches d'analyse. L'objet de notre étude est la conception de techniques de représentation et plus généralement d'interaction adaptées aux tâches de l'utilisateur manipulant des données temporelles.

Nous focalisons nos travaux sur des techniques de représentation visuelle de données temporelles. En effet, l'information temporelle est traduite mentalement en représentation spatiale. Ainsi, les techniques de représentation spatiale sont les plus naturelles pour des données temporelles. De plus, des lois de psychologie de la perception soulignent les risques de perception incorrecte d'une représentation séquentielle d'une donnée temporelle, comme un rendu sonore ou une représentation visuelle dynamique, séquentielle dans le temps :

Selon la loi de Vierordt<sup>1</sup> [Nair 99] dite aussi loi de perception du temps, *les intervalles courts sont surestimés et les intervalles longs sont sous estimés*. Ainsi lorsque la présentation des intervalles de temps est séquentielle comme dans une interface sonore, les intervalles de temps risquent de ne pas être perçus correctement.

Selon la loi de Weber<sup>2</sup> [Nair 99], *la différence entre les durées de deux stimulus est une proportion constante de la plus petite des deux durées*. Là encore, la loi de Weber souligne l'inadéquation d'une présentation séquentielle de données temporelles en mettant en exergue les risques de perception incorrecte.

---

Citations :

<sup>1</sup> Vierordt's law : short intervals are overestimated and long intervals are underestimated.

<sup>2</sup> Weber's law : the just noticeable difference between two stimulus durations is a constant proportion of the shorter of the two values.

Bien que nous n'excluons pas l'usage synergique du son avec une représentation visuelle d'une donnée temporelle, nous avons restreint notre champs d'étude aux techniques interactives de représentation **visuelle statique** d'une donnée temporelle. Les caractéristiques des données manipulées et la variété des tâches utilisateur dessinent les contours d'un vaste champs d'étude à maîtriser pour concevoir des techniques d'interaction dédiées à l'analyse visuelle de données temporelles.

Nos objectifs se situent dans ce contexte.

## **2 Objectifs**

L'objectif global de notre étude est d'établir des éléments utiles à la conception et à la réalisation de techniques d'interaction pour l'analyse de données temporelles. Plus précisément, notre étude sur la conception et la réalisation logicielle vise un double objectif :

1) comprendre et structurer l'espace problème des techniques d'interaction dédiées aux données temporelles. Le modèle visé doit permettre aux concepteurs ergonomes et psychologues, tout comme eux concepteurs de logiciel, de raisonner sur le domaine des techniques d'interaction dédiées aux données temporelles ; de par la variété des tâches de manipulation de données temporelles et des domaines d'application, de très nombreuses techniques de visualisation et plus généralement d'interaction dédiées aux données temporelles ont été conçues et développées. Il convient dans ces conditions d'organiser le savoir expérimental dans un cadre conceptuel. De là, un espace problème structurant permet de raisonner, comparer, et évaluer les techniques existantes ou à concevoir. Nous estimons que ce premier objectif de compréhension et de structuration de l'espace des possibilités est un point contributif important au domaine.

2) fournir un espace solution à la fois conceptuel et opérationnel. Au niveau conceptuel, nous visons à fournir des éléments de conception centrée sur l'utilisateur et ses tâches, qui intègrent les caractéristiques des données étudiées. Etablir une taxonomie exhaustive des tâches utilisateur de manipulation de données temporelles semble difficilement réalisable de par l'étendue des possibilités. Sans viser l'exhaustivité, notre objectif est d'établir une liste organisée de tâches sur laquelle nous nous appuyons pour la conception de techniques d'interaction. Au niveau opérationnel, notre objectif est la conception logicielle et la mise en

œuvre d'un outil informatique pour assister l'utilisateur à réaliser un ensemble de tâches de manipulation de données temporelles. Plus qu'une ou plusieurs techniques d'interaction, notre objectif est ici la définition d'une plate-forme d'accueil de techniques complémentaires vis-à-vis des tâches utilisateur.

### **3 Structure du mémoire**

L'organisation du mémoire reflète notre démarche d'analyse, de l'espace des possibilités aux solutions de conception et de réalisation. La première partie composée de deux chapitres (Chapitres 2 et 3) décrit l'existant et son organisation selon notre espace taxinomique (Chapitre 3). Les possibilités étant cernées, la deuxième partie du mémoire est composée de deux chapitres, dédiés aux éléments utiles à la conception (Chapitre 4) et à la réalisation (Chapitre 5).

La première partie du mémoire pose la terminologie et décrit l'espace des possibilités.

Le Chapitre 2 est dédié à la représentation du temps. Les données temporelles sont d'une part manipulées par des utilisateurs et d'autre part traitées par un système informatique. Nous structurons le Chapitre 2 selon ces deux points de vue. Ainsi, nous présentons d'abord la représentation mentale (niveau humain) du temps en nous reposant sur des résultats issus de travaux en psychologie de la perception. Nous présentons par la suite la représentation informatique (niveau système) du temps en nous référant aux travaux en base de données temporelles. Le Chapitre 2 fait donc le point sur la terminologie et sur les référentiels applicables à la dimension temporelle d'une donnée. Ainsi, lors de la conception de techniques d'interaction (Chapitre 4), nous exploitons les résultats de ce chapitre qui explicite les caractéristiques des données temporelles. Il est à noter que les résultats présentés dans le Chapitre 2 devraient être considérés dans le contexte de cette thèse, c'est-à-dire pour caractériser les données temporelles telles qu'elles sont utilisées en base de données.

Au Chapitre 3, nous proposons notre cadre taxinomique des techniques d'interaction existantes. Les dimensions de ce cadre taxinomique reposent sur les étapes du processus de visualisation de données défini par Chi [Chi 00]. Notre revue des techniques existantes selon notre cadre taxinomique met en exergue le peu d'études pour deux cas de notre taxinomie,

deux configurations du processus de visualisation. Ces deux configurations font l'objet du chapitre suivant dédié à la conception.

Après avoir identifié les caractéristiques des données temporelles et proposé notre taxinomie des techniques existantes, la deuxième partie de ce mémoire définit l'espace solution. L'exposé de l'espace solution comprend deux chapitres. Il s'agit de solutions de conception de techniques (Chapitre 4) et de réalisation logicielle (Chapitre 5).

Le Chapitre 4 explore la conception de techniques d'interaction, en présentant notre approche de conception des techniques d'interaction dédiées à des données temporelles. Nous focalisons sur les deux configurations du processus de visualisation peu couvertes par les travaux de la littérature tout en garantissant l'utilisabilité des techniques développées. Pour cela, nous adoptons une approche de conception centrée tâches utilisateur. Nous présentons une liste organisée de tâches utilisateur en termes de manipulation de données temporelles. Cette liste n'est ni exhaustive ni générale. En effet, nous l'avons établie grâce à une collaboration étroite avec le laboratoire SEIGAD (Systèmes Environnementaux, Information Géographique et Aide à la Décision) de Géographie de l'Université de Grenoble. Notre démarche de conception repose donc sur l'identification des besoins et des tâches des utilisateurs dans le cadre d'un domaine d'application particulier mais aussi des caractéristiques des données temporelles (Chapitre 1). Nous avons ensuite traduit ces besoins en termes de propriétés ergonomiques. Les propriétés ergonomiques représentent des contraintes générales qui doivent être vérifiées par la technique d'interaction à concevoir. Nous illustrons ensuite notre démarche de conception par la proposition de cinq techniques que nous avons conçues pour l'analyse visuelle de données temporelles.

Le Chapitre 5 présente alors INVEST (Interactive Visualization and Explorative System of Temporal data), notre plate-forme logicielle de gestion de vues multiples de données temporelles. En particulier, INVEST inclut les cinq techniques d'interaction présentées au Chapitre 4. Aussi, dans une première partie, nous exposons les résultats de deux expérimentations que nous avons menées pour évaluer l'utilisabilité des différentes techniques présentées dans le Chapitre 4. Outre l'amélioration des techniques elles-mêmes, les résultats expérimentaux obtenus ont été exploités pour définir et développer le moteur de sélection des techniques à partir des tâches utilisateur. Nous présentons ensuite, dans la

## ***Chapitre 1. Introduction***

---

---

deuxième partie de ce chapitre, l'architecture logicielle d'INVEST, selon le modèle d'architecture conceptuelle PAC-Amodeus proposé par Nigay [Nigay 94] pour la conception de systèmes interactifs.

Nous concluons alors par une synthèse et une analyse critique de notre contribution : originalités, points forts et faiblesses y sont envisagés. Nous élargissons ensuite le cadre de l'étude, par l'esquisse de perspectives tant techniques qu'applicatives.

Une annexe est incluse à la fin du mémoire. Celle-ci définit les critères d'ergonomie établis par notre équipe IIHM du laboratoire CLIPS-IMAG pour caractériser la notion d'utilisabilité.

## *Chapitre 2*

### *Temps : perception humaine et représentation informatique*

Qu'est-ce que le temps ?

**“Si personne ne me le demande, je le sais ; mais si on me le demande et que je veuille l'expliquer, je ne le sais plus.”**

**Saint Augustain**



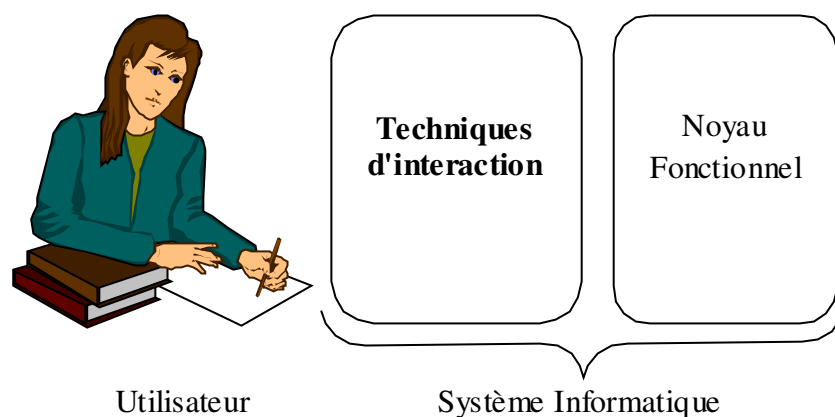
Table des matières

**Chapitre 2.** Temps : perception humaine et représentation informatique.

<u>1</u>	<u>Introduction</u> .....	17
<u>2</u>	<u>Perception humaine du temps</u> .....	18
<u>2.1</u>	<u>Processus de perception du temps</u> .....	18
<u>2.1.1</u>	<u>Perception basée sur les événements externes</u> .....	18
<u>2.1.2</u>	<u>Perception basée sur les événements internes</u> .....	19
<u>2.2</u>	<u>Représentation mentale du temps</u> .....	20
<u>2.2.1</u>	<u>Structure de l'espace temps</u> .....	20
<u>2.2.2</u>	<u>Éléments de l'espace temps</u> .....	21
<u>2.3</u>	<u>Synthèse</u> .....	24
<u>3</u>	<u>Représentation informatique du temps : point de vue base de données</u> .....	25
<u>3.1</u>	<u>Information temporelle</u> .....	26
<u>3.1.1</u>	<u>Définition d'une donnée temporelle</u> .....	26
<u>3.1.2</u>	<u>Dimension temporelle</u> .....	27
<u>3.2</u>	<u>Modèle du temps</u> .....	28
<u>3.2.1</u>	<u>Partition du temps</u> .....	28
<u>3.2.2</u>	<u>Unité d'observation</u> .....	29
<u>3.2.3</u>	<u>Calendriers</u> .....	31
<u>3.2.4</u>	<u>Types temporels</u> .....	32
<u>3.3</u>	<u>Modèle d'historique</u> .....	34
<u>3.3.1</u>	<u>Domaine temporel</u> .....	34
<u>3.3.2</u>	<u>Historisation des valeurs des données</u> .....	35
<u>3.4</u>	<u>Synthèse</u> .....	36
<u>4</u>	<u>Conclusion</u> .....	37

## 1 Introduction

Cette thèse concerne la conception de techniques de visualisation et plus généralement d'interaction pour des données temporelles. La Figure 2 rappelle, s'il en était besoin, le rôle d'une technique d'interaction dans un système informatique : un intermédiaire matériel et logiciel entre l'utilisateur et le noyau fonctionnel. Nous entendons par Noyau Fonctionnel, le logiciel modeler des concepts de la tâche informatisée manipulant des données temporelles. La spécificité de nos travaux repose sur la dimension temporelle des données considérées. Aussi ce chapitre est-il dédié à l'étude du temps selon deux facettes : (1) l'utilisateur et la perception humaine du temps et (2) le Noyau Fonctionnel et la représentation informatique du temps.



*Figure 2* : Technique d'interaction, intermédiaire entre l'utilisateur et le Noyau Fonctionnel. Dans ce chapitre, la première partie, intitulée "perception humaine du temps", expose des résultats issus de la psychologie de la perception. Notre objectif est de présenter des éléments expliquant la perception humaine du temps, sans pour autant viser à l'exhaustivité ni envisager de contribuer au domaine de la psychologie. Nous concluons cette partie par une synthèse des caractéristiques que nous retenons pour nos travaux : ces caractéristiques ont un impact sur l'utilisabilité des techniques d'interaction et/ou ont inspiré les représentations informatiques des données temporelles, objet de la deuxième partie de ce chapitre. Dans cette deuxième partie, nous nous concentrons sur le Noyau Fonctionnel du système informatique et nous étudions les représentations informatiques des données temporelles. A cette fin, nous nous reposons sur les travaux développés en bases de données temporelles. Notons ici qu'il faut considérer les résultats exposés dans ce chapitre dans le contexte de cette thèse, c'est-à-dire pour caractériser les données temporelles telles qu'elles sont utilisées en base de données.

L'étude dans ce chapitre des deux extrêmes de la Figure 2, l'utilisateur et le Noyau Fonctionnel, nous conduira naturellement à étudier au Chapitre 3, les techniques d'interaction, intermédiaires entre ces deux extrêmes.

## **2 Perception humaine du temps**

Adoptant une terminologie issue de l'informatique, nous considérons la perception du temps comme un processus dont la production est une représentation mentale du temps. Ce processus est influencé par deux types d'événements : les événements externes à l'utilisateur comme le lever ou le coucher du soleil et les événements internes ou biologiques comme le rythme cardiaque ou respiratoire. Dans cette partie, nous présentons d'abord le processus de perception : le processus basé sur des événements externes puis celui basé sur des événements internes. Le mécanisme de perception étant cerné, nous étudions ensuite son résultat : la représentation mentale du temps. Là encore nous adoptons une terminologie informatique, et nous considérons la représentation mentale du temps comme un espace d'informations que nous analysons selon sa structure puis ses éléments constitutifs. Nous le constaterons dans la partie suivante de ce chapitre, intitulée "représentation informatique du temps", les représentations informatiques du temps s'inspirent largement de ces représentations mentales.

### **2.1 Processus de perception du temps**

Une revue de la littérature permet de conclure que peu de travaux en psychologie sont dédiés à la perception du temps tandis que les perceptions visuelle ou encore auditive font l'objet de nombreuses études. L'homme voit des couleurs et entend des sons grâce aux sens humains de perception. A l'opposé, le processus de perception du temps n'est pas associé à un sens particulier. Dans ce paragraphe, pour étudier le processus de perception du temps et à des fins analytiques, nous considérons d'abord le processus basé sur les événements externes à l'homme puis celui basé sur les événements internes. Néanmoins, cette dichotomie est purement analytique et la perception humaine du temps est un processus complexe influencé par les deux types d'événements [Chenet 00].

#### **2.1.1 Perception basée sur les événements externes**

Comme annoncé en introduction, le temps n'est pas associé à un sens particulier. La question concerne donc l'existence même du temps. L'homme ne perçoit pas le temps lui-même, mais

des changements ou des événements et leurs relations dans le temps, par exemple, le lever ou le coucher du soleil, la naissance, etc. Un événement est "un fait auquel vient aboutir une situation" [petitRobert]. La situation est l'état résultant de l'événement. Un événement est donc défini par un état ou fait, et un instant à partir duquel le fait prend naissance. Il peut lui-même engendrer un autre événement. Définissant un événement par le couple (valeur d'une donnée, instant), où la valeur d'une donnée désigne le fait ou état, et l'instant, le point du temps à partir duquel le fait prend naissance, nous concluons qu'un événement n'est donc pas une composante du temps, mais il est défini par rapport à celui-ci et influence sa perception.

En conclusion, selon un processus de perception basé sur les événements externes, le temps n'est pas considéré isolément, mais perçu à travers des événements, externes à l'homme. De plus les événements affectent la perception du temps : des qualités et des caractéristiques différentes sont associées au temps en fonction des événements perçus. Par exemple, l'échelle du temps avec laquelle on observe le cycle de vie d'une cellule est différente de celle avec laquelle on observe le cycle de vie d'un astre.

Le temps dont la perception est influencée par des événements externes est qualifié de *temps réel*.

### **2.1.2 Perception basée sur les événements internes**

L'homme perçoit aussi le temps grâce à plusieurs rythmes biologiques de périodicités particulières. Le processus de perception est alors basé sur des événements dits internes. Par exemple, les battements du cœur ainsi que la respiration suivent des rythmes de différentes périodicités et sont commandés par un système interne (biologique), ou "horloge interne" [CNRS 01, Damasio 02]. De nombreuses études confirment l'existence d'une "horloge interne". Des études expérimentales ont même permis de localiser cette horloge : l'horloge est circonscrite au niveau du cervelet [Nair 99, Ivry 00, Jueptner 01]. Cette horloge permet d'estimer différents aspects du temps comme son passage ou encore les périodicités des rythmes biologiques, et d'anticiper les événements à venir.

Le temps dont la perception est influencée par l'horloge interne est qualifié de *temps biologique*.

## **2.2 Représentation mentale du temps**

Au paragraphe précédent, nous avons étudié le processus de perception du temps influencé par des événements externes ainsi que par l'horloge interne. Le résultat de ce processus est une représentation mentale du temps, objet de ce paragraphe.

Des études expérimentales, nous l'avons dit précédemment, ont montré que l'horloge interne est localisée au niveau du cervelet [Harrington 99] : à ce niveau, selon une hypothèse présentée par Herz [Herz 01] le domaine temporel est traduit en une dimension spatiale, et donc les informations temporelles sont mentalement représentées de façon spatiale. De cette hypothèse, nous concluons qu'une technique d'interaction reposant sur une représentation spatiale des données temporelles à l'écran peut être naturelle car la correspondance entre la représentation à l'écran et la représentation mentale est directe : en effet la distance d'évaluation comme définie par Norman [Norman 86] dans sa théorie de l'action est alors réduite.

Partant de cette hypothèse, nous considérons la représentation mentale du temps comme un espace d'informations noté *espace temps*. Adoptant la définition de Vernier [Vernier 01] adaptée de [Shneiderman 96], un espace d'informations et donc l'espace temps, repose sur une structure générale où chaque élément est une valeur ou une structure contenant des valeurs. Aussi dans ce paragraphe, nous étudions l'espace temps selon sa structure globale puis selon ses éléments constitutifs.

### **2.2.1 Structure de l'espace temps**

Nous distinguons trois structures de l'espace temps : cyclique, linéaire et logarithmique.

Aux événements externes ou internes, qui participent à la perception du temps, est associé un mode d'avancement dans le temps. Par exemple l'alternance de la nuit et du jour, ainsi que des saisons, les rythmes des battements du cœur et la respiration impliquent une perception du temps comme une succession de périodes : le temps est alors perçu de manière cyclique. Dans la Figure 3, les cercles traduisent le caractère cyclique du temps perçu [Freeman 00].

Une autre représentation mentale du temps est linéaire. En accord avec le modèle de Descartes (géométrie analytique), le temps est alors représenté par des points discrets sur une

ligne [Freeman 00]. Notons néanmoins que les représentations linéaire et cyclique du temps ne sont pas contradictoires, comme le schématise la Figure 3 :

- (1) Les cercles traduisent le caractère cyclique du temps.
- (2) La flèche (axe temps réel) représente le temps linéaire et indique la direction d'avancement du flux temporel. Ainsi le temps linéaire est une succession de cycles. Le modèle cyclique peut alors être interprété comme étant une extension ou un cas particulier du modèle linéaire.

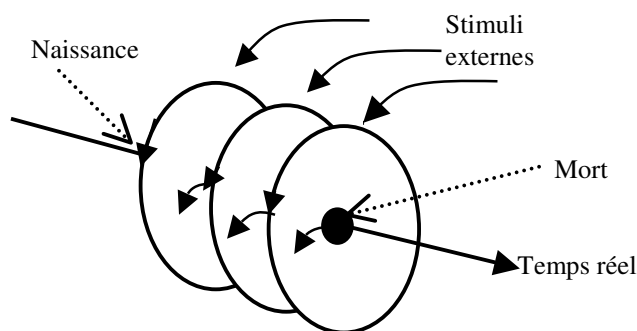


Figure 3 : Représentations linéaire et cyclique du temps [Freeman 00].

Outre les deux représentations du temps, cyclique et linéaire, il est proposé dans [Church 99]<sup>3</sup> une représentation mentale du temps approximativement logarithmique. Ce type de représentation permet d'expliquer le fait que notre perception du temps varie selon les situations. Ainsi dans des situations d'attente, nous avons souvent l'impression que le temps avance très lentement. Par contre, dans des situations de joie ou des situations de stress (avec une durée fixée comme un examen), nous avons l'impression que le temps passe très vite. Comme pour la représentation cyclique, la représentation logarithmique n'est pas à opposer à la représentation linéaire. Dans [Church 99], les deux représentations sont combinées.

### **2.2.2 Éléments de l'espace temps**

Dans ce paragraphe nous étudions les éléments du *temps réel*. Nous étudions successivement les relations des éléments qui le composent puis leurs granularités.

---

Citation :

<sup>3</sup> Staddon and Higa conclude that the psychological representation of time is approximately a logarithmic, rather than a linear function of time.

➤ **Relations entre éléments**

Comme expliqué au paragraphe 2.1.1, nous ne percevons pas le temps même, mais des changements ou des événements et leurs relations dans le temps. Les informations du monde externe sont perçues de façon continue selon un sens de perception (vue, ouïe, toucher, etc.). La perception du temps étant affectée par ces événements, le temps est donc perçu comme continu. Le temps est alors isomorphe aux nombres réels. Chaque nombre réel correspond à un point dans le temps dit instant. Entre deux instants, il existe un instant. Néanmoins le cerveau discrétise les informations pour pouvoir les traiter. Aussi l'estimation du temps peut aussi être modélisée par un processus basé sur des impulsions discrètes [Harrington 99]. La discrétisation des moments temporels nous permet de nous situer dans le temps. Nous considérons ainsi deux jours consécutifs comme disjoints et différents. Dans ce mémoire, nous retenons la vision discrète du temps plus facilement modélisable informatiquement.

Néanmoins, que le temps soit perçu comme continu ou discret, le présent correspond au moment de réalisation de l'événement en cours. Ce qui est défini avant ce moment est dit passé, et ce qui est défini après ce moment est dit futur. La perception d'un point temporel dans le passé passe forcément par la mémoire. Ainsi si une personne récapitule ce qui est arrivé à son environnement ou à elle-même entre les instants  $t-1$  et  $t+1$ , elle ne peut que conclure à l'existence du passé. Ce qui est futur à  $t$ , devient présent à  $t+1$ , et passé à  $t+2$ . Ainsi certains événements sont perçus comme passés non pas à cause de propriétés intrinsèques mais parce qu'ils ont été associés à d'autres événements considérés comme passés.

La notion de passé, présent et futur décline de la non réversibilité du temps [Chenet 00]. Pour comprendre cette propriété, nous opposons le temps à l'espace physique. L'espace peut être parcouru dans tous les sens sans mettre en cause ses constituants (horizontal, vertical et profondeur). Ceci est dû au fait que les composantes de l'espace physique coexistent, formant ainsi un milieu homogène. Une personne peut faire un aller et retour Grenoble-Paris et ainsi revenir au même point de départ. La direction du mouvement dans l'espace peut donc être inversée. A l'opposé, cette même personne ne peut pas revenir dans le temps et inverser sa direction.

L'importance de la notion de *passé, présent et futur* réside dans le fait qu'elle nous permet de définir un espace, dit ligne ou flèche du temps, dans lequel nous situons des événements. Les

trois composantes du temps (passé, présent et futur) ne coexistent pas. Un événement passé précède obligatoirement un événement présent qui lui-même précède un événement futur. Le temps peut alors être traduit en terme de nombres qui sont ordonnés selon le prédicat de comparaison (<). Ainsi une relation d'ordre permet de distinguer l'avant de l'après. Cette relation d'ordre est absolue et il convient de la distinguer de la relation d'ordre relative comme celle liée aux concepts de *demain*, *maintenant* et *hier*. En effet, ces concepts sont définis par rapport au point temporel en cours (le présent), et réfèrent une position dans l'une des trois composantes du temps. *Demain* représente le prochain point dans le temps, en considérant le jour comme étant l'unité de mesure utilisée.

Les points temporels se présentent successivement pour former une ligne de temps (LDT) [Callendar 01]. Ces points sont adjacents, provenant du passé et allant vers le futur. A un point donné de cette ligne temps, il peut y avoir plusieurs événements. Par rapport à cette ligne temps, la simultanéité concerne les événements et non pas le temps. Les secondes sont successives, mais en une seconde nous respirons et notre cœur bat.

Les relations entre événements dans le temps peuvent être représentées spatialement. Ainsi, les relations "précède" et "succède" se traduisent par "avant" et "après" et sont représentées par des positions dans un espace. Pour modéliser les relations temporelles entre deux événements, les relations d'Allen [Allen 91], définies pour les documents multimédias, sont le plus souvent utilisées. Ces relations spécifient le degré de séquentialité ou de parallélisme de deux événements dans le temps. Par exemple, la relation "rencontre", définie entre deux événements A et B, exprime le fait que B commence à l'instant suivant l'instant de fin de A. Les relations d'Allen ne prennent en compte que les instants de début et de fin des deux événements. D'autres relations entre événements sont nécessaires à modéliser. Elles correspondent en effet à des tâches utilisateurs que nous étudions au paragraphe 2 du Chapitre 4. C'est le cas de la relation de causalité entre événements : le fait que deux événements A et B se suivent ne signifie pas forcément que A implique B ou que A affecte B.

➤ ***Granularité d'un élément***

Le présent est défini de telle sorte que l'événement en cours ait une sémantique. Le présent correspond à tout l'espace temporel nécessaire à la réalisation de l'événement en cours. Le présent est donc défini par une durée. Pendant cette durée, deux événements e1 et e2 sont tous



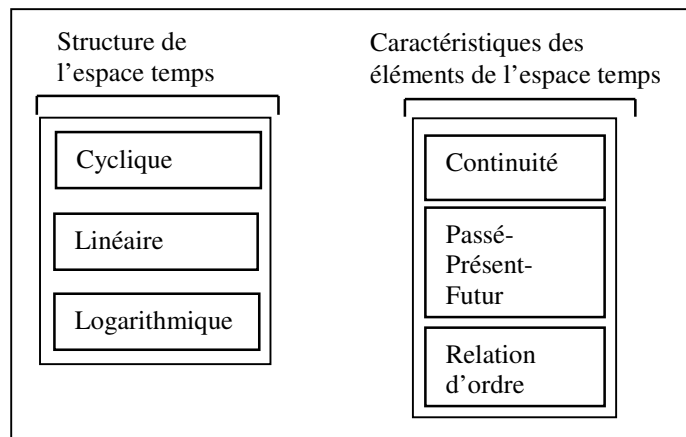
les deux perçus comme présent. La durée est donc la matière première du temps. L'évaluation de la durée dépend du traitement que l'on applique aux événements, ces derniers étant qualifiés d'informations non temporelles. La durée estimée dépendrait de la quantité d'informations non temporelles traitées pendant la période à évaluer [Ornstein d'après Pouthas 99]. La durée est une information dérivée d'autres informations. Par exemple, pour prévoir où il doit frapper la balle, le joueur de tennis doit intégrer la vitesse et la distance qui le sépare d'elle. L'évaluation d'une durée doit donc passer par une évaluation d'informations non temporelles. Ceci a une incidence directe sur le mode de représentation des données temporelles. A priori, le mode de représentation des valeurs des données (dimension structurelle étudiée au paragraphe 3.1) peut influencer certaines tâches utilisateur se rapportant à la dimension temporelle. Dans le Chapitre 5, nous étudierons en particulier l'influence que les caractéristiques de l'espace de représentation de la dimension structurelle peuvent avoir sur la tâche de navigation selon la dimension temporelle d'une donnée.

A l'opposé, comme l'indique [Chenet 00], d'autres études considèrent que le présent n'existe pas ou qu'il est instantané : "Le moment où je parle est déjà loin de moi". Le flux informationnel (sémantique) lié à l'événement en cours est alors ignoré. Le temps est alors indépendant des événements. Ceci remet en cause le processus de perception basé sur les événements externes ou internes exposé au paragraphe 2.1. Nous concluons que ces deux modélisations du présent ne sont pas contradictoires. C'est le niveau d'abstraction ou l'échelle d'observation qui change. Une année peut être considérée comme étant une durée ou comme étant un point (non décomposable) en fonction du phénomène observé.

### **2.3 Synthèse**

La prise en compte de la dimension temporelle est centrale à nos travaux sur la visualisation de données temporelles. Afin de cerner les caractéristiques du temps, nous nous sommes basés sur les travaux de recherche en psychologie de la perception. La Figure 4 organise les concepts que nous retenons pour la suite de ce manuscrit. Nous signalons ici que d'autres concepts relatifs au temps pourraient être pertinents dans un contexte autre que celui de la visualisation du résultat d'une requête base de données temporelles. En considérant le temps comme un espace d'informations, ces concepts se rapportent à deux dimensions : la structure de l'espace temps et les éléments unitaires de cet espace. Nous exploitons ces concepts lors de

l'étude de l'existant sur la visualisation de données temporelles (Chapitre 3) et de la conception de techniques d'interaction (Chapitre 4).



*Figure 4* : Perception humaine du temps : concepts retenus.

Après avoir étudié la représentation humaine du temps, il convient maintenant de focaliser sur la représentation informatique. Pour cela nous nous appuyons en particulier sur les travaux liés aux Bases de Données Temporelles (BDT). Néanmoins, nous ne prétendons pas faire un état de l'art sur la conception de Système de Gestion de Base de Données (SGBD) temporelles. Notre objectif est de discuter de la représentation informatique du temps et de la gestion de données temporelles. Nous constaterons dans le paragraphe suivant que les représentations informatiques sont souvent inspirées des représentations humaines du temps.

### **3 Représentation informatique du temps : point de vue base de données**

Plusieurs travaux de recherche en base de données se sont intéressés à la définition et au développement de modèles pour la gestion et la manipulation de données temporelles [Chomiki 94, Schmidt 95, Snodgrass 95, Steiner 98]. La prise en compte de l'évolution des données est dans les bases de données classiques, à la charge du développeur de l'application, ce qui accroît la complexité de développement. Les travaux en base de données temporelles visent à intégrer toutes les fonctionnalités nécessaires à la prise en compte du temps dans les Systèmes de Gestion de Bases de Données (SGBD). Nous nous basons essentiellement sur Tempos (Temporal Extension Models for Persistent Objects Systems) [Canavaggio 97, Dumas 00] qui est une plate-forme pour le développement d'applications temporelles au-

dessus de SGBD à objets. Le modèle Tempos étend le standard défini par l'ODMG (Object Database Management Group) [Cattell 00] pour la prise en compte de la dimension temporelle des données. Il inclut aussi TempODL en tant que langage de définition de schémas temporels et TempOQL comme langage de requêtes.

Le temps peut être associé aux attributs, permettant ainsi de garder une photographie de toutes les mises à jour de l'attribut dont on veut observer l'évolution. Cet ensemble de photographies que nous appelons historique représente la trajectoire de l'attribut dans le temps. L'état d'évolution d'un attribut peut être l'un des trois cas suivants :

- Constant, si la valeur de l'attribut ne varie jamais dans le temps,

S'il évolue, il est :

- Fugitif, si nous ne nous intéressons qu'à sa valeur la plus récente qui peut traduire un nouvel état (une évolution de la donnée) ou une correction d'une valeur erronée,
- Historique, si nous voulons garder la trajectoire de son évolution dans le temps. Tout au long de cette thèse, nous dénotons par donnée temporelle l'historique de l'évolution d'une donnée dans le temps.

## **3.1 Information temporelle**

### **3.1.1 Définition d'une donnée temporelle**

Une donnée temporelle est une séquence de couple  $\langle t, v \rangle$  où  $t$  dénote l'instant d'observation et  $v$  la valeur de la donnée à cet instant. La valeur  $t$  est dite valeur temporelle,  $v$  valeur structurelle et  $\langle t, v \rangle$  est dit instantané. La valeur temporelle peut être exprimée selon différentes granularités : jour, heure, ou minute. De même, la valeur structurelle a un type qui la caractérise. Elle peut être un nombre, une couleur, un objet, etc. Afin de délimiter notre cadre d'étude, nous ne considérons que des valeurs quantitatives. La Figure 5 présente un extrait des valeurs d'une donnée temporelle quantitative réelle sous une représentation alphanumérique. Chaque ligne représente un instantané. La valeur structurelle représente les valeurs du polluant NO<sub>2</sub> et la valeur temporelle représente l'instant d'observation correspondant. La valeur temporelle (lecture de droite vers la gauche) de l'instantané en gras dans la Figure 5 représente la deuxième quinzaine de minutes (**2**), à minuit (**00**), du 26<sup>ème</sup> jours (**26**) du mois de janvier (**01**) de l'an 2000 (**2000**).

Date	Valeur du Polluant NO2
<b>2000/01/26/00/2</b>	<b>53.48</b>
2000/01/26/00/3	59.21
2000/01/26/01/0	55.39
2000/01/26/01/1	32.47
2000/01/26/01/2	45.84
2000/01/26/01/3	26.74
2000/01/26/02/0	40.11
2000/01/26/02/1	57.3
2000/01/26/02/2	43.93
2000/01/26/02/3	30.56
2000/01/26/03/0	40.11
2000/01/26/03/1	22.92
2000/01/26/03/2	34.38
2000/01/26/03/3	40.11
2000/01/26/04/0	26.74
2000/01/26/04/1	32.47
2000/01/26/04/2	24.83
2000/01/26/04/3	26.74
2000/01/26/05/0	59.21
2000/01/26/05/1	19.1

Figure 5 : Extrait des valeurs d'une série de mesure de NO2.

### 3.1.2 Dimension temporelle

L'évolution des valeurs peut être observée selon plusieurs dimensions temporelles. La distinction la plus notable consiste à spécifier deux dimensions temporelles à savoir le temps de validité et le temps de transaction [Chomiki 94, Jensen 98, Snodgrass 95].

Le temps de validité (*valid time*) concerne le temps pendant lequel un événement est valide dans la réalité indépendamment de toute action d'enregistrement de cet événement dans une base de données. Le temps de validité peut être dans le futur si on prévoit que le fait sera valide dans la réalité à une date future. Cette dimension permet donc de refléter le passé et d'anticiper la réalisation des faits dans le futur. Le temps de validité correspond à la valeur temporelle saisie au niveau de la base de données.

Le temps de transaction (*transaction time*) correspond au temps où une donnée est enregistrée dans la base de données. Il est borné du fait son domaine temporel (paragraphe 3.3) étant défini de la date de création de la base de données jusqu'au présent. Aucune information ne

peut être stockée avant la création de la base et ne peut pas dépasser le présent puisque le temps système est toujours borné par le présent. La valeur temporelle est fournie par le système, seule la valeur structurelle est fournie par l'utilisateur. Dans le cas où cette valeur structurelle est erronée, elle est enregistrée par le système par rapport au temps de transaction et l'utilisateur ne peut pas la modifier. L'utilisateur doit saisir une nouvelle valeur pour exprimer la correction.

Nous venons de fixer la définition d'une donnée temporelle. Il convient maintenant d'étudier la représentation du temps en base de données temporelles. Dans cet objectif, nous étudions le modèle du temps implémenté dans Tempos [Canavaggio 97, Scholl 98, Dumas 00].

## **3.2 Modèle du temps**

Pour étudier le modèle du temps, nous étudions successivement la partition du temps, la notion d'unité d'observation, la notion de calendrier et les types temporels définis dans Tempos.

### **3.2.1 Partition du temps**

Dans la littérature, les travaux sur la modélisation du temps se sont concentrés sur plusieurs modèles, le plus commun étant le modèle linéaire et discret. Avec un modèle linéaire (étudié au paragraphe 2.2.1) et discret (étudié au paragraphe 2.2.2), le temps est représenté comme une droite partitionnée en une suite d'intervalles consécutifs et disjoints comme l'illustre la Figure 6. Chaque intervalle est considéré comme un point non décomposable (tout instant de la droite des réels est approximé par le point qui le contient) et numéroté par un entier. Ce point non décomposable est appelé *chronon* et il représente la quantité de temps (ou la durée) la plus petite que l'on peut représenter. Le chronon est une modélisation informatique du concept d'*Élément de l'espace temps* (étudié au paragraphe 2.2.2). L'ordre entre les entiers correspond à la succession des points dans le temps conformément à la relation d'ordre (étudiée au paragraphe 2.2.2). Les points sont numérotés par des entiers positifs ou nuls pour prendre en compte le passé et le futur (étudiés au paragraphe 2.2.2). Ceci suppose bien entendu que la droite du temps est bornée par un instant d'origine de numéro 0 comme le montre la Figure 6.

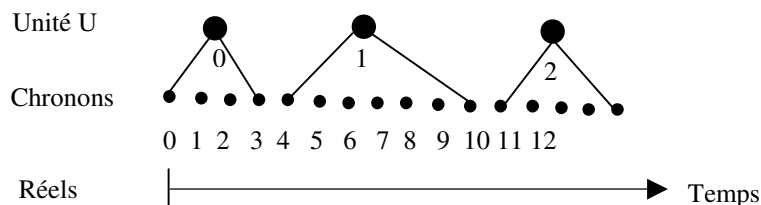


Figure 6 : Partition du temps.

Une partition particulière que nous considérons est celle des chronons. Toute autre partition est construite sur l'ensemble des chronons. Chaque partition possible de la droite des réels définit une unité d'observation.

### 3.2.2 Unité d'observation

Une unité d'observation [Canavaggio 97] (dite aussi granularité d'observation [Dumas 00]) permet de fixer le niveau d'abstraction de l'observation des instants dans le temps. Elle permet ainsi d'interpréter un intervalle de temps comme un instant ponctuel. Dans la Figure 6, l'intervalle des chronons [0..3] est approximé par le grain 0 de l'unité U. Une unité d'observation est définie par rapport à un calendrier [Chandra 94]. Ce dernier consiste en un ensemble de repères significatifs dans un contexte donné. Une définition complète des calendriers est présentée au paragraphe 3.2.3. Des exemples d'unités d'observation sont année, mois, jour, heure et minute. L'utilisation des unités d'observation définit une structure multigranulaire du temps [Canavaggio 97, Dumas 00] : une année est composée de 12 mois, un mois de 28, 29, 30 ou 31 jours, un jour de 24 heures, etc.

Dans la suite, nous étudions les relations qui peuvent exister entre les unités d'observation. Nous présentons trois relations particulières : relation de régularité, relations d'ordre entre unités d'observation et unités comparables.

#### 3.2.2.1 Relation de régularité entre unités d'observation

Une unité d'observation est dite régulière si les grains de la partition qui la définissent ont tous la même taille, comme les jours, les heures, les minutes ou les secondes. Dans le cas inverse, c'est-à-dire lorsque les grains ont des tailles variables comme pour l'année ou les mois, l'unité d'observation est dite irrégulière. Il est à noter que l'unité chronon représente par

construction, une unité d'observation régulière et que si U1 est régulière par rapport à U2 et que U2 est régulière par rapport aux chronons alors U1 est régulière par rapport aux chronons.

### 3.2.2.2 Relations d'ordre entre les unités d'observation

Une unité d'observation U2 est dite *plus-fine* qu'une seconde unité d'observation U1 si tout grain de U1 est construit à partir d'un ensemble de grains consécutifs de U2. Dans ce cas, U1 est dite *plus-grossière* que U2. Nous disons ainsi que l'unité minute est *plus-grossière* que l'unité seconde (toute minute est composée de 60 secondes) et que l'unité seconde est *plus-fine* que l'unité minute.

### 3.2.2.3 Unités comparables

Deux unités d'observation sont dites comparables si l'une d'elle réalise une partition de l'autre comme le cas de l'unité jour et heure. Dans le cas contraire, elles sont dites non comparables : c'est le cas des unités U0 et U2 à la Figure 7. Un exemple d'unités non comparables est le mois et la semaine. En effet, un mois ne contient pas un nombre exact de semaines.

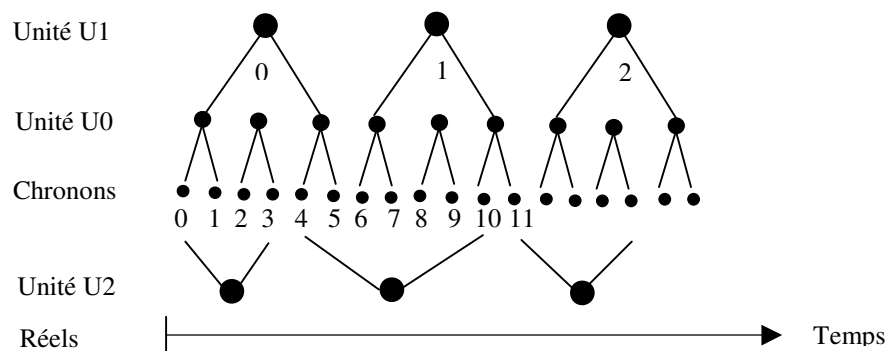


Figure 7 : Unités comparables et non comparables.

La conversion vers une unité U', d'une valeur temporelle exprimée dans une unité U ne peut se faire que si les unités U et U' sont comparables. Nous parlons d'expansion pour le cas de passage d'une unité U' vers une unité U plus fine, notée  $\epsilon_{u', u}$  à la Figure 8. Par exemple, U' correspond au jour tandis que U à l'heure. De même, nous parlons d'approximation pour le cas du passage d'une unité U à une unité U' plus grossière et est notée  $\alpha_{u, u'}$  à la Figure 8. Par exemple, U correspond à l'heure tandis que U' au jour.

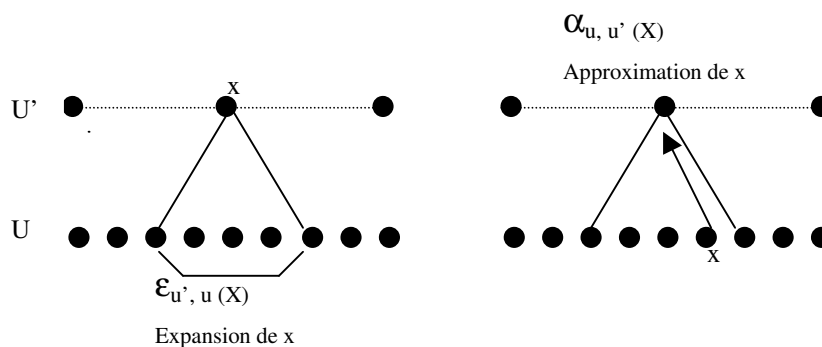


Figure 8 : Conversion entre unités comparables.

### 3.2.3 Calendriers

La définition de la ligne temps (LDT), nous amène à fixer un point d'origine et une unité d'observation qui serviront à repérer les événements dans cet espace. Ainsi un instant  $I$  est défini par la distance qui le sépare de ce point d'origine. En considérant la seconde, par exemple, comme étant l'unité d'observation, un instant  $I$  sera défini par le nombre de secondes qui le séparent du point origine. Se limiter ainsi à l'utilisation des distances par rapport à l'origine rend difficile le repérage dans le temps. Aussi dans le but de situer un événement quelconque sans ambiguïté, des échelles de temps et des systèmes de numérotation ont été définis : les calendriers.

Plusieurs calendriers ont été développés pour fixer diverses interprétations du temps. Les calendriers sont fondés sur des traits astronomiques. Le mois est l'intervalle de temps séparant deux nouvelles lunes. Le calendrier Grégorien (de type solaire), par exemple, décompose la ligne temps en années, mois, jours, etc. D'autres types de calendriers peuvent être définis pour spécifier des régularités particulières, par exemple, les jours ouvrables, ou tous les lundis. Nous pouvons aussi définir un calendrier qui décompose le temps en années, semestres, deux mois, dizaines de jours, jours, trois heures, etc.

Une même valeur temporelle définie par rapport à un calendrier donné peut être représentée sous plusieurs formes, dites représentations externes, comme le témoigne le corpus suivant :

- "5/7/1999" (dans le format Européen)
- "7/5/1999" (dans le format Américain)
- "5 Juillet 1999" (en Français)
- "July 5, 1999" 5 (en Anglais)



- "1999-07-05"(selon le standard ISO-8601)

Cette variété est due à divers facteurs : calendrier utilisé, contexte linguistique et culturel, habitudes personnelles, etc. Les propriétés des représentations externes des valeurs temporelles dépendant du contexte linguistique et culturel, sont regroupées dans Tempos sous la notion de format. Ainsi, nous parlons du format Européen des dates. Ce format interprète les dates dans le système JMA (Jour Mois Année). Par exemple, la chaîne de caractères "5/7/1999" dénote le cinquième jour du mois de Juillet de l'année 1999. Par ailleurs, les formes externes de ce format dénotent des instants observés à la granularité du jour. Dans ce manuscrit, nous nous limitons aux trois formats suivants : Européen, Américain et ISO.

### **3.2.4 Types temporels**

Les modèles temporels développés en base de données temporelles définissent plusieurs types temporels. Nous présentons ici ceux implémentés dans Tempos [Canavaggio 97, Dumas 00].

- Instant : un instant ou temps absolu (*absolute time*) est un grain particulier de la droite du temps. Certains modèles du temps considèrent un instant comme atomique et il n'a pas de durée relativement à son unité d'observation. D'autres modèles comme [Zaniolo 97] considèrent un instant comme une durée. En particulier, ils considèrent le chronon non pas comme un point mais comme un segment de la droite du temps et plus précisément comme étant la plus petite durée à représenter. Ces deux cas ne sont que des implémentations des deux visions sur les éléments du temps, c'est-à-dire élément temporel avec durée et élément temporel sans durée que nous avons étudiés au paragraphe 2.2.2.

Exemple : "1997" est un instant décrit dans l'unité Année.

- Durée : la durée est considérée comme un intervalle non-ancré sur la droite du temps. La durée correspond dans le standard SQL2<sup>4</sup> à un intervalle de temps : une durée signée, définie comme une quantité de temps avec une longueur déterminée mais sans avoir des bornes sur la droite du temps [Zaniolo 97]. Une durée est à distinguer d'une distance, cette dernière étant toujours positive. La valeur d'une durée dont l'unité d'observation est irrégulière n'est pas définie avec précision. Pour des unités régulières, il est possible de

---

<sup>4</sup> SQL2 (Structured Query Language 2) est une extension du langage SQL.

faire des conversions de durées d'une unité vers une autre. Par exemple la durée 2 exprimée dans l'unité heure est égale à la durée 120 exprimée dans l'unité minute.

- Ensemble d'instants (EDI) : un ensemble d'instants (définis dans la même unité) peut être représenté, selon le cas, sous forme d'intervalle, de séquence ou d'une séquence d'intervalles (D-séquence) :

- Intervalle : c'est l'ensemble de grains adjacents deux à deux, délimités par deux instants particuliers représentant les deux bornes de l'intervalle.

Par exemple : [ '01/03/1986' ... '10/06/1997' ]. L'unité d'observation d'un intervalle est celle de ses grains. Dans l'exemple c'est l'unité jour.

- Séquence : nous parlons d'une séquence d'instants périodiques dans le cas particulier où un ensemble d'instants est observé selon la même unité et en ordre chronologique tel que deux instants consécutifs sont séparés par la même durée non nulle : cette durée représente la période de la séquence [Tuzhilin 95]. C'est une implémentation de la perception cyclique du temps présentée au paragraphe 2.2.1.

Par exemple : [ '03/10/06/1999', '05/10/06/1999', '07/10/06/1999', '09/10/06/1999', '11/10/06/1999' ]. Dans l'exemple, l'unité d'observation est l'heure et la durée qui sépare deux instants consécutifs est égale à 2 heures.

- D-séquence : c'est une séquence finie d'intervalles non vides de même unité d'observation et qui sont deux à deux disjoints et non contigus, et ordonnés selon la relation "précède" : la borne supérieure d'un intervalle de rang i dans la séquence est strictement inférieure aux bornes inférieures de tous les intervalles subséquents.

Par exemple : { ['01/03/1986'...'10/06/1986'], ['09/02/1991'...'10/06/1992'], ['01/04/1994'...'22/09/1997'] }. L'unité d'observation d'un élément temporel est celle des intervalles qui le composent. Dans l'exemple c'est l'unité jour.

La Figure 9 fournit un modèle UML [UML 03] des types temporels définis dans Tempos [Dumas 00]. Ces types temporels complètent ceux déjà fournis par le modèle ODMG [Cattell 00] à savoir Date, Timestamp, Interval, et Time. La sémantique de ces types

temporels est définie par rapport à des lignes de temps et des granularités fixées par le modèle ODMG.

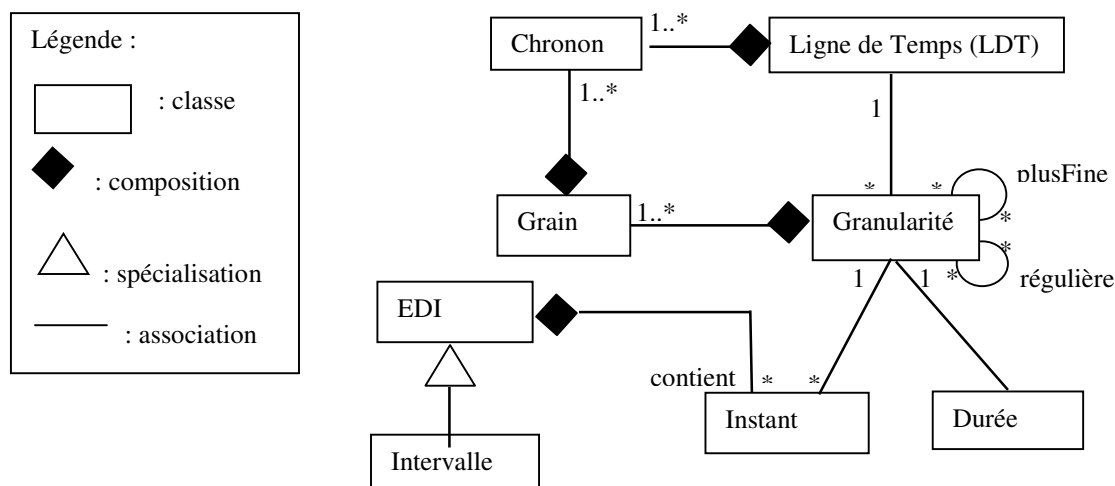


Figure 9 : Modélisation UML des types temporels [Dumas 00].

### 3.3 Modèle d'historique

Le paragraphe précédent a présenté le modèle du temps défini dans Tempos. Afin de montrer comment les données temporelles sont définies selon ce modèle, nous présentons dans la suite la notion de domaine temporel puis l'historisation des valeurs des données.

#### 3.3.1 Domaine temporel

Le domaine temporel d'une donnée temporelle est l'ensemble des instants observés dans la même unité d'observation. Il fixe l'ensemble des moments d'observation des données. Le domaine temporel peut être fixé explicitement par l'utilisateur ou aussi déterminé dynamiquement à partir des instants estampillant les valeurs structurelles.

Le domaine temporel est défini par l'union du domaine effectif et du domaine potentiel de l'historique [Canavaggio 97, Dumas 00]. Le domaine effectif d'un historique est construit par les instants pour lesquels les mises à jour ont fourni des données, c'est-à-dire les instants de saisie de nouvelles valeurs. Le domaine potentiel d'un historique est construit par les instants pour lesquels les mises à jour n'ont pas fourni de données.

Nous distinguons deux types d'objets temporels : les objets temporels dits ancrés dont le domaine temporel est défini explicitement et est stocké physiquement comme l'évolution

dans le temps d'un attribut salaire. Un domaine temporel dit non-ancré est défini à partir de son temps d'exécution tel que les objets multimédias. La vidéo est un objet temporel non-ancré. Si nous considérons une vidéo comme une succession d'images, le temps estampillant ses images n'est connu qu'au moment de l'exécution de la vidéo, en se basant sur le séquençement des images. Nous ne pouvons pas définir le domaine temporel de tel objet à l'avance, ni stocker ses valeurs dans la base par rapport à un domaine temporel.

Le domaine temporel de la donnée représentée à la Figure 5 est défini par l'ensemble des 15Minutes (l'unité d'observation est 15Minutes) compris entre 2000/01/26/00/2 et 2000/01/26/05/1.

### 3.3.2 Historisation des valeurs des données

Les instants correspondant aux instants de saisie de nouvelles valeurs (dimension structurelle) représentent le domaine temporel effectif. Entre deux instants de saisie, il peut y avoir des instants pour lesquels on ne connaît pas la valeur des données. Ces valeurs sont alors calculées selon une fonction d'interpolation (dite aussi modalité d'interpolation). Tempos inclut trois fonctions d'interpolation :

**Discret** : les valeurs de l'historique aux instants du domaine potentiel sont toutes égales à la valeur neutre du type de cette donnée, ou à une autre *valeur par défaut* fournie par l'utilisateur. Cette fonction d'interpolation est utilisée pour modéliser des phénomènes comme la production d'une usine (voir la Figure 10). Ainsi, une absence de production au 14/01/2001 et au 15/01/2001 implique que la valeur est égale à zéro (donnée quantitative).

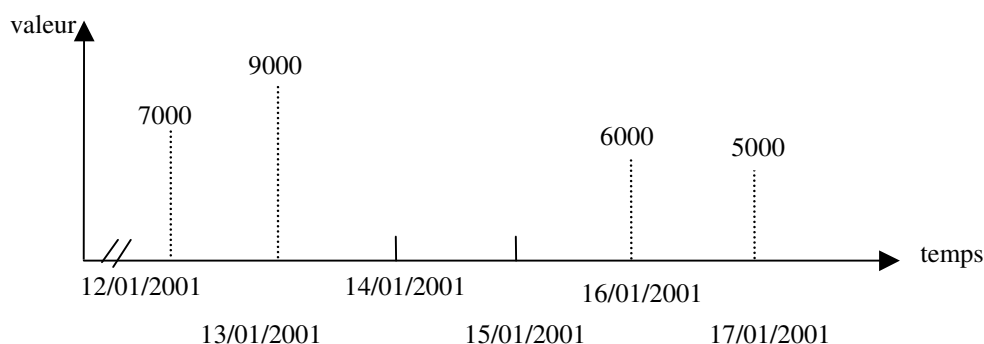


Figure 10 : La production d'une usine : phénomène discret.

**En escalier** : la donnée considérée évolue de telle manière que ses valeurs restent stables entre deux instants dans le domaine effectif. Ceci est par exemple le cas du salaire d'un employé. Le salaire demeure le même tant qu'il n'y a pas eu une nouvelle valeur (Figure 11).

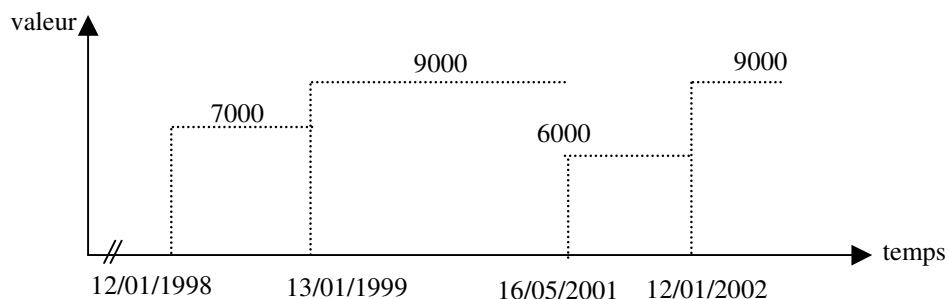


Figure 11 : Le salaire d'un employé : un phénomène en escalier.

**Interpolée** : entre deux valeurs successives dans le domaine effectif, la valeur de la donnée est calculée par une fonction d'interpolation. Ceci est par exemple le cas du débit d'un fleuve. A la Figure 12, cette fonction est linéaire. Cette fonction d'interpolation ne s'applique qu'à des valeurs numériques.

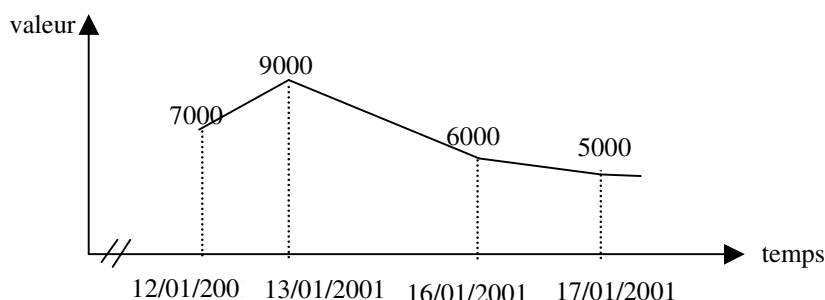


Figure 12 : Le débit d'une fleuve : un phénomène interpolé par une fonction linéaire.

### 3.4 Synthèse

Dans ce paragraphe, nous avons présenté une représentation informatique des données temporelles, et non un état de l'art des travaux. Nous nous sommes focalisés sur le modèle Tempos [Canavaggio 97, Dumas 00]. Le modèle Tempos étend le standard défini par l'ODMG pour la prise en compte de la dimension temporelle des données.

La représentation informatique du temps s'est beaucoup inspirée de la perception humaine du temps. La Figure 13 met en correspondance les concepts retenus liés à la perception humaine du temps et leurs concrétisations au niveau de la représentation informatique selon le modèle Tempos. Comme le montre cette figure, nous retrouvons les perceptions cyclique et linéaire

du temps concrétisées dans le monde informatique en utilisant un modèle linéaire du temps et des opérateurs temporels pour identifier les périodicités des données. La perception logarithmique du temps n'a pas été considérée dans le modèle Tempos. Nous notons aussi que la représentation multigranulaire du temps étudiée au paragraphe 3.2.2 n'a pas été considérée dans les travaux sur la perception humaine du temps. La représentation multigranulaire du temps peut être nécessaire si l'utilisateur s'intéresse aux relations entre valeurs temporelles définies dans différentes unités d'observation.

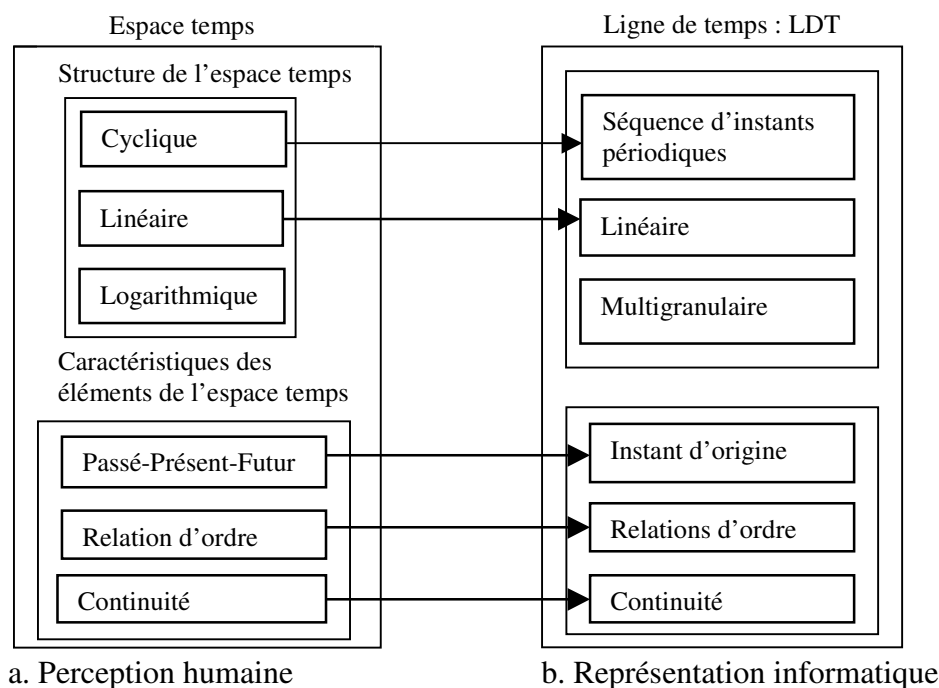


Figure 13 : Représentation du temps.

## 4 Conclusion

Notre étude vise à exploiter et respecter les caractéristiques du temps durant le processus de conception des techniques de visualisation de données temporelles. Dans ce chapitre, nous avons présenté une étude du temps selon deux facettes : la perception humaine et la représentation informatique.

- Nous avons considéré la perception du temps comme un processus dont la production est une représentation mentale. De plus, nous avons considéré cette représentation mentale du temps comme un espace d'informations, ce qui nous a permis de distinguer deux dimensions : 1) la structure de cet espace et 2) les caractéristiques de ses éléments unitaires. La Figure 4 (page 25) présente les

concepts que nous avons retenus par rapport aux deux dimensions de l'espace temps.

- Pour l'étude de la représentation informatique des données temporelles, nous nous sommes tournés vers les travaux en bases de données temporelles. Pour se faire, nous nous sommes basés sur le modèle Tempos [Canavaggio 97, Dumas 00]. Nous l'avons souligné tout au long de ce chapitre, la représentation informatique du temps s'est beaucoup inspirée de la perception humaine du temps.

Ce chapitre établit donc les fondements à nos travaux en caractérisant une donnée temporelle, tant du point de vue de la perception humaine que du point de vue informatique. Avant de présenter notre travail de conception des techniques de visualisation de données temporelles, il convient tout d'abord de recenser les techniques développées dans la littérature. Nous consacrons le Chapitre 3 à l'étude des techniques de visualisation de données temporelles proposées dans la littérature.

## *Chapitre 3*

### *Techniques de visualisation de données temporelles*

**“La raison peut se comparer à une montre : on ne voit point marcher l'aiguille ; elle marche cependant, et ce n'est qu'au bout de quelque temps qu'on s'aperçoit du chemin qu'elle a fait.”**

**Jean le Rond d'Alembert  
Philosophe et mathématicien  
Français**



Table des matières

**Chapitre 3.** Techniques de visualisation de données temporelles.

1	<a href="#">Introduction</a> .....	41
2	<a href="#">Processus de visualisation de données</a> .....	41
3	<a href="#">Processus de visualisation de la dimension structurelle</a> .....	45
3.1	<a href="#">Techniques orientées pixels</a> .....	46
3.2	<a href="#">Techniques à base d'éléments graphiques</a> .....	50
3.3	<a href="#">Synthèse</a> .....	52
4	<a href="#">Processus de visualisation de la dimension temporelle</a> .....	52
4.1	<a href="#">Espace temps</a> .....	53
4.2	<a href="#">Point de vue sur l'espace temps</a> .....	55
4.2.1	<a href="#">Formes cycliques du temps</a> .....	56
4.2.2	<a href="#">Formes linéaires du temps</a> .....	59
4.2.3	<a href="#">Forme logarithmique du temps</a> .....	63
4.2.4	<a href="#">Forme multigranulaire du temps</a> .....	63
4.2.5	<a href="#">Relations entre formes de représentation du temps</a> .....	64
4.3	<a href="#">Synthèse</a> .....	65
5	<a href="#">Techniques de visualisation d'une donnée temporelle</a> .....	66
5.1	<a href="#">Fusion complète ou fusion Point de vue sur la donnée</a> .....	68
5.2	<a href="#">Fusion Espace de visualisation</a> .....	71
5.3	<a href="#">Fusion Point de vue sur l'espace de visualisation</a> .....	74
5.4	<a href="#">Temps puis Espace de visualisation de la dimension structurelle</a> .....	75
5.5	<a href="#">Temps puis Point de vue sur l'espace de visualisation de la dimension structurelle</a> .....	77
5.6	<a href="#">Processus complètement séparés</a> .....	79
5.7	<a href="#">Synthèse</a> .....	79
6	<a href="#">Techniques de visualisation de plusieurs données temporelles</a> .....	81
6.1	<a href="#">Utilisation d'un seul espace temps</a> .....	82
6.2	<a href="#">Utilisation d'un espace temps par donnée</a> .....	86
6.3	<a href="#">Synthèse</a> .....	87
7	<a href="#">Conclusion</a> .....	88

## **1 Introduction**

Après avoir étudié les représentations mentales et informatiques du temps, nous consacrons ce chapitre aux techniques de visualisation de données temporelles. Une revue des travaux de visualisation de données temporelles proposées dans la littérature montre d'une part le dynamisme de cet axe de recherche et d'autre part le manque de standard et de cadre conceptuel pour classer ces travaux. Ce chapitre vise à recenser et à organiser dans une taxonomie les techniques de visualisation de données temporelles existantes.

Nous basons notre taxonomie sur les étapes du processus de visualisation proposé par Chi [Chi 00, Chi 02b]. L'originalité de nos travaux réside ici dans le fait que nous appliquons ce processus de visualisation aux deux dimensions structurelle et temporelle d'une donnée temporelle.

Ce chapitre est organisé comme suit : nous présentons d'abord le processus de visualisation de Chi. Nous étudions par la suite le processus de visualisation de la dimension structurelle d'une donnée, puis celui de sa dimension temporelle. Nous présentons enfin notre taxonomie de techniques de visualisation par rapport au processus de visualisation des deux dimensions d'une donnée temporelle.

## **2 Processus de visualisation de données**

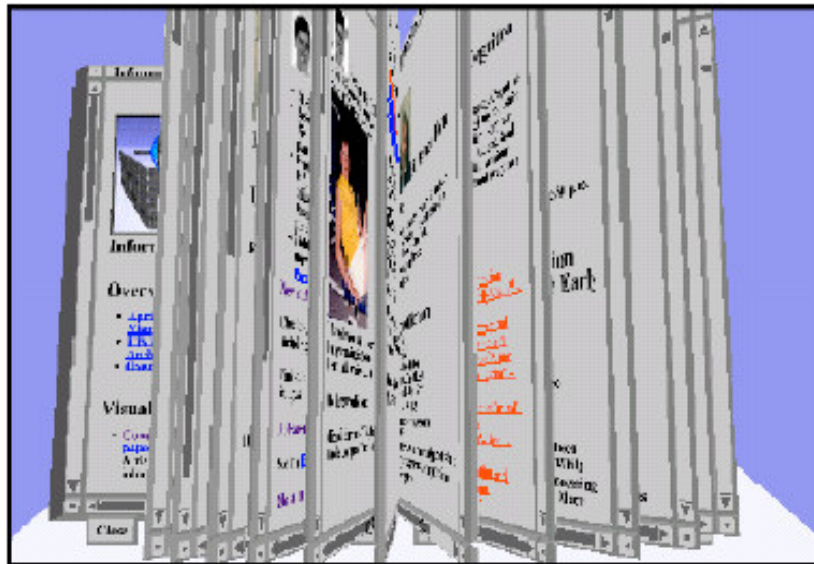
Dans ce paragraphe, nous présentons le processus de visualisation dit *modèle d'état de données* proposé par Chi [Chi 00, Chi 02b]. D'après Chi [Chi 02], le *modèle d'état de données* est équivalent, en terme de pouvoir d'expression, au *modèle de flux de données*<sup>5</sup>. Le *modèle de flux de données* est classiquement utilisé en visualisation de données. Par exemple, dans [Baudel 02] le modèle de flux de données est utilisé pour caractériser les techniques de visualisation compactes. Ces techniques de visualisation sont dites compactes car elles permettent de visualiser un grand espace de données dans un espace de représentation réduit.

---

Citation:

<sup>5</sup> In terms of expressiveness anything that can be represented using the Data Flow Model can also be represented using Data State Model, and vice versa.

Nous illustrons ce processus par un exemple qui est la technique webBook. La technique webBook [Card 96] présentée à la Figure 14 permet de visualiser une collection de pages web sous la forme d'un livre dans un espace de représentation en trois dimensions.



*Figure 14 : Technique WebBook [Card 96].*

Comme le montre la Figure 15a, Chi [Chi 00, Chi 02a , Chi 02b] propose de décomposer le processus de visualisation en quatre niveaux de données et trois types de transformation<sup>6</sup>. Dans ce manuscrit, nous adoptons la définition proposée par Vernier [Vernier 01] pour structurer le processus de Chi en quatre étapes : *donnée*, *point de vue sur la donnée*, *espace* et *point de vue sur l'espace*. Nous appelons les deux dernières étapes *espace de visualisation* et *point de vue sur l'espace de visualisation*. A la différence de Vernier [Vernier 01] qui définit une étape du processus de visualisation par un niveau de donnée, nous la définissons par le couple niveau de donnée et la transformation qui a produit ce niveau de donnée (Figure 15c). Dans la suite, nous présentons chacune des quatre étapes du processus de visualisation de Chi.

---

<sup>6</sup> Les quatre niveaux de données sont respectivement nommés dans le texte original en anglais : Data, Analytical Abstraction, Visualization Abstraction et View. Les trois types de transformations sont : Data Transformation, Visualization Transformation et Visual Mapping Transformation

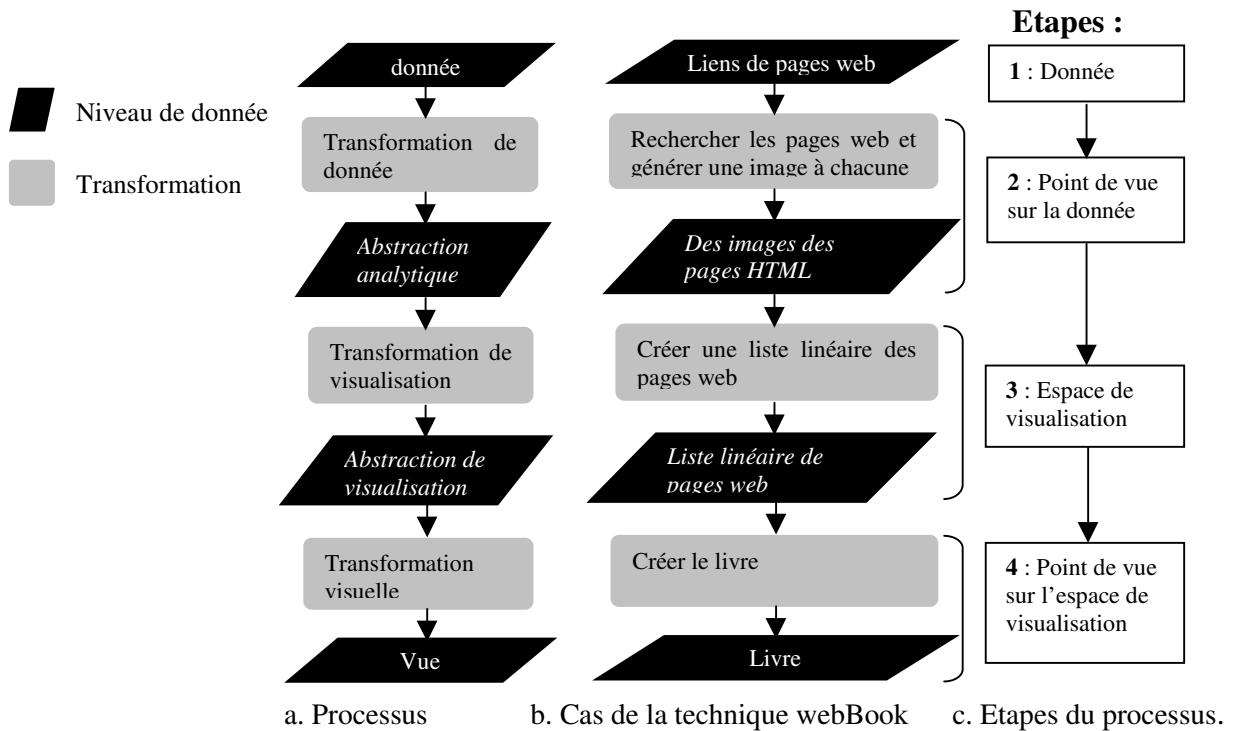


Figure 15 : Processus de visualisation de données de Chi [Chi 00].

**Donnée** : l'étape *donnée* représente les valeurs de la donnée (dans l'état brut) à visualiser. Comme nous l'avons déjà défini au chapitre précédent, une donnée temporelle est une association de valeurs structurelles (sa dimension structurelle) et de valeurs temporelles (sa dimension temporelle). La Figure 16 souligne cette distinction. Les valeurs de la dimension structurelle ont un type qui les caractérise. Nous distinguons en particulier des données 1D qui peuvent être de nature quantitative, nominale ou ordonnée comme l'ont défini Card et Mackinlay [Card 97]. Les données 2D sont par nature définies dans un espace en deux dimensions comme les images ou les cartes géographiques. Les données 3D sont par exemple les objets du monde réel.

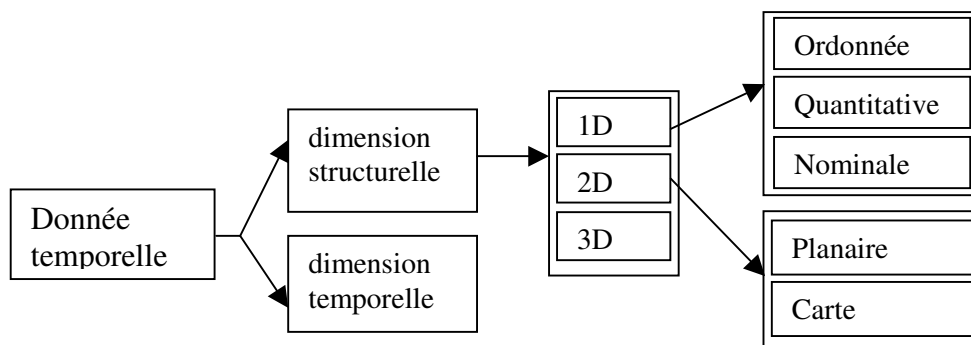


Figure 16 : Dimensions d'une donnée temporelle.

**Point de vue sur la donnée :** cette étape est définie par la transformation des valeurs de la donnée à visualiser. Cette transformation est spécifique à un certain type de structures de données. Elle génère, généralement par extraction, une abstraction analytique ou méta-données (informations concernant les données) à partir des valeurs de la donnée à visualiser. Le *point de vue sur la donnée* définit ainsi le type d'information à représenter au niveau d'une technique de visualisation. Dans l'exemple de la Figure 15b, la transformation de la donnée consiste en la recherche des pages web définies par des URL et la génération de l'ensemble des images correspondantes.

**Espace de visualisation :** l'étape *espace de visualisation* est définie par une transformation de visualisation qui traduit une représentation analytique en valeurs directement visualisables. Dans l'exemple de la Figure 15b, la transformation de visualisation consiste à ordonner l'ensemble des pages web obtenues à partir de l'étape *point de vue sur la donnée* en une liste linéaire de pages web. De plus, cette transformation fixe les caractéristiques de l'espace de représentation comme le nombre de dimensions de l'espace graphique utilisé. Le nombre de dimensions de la donnée à visualiser n'est pas toujours équivalent au nombre de dimensions de son espace de représentation. Une donnée hiérarchique ou temporelle peut être représentée dans un espace en deux dimensions (2D) ou encore dans un espace en trois dimensions (3D). Les caractéristiques de l'espace de représentation dépendent des caractéristiques de la dimension structurelle, de la dimension temporelle ainsi que des tâches utilisateur à considérer. Les tâches seront étudiées dans le chapitre suivant.

**Point de vue sur l'espace de visualisation :** le *point de vue sur l'espace de visualisation* définit le rendu perceptible par l'utilisateur. Ce rendu est le résultat d'une transformation qui traduit les valeurs directement visualisables en une représentation graphique (vue). Dans l'exemple de la Figure 15b, la transformation visuelle crée la forme du livre à partir de la liste linéaire des pages web. Ainsi, l'étape *espace de visualisation* fixe les caractéristiques de l'espace de représentation, et par rapport à ces caractéristiques l'étape *point de vue sur l'espace de visualisation* traduit les valeurs des données en signes graphiques. Aussi, à ce niveau une fonction de déformation peut être utilisée pour gérer un grand espace de données [Vernier 01].

En plus de la décomposition du processus de visualisation en quatre niveaux de données et trois types de transformation, Chi propose de caractériser les systèmes de visualisation par des

opérateurs spécifiques à chaque niveau de donnée et chaque type de transformation. Nous situons les opérateurs définis par Chi par rapport aux quatre étapes de la Figure 15c. Les opérateurs spécifiques à l'étape *donnée* concernent par exemple le filtrage de données, l'ajout d'un nouvel ensemble de données et la transformation de Fourier pour les images. Les opérateurs spécifiques à l'étape *point de vue sur l'espace de visualisation* concernent par exemple la rotation, la translation, l'agrandissement, et le positionnement et l'orientation de la caméra pour une scène 3D. D'autres opérateurs sont définis entre les deux extrémités *donnée* et *point de vue sur l'espace de visualisation*.

Ayant présenté le processus de visualisation de Chi, nous appliquons ce modèle à la dimension structurelle puis à la dimension temporelle d'une donnée.

### **3 Processus de visualisation de la dimension structurelle**

Nous l'avons souligné au paragraphe 2, les valeurs structurelles sont définies par un type qui les caractérise. Elles peuvent être quantitatives, qualitatives, des objets 2D ou autres. Ainsi, l'étude des deux étapes *donnée* et *point de vue sur la donnée* du processus de visualisation de la dimension structurelle revient à l'étude de ces deux étapes dans le cas du processus de visualisation de données en général. Nous ne reprenons donc pas ces deux étapes du processus de visualisation de la dimension structurelle. Nous excluons aussi l'étude de l'étape *espace de visualisation* pour deux raisons : 1) parce que la transformation de visualisation de l'étape *espace de visualisation* est définie par rapport aux résultats de l'étape *point de vue sur la donnée* et 2) parce que les caractéristiques de l'espace de représentation (nombre de dimensions de l'espace graphique par exemple) sont la plupart du temps définies par rapport aux deux dimensions structurelle et temporelle de la donnée à visualiser. Nous étudions l'étape *espace de visualisation* de la dimension temporelle dans le paragraphe 4.1. Comme le montre la Figure 16, le processus de visualisation de la dimension structurelle est général.

Dans ce paragraphe, nous nous limitons à l'étude du rendu perceptible par l'utilisateur, c'est-à-dire l'étape *point de vue sur l'espace de visualisation de la dimension structurelle*. Nous considérons cette étape dans le contexte de la visualisation de données temporelles. Nous écartons donc de cette étude les travaux sur la visualisation de données non-temporelles comme ceux sur la visualisation de documents XML [Jacquemin 02a, Jacquemin 02a].

La transformation visuelle de l'étape *point de vue sur l'espace de visualisation* (Figure 17) est définie par une transformation de l'espace de représentation par des fonctions de déformation par exemple, et d'une transformation des valeurs des données en signes graphiques. Bruley [Bruley 99] distingue deux types de transformation des valeurs des données selon la granularité de l'élément utilisé. Il distingue des techniques à base de pixels et des techniques à base d'éléments graphiques. La transformation des valeurs des données en éléments graphiques ou en pixels ne couvre évidemment pas l'étape *point de vue sur l'espace de visualisation de la dimension structurelle*. D'autres critères doivent être pris en compte, par exemple les tâches concrètes d'interaction (zoom, sélection des éléments de l'espace de représentation, etc.).

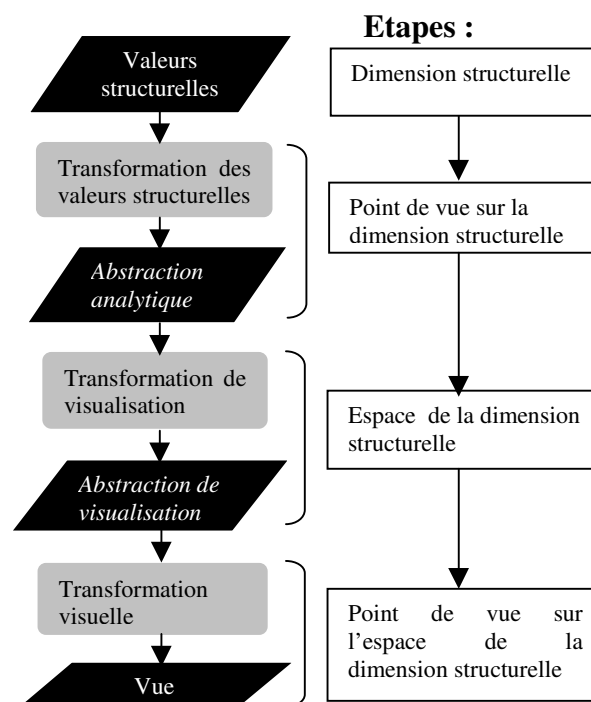


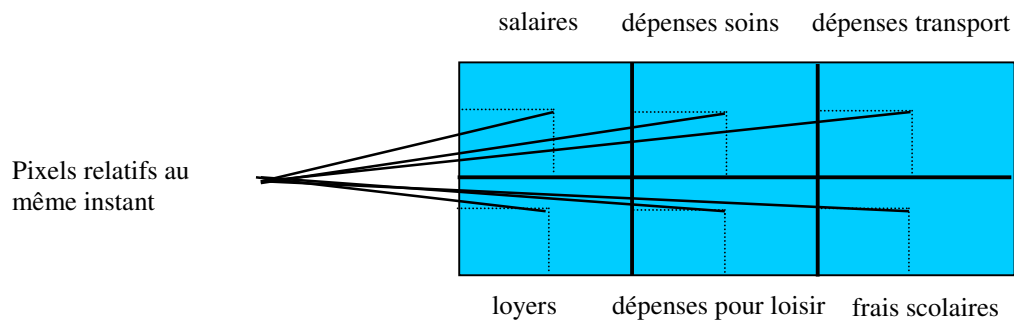
Figure 17 : Processus de visualisation de la dimension structurelle.

Dans la suite, nous présentons chacune des deux classes (pixel et élément graphique) de techniques de représentation de la dimension structurelle.

### 3.1 Techniques orientées pixels

Chaque entité de l'espace de données est représentée par un pixel dont la couleur traduit la valeur d'un attribut. L'utilisation de pixels permet la visualisation d'un très grand espace de données. En effet, un écran de résolution 1024x768 permet de visualiser 786432 valeurs.

Comme le montre la Figure 18, la visualisation de plusieurs données est traduite en terme de plusieurs fenêtres tel que chacune soit relative à une donnée particulière. Par exemple, si la visualisation concerne l'historique du salaire d'un employé, l'historique de ses dépenses de soins et l'historique de ses dépenses de transport, une fenêtre est associée à chaque donnée, et les valeurs des pixels aux positions (x,y) sont toutes relatives au même instant temporel.



*Figure 18* : Représentations à base de pixels de plusieurs données : une fenêtre est associée à chaque donnée. Sur l'exemple, six données sont représentées par six fenêtres.

Daniel Keim [Keim 96a] distingue deux classes de techniques à base de pixels selon que le mode d'arrangement est dépendant ou indépendant de la requête formulée. La classe des *techniques de visualisation indépendantes d'une requête* est dédiée à des données ordonnées selon un critère. En effet, ces techniques sont dites indépendantes d'une requête car elles utilisent un critère d'ordre intrinsèque aux données. Ces techniques sont adaptées pour visualiser des données temporelles, le temps étant le critère d'ordre. Le critère d'ordre est utilisé pour placer les pixels au niveau de l'espace de représentation. La façon la plus simple est de placer les pixels selon un parcours horizontal (Figure 20a) ou vertical (Figure 20b). Ceci pose le problème de la représentation d'une distribution unidimensionnelle, le temps par exemple, dans un espace graphique en deux dimensions. Comme le montre la Figure 19, la distance qui sépare les deux points C et D (respectivement C et F) dans la représentation unidimensionnelle (à gauche dans la Figure 19) est beaucoup moins grande (respectivement beaucoup plus grande) que la distance qui les sépare dans l'espace en deux dimensions (à droite dans la Figure 19).



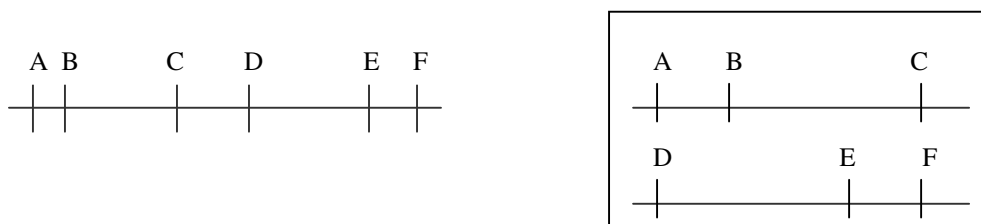


Figure 19 : Illustration du problème de représentation d'une distribution unidimensionnelle dans un espace graphique en deux dimensions.

Afin de résoudre ce problème, des algorithmes de remplissage d'espace, comme ceux proposés par Peano et Hilbert (Figure 20c) [Moon 96] et Morton (Figure 20d) [Keim 96a] sont souvent utilisés. Ils garantissent que les éléments qui sont proches dans une distribution unidimensionnelle, sont la plupart du temps proches dans un espace de représentation visuel à deux dimensions [Keim 96a, Keim 96b].

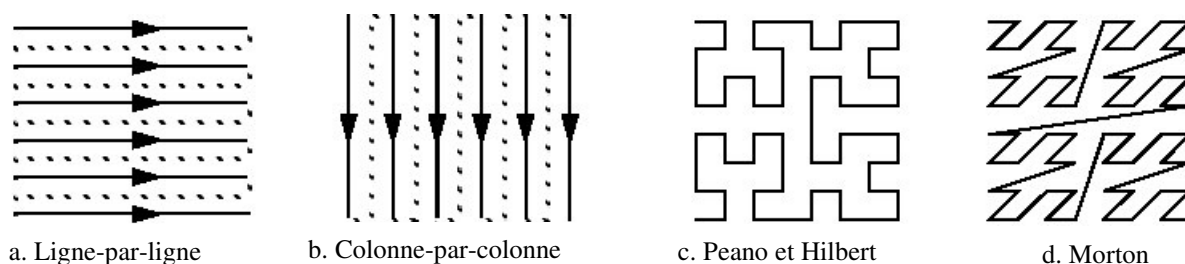


Figure 20 : Modes de placement des pixels [Keim 96a].

La Figure 21 présente un exemple de visualisation du taux de change du dollar Américain du mois Janvier 1987 au mois de Mars 1993 [Keim 96a]. A gauche dans la Figure 21a, la visualisation est générée par l'algorithme de Peano et Hilbert, tandis qu'à droite, Figure 21b, elle est générée par l'algorithme de Morton. Pour interpréter correctement les données visualisées aux Figure 21a et 21b, l'utilisateur doit donc connaître l'algorithme de placement.

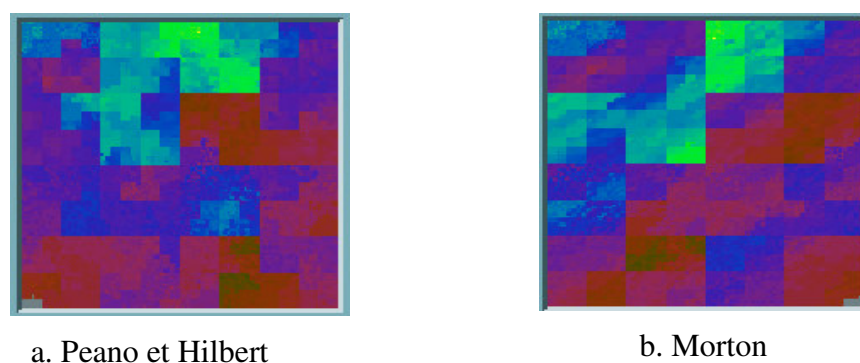
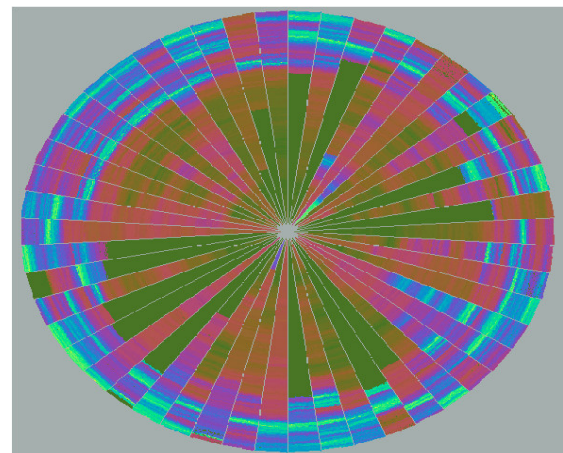
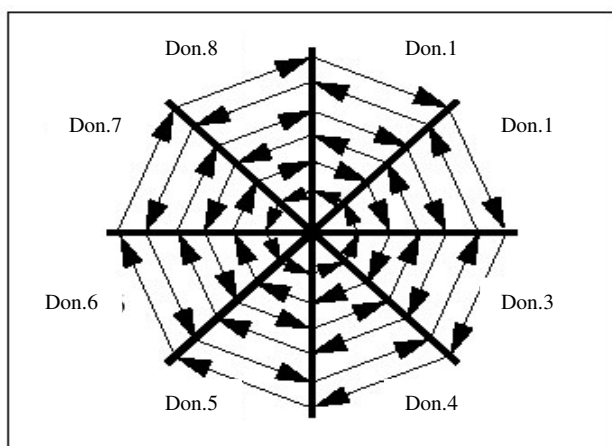


Figure 21 : Visualisation du taux de change du dollar Américain [Keim 96a].

La classe des *techniques de visualisation indépendantes de la requête* visualise les valeurs d'une donnée en les traduisant directement en couleurs de pixels. A l'inverse, les *techniques de visualisation dépendantes de la requête* visualisent les données en fonction de leur degré de pertinence par rapport à une requête utilisateur. Par exemple, pour un ensemble de données initial D, et un sous-ensemble E représentant le résultat d'une requête utilisateur appliquée sur D, les techniques de cette classe visualisent les distances des valeurs de D par rapport à la requête au lieu de visualiser directement les valeurs de E. L'utilisateur peut donc situer l'ensemble E (le sous-ensemble de D qui est le plus pertinent par rapport à la requête) dans son contexte (l'ensemble de données initial). L'objectif est de donner à l'utilisateur un retour d'information et lui permettre de diriger sa recherche dans l'ensemble de données initial. Dans ce cas, plusieurs modes d'arrangement des données sont possibles. Il s'agit par exemple de visualiser les données les plus pertinentes au centre. La technique Cercle-Segments (Circle-Segments Technique) [Ankerst 96] est un exemple de technique qui utilise cette approche. L'idée de base est de représenter N données (N = 8 données dans la Figure 22 à gauche) en même temps sous la forme d'un cercle partagée en N segments. Chaque segment est dédié à une donnée. A l'intérieur de chaque segment, les valeurs des distances de pertinence des données par rapport à la requête sont représentées à partir de l'origine en suivant la direction des flèches comme le montre la Figure 22a. Un exemple concret de visualisation des valeurs de 50 actions du mois de Janvier 1974 au mois d'Avril 1995 est représenté à la Figure 22b. Chaque segment correspond aux valeurs dans le temps d'un stock particulier.



a. Principe de la technique Cercle-Segments.

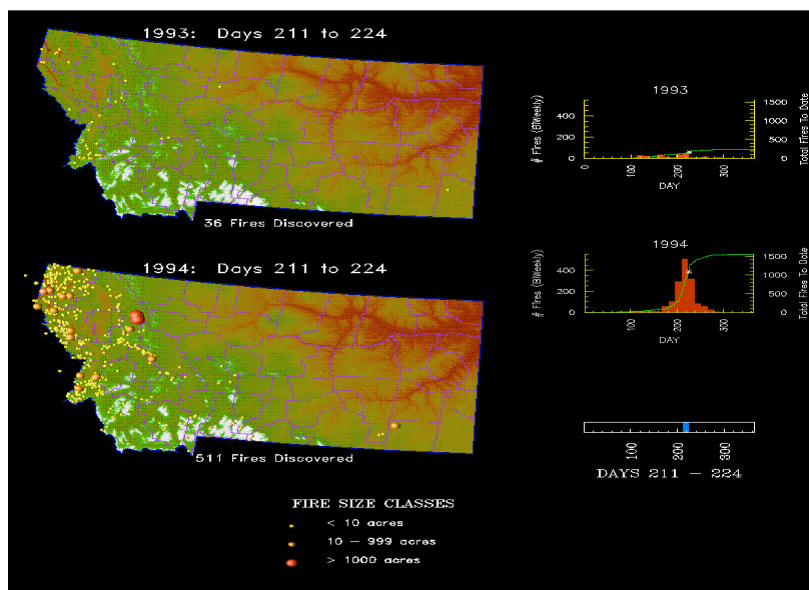
b. Visualisation de 50 données temporelles.

Figure 22 : Technique Cercle-Segments [Ankerst 96].

### 3.2 Techniques à base d'éléments graphiques

Avec les techniques à base d'éléments graphiques, chaque entité est représentée par un objet graphique dont on fait varier une ou plusieurs variables rétinienne (forme, taille, orientation, etc.) [Bertin 67, Mackinlay 86]. Nous distinguons deux classes de techniques selon que les éléments graphiques utilisés sont définis par rapport à un domaine d'application ou non.

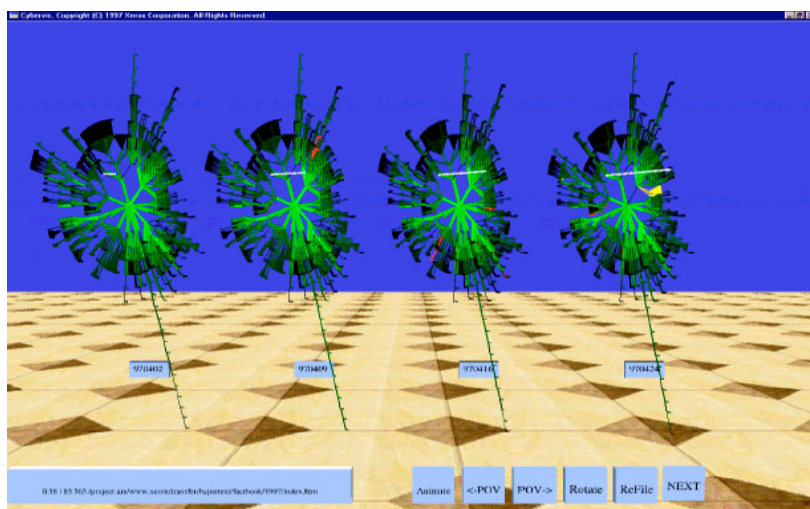
La technique de la Figure 23 est un premier exemple de technique qui utilise des éléments graphiques indépendamment d'un domaine d'application. Les données représentant les superficies de zones touchées par des incendies entre le 211<sup>e</sup> et 224<sup>e</sup> jour en 1993 et en 1994, sont représentées par des sphères sur le fond de deux cartes géographiques à gauche dans la Figure 23, et en histogrammes à droite dans la Figure 23. Cette technique peut ainsi être réutilisée pour visualiser des données quantitatives et spatialement référencées comme les productions agricoles ou les exportations de chaque ville d'un pays donné.



**Data Explorer (DX)** [Explorer 03], permet de visualiser des données de différents types. La Figure 23, présente une utilisation géographique de DX pour visualiser des données à partir d'une base de données oracle. Il s'agit de visualiser l'historique des incendies dans le temps.

Figure 23 : Data Explorer (DX) [Explorer 03].

La technique Time Tube [Chi 98] est un deuxième exemple de cette classe de techniques. La Figure 24, présente une visualisation de documents pendant le mois d'Avril 1997. La technique Time Tube se base sur l'utilisation de la technique de visualisation "arbre en disque" (Disk Tree Visualization). Elle utilise quatre représentations en disque en arbre, chacune correspond à une semaine du mois d'Avril.



**Time Tube** [Chi 98], est une technique de visualisation basée sur l'utilisation d'une représentation "arbre en disque" (disk tree) pour la visualisation des valeurs des données. La troisième dimension de l'espace est utilisée pour représenter le temps.

Figure 24 : La technique Time Tube [Chi 98].

Dans la deuxième classe de techniques, les formes des éléments graphiques sont définies par rapport au domaine d'application considéré. Ainsi, pour chaque type de données, des signes graphiques ont été créés. Le vent, par exemple, est souvent représenté, dans les applications géographiques, par des flèches dont la direction indique la direction du vent et la longueur indique la valeur enregistrée. La Figure 25 montre un exemple d'une telle visualisation : il s'agit de l'intensité du vent au 15 Octobre de l'an 2000.

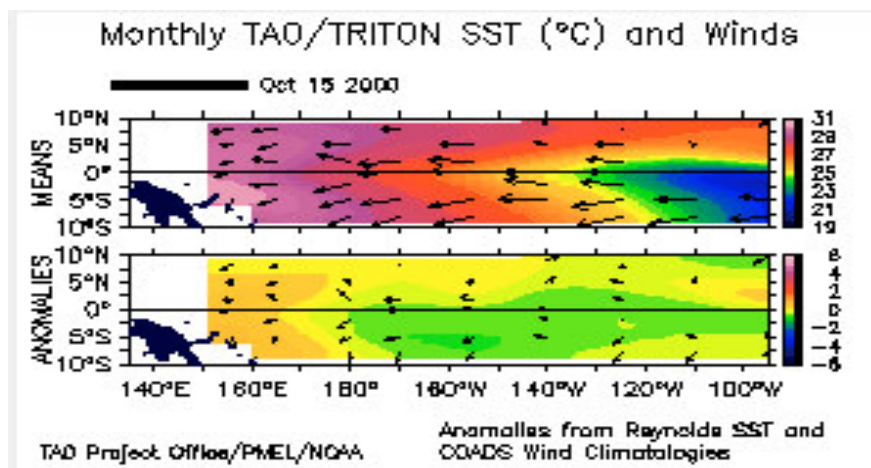


Figure 25 : Visualisation de l'intensité du vent [TAO 03].

Nous visons à concevoir des techniques de visualisation indépendantes d'un domaine d'application. Bien que notre domaine d'application privilégié est la géographie, notre objectif est de concevoir des techniques génériques réutilisables. Aussi, le choix des objets graphiques pour traduire les valeurs des données ne dépendra pas du domaine applicatif mais des caractéristiques des données (quantitatives, qualitatives, ordonnées, etc.) à visualiser.

### 3.3 Synthèse

Dans ce paragraphe, nous avons distingué deux classes de techniques de visualisation de données temporelles selon la granularité de l'élément utilisé pour traduire les valeurs des données : élément graphique ou pixel. Il convient de noter que dans le cas de la dimension structurelle d'une donnée temporelle, le processus de visualisation est très général. Nous nous sommes limité à l'étude de la transformation visuelle de l'étape *point de vue sur l'espace de visualisation* de ce processus de visualisation. Au contraire, nous étudions dans la suite en détail le processus de visualisation de la dimension temporelle.

## 4 Processus de visualisation de la dimension temporelle

Comme nous l'avons précisé dans l'introduction de ce chapitre, nous considérons dans cette partie la représentation de la dimension temporelle d'une donnée comme un espace à part entière et indépendant. Appliquant le processus de visualisation de Chi, nous obtenons alors le processus de la Figure 26.

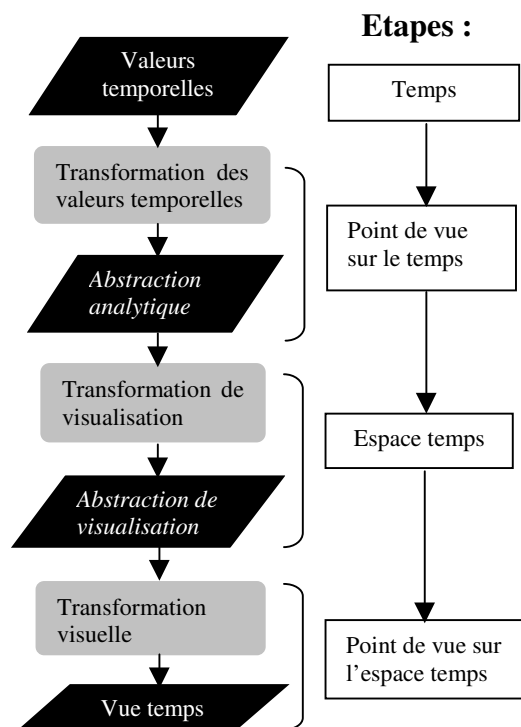


Figure 26 : Processus de visualisation de la dimension temporelle.

L'étape *temps* considère les valeurs temporelles initiales à visualiser. Dans le chapitre précédent, nous avons présenté une modélisation informatique des données temporelles. Les valeurs temporelles sont définies par des caractéristiques comme l'unité d'observation, le

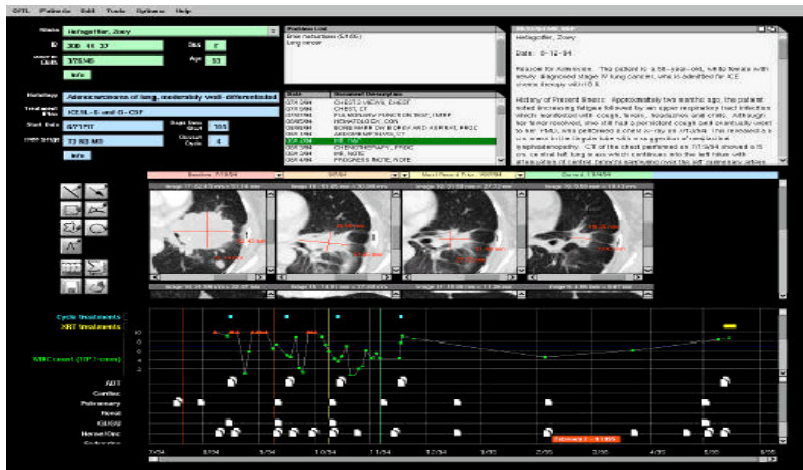
calendrier utilisé ou encore le domaine temporel. A partir des valeurs temporelles, la transformation des valeurs temporelles de l'étape *point de vue sur le temps* génère, généralement par extraction, une abstraction analytique ou méta-données (informations concernant les valeurs temporelles). En considérant le temps comme espace de données, cette transformation génère une représentation de la structure globale de l'espace temps étudiée au Chapitre 2, c'est-à-dire une forme linéaire, cyclique ou logarithmique du temps. Une représentation hiérarchique du temps est aussi utilisée pour concrétiser la modélisation multigranulaire (composition des unités d'observation) du temps étudiée au Chapitre 2. En résumé, l'étape *point de vue sur le temps* consiste à analyser la structure du domaine temporel et les caractéristiques de l'unité d'observation pour décider si le temps peut être représenté sous une forme cyclique, linéaire ou autre.

Les deux étapes *temps* et *point de vue sur le temps* concernent le traitement des données à visualiser et non pas le rendu perceptible par l'utilisateur. A l'opposé, les deux étapes suivantes *espace temps* et *point de vue sur l'espace temps* déterminent le rendu perceptible par l'utilisateur. Nous les étudions successivement dans les paragraphes suivants.

## **4.1 Espace temps**

L'étape *espace temps* est définie par une transformation de visualisation et le résultat de cette transformation qui est une abstraction de visualisation (Figure 26). La transformation de visualisation traduit une représentation analytique en valeurs directement visualisables (l'abstraction de visualisation). Par rapport aux valeurs temporelles, la transformation de visualisation traduit par exemple une suite de valeurs définies par rapport à l'unité d'observation Jour et ordonnées selon une relation d'ordre (la représentation analytique de l'étape *point de vue sur le temps*) en un tableau à deux dimensions (abstraction de visualisation de l'étape *espace temps*). Les valeurs de ce tableau sont définies par exemple par l'unité d'observation Heure. Les deux dimensions du tableau correspondent aux unités d'observation Jour et Heure. De plus, lors de l'étape *espace temps* le nombre de dimensions de l'espace de représentation (l'espace géométrique) dans lequel les valeurs temporelles seront visualisées est fixé. Nous distinguons les espaces à deux dimensions de ceux à trois dimensions.

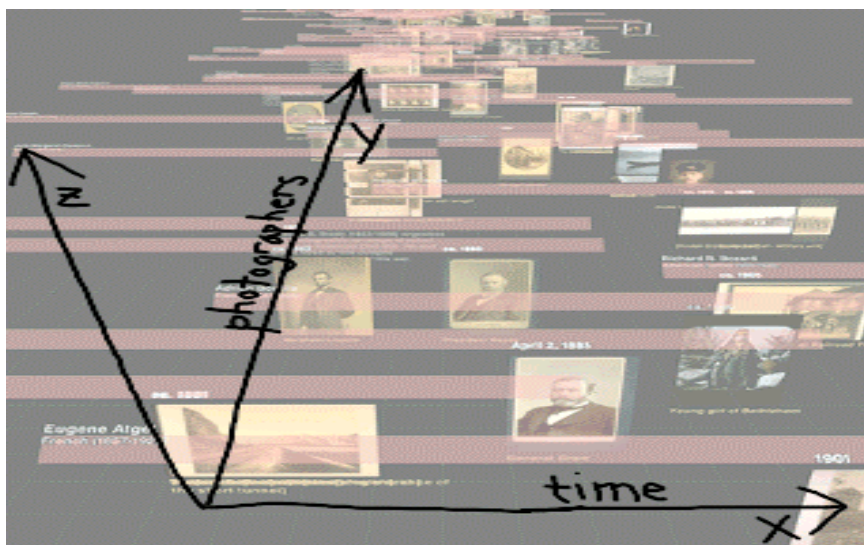




OITL (Oncology Imaging TimeLine) [Bui 98]. La Figure 3 représente l'interface du système, utilisée par les radiologistes et les médecins pour observer et étudier l'historique des images radio des patients.

Figure 27 : Système OITL [Bui 98].

Indépendamment du nombre de dimensions de l'espace de représentation final, le temps peut être défini au niveau de l'étape *espace temps* par 1, 2 ou N dimensions. Le système OITL [Bui 98] par exemple, représente le temps par une dimension (1D) dans un espace final en deux dimensions sous la forme d'un navigateur référencé par l'étiquette *ligne temps* dans la Figure 27. Le système OITL permet de visualiser l'historique médical d'un patient : les images radio, les différents rapports médicaux, etc. Par contre, la technique de la Figure 28, développée par Kullberg [Kullberg 95], représente le temps par une dimension dans un espace final en trois dimensions. Cette technique consacre l'axe des X au temps, et celui des Y aux photographies correspondants. Dans les deux exemples présentés aux Figure 27 et 28, la transformation de visualisation de l'étape *espace temps* produit une liste ordonnée (une seule dimension) des valeurs temporelles. Pour cette raison, le temps est représenté dans les deux exemples par une seule dimension de l'espace de représentation.



Dynamic Timelines [Kullberg 95], développé au Media Lab. du MIT est une technique 3D pour la visualisation d'historique de photographies.

Figure 28 : Visualisation 3D d'un historique de photographies [Kullberg 95].

Dans la Figure 29, l'espace temps est défini par deux dimensions, l'une pour représenter les heures et l'autre pour représenter les jours, dans un espace final en trois dimensions. La représentation du temps par deux dimensions est le résultat de la transformation de visualisation qui dans ce cas a produit un tableau en deux dimensions de valeurs temporelles, défini par deux dimensions Heure et Jour. Dans cet exemple, les données visualisées sont quantitatives. Elles sont traduites en objets graphiques telle que leurs hauteurs (la dimension verticale de l'espace 3D) et leurs colorations sont proportionnelles aux valeurs structurelles correspondantes.

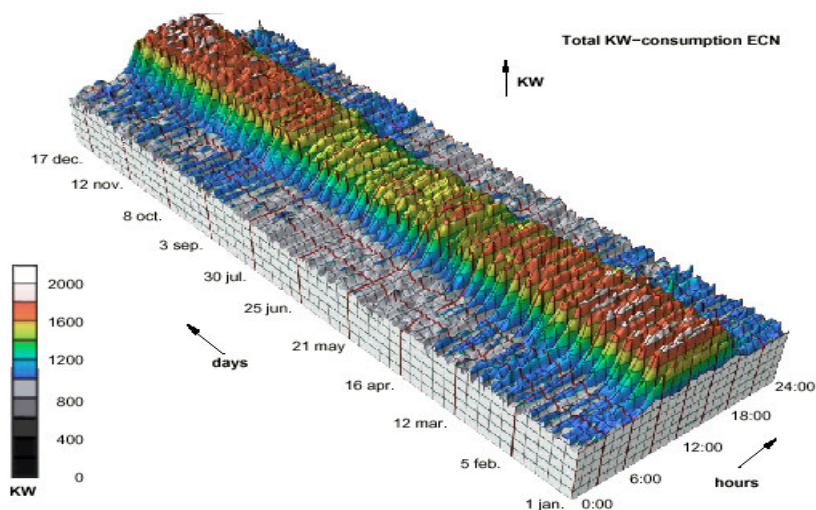


Figure 29 : Utilisation de deux axes pour représenter le temps [Wijk 99].

## 4.2 Point de vue sur l'espace temps

Le *point de vue sur l'espace temps* définit le rendu perceptible de la dimension temporelle au niveau d'une technique de visualisation. Nous étudions donc les formes de représentation du temps. Ces dernières correspondent aux perceptions linéaire, cyclique ou logarithmique et à la représentation multigranulaire du temps. Le choix d'une forme de représentation du temps plutôt que d'une autre est le résultat de l'étape *point de vue sur le temps*. En effet, comme nous l'avons présenté dans l'introduction du paragraphe 4, l'étape *point de vue sur le temps* identifie les caractéristiques des valeurs temporelles à visualiser. Une représentation graphique du temps sous une forme cyclique est utilisée lorsque l'étape *point de vue sur le temps* montre que les valeurs temporelles sont cycliques (périodiques). Nous étudions dans la suite chacune des formes de représentation du temps.



### 4.2.1 Formes cycliques du temps

Nous distinguons dans la littérature deux modes de représentation cyclique du temps : représentation en spirale et représentation circulaire.

La Figure 30 présente un exemple de représentation en spirale : la visualisation concerne l'intensité solaire. La représentation en spirale (Figure 30b) fait apparaître des cycles et facilite l'observation des événements. Par exemple, il est plus facile de voir les périodes de nuages dans la représentation en spirale de la Figure 30b que dans la représentation linéaire de la Figure 30a. Les périodes de nuages correspondent aux zones colorés en noir au niveau des arcs de la spirale représentant les jours (étiquette Jour dans la Figure 30b). Nous revenons sur la représentation linéaire du temps au paragraphe 4.2.2.

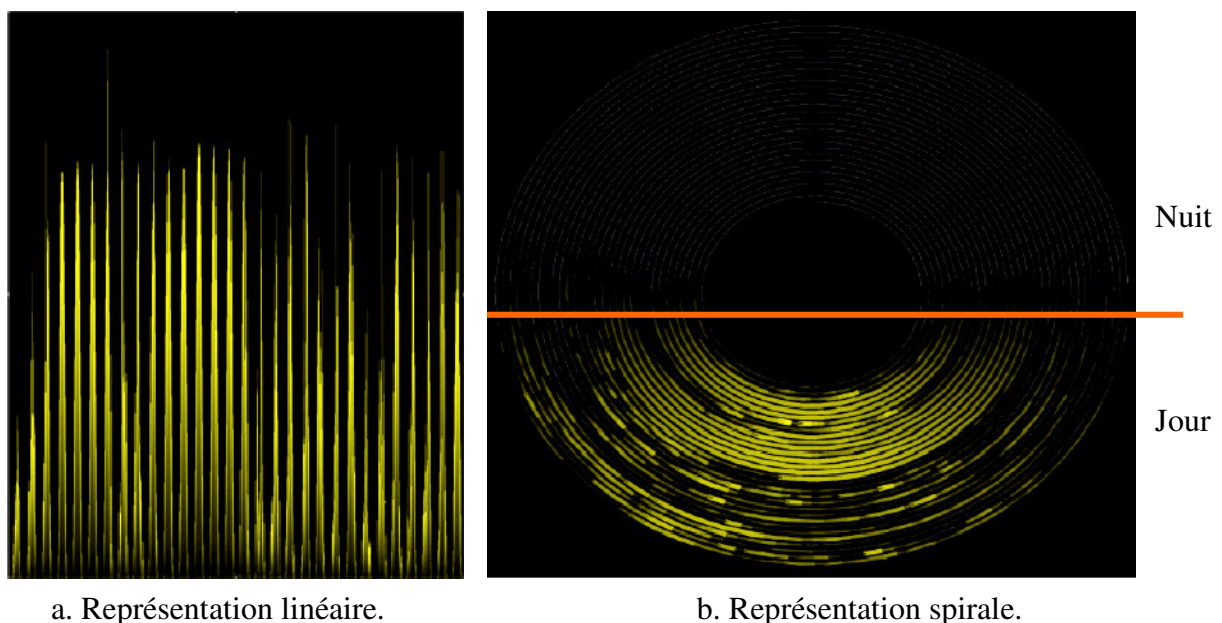


Figure 30 : Visualisation de l'intensité solaire dans le temps [Weber 01].

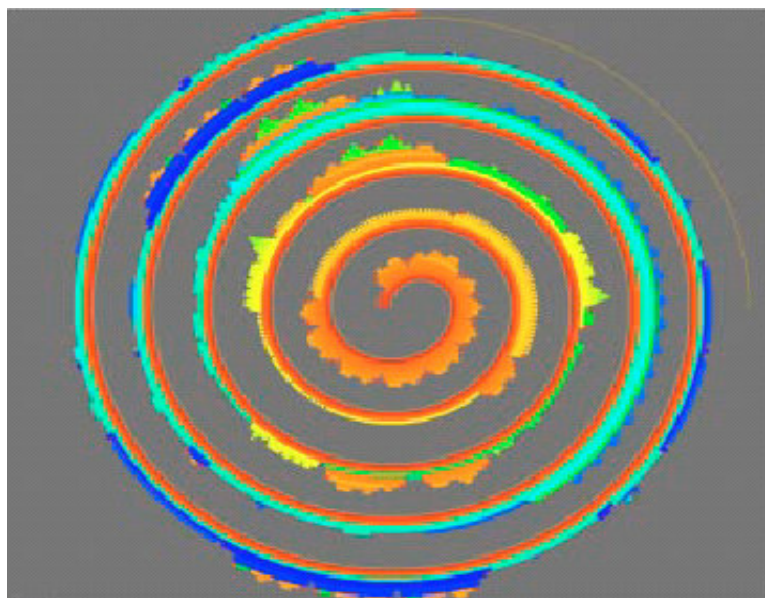
Une spirale peut être définie par une fonction  $r = f(\varphi)$ . La principale caractéristique d'une spirale est que la fonction  $f$  est une fonction monotone ( $f$  strictement croissante). Une spirale est définie par la fonction suivante :

$$r = f(\varphi), \quad df / d\varphi > 0, \quad \varphi \in \mathbb{R}^+.$$

D'autres fonctions conduisent à la définition de spirales de types différents. Nous distinguons principalement deux modes de représentation d'une spirale pour la visualisation de données temporelles : la représentation logarithmique et la représentation d'Archimède. Le principe

général d'une représentation en spirale est de représenter un axe d'une façon continue, qui commence à partir d'un point origine, autour duquel il tourne en spirale et progresse. Cet axe peut être vu comme une succession de formes "circulaires" que nous appelons *partitions circulaires* ou *périodes*. Les données sont par la suite représentées sur les partitions circulaires.

La différence de la représentation logarithmique d'une spirale par rapport à la représentation d'Archimède est le rôle même des périodes. Dans la représentation logarithmique, l'axe sert de support aux données sans distinguer les périodes. Ainsi, le nombre d'éléments représentés varie d'une partition circulaire à une autre. L'inconvénient majeur de cette représentation est que des éléments non reliés sémantiquement seront très proches les uns des autres dans l'espace de représentation (l'axe de la spirale). La Figure 31 présente une visualisation des traces d'un programme dans le temps avec une représentation logarithmique d'une spirale [Renieris 99]. Il est difficile avec ce type de représentation d'observer les cycles. Le temps est considéré comme linéaire plus que cyclique. La forme logarithmique d'une spirale est utilisée dans cet exemple pour mieux gérer l'espace de représentation dans le cas de la visualisation d'une grande quantité de données temporelles.



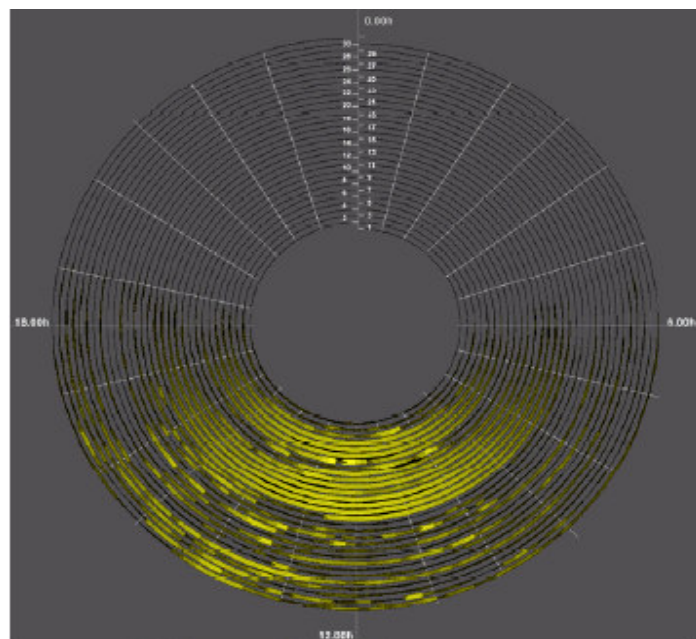
*Figure 31* : Représentation logarithmique d'une spirale [Renieris 99].

La représentation d'Archimède est utilisée en visualisation de données temporelles pour renforcer la perception des cycles dans le temps. Elle est définie par l'équation suivante [Weber 01] :

$$r = \alpha\varphi ,$$

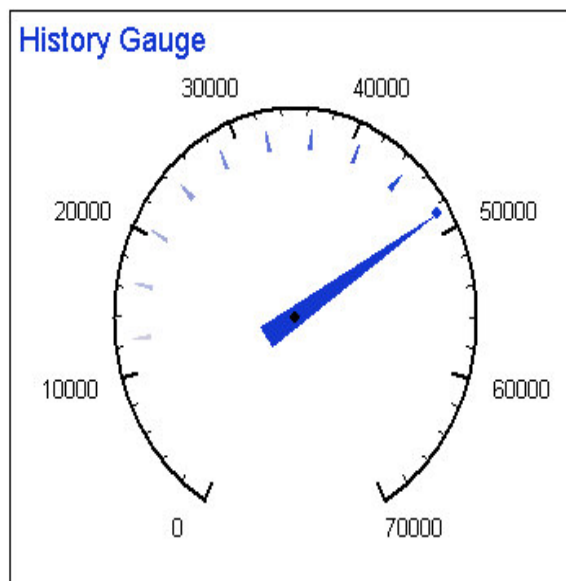
tel que  $x = r \cos \varphi$  et  $y = r \sin \varphi$   
 $r = \text{sqrt}(x^2 + y^2)$  et  $\varphi = \tan^{-1}(y/x)$

Comme le montre la Figure 32, la spirale d'Archimède possède la propriété suivante : chaque rayon commençant de l'origine croise deux arcs consécutifs de la spirale selon une distance constante égale à  $2\pi\alpha$ . Les cycles sont ainsi visibles.



*Figure 32* : Représentation de la spirale d'Archimède [Weber 01].

Bien que la représentation en spirale soit la plus courante, la perception cyclique du temps peut aussi être traduite en terme de représentation circulaire. La technique Gauge de la Figure 33 est un exemple de technique qui visualise l'espace temps sous une forme circulaire. La Figure 33 présente l'historique des états d'un processus en exécution. Dans cet exemple les valeurs de l'état évoluent d'une manière uniforme.



La technique Gauge [Matkovié 02] a été développée pour visualiser les états de processus.

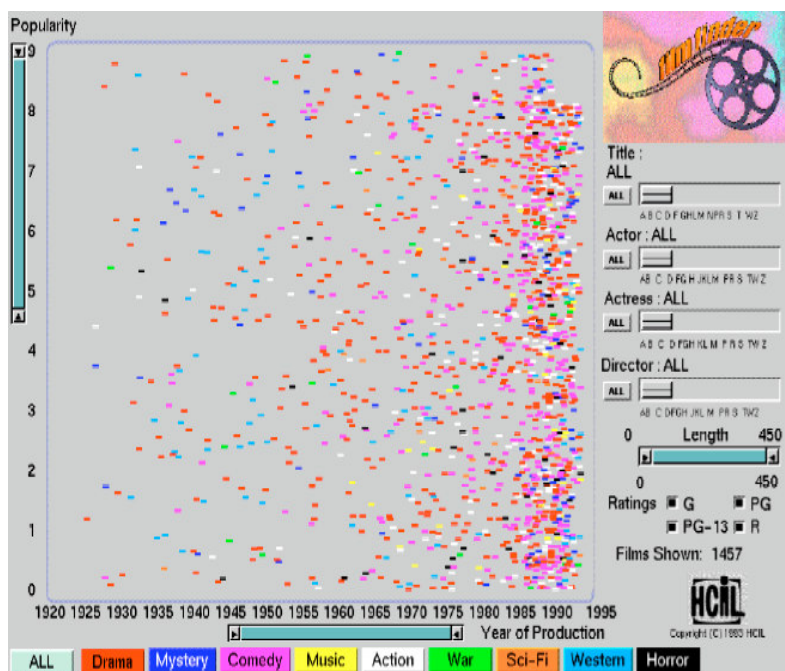
Figure 33 : Technique Gauge [Matkovié 02].

## 4.2.2 Formes linéaires du temps

Nous distinguons deux modes de représentation de la linéarité du temps : explicite ou implicite en fonction que le passage de la représentation des données à un instant  $t$  à leur représentation à l'instant  $(t+1)$  est réalisé suite à une intervention explicite de l'utilisateur ou contrôlé par le système. Nous présentons dans la suite ces deux modes de représentation de la linéarité du temps.

### 4.2.2.1 Représentation explicite

La représentation explicite de la linéarité du temps est définie par l'utilisation d'une ligne temps. La ligne permet de positionner les événements dans le temps et d'observer leurs relations. Le système Starfield [Jog 95, Cailleteau 99] de la Figure 34, développé pour visualiser des informations relatives à des films (date d'apparition, popularité, liste d'acteurs et actrices, etc.), utilise une représentation explicite du temps (ligne temps). Les valeurs structurelles sont représentées dans un espace séparé de l'espace temps, et sont traduites en coloration de rectangles.



**Starfield** [Jog 95, Cailleteau 99] est dédié à la visualisation d'une base de données de films. Bien que ce système n'ait pas été conçu pour la visualisation de données temporelles, il utilise une représentation explicite du temps pour positionner les films par rapport à leurs dates d'apparition.

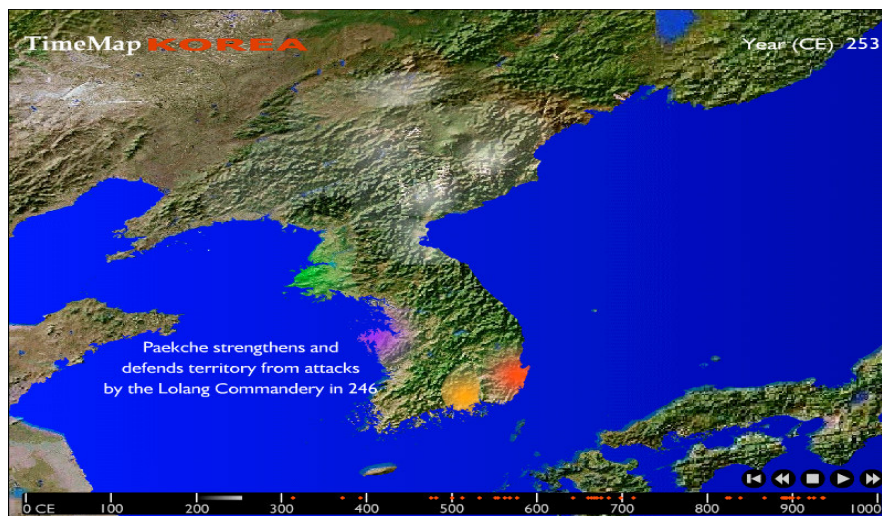
Figure 34 : Système Starfield [Jog 95, Cailleteau 99].

#### 4.2.2.2 Représentation implicite

La représentation implicite ou animée associe les valeurs temporelles d'une donnée au temps d'exécution. Une technique animée se caractérise par une transition qui fait passer la visualisation d'un état à un autre. Ce type de représentation est essentiellement utilisé pour visualiser des cartes animées. Une approche pour concevoir des techniques animées, dite animation par images (*frame-based animation*) Hansen [Hansen 01], consiste à créer des images individuelles et à les afficher rapidement ce qui crée une illusion de mouvement ou de changement. Une taxonomie détaillée des techniques d'animation est présentée dans [Thalman 94].

Le système TimeMap Viewer [Johnson 02] de la Figure 35 permet de générer des animations de données spatio-temporelles. Chaque image correspond à un état d'une représentation spatiale en deux dimensions. L'animation est utilisée pour représenter la dimension temporelle des données. La Figure 35 présente un état d'une animation générée par TimeMap Viewer pour une application archéologique. Cet état correspond à la visualisation des données pour l'année 253.

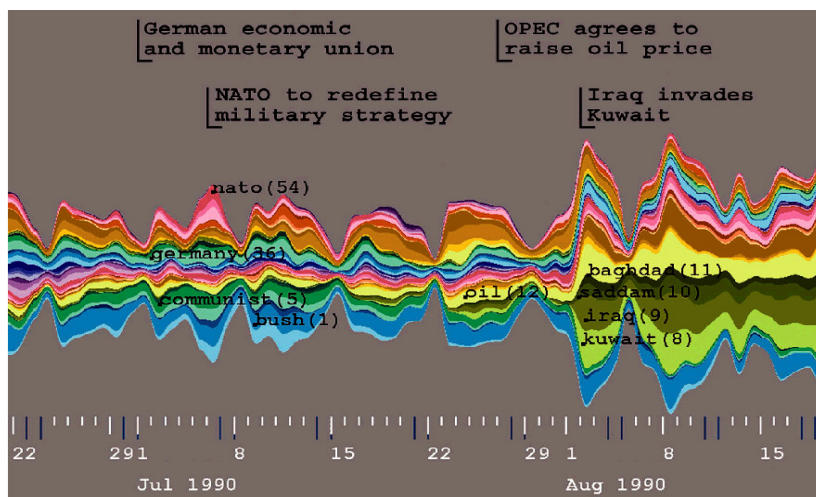




**TimeMap Viewer** [Johnson 02] est un système pour la visualisation de données spatio-temporelles. Ce système utilise l'animation pour représenter la dimension temporelle.

Figure 35 : Système TimeMap Viewer [Johnson 02].

Par opposition à l'animation utilisée dans le système TimeMap Viewer de la Figure 35, l'animation de la technique ThemeRiver [Havre 99] de la Figure 36 ne fait pas disparaître l'état (i-1) suite à l'apparition de l'état i. La technique ThemeRiver utilise la métaphore d'une rivière pour représenter une animation des valeurs temporelles : le temps passe comme le flux d'une rivière. L'animation dans la technique ThemeRiver est ainsi horizontale alors qu'elle est en profondeur pour le système TimeMap Viewer.



**ThemeRiver** [Havre 99, Havre 00] est une technique pour visualiser la variation de thèmes traités dans le temps dans une grande collection de documents. Elle permet de visualiser plusieurs données à la fois, le temps passe selon la métaphore de la rivière. La Figure 36 montre par exemple que le thème le plus étudié au mois d'août 1990 concerne la guerre de l'Irak contre le Kuwait.

Figure 36 : Technique ThemeRiver [Havre 99, Havre 00].

La représentation implicite du temps peut être utile pour visualiser des données capturées en temps réel. En effet, les tâches utilisateur (voir Chapitre 4) ne concernent pas la détection de moments temporels particuliers, mais l'observation de l'évolution des données dans le temps. La visualisation des performances d'un processeur de la Figure 37 est un exemple de ce type

de représentation. L'axe horizontal sert à représenter le temps tandis que l'axe vertical sert à représenter les valeurs de la donnée (occupation du processeur). Un autre exemple concerne la visualisation des secousses sismiques (séismogramme) ou du rythme cardiaque (cardiogramme). Notre étude ne portera pas sur la représentation implicite du temps et nous considérons que les données à visualiser sont disponibles avant l'utilisation de la technique de visualisation.

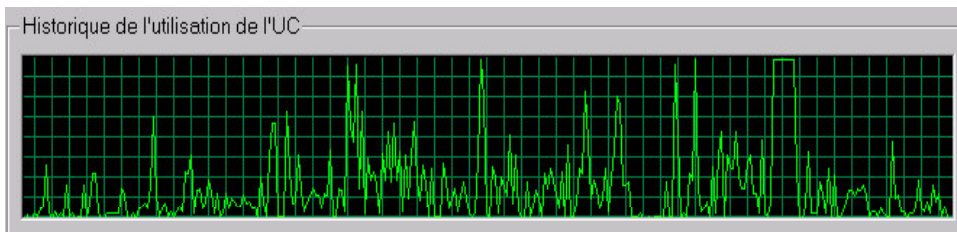
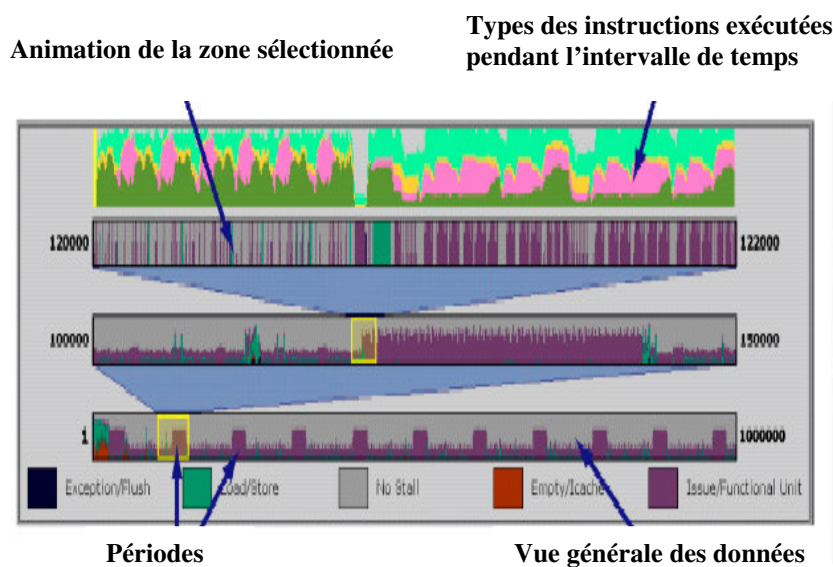


Figure 37 : Historique de l'utilisation de l'unité centrale.

Les représentations explicite et implicite de la linéarité du temps peuvent être complémentaires. La technique de la Figure 38 utilise le principe d'observabilité du focus dans son contexte. Le contexte est visualisé selon une représentation explicite en utilisant une ligne temps. Une première sélection d'une zone du contexte est représentée par une seconde ligne temps traduisant un agrandissement de la zone sélectionnée. De même, une troisième ligne est le résultat d'une sélection au niveau de la ligne temps précédente. Enfin, la technique utilise une animation (représentation implicite du temps) pour visualiser les valeurs sélectionnées au niveau de la troisième ligne temps.

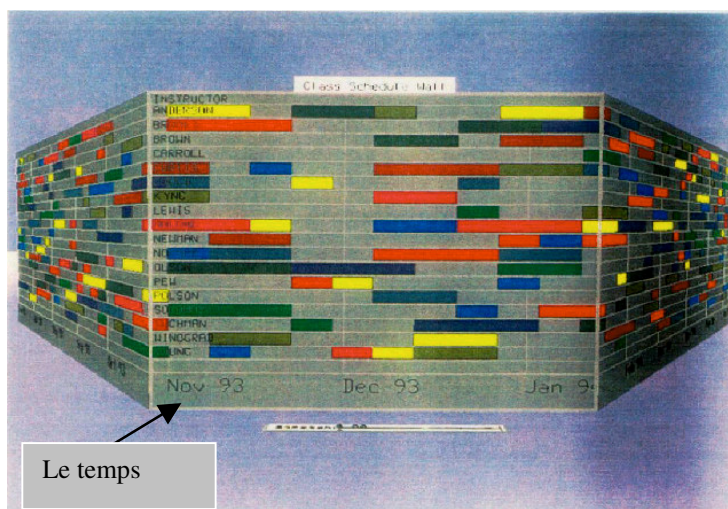


Cette technique [Stolte 99] permet de visualiser le comportement des applications exécutées sur des processeurs superscalaires (superscalar processors). Elle utilise à la fois les deux représentations statique et dynamique de la linéarité du temps.

Figure 38 : Complémentarité entre représentations explicite et implicite [Stolte 99].

### 4.2.3 Forme logarithmique du temps

La représentation logarithmique du temps n'est généralement pas concrétisée au niveau des techniques de visualisation de données temporelles proposées dans la littérature. La forme du Mur en Perspective [Mackinlay 91] de la technique présentée à la Figure 39 peut être considérée comme une représentation logarithmique du temps. Cette technique conçue pour gérer de grand espace de données, utilise une déformation de l'espace en un mur à trois faces. Une face avant pour visualiser les données d'une manière non déformée et détaillée. Deux faces en perspectives pour visualiser une grande quantité de données d'une manière déformée et moins détaillée. Par rapport à la représentation linéaire du temps, la face à gauche sert à représenter le passé, la face d'avant pour représenter le présent et celle de droite pour représenter le futur. Par contre, si nous considérons une représentation logarithmique du temps, la face avant représente un avancement normal (avec un pas fixe) dans le temps, alors que les deux dimensions de gauche et de droite représentent un avancement plus ou moins rapide dans le temps.



**Perspective wall** [Mackinlay 91] est une technique basée sur la déformation de l'espace de représentation sous une forme d'un mur en perspective. Elle permet en particulier de visualiser une très grande quantité de données. La Figure 39 montre une collection de fichiers visualisés par rapport à leurs types (axe vertical) et leurs dates de mise à jour (axe horizontal).

Figure 39 : Technique Mur en Perspective [Mackinlay 91].

### 4.2.4 Forme multigranulaire du temps

La vision multigranulaire s'appuie sur les relations existant entre les unités d'observation (Chapitre 2). La métaphore du calendrier est utilisée dans la Figure 40 [Wijk 99] pour représenter la structure multigranulaire du temps. Une année est composée de 12 mois, un mois est composé de 28, 29, 30 ou 31 jours, un jour est composé de 24 heures, etc.



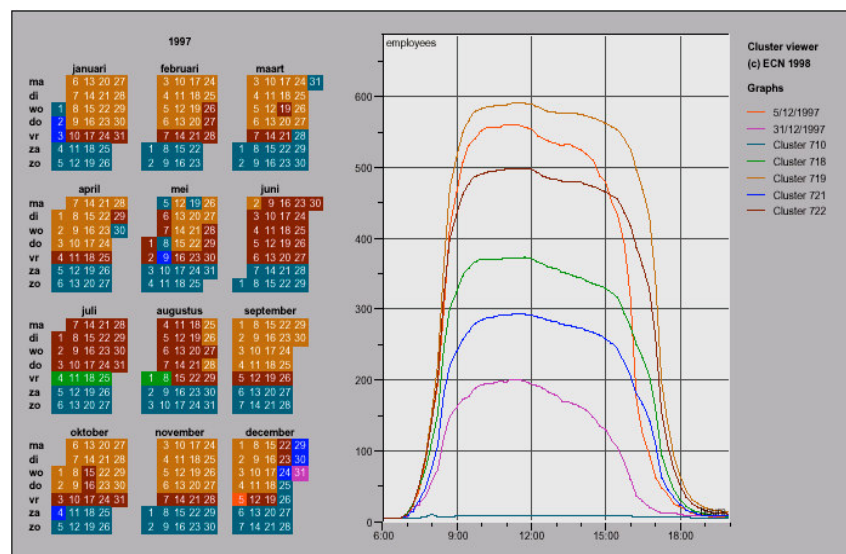
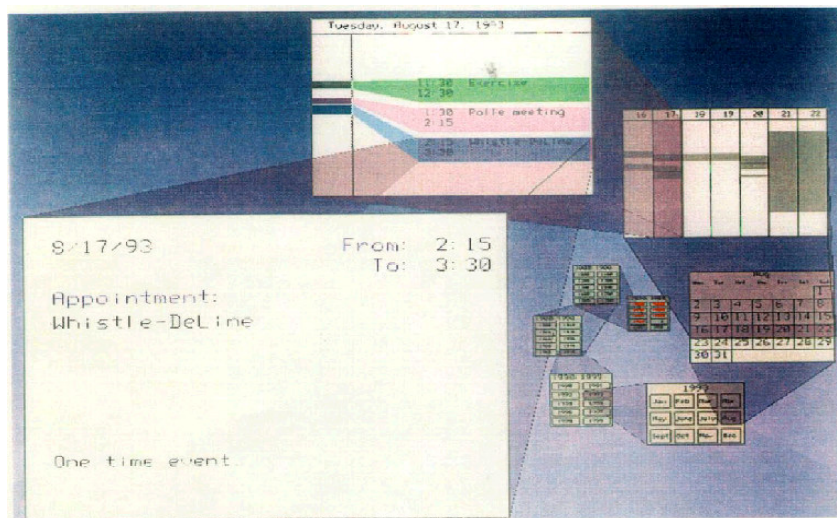


Figure 40 : Utilisation de la métaphore de calendrier [Wijk 99].

#### 4.2.5 Relations entre formes de représentation du temps

L'utilisation d'une forme de représentation du temps plutôt que d'une autre dépend des caractéristiques des valeurs temporelles et des tâches utilisateur. Certaines techniques de visualisation de données temporelles utilisent simultanément plusieurs formes de représentation du temps. La technique Spiral Calendar Visualizer [Mackinlay 94] de la Figure 41, intègre par exemple une représentation hiérarchique ainsi qu'une représentation cyclique du temps. Les valeurs de chaque unité d'observation y sont représentées selon une vue en calendrier (représentation multigranulaire du temps). Ainsi, les années sont représentées par une première vue, une année (1993 dans l'exemple) est projetée et les mois de cette année sont affichés selon une autre vue, un de ces mois (le mois d'Août) est projeté, etc. Les projections des valeurs temporelles sont représentées en spirale (représentation cyclique du temps). L'utilisation simultanée de ces deux formes de représentation renforce la perception de l'utilisateur de la structure multigranulaire des valeurs temporelles et fait éventuellement apparaître des cycles.



*Spiral Calendar Visualizer* [Mackinlay 94] permet de visualiser des données temporelles sous la forme de calendrier. Cette technique exploite la représentation multigranulaire du temps pour visualiser les valeurs des données par plusieurs vues, chacune relative à une unité d'observation. Les vues sont connectées entre-elles selon une forme spirale.

Figure 41 : Technique Spiral Calendar Visualizer [Mackinlay 94].

### 4.3 Synthèse

Dans ce paragraphe, nous avons présenté l'étape *espace de visualisation* et l'étape *point de vue sur l'espace de visualisation* du processus de visualisation de la dimension temporelle. A ce niveau, la transformation de visualisation de l'étape *espace temps* traduit une représentation analytique des valeurs temporelles en valeurs directement visualisables. De plus, elle fixe le nombre de dimensions de l'espace géométrique dans lequel est représentée la dimension temporelle.

L'état perceptible de la représentation de la dimension temporelle est le résultat de l'étape *point de vue sur l'espace temps*. Cette étape implémente une transformation visuelle qui traduit les valeurs temporelles en représentation visuelle. La représentation visuelle est définie par une forme de représentation du temps. Nous trouvons ainsi la forme linéaire du temps (avec une représentation explicite ou implicite), la forme cyclique (avec une représentation en spirale ou circulaire), la forme logarithmique et la forme multigranulaire. Nous retenons la forme de représentation du temps comme critère de classification des techniques de visualisation de données temporelles.

Après avoir détaillé indépendamment les deux processus de visualisation des dimensions structurelle et temporelle d'une donnée, il convient maintenant d'étudier le processus de visualisation des données temporelles dans son ensemble. Il s'agit ici d'identifier les points d'ancrages entre le processus de visualisation de la dimension temporelle et celui de la

dimension structurelle. Une technique peut visualiser une ou plusieurs données temporelles. Nous présentons tout d'abord une taxonomie des techniques de visualisation d'une seule donnée temporelle puis de plusieurs données temporelles.

## **5 Techniques de visualisation d'une donnée temporelle**

Nous considérons que l'espace de représentation d'une donnée temporelle peut être décomposé en deux espaces, un espace pour la dimension structurelle et un espace pour la dimension temporelle. Dans les deux paragraphes précédents, nous avons étudié séparément chaque dimension et détaillé son processus de visualisation. Dans ce paragraphe, nous étudions les techniques de visualisation de données temporelles au regard du processus de visualisation. Ce dernier étant composé des deux processus de visualisation étudiés précédemment, l'enjeu est ici d'identifier les points d'ancrage.

Le processus de visualisation de Chi [Chi 00] est défini pour une seule donnée. En considérant les deux dimensions d'une donnée temporelle, nous étendons ce modèle pour la visualisation de deux données. La Figure 42 présente les différentes configurations possibles entre le processus de visualisation de la dimension temporelle et celui de la dimension structurelle. Ces configurations permettent de distinguer et de classer les techniques de visualisation de données temporelles et constituent donc le critère sur lequel repose notre taxonomie. Ces cas identifiés varient entre deux extrémités : processus complètement fusionnés (Figure 42a) et processus complètement séparés (Figure 42g). Nous illustrons chaque cas par rapport à des techniques existantes sur la visualisation de données temporelles.

Pour définir les configurations, nous nous repons sur la sémantique des opérateurs définis par le processus de visualisation. Rappelons que ces opérateurs sont spécifiques à chacune des quatre étapes du processus de visualisation. Par exemple, les opérateurs spécifiques à l'étape *point de vue sur l'espace de visualisation* concernent la rotation, la translation ou l'agrandissement des objets graphiques. Les deux processus relatifs aux deux dimensions d'une donnée temporelle se rejoignent au niveau de l'étape dans laquelle intervient au moins un opérateur qui est commun aux deux dimensions à la fois. Nous notons cette étape, étape-charnière. Si par exemple, la translation des éléments de la représentation de la dimension structurelle affecte la représentation de la dimension temporelle, ceci implique selon notre définition que l'étape *point de vue sur l'espace de visualisation* est charnière aux deux

processus. Notons néanmoins qu’au niveau d’une étape-charnière certains opérateurs restent spécifiques à chacun des deux processus. Par exemple, il peut exister des opérateurs spécifiques à l’un des deux processus sans affecter l’autre. Par exemple, la rotation d’un objet graphique (représentation d’une valeur de la dimension structurelle) peut ne pas affecter la dimension temporelle. Notons aussi que les opérateurs qui interviennent au niveau des étapes précédant l’étape charnière sont spécifiques à chaque dimension. En considérant le cas de la Figure 42d par exemple, les opérateurs de la transformation de visualisation du processus de visualisation (étape *espace*) de la dimension temporelle ne doivent en aucun cas affecter le processus de visualisation de la dimension structurelle. Dès l’instant que les deux processus sont fusionnés, nous considérons que les étapes suivant l’étape charnière sont fusionnées.

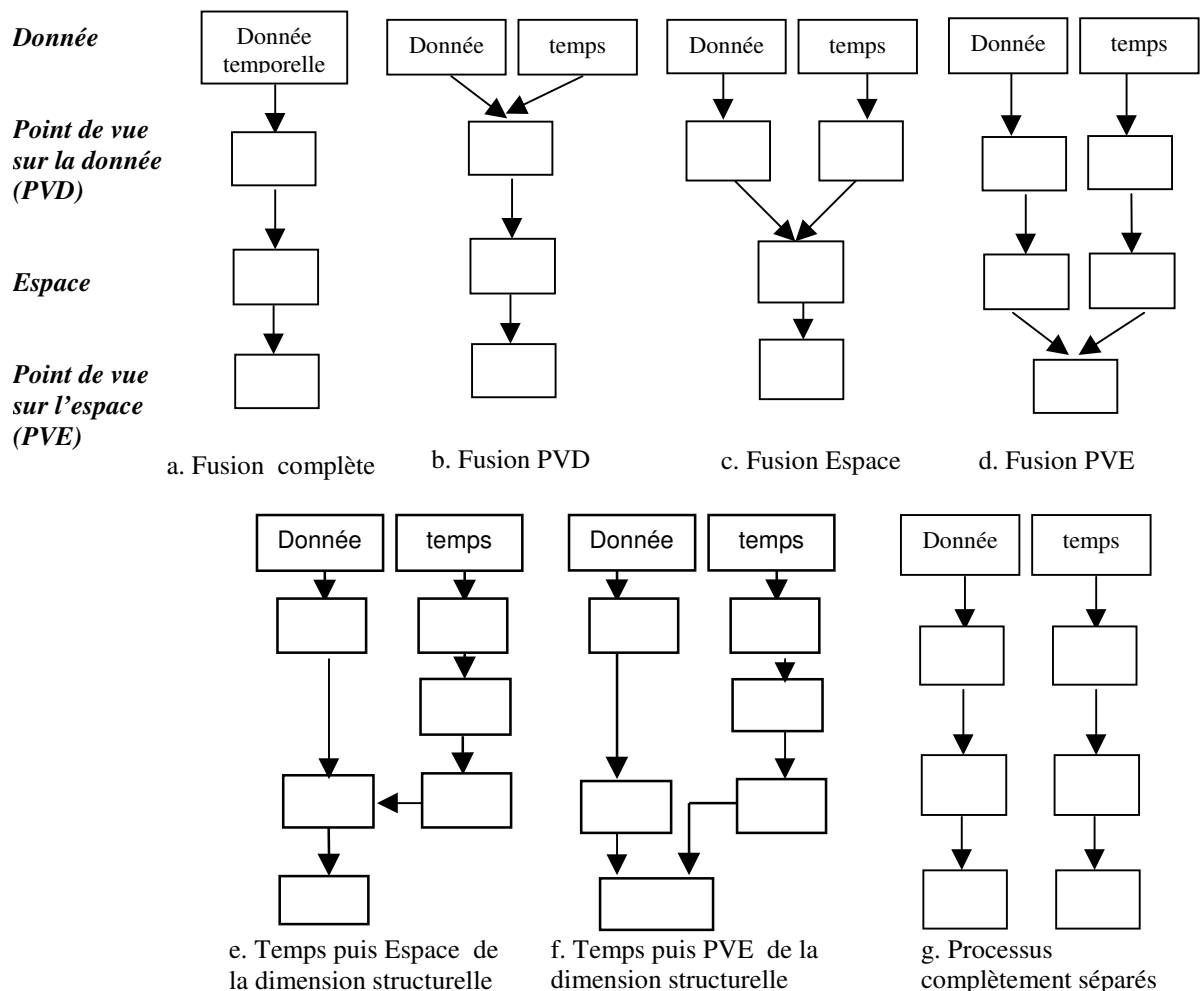


Figure 42 : Processus de visualisation d’une donnée temporelle.

Dans la suite, nous détaillons chaque modèle de processus de visualisation de la Figure 42 et l’illustrons par des techniques existantes.

## **5.1 Fusion complète ou fusion Point de vue sur la donnée**

Certaines techniques de visualisation de données temporelles résultent d'une fusion complète des deux processus, le processus de la dimension temporelle et celui de la dimension structurelle. Nous distinguons les deux cas de la Figure 42a ou 42b. Le modèle de la Figure 42a est souvent utilisé pour visualiser des données dont le domaine temporel est non-ancré (paragraphe 3.3 du Chapitre 2). La représentation du temps est dans ce cas implicite. Des exemples incluent les données multimédias, ou les données capturées et visualisées en temps réel. A l'opposé, le modèle de la Figure 42b est utilisé pour visualiser des données temporelles dont le domaine temporel est ancré. Le temps est alors explicite ce qui justifie que nous distinguons données et temps à la Figure 42b. Dans les deux cas (Figure 42a et Figure 42b), la représentation linéaire du temps, sous la forme d'un axe horizontal, est souvent considérée. Comme l'a souligné Tufte [Tufte 83, p. 186] nos yeux sont adaptés à détecter des déviations de l'horizon. Les séries temporelles (données temporelles) représentées horizontalement sont plus faciles à percevoir et interpréter<sup>7</sup>.

Une revue des techniques de visualisation existantes correspondant aux cas de la Figure 42a et 42b nous permet d'identifier deux classes de techniques :

- ❑ la dimension temporelle est secondaire mais sa représentation est explicite,
- ❑ la représentation de la dimension temporelle n'est pas explicite.

La première classe regroupe les techniques qui considèrent la dimension temporelle secondaire. Dans ce cas les tâches utilisateur (observer l'évolution des valeurs par exemple) se rapportent seulement à la dimension structurelle. Néanmoins, la représentation de la dimension temporelle est nécessaire et explicite car l'utilisateur doit pouvoir situer les objets graphiques représentant les valeurs des données par rapport aux moments temporels correspondants. Dans ce cas, des représentations en diagrammes sont souvent utilisées. Ces représentations ne sont valables que pour des données quantitatives. Plusieurs types de diagrammes sont décrits dans la littérature, en particulier en statistique. La Figure 43 présente trois types de diagrammes. Il s'agit d'une représentation en courbe (Figure 43a), d'une représentation en histogramme (Figure 43b) ou d'une représentation en diagramme boursier à

---

<sup>7</sup> Citation :

Our eye is mutually practiced in detecting deviations from the horizon and graphic design should take advantage of this fact. Horizontally stretched time-series are more accessible to the eye.

la Figure 43c (nom extrait d'Excel [Microsoft 03]). Dans ce type de représentation, le temps est souvent représenté sur l'axe des X et les valeurs sur l'axe des Y. La représentation en diagrammes présente un inconvénient dans le cas de la visualisation d'un domaine temporel de grande taille. La plupart des outils de visualisation de données temporelles commercialisés comme DataDesk, ActivStats [DataDesk 03], ViSta [Young 97], eviews [eviews 03], mathworks [mathworks 03], jwave [jwave 03], ou Browser [Browser 03] sont basés sur des représentations en diagrammes.

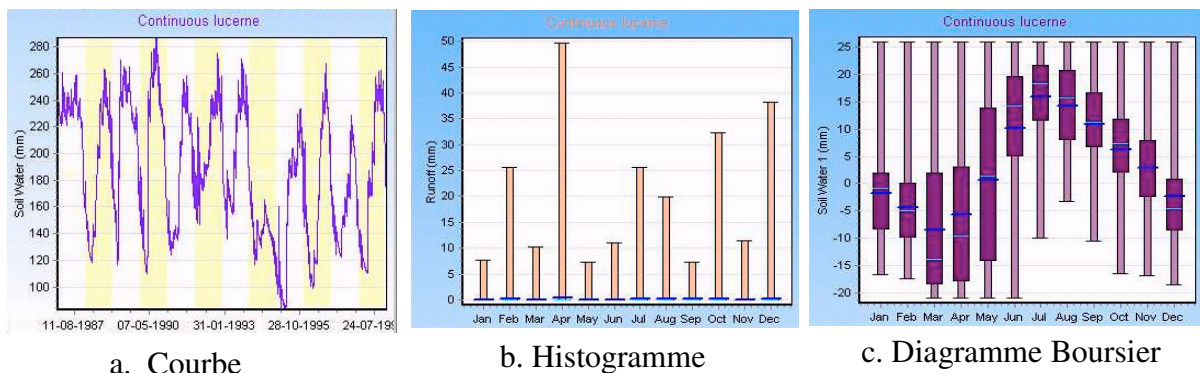


Figure 43 : Représentation de données temporelles en diagrammes [Browser 03].

Un diagramme peut être construit à partir d'éléments graphiques (rectangle, segment, etc.) ou de pixels. La Figure 44 présente deux représentations en histogramme [Hao 01]. A gauche (Figure 44a), la représentation est construite à base d'éléments graphiques, et à droite (Figure 44b) elle est construite à base de pixels. Dans les deux cas, le temps (observé à l'unité du Mois) est représenté sur l'axe horizontal, et les valeurs (prix des achats en Dollar) sur l'axe vertical.

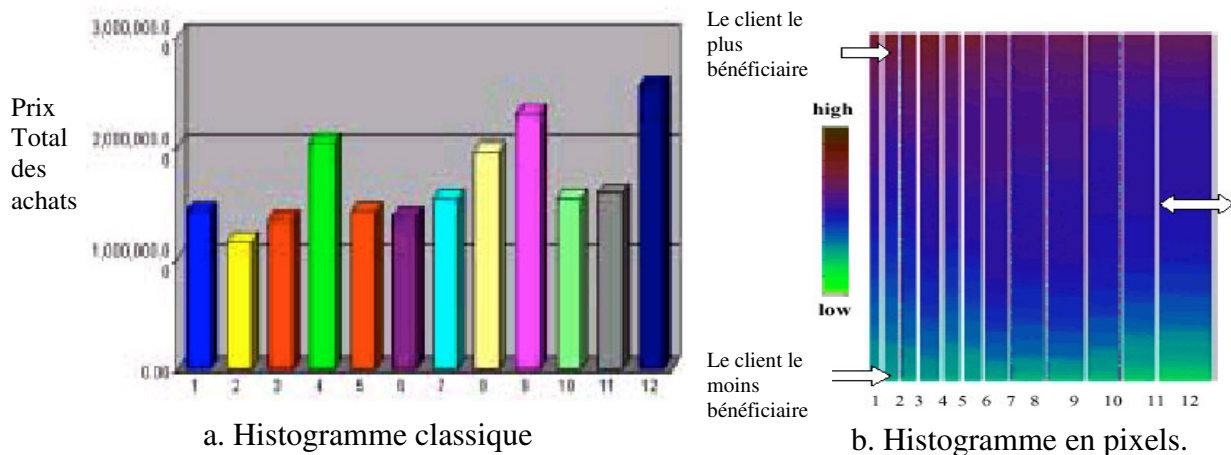


Figure 44 : Représentation en histogramme [Hao 01].



La deuxième classe regroupe les techniques dont la représentation de la dimension temporelle n'est pas explicite. Peu de travaux en visualisation de données temporelles ne considèrent pas de représentation du temps. La représentation de Tufte [Tufte 83] des pertes successives en hommes de l'armée française dans la campagne de Russie entre 1812 et 1813 (Figure 45) est un exemple de technique qui n'utilise pas de représentation explicite de la dimension temporelle. L'objectif est d'avoir une représentation globale de l'évolution de la dimension structurelle sans viser à positionner les valeurs structurelles par rapport aux moments correspondants. Le processus de visualisation de la donnée correspond au processus de visualisation de la dimension structurelle. Comme la dimension temporelle n'est pas explicitement représentée, les deux étapes *espace de visualisation* et *point de vue sur l'espace de visualisation* du processus de visualisation de la dimension temporelle ne sont pas considérées.

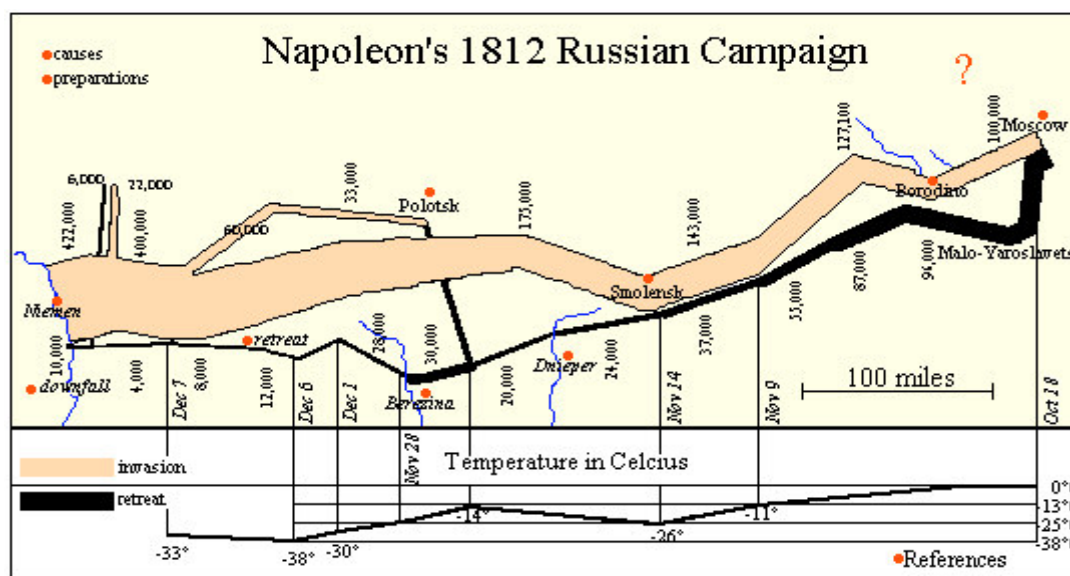
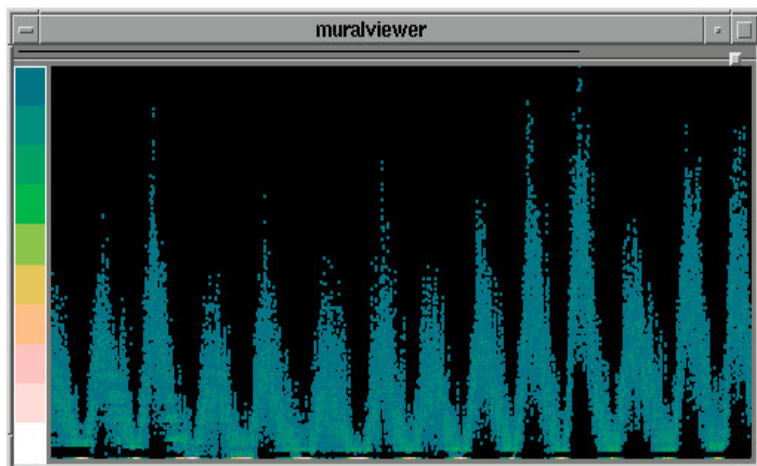


Figure 45 : Pertes de l'armée Française dans la campagne de Russie [Tufte 83].

La technique Information Mural [Jerding 97] est un deuxième exemple de technique pour lequel la représentation de la dimension temporelle n'est pas explicite. Cette technique permet la visualisation d'un grand espace de données : le nombre de valeurs à représenter est supérieur au nombre de pixels disponibles. Elle propose de transformer l'espace initial défini par  $M \times N$  pixels en un espace réduit de  $I \times J$  pixels tout en conservant une vue semblable à l'espace initial. La Figure 46 représente une visualisation par Information Mural du nombre de tâches solaires quotidiennement observées par les astronomes depuis 1850 ce qui est de l'ordre de 52000 valeurs. La transformation, appliquée à la dimension structurelle, vise à

garder une représentation réduite et similaire de l'espace réel des données. Cette transformation ne peut pas être appliquée à la dimension temporelle. Ainsi, l'objectif est d'avoir une représentation grossière de l'évolution de la donnée dans le temps.



**Information Mural**  
[Jerding 97] a été utilisé pour la visualisation de différents types de données, en particulier pour la visualisation des traces de programmes dans le temps, la visualisation de données astronomiques ou de données géographiques.

Figure 46 : Technique Information Mural [Jerding 97].

## 5.2 Fusion Espace de visualisation

La fusion des deux processus de visualisation au niveau de l'étape *espace de visualisation* (Figure 47) implique que l'espace de représentation de la dimension structurelle est dépendant de l'espace de représentation de la dimension temporelle. Ainsi, les caractéristiques de l'espace de représentation de la dimension temporelle et certains opérateurs qui s'y appliquent, affectent directement l'espace de représentation de la dimension structurelle, et inversement. Nous distinguons deux classes de techniques définies par le processus de la Figure 47 selon que l'espace de représentation de la dimension structurelle est fusionné dans l'espace de représentation de la dimension temporelle, ou non.

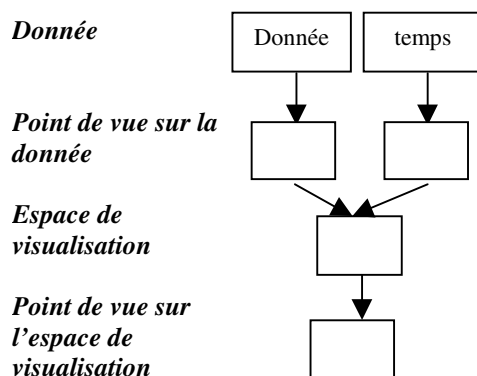


Figure 47 : Fusion des processus au niveau de l'étape *espace de visualisation*, extrait de la Figure 42.



Dans une première classe de techniques, nous regroupons les techniques dont l'espace de représentation de la dimension temporelle définit la structure et la forme de l'espace global de visualisation de la donnée. La technique TimeLine [TimeLine 03] de la Figure 48 en est un exemple. La dimension temporelle est représentée en spirale. Les opérateurs de l'étape *espace de visualisation* de la dimension temporelle affectent directement l'étape *espace de visualisation* de la dimension structurelle. Une rotation de l'espace de représentation de la dimension temporelle (la ligne temps dans l'exemple de la technique Timeline de la Figure 48a) implique une rotation de l'espace de représentation de la dimension structurelle. Les deux espaces sont obligatoirement fusionnés en un seul espace graphique. Si la dimension temporelle (réciproquement la dimension structurelle) est représentée dans un espace tridimensionnel, la dimension structurelle (réciproquement la dimension temporelle) l'est obligatoirement. Une représentation de la dimension structurelle (vue de l'intérieur du TimeLine) est présentée dans la Figure 48b.

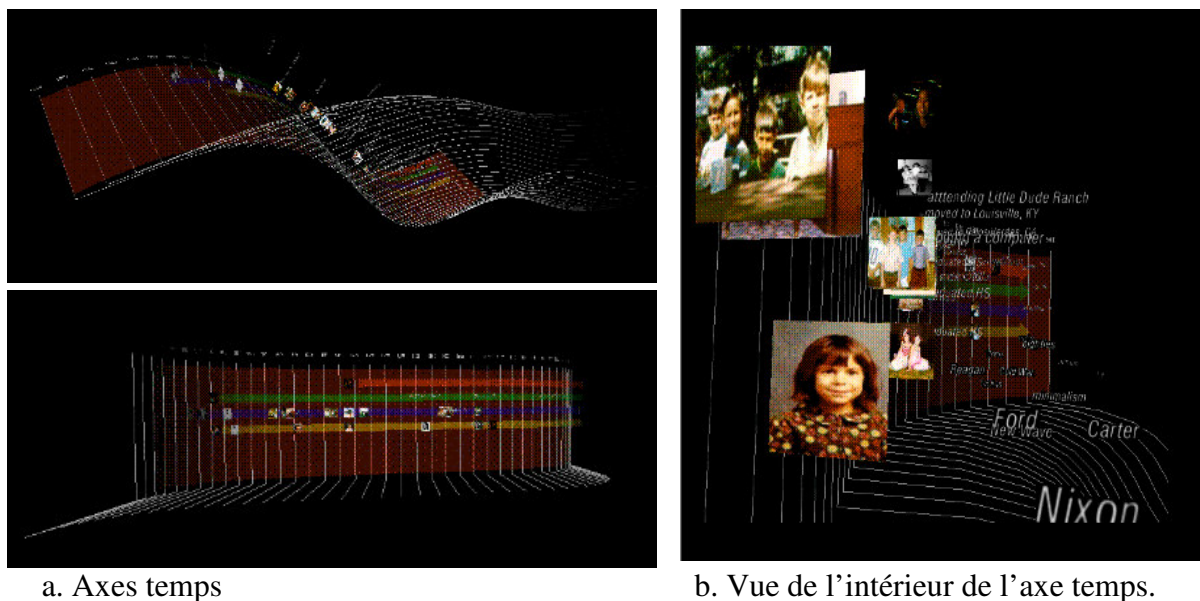


Figure 48 : L'axe temps structure la représentation des données [TimeLine 03].

La technique SpiraClock [Dragicevic 02] de la Figure 49 est un deuxième exemple de cette classe de techniques où la représentation de la dimension temporelle définit la structure de l'espace perceptible par l'utilisateur. Cette technique utilise la métaphore d'une montre pour représenter le temps sous la forme en spirale. Elle visualise des distances temporelles à des événements au fur et à mesure que ces derniers s'approchent. La Figure 49 représente cinq états de la technique SpiraClock à cinq moments différents. L'image au milieu indique qu'un

premier bus va partir à 9h45 et qu'il faut 5 minutes pour atteindre la première station de bus. Elle indique aussi qu'un deuxième bus va partir à 10h30 et qu'il faut 20 minutes pour atteindre la deuxième station de bus.

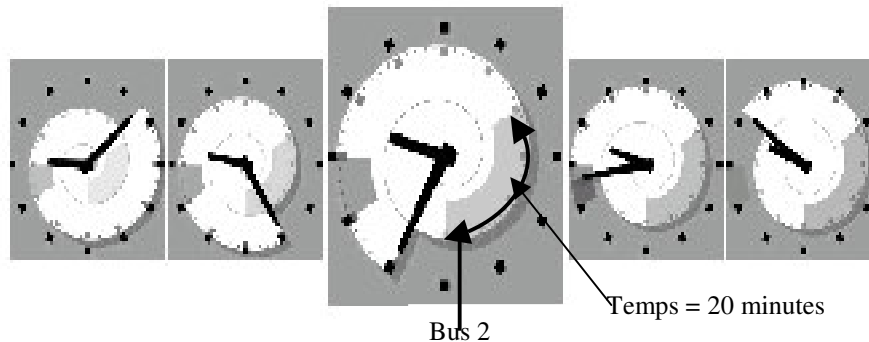


Figure 49 : Technique SpiraClock [Dragicevic 02].

Dans une deuxième classe, nous regroupons les techniques dont la représentation de la dimension structurelle est séparée de celle de la dimension temporelle tout en ayant l'étape *espace de visualisation* en commun. La dimension temporelle et la dimension structurelle sont obligatoirement visualisées dans le même espace de représentation. La technique Multi-scale Timeline Slider [Richter 99] de la Figure 50 sépare la représentation du temps de la représentation des valeurs structurelles. Cette technique consiste en une série de lignes temps. Chacune d'entre-elles représente un agrandissement d'une zone sélectionnée dans la ligne temps précédente. Le temps et les valeurs structurelles sont représentés séparément mais dans un même espace graphique (ceci est dû à la fusion au niveau de l'étape *espace de visualisation*).

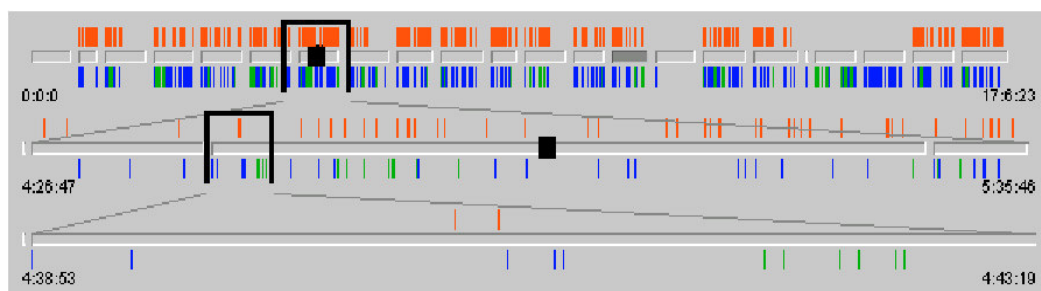


Figure 50 : A multi-scale Timeline [Richter 99].

Le deuxième exemple de cette classe de techniques de visualisation est celui présenté dans [Rekimoto 99]. Il s'agit d'un outil qui permet de sauvegarder dans le temps des vues de l'état du système (historique des pages web visitées, l'historique des mails reçus ou envoyés, etc.). L'interface de visualisation et d'interaction avec les données est représentée à la Figure 51.

Comme il s'agit de représenter une grande quantité d'informations, seule la vue correspondante à l'instant sélectionné au niveau du navigateur temps est affichée. Les valeurs temporelles et les valeurs structurales sont affichées au niveau du même espace de représentation.



**TimeScape [Rekimoto 99]** est un environnement pour la visualisation dans le temps des vues de l'état du système (historique des pages web visitées, l'historique des mails reçus ou envoyés, etc.).

Figure 51 : Visualisation de l'état d'un PC avec TimeScape [Rekimoto 99].

### 5.3 Fusion Point de vue sur l'espace de visualisation

La fusion des deux processus de visualisation dans une étape ultérieure à l'étape *espace de visualisation* permet de distinguer l'espace de représentation de la dimension temporelle de celui de représentation de la dimension structurale. Ainsi, le temps peut par exemple être représenté dans un espace en deux dimensions et les valeurs structurales dans un espace en trois dimensions, ou inversement.

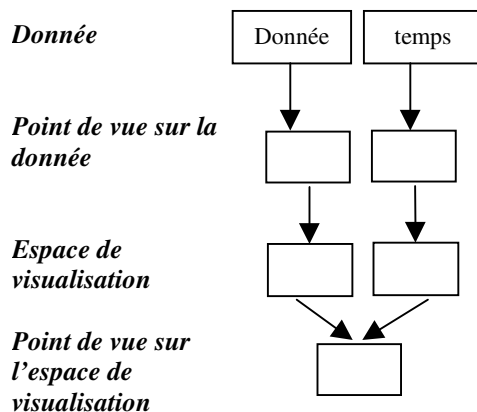
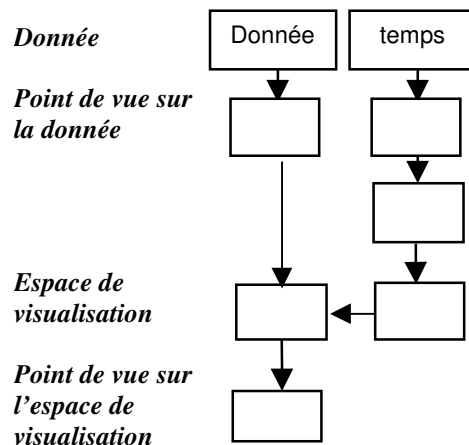


Figure 52 : Fusion des processus au niveau de l'étape *point de vue sur l'espace de visualisation*, extrait de la Figure 42.

Aucune technique de visualisation dans la littérature n'est conçue selon le processus de visualisation de la Figure 52 où la fusion est réalisée au niveau de l'étape *point de vue sur l'espace de visualisation*. En effet, le point de vue sur l'espace de la dimension temporelle ne dépend en aucun cas de celui de la dimension structurelle, alors que la réciproque est fautive. Les valeurs structurelles sont toujours définies par rapport à des valeurs temporelles, c'est-à-dire que le *point de vue sur l'espace de visualisation* de la dimension structurelle ne sera défini qu'une fois celui de la dimension temporelle défini. Ainsi, la définition des deux points de vue s'effectue séquentiellement et ne peut pas se faire en parallèle. La fusion au niveau de l'étape *point de vue sur l'espace de visualisation* est donc impossible. Nous présentons dans la suite les deux modèles de processus possibles pour lesquelles le *point de vue sur l'espace de visualisation* de la dimension temporelle et celui de la dimension structurelle sont définis séquentiellement.

#### **5.4 Temps puis Espace de visualisation de la dimension structurelle**

Certaines techniques de visualisation de données temporelles utilisent l'espace temps comme support pour la représentation des valeurs structurelles. L'avantage de ce type de représentation est l'association directe des valeurs structurelles aux valeurs temporelles correspondantes. Cette mise en correspondance réduit (ou élimine) la discontinuité visuelle entre les deux espaces de représentation. Les valeurs temporelles et les valeurs structurelles sont ainsi visualisées dans le même espace de représentation, en particulier celui du temps. Les caractéristiques de cet espace sont tout d'abord fixées par rapport à la dimension temporelle, les valeurs structurelles sont par la suite visualisées en respectant ces caractéristiques. Si le temps est représenté dans un espace en deux dimensions, les valeurs structurelles le seront aussi. Pour cette raison, l'espace de représentation de la dimension structurelle est défini une fois que le processus de visualisation du temps est complètement achevé (Figure 53). Il est important de noter que ce modèle est différent de celui où la fusion est réalisée à l'étape *espace de visualisation* (Figure 47). En effet, avec ce modèle, le processus de visualisation de la dimension temporelle est complètement indépendant de celui de visualisation de la dimension structurelle. Ce dernier peut être modifié ou remplacé par un autre tout en gardant le processus de visualisation de la dimension temporelle.



*Figure 53 : Processus temps puis l'étape espace de visualisation de la dimension structurelle, extrait de la Figure 42.*

La technique *représentation en Spirale* [Carlis 98] de la Figure 54 est conçue selon le modèle de processus de visualisation de la Figure 53. Cette technique considère le fait que les données sont périodiques. Elle utilise explicitement la relation de dépendance entre deux unités d'observation : unité grossière et unité fine (Chapitre 2). Les valeurs de l'unité grossière sont tout d'abord représenté sous la forme d'une spirale puis les valeurs de l'unité fine sont représentées par des rayons au-dessus de la spirale. Dans l'exemple de la Figure 54, chaque partition circulaire de la spirale correspond à une année (unité grossière), et chaque rayon correspond à un mois de l'année (unité fine). Les valeurs des données observées à l'unité d'observation Mois sont ensuite représentées par des éléments graphiques et sont placées directement à l'intersection de l'axe temps (la spirale) et le rayon correspondant. Nous soulignons l'existence d'une indépendance visuelle entre la ligne temps et les objets graphiques (disque plein dans la Figure 54) traduisant les valeurs structurelles. En effet, le processus de visualisation de la dimension structurelle peut être remplacé par un autre processus tout en conservant celui de la dimension temporelle. Ainsi, d'autres transformations visuelles peuvent être utilisées pour traduire les valeurs des données en signes graphiques.

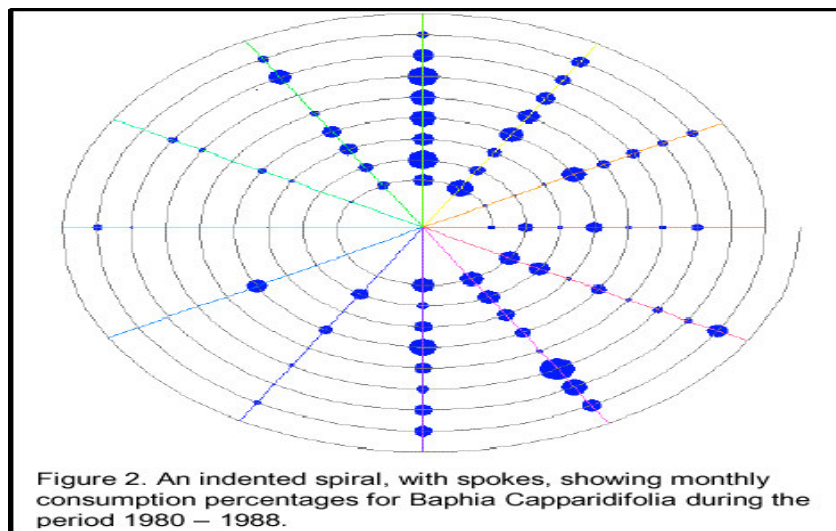


Figure 54 : L'axe temps utilisé comme support pour représenter les données [Carlis 98].

La technique Mur en Perspective [Mackinlay 91] de la Figure 39 (page 63) est un second exemple de technique conçue selon ce modèle de processus de visualisation. La déformation en trois faces, une en avant et deux en perspectives, définit la structure de l'espace de représentation de la dimension temporelle. L'espace de représentation de la dimension structurelle est par la suite défini en respectant la structure de l'espace temps. Là encore, un autre processus pour la visualisation de la dimension structurelle peut être utilisé.

## 5.5 Temps puis Point de vue sur l'espace de visualisation de la dimension structurelle

Comme le montre la Figure 55, l'étape *espace de visualisation* de la dimension structurelle est définie indépendamment du processus de visualisation de la dimension temporelle. La dimension temporelle est ainsi représentée dans un espace indépendant de celui de la dimension structurelle. Le temps peut par exemple être représenté dans un espace en deux dimensions alors que les valeurs structurelles sont représentées dans un espace en trois dimensions. Cette distinction permet une séparation explicite entre les tâches utilisateurs se rapportant à chacune des deux dimensions d'une donnée temporelle. La navigation dans le temps est réalisée par une navigation dans l'espace de représentation de la dimension temporelle, et la navigation dans l'espace des valeurs structurelles est réalisée au niveau de l'espace de représentation correspondant. L'inconvénient de ce mode de représentation est la possible discontinuité visuelle entre les deux espaces de représentation. Un mauvais positionnement des espaces l'un par rapport à l'autre peut rendre difficile, à l'utilisateur, le

positionnement de l'objet graphique traduisant une valeur structurale par rapport à l'instant correspondant.

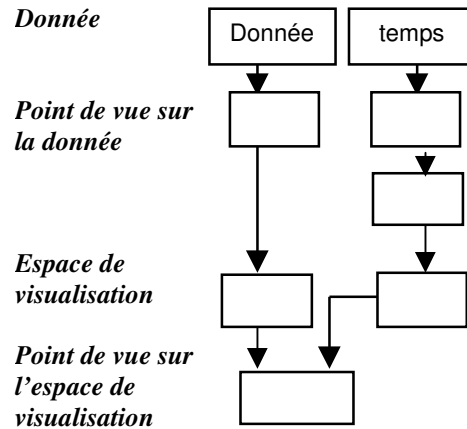
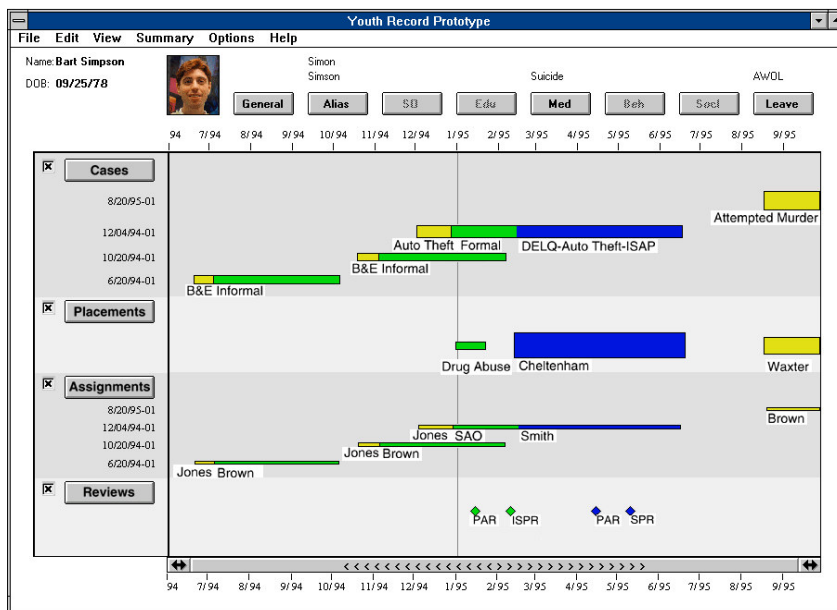


Figure 55 : Processus temps puis l'étape point de vue sur l'espace de visualisation de la dimension structurale, extrait de la Figure 42.

La technique Lifelines [Plaisant 96] de la Figure 56 est un exemple de technique développée selon ce modèle de processus de visualisation. Comme le montre la Figure 56, seul le point de vue sur l'espace de visualisation des valeurs structurales dépend du temps. Les deux espaces de représentation sont indépendants. Ainsi, les valeurs structurales pourraient être visualisées dans un espace en trois dimensions, sans que l'espace temps soit modifié. Dans cet exemple, les deux dimensions sont définies dans deux espaces, chacun à deux dimensions.



*LifeLines* est une technique destinée à la visualisation de biographie. Cette technique a été utilisée dans des applications médicales [Plaisant 98] et des applications légales [Harris 99].

Figure 56 : Technique LifeLines [Plaisant 96].

## 5.6 Processus complètement séparés

Aucune technique de visualisation d'une donnée temporelle de la littérature n'utilise le modèle tel que les deux processus de visualisation sont complètement séparés (Figure 57). La séparation complète entre les deux processus de visualisation est impossible car les valeurs structurelles sont toujours définies par rapport à des valeurs temporelles.

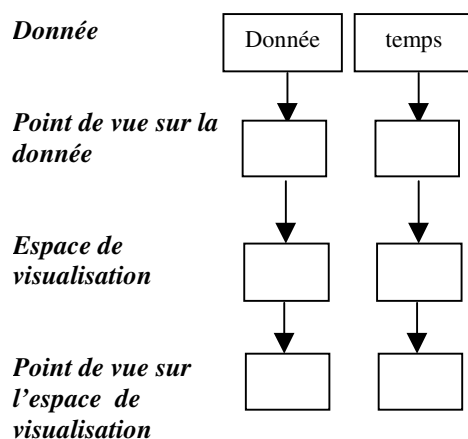


Figure 57 : Processus complètement séparés, extrait de la Figure 42.

## 5.7 Synthèse

Dans ce paragraphe, nous avons présenté plusieurs modèles de processus de visualisation d'une donnée temporelle (Figure 42, page 67) en considérant les points d'ancrage entre le processus de visualisation de la dimension structurelle et celui de la dimension temporelle. Ces modèles varient entre deux extrémités : processus complètement fusionnés et processus complètement séparés. Nous classons dans le Tableau 1 les techniques existantes selon notre taxonomie. Comme le montre ce tableau, la plupart des techniques existantes correspondent aux deux premiers modèles de processus de visualisation de la Figure 42 (les deux premières lignes du tableau ci-dessous). Il convient donc d'explorer plus en détail les deux derniers modèles de processus. Dans ces deux modèles, le processus de visualisation de la dimension temporelle est tout d'abord achevé, puis soit l'étape *espace de visualisation* ou l'étape *point de vue sur l'espace de visualisation* de la dimension structurelle est réalisée.

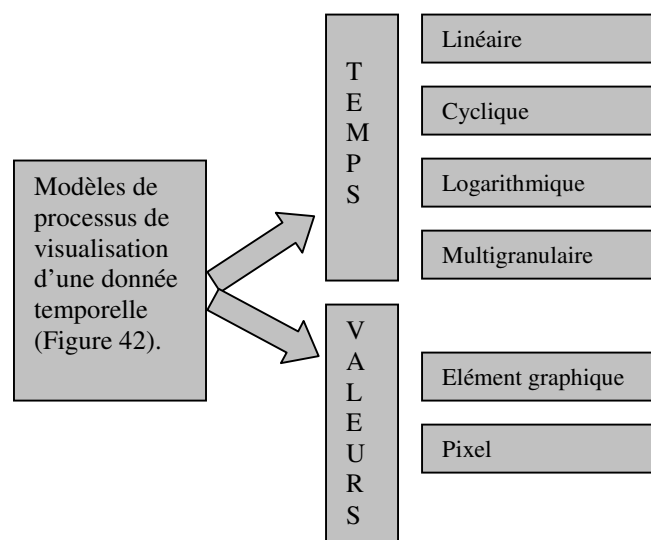


Fusion complète/ fusion au niveau de l'étape <i>point de vue sur la donnée</i>	Data Explorer (DX) [Explorer 03], Représentation en diagrammes : [Fredrikson 99] [Kosara 00] [Theus 96] [TAO 03] [Brower 03] [Hao 01], Carte figurative des pertes de l'armée Française [Tufté 83], Information Mural [Jerding 97], TimeMap [TimeMap 03].
Fusion au niveau de l'étape <i>espace</i>	Time Tube [CPM+], [Wijk 99], Dynamic Timelines [Kullberg 95], [Visual 02], Spirale [Weber 01], Spirale [Renieris 99], Worlds Within Worlds [Feiner 90], Spiral Calendar Visualizer [Mackinlay 94], TimeLine [TimeLine 03], SpiraClock [Dragicevic 02], multi-scale Timeline [Richter 99], TimeScape [Rekimoto 99], BITS [Dias 95], [Stolte 99].
Fusion au niveau de l'étape <i>point de vue sur l'espace</i>	Cas impossible
<i>Processus temps</i> puis l'étape <i>espace de visualisation</i> de la dimension structurelle	Représentation spirale [Carlis 98], Mur en perspective [Mackinlay 91]
<i>Processus temps</i> puis l'étape <i>point de vue sur l'espace de visualisation</i> de la dimension structurelle	LifeLines [Plaisant 96], Représentation en calendrier [Wijk 99]
Processus complètement séparés	Cas impossible

Tableau 1 : Techniques de visualisation d'une seule donnée temporelle.

En résumé, une technique de visualisation est définie par des caractéristiques de la dimension structurelle, des caractéristiques de la dimension temporelle, et par le modèle de processus de visualisation utilisé. Concernant la dimension structurelle, nous retenons la granularité de l'élément graphique utilisé. Nous distinguons ainsi les techniques à base d'éléments

graphiques des techniques à base de pixels. Par rapport à la dimension temporelle, nous considérons la forme de représentation du temps comme critère de classification des techniques de visualisation. Le temps peut être représenté sous une forme linéaire, cyclique, logarithmique ou multigranulaire. Enfin, nous considérons le processus de visualisation structurant la dimension structurelle et la dimension temporelle (Figure 42) comme critère de classification des techniques existantes. La Figure 58 récapitule ces trois niveaux de classification des techniques de visualisation d'une donnée temporelle.



*Figure 58* : Critères de classification de techniques de visualisation d'une donnée temporelle. Ayant présenté notre taxonomie de visualisation d'une seule donnée, nous étendons notre étude aux cas des techniques de visualisation de plusieurs données temporelles.

## **6 Techniques de visualisation de plusieurs données temporelles**

L'étude des techniques de visualisation de plusieurs données temporelles proposées dans la littérature fait apparaître deux classes de techniques en fonction du nombre d'espaces temps utilisés. Une première classe considère une seule représentation de la dimension temporelle pour visualiser plusieurs dimensions structurelles des données correspondantes. Une deuxième classe considère une représentation de la dimension temporelle relative à chaque dimension structurelle des données à visualiser. Nous étudions ces deux classes de techniques de visualisation dans la suite.

## 6.1 Utilisation d'un seul espace temps

L'utilisation d'une seule représentation de l'espace temps pour visualiser plusieurs données temporelles garantit la mise en correspondance des dimensions structurelles aux mêmes instants temporels. Il s'agit en particulier d'observer le comportement des données et d'étudier leurs relations à des points temporels précis. L'utilisation d'une seule représentation de l'espace temps suppose que les différentes données ont le même domaine temporel, c'est-à-dire qu'elles sont estampillées par les mêmes instants. Dans le cas contraire, la technique doit construire l'espace temps défini par l'union des espaces temps correspondant aux données à visualiser tout en faisant des transformations pour ramener les valeurs temporelles à la même unité d'observation, si elles sont définies dans différentes unités (Chapitre 2).

Nous distinguons deux sous-classes de techniques de visualisation de plusieurs données temporelles par rapport à un seul espace temps selon qu'elles utilisent un ou plusieurs espaces pour représenter les différentes dimensions structurelles.

L'application TimeSearcher [Hochheiser 02a, Hochheiser 02b] de la Figure 59 est un exemple de la première sous-classe de techniques où toutes les dimensions structurelles des données sont visualisées dans un même espace graphique et par rapport à un seul espace temps. Par exemple, dans la Figure 59, les courbes sont tracées dans le même repère.

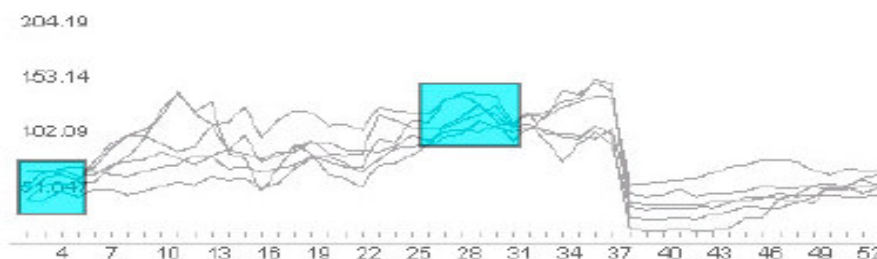


Figure 59 : TimeSearcher [Hochheiser 02a, Hochheiser 02b].

La deuxième sous-classe de techniques utilise un espace de représentation pour chacune des dimensions structurelles. Les espaces de représentation des dimensions structurelles sont par la suite placés par rapport à un seul espace temps. Des exemples de cette classe de techniques sont le système Browser [Browser 03] et la technique AsbruView [Kosara 99, Miksch 98]. Le système Browser [Browser 03] de la Figure 60 par exemple, visualise chaque dimension

structurale dans un espace séparé. Les représentations des dimensions structurelles ne sont pas superposées, et elles sont placées par rapport à un seul axe temps.

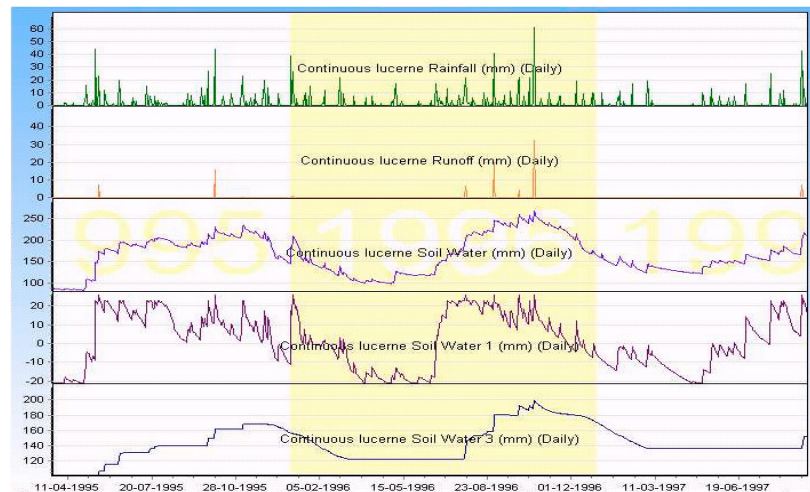


Figure 60 : Un seul axe temps pour représenter plusieurs données [Browser 03].

D'autres techniques de visualisation de cette deuxième sous-classe n'utilisent pas de représentation explicite d'un espace temps. L'approche utilisée consiste à décomposer le temps en éléments (instants ou intervalles), puis à traduire les valeurs structurelles de chaque élément temporel en une représentation visuelle, et enfin à placer les représentations selon l'ordre de succession des éléments temporels. La distinction entre les moments temporels se fait par une distinction des représentations des dimensions structurelles. La Figure 61 présente une visualisation de trois données (oxydes de nitrogène, monoxyde de carbone et hydrocarbure réactif). Le temps est décomposé en quatre intervalles. Le premier intervalle de temps, par exemple, est défini de minuit à trois heures du matin.

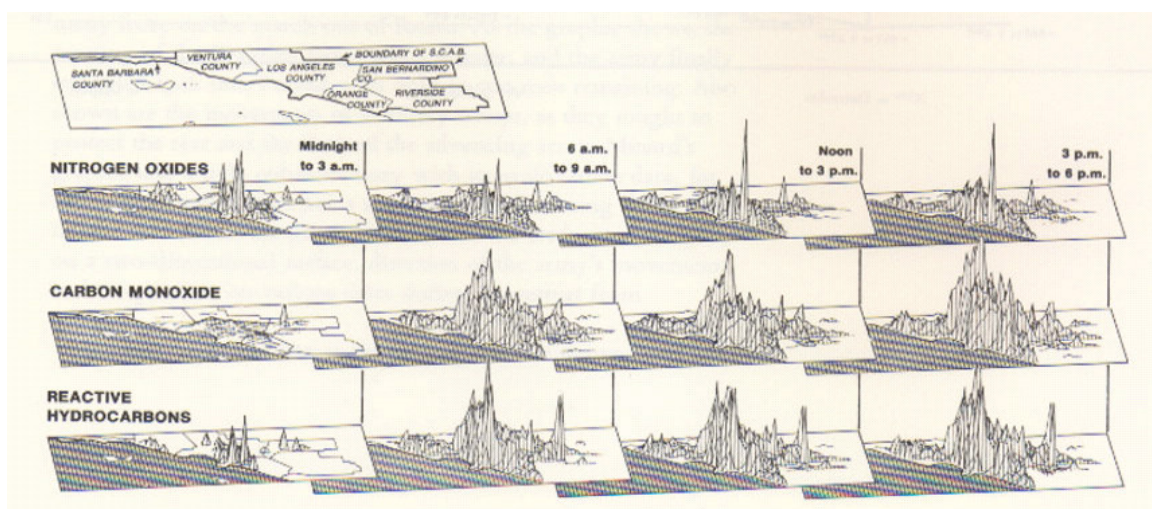
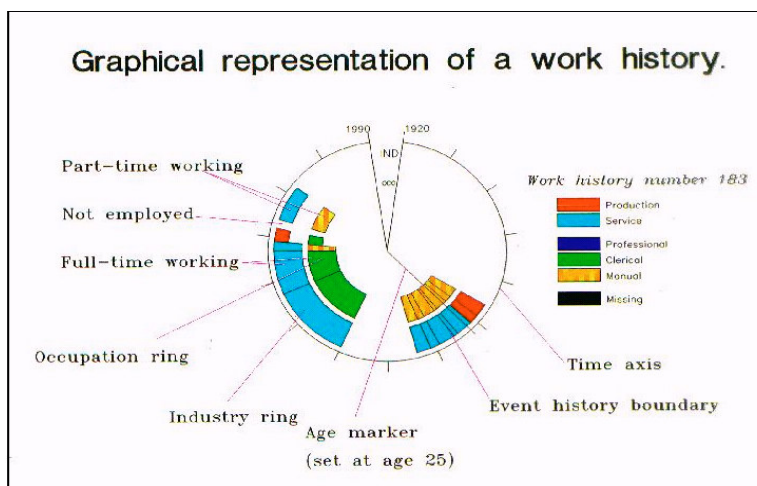


Figure 61 : Représentation du temps à travers la représentation des données [Tufté 83].

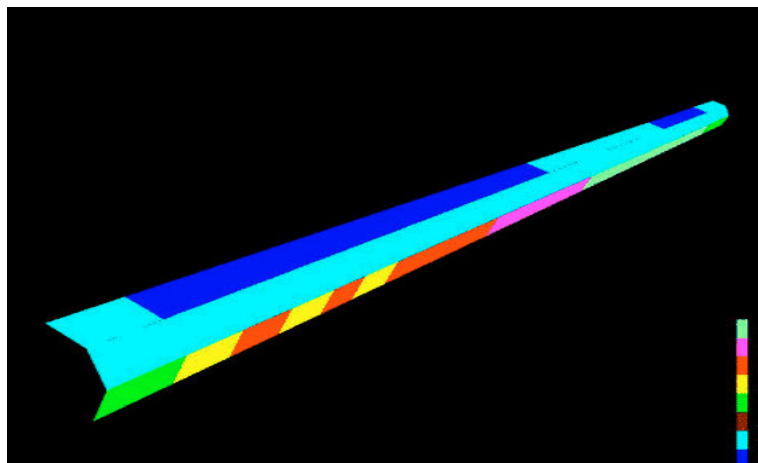
Les trois techniques précédentes (Figure 59, 60 et 61) utilisent une représentation linéaire du temps. Dans la Figure 62, la technique Tulip [Tulip 02], utilise une représentation cyclique (circulaire) de l'axe temps pour représenter plusieurs dimensions structurelles. La couleur est utilisée pour distinguer les différentes dimensions structurelles des données visualisées. L'utilisation d'un seul espace temps pour représenter plusieurs dimensions structurelles est donc indépendante des perceptions linéaire, cyclique, logarithmique ou multigranulaire du temps.



**Tulip** [Tulip 02] est une technique pour la visualisation de plusieurs données quantitatives par rapport à un seul axe temps. Le temps est représenté en cercle. La couleur est utilisée pour différencier entre les différentes données.

Figure 62 : Représentation de plusieurs dimensions structurelles par rapport à un seul espace temps sous une forme cyclique [Tulip 02].

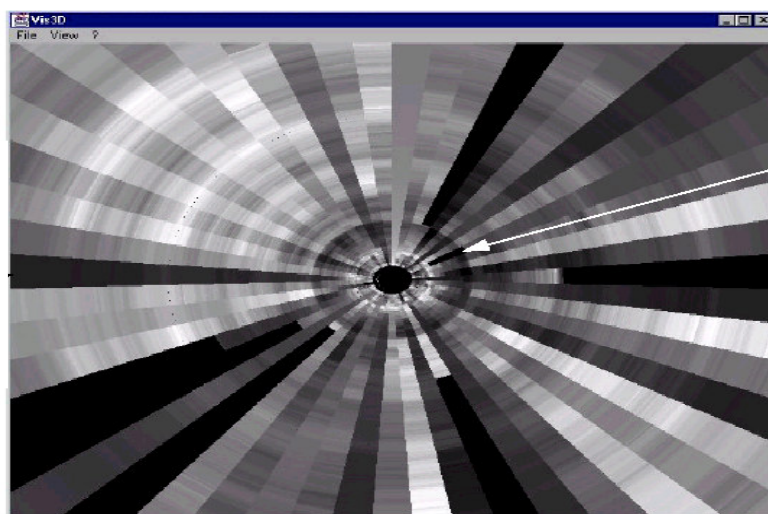
Aussi, indépendamment du nombre de dimensions de l'espace final de représentation, une seule dimension temps peut être utilisée pour représenter plusieurs dimensions structurelles. Dans les quatre exemples précédents, le temps est représenté dans un espace en deux dimensions. La technique Lexis Pencils [Brian 03] de la Figure 63 utilise la forme d'un stylo pour représenter le temps dans un espace en trois dimensions. L'axe temps est représenté par l'axe défini par le stylo même. Le stylo est représenté par différentes facettes, chacune est utilisée pour représenter une dimension structurelle. Les valeurs sont représentées par des variables rétinienne comme la couleur et la texture. L'exemple présenté dans la Figure 63 est une visualisation de l'historique de la vie d'un couple. Le temps varie de la gauche vers la droite commençant de la date du mariage. La face en haut du stylo représente l'historique du travail de la femme, celle au milieu représente l'historique du travail de l'homme, et la face en bas représente l'âge des enfants. Les valeurs des données sont traduites en couleurs. Pour la facette en bas par exemple, aucun enfant est codé en vert, en jaune pour un enfant âgé de moins un an, en rouge pour un enfant âgé de 1 à 5 ans, en magenta de 5 à 11, etc.



**Lexis Pencils** [Brian 03], développée par Francis Brian et Pritchard John à Lancaster University, permet de visualiser plusieurs données temporelles. Le temps est représenté sous la forme d'un stylo dans un espace en trois dimensions. Chaque facette du stylo sert à visualiser une dimension structurelle.

Figure 63 : Technique Lexis Pencils [Brian 03].

La visualisation des dimensions structurelles sur les facettes externes du stylo avec la technique Lexis Pencils nécessite que l'utilisateur tourne le stylo dans l'espace de représentation pour observer une donnée particulière. Il est ainsi difficile de comparer, par exemple, les données entre elles. La technique Data Tube [Ankerst 00] de la Figure 64 visualise les données au niveau des facettes internes d'un tube. Le tube représente l'axe temps. Cette technique permet ainsi d'observer l'évolution de toutes les données en même temps. La Figure 64 présente une visualisation de 50 données (prix des actions de Frankfurter Aktien Index) de Janvier 1974 à Avril 1995. Chaque colonne de la représentation correspond à une donnée. Il convient de noter qu'il ne s'agit pas d'une représentation en pixels comme pour la technique Cercle-Segments de la Figure 22. Avec la technique Data Tube, chaque valeur est représentée par un segment (élément graphique) dont la couleur est définie dans un niveau de gris proportionnellement à la valeur.



**Data Tube** [Ankerst 00, Ankerst 01] est une technique en 3D pour visualiser un grand nombre données. Le temps est représenté par un tube à l'intérieur duquel les dimensions structurelles sont représentées.

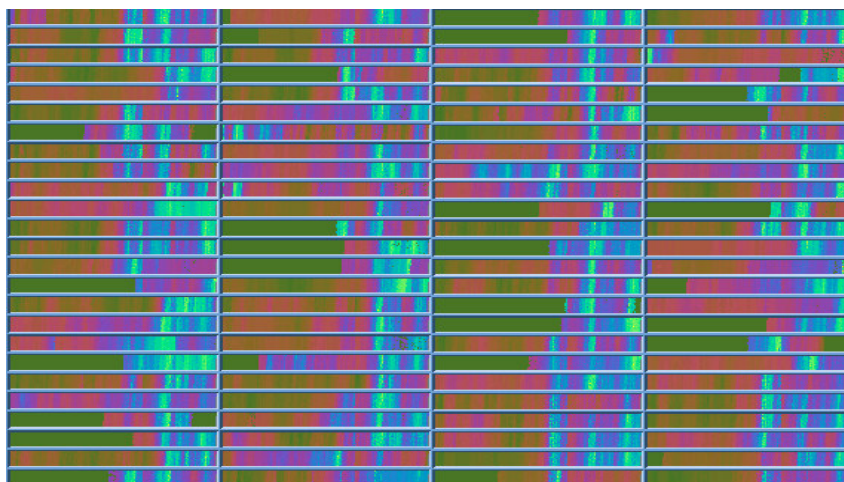
Figure 64 : Technique Data Tube [Ankerst 00, Ankerst 01].



En synthèse, nous distinguons deux sous-classes de techniques de visualisation de plusieurs données temporelles par rapport à un seul espace temps. Tandis que la première sous-classe utilise un seul espace pour représenter les différentes dimensions structurelles, la deuxième sous-classe utilise un espace de représentation pour chacune des dimensions structurelles.

## **6.2 Utilisation d'un espace temps par donnée**

La représentation d'un espace temps pour chaque donnée permet de ne représenter que le domaine temporel associé à la donnée en question. L'important est la comparaison des différentes formes du comportement des données et non pas leurs valeurs par rapport aux mêmes instants temporels. La Figure 65 présente une visualisation de cent données temporelles, à chacune est associée une portion rectangulaire définie par un repère modélisant l'espace temps. Les valeurs des données sont traduites en coloration de pixels.



*Figure 65* : Utilisation d'un espace temps par donnée [keim 96b].

L'objectif de ce type de représentation est d'observer l'évolution des données dans le temps sans viser à comparer leurs valeurs à des moments temporels précis. A l'opposé, la technique Cave Plots [Becker 94] de la Figure 66 est conçue pour faciliter la comparaison de deux données par rapport aux mêmes instants. Elle utilise deux représentations du même axe temps représentées en parallèle pour visualiser deux données X et Y (Figure 66). Les valeurs de la donnée X sont représentées de bas en haut, celles de la donnée Y de haut en bas. La duplication de l'axe temps (en bas et en haut), sert à faciliter le positionnement des valeurs des deux dimensions structurelles par rapport aux mêmes valeurs de la dimension temporelle.

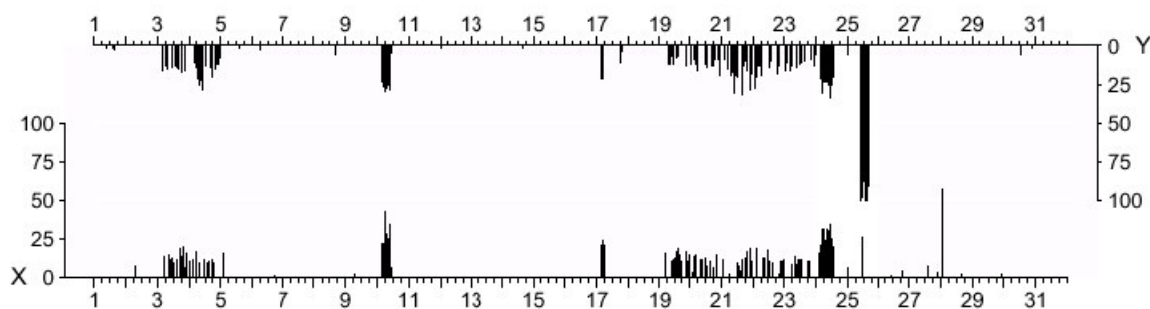


Figure 66 : Technique Cave plots [Becker 94].

### 6.3 Synthèse

Dans ce paragraphe, nous avons présenté une étude des techniques de visualisation de plusieurs données temporelles. Cette étude fait ressortir deux approches utilisées pour la conception de techniques multi-données temporelles. La première approche utilise une seule représentation de l'espace temps tandis que la deuxième utilise une représentation d'un espace temps par donnée visualisée.

Nous classons dans le Tableau 2 les techniques existantes étudiées selon ces deux approches. Comme le confirme le tableau, la plupart des techniques de visualisation de plusieurs données temporelles utilisent une seule représentation de l'espace temps. En effet, l'utilisation d'un seul espace temps garantit le fait que toutes les données soient positionnées par rapport aux mêmes valeurs temporelles. La comparaison des données devient plus immédiate. Le choix de l'utilisation d'un ou de plusieurs espaces temps dépend donc des tâches utilisateurs. Dans le cadre de cette thèse, nous n'excluons aucune de ces deux possibilités. Les tâches utilisateurs seront étudiées dans le chapitre suivant.

Un seul espace temps	Browser [Browser 03], AsbruView [Kosara 99, Miksch 98], TimeSeracher [Hochheiser 02a, Hochheiser 02b], Tulip [Tulip 02], Lexis Pencils [Brian 03], ThemeRiver [Havre 99, Havre 00], Data Tube [Ankerst 00].
Un espace temps par donnée	Cave plots [Becker 94], cercle-segments [Keim 96b].

Tableau 2 : Techniques de visualisation de plusieurs données temporelles.



## **7 Conclusion**

Dans ce chapitre, nous avons présenté une revue des techniques de visualisation de données temporelles. Une donnée temporelle est définie par deux dimensions : une dimension temporelle et une dimension structurelle. A partir de cette définition, nous avons considéré que l'espace de représentation d'une donnée temporelle est défini par une composition de deux espaces relatifs à ces deux dimensions. La représentation de chaque dimension est le résultat d'un processus de visualisation. Dans ce chapitre, nous avons considéré le processus de visualisation "*modèle d'état de donnée*", proposé par Chi [Chi 00, Chi 02]. Nous avons d'abord présenté ce processus de visualisation appliqué à la dimension structurelle et à la dimension temporelle en les illustrant par des exemples de techniques de visualisation. Nous avons ensuite proposé une taxonomie de techniques de visualisation d'une donnée temporelle en nous basant sur les deux processus de visualisation mis en jeu. Un premier trait de classification est donc la relation existante entre les deux processus (dimension temporelle et dimension structurelle). Nous avons identifié 7 modèles de processus de visualisation d'une donnée temporelle (Figure 42, [page 67](#)). Le Tableau 1 ([page 80](#)) résume les travaux de visualisation d'une seule donnée temporelle par rapport aux modèles de la Figure 42. Comme le montre ce tableau, deux modèles de processus de la taxonomie sont peu couverts par les travaux de la littérature. Dans ces deux modèles, le processus de visualisation de la dimension temporelle est achevé indépendamment du processus de visualisation de la dimension structurelle. Il convient donc de couvrir ces deux modèles de processus de visualisation d'une donnée temporelle. Ce travail fera l'objet du chapitre quatre.

## *Chapitre 4*

# *Conception de techniques d'interaction avec des données temporelles*

“A graphic is never an end in itself; it is a moment in the process of decision-making.”

**Jacques Bertin**  
**Graphic Information**  
**Processing**

Table des matières

**Chapitre 4.** Conception de techniques d'interaction avec des données temporelles.

<a href="#">1</a>	<a href="#">Introduction</a>	91
<a href="#">2</a>	<a href="#">Approche centrée tâches utilisateur</a>	92
<a href="#">2.1</a>	<a href="#">Tâches d'interaction avec la source de données</a>	94
<a href="#">2.2</a>	<a href="#">Tâches d'analyse de données</a>	96
<a href="#">2.2.1</a>	<a href="#">Tâches de navigation</a>	96
<a href="#">2.2.2</a>	<a href="#">Tâches d'observation</a>	97
<a href="#">2.2.3</a>	<a href="#">Tâches de comparaison</a>	102
<a href="#">2.2.4</a>	<a href="#">Tâches de manipulation</a>	105
<a href="#">2.3</a>	<a href="#">Synthèse</a>	106
<a href="#">3</a>	<a href="#">Processus de conception de l'espace temps</a>	106
<a href="#">3.1</a>	<a href="#">Lignes temps proposées dans la littérature</a>	107
<a href="#">3.2</a>	<a href="#">Vue en mur en perspective pour représenter le temps</a>	110
<a href="#">3.2.1</a>	<a href="#">Gestion d'un grand espace d'informations temporelles</a>	110
<a href="#">3.2.2</a>	<a href="#">Caractéristiques du temps</a>	111
<a href="#">3.2.3</a>	<a href="#">Tâches utilisateur se rapportant à la dimension temporelle</a>	112
<a href="#">3.3</a>	<a href="#">Dispositif physique pour l'interaction avec l'espace temps</a>	113
<a href="#">4</a>	<a href="#">Techniques de visualisation de données temporelles</a>	115
<a href="#">4.1</a>	<a href="#">Technique cercles concentriques</a>	115
<a href="#">4.1.1</a>	<a href="#">Représentation graphique</a>	115
<a href="#">4.1.2</a>	<a href="#">Processus de visualisation</a>	118
<a href="#">4.1.3</a>	<a href="#">Tâches utilisateur</a>	119
<a href="#">4.2</a>	<a href="#">Technique de représentation en étoiles</a>	121
<a href="#">4.2.1</a>	<a href="#">Représentation graphique</a>	121
<a href="#">4.2.2</a>	<a href="#">Processus de visualisation</a>	123
<a href="#">4.2.3</a>	<a href="#">Tâches utilisateur</a>	123
<a href="#">4.3</a>	<a href="#">Représentation en diagrammes superposés</a>	125
<a href="#">4.3.1</a>	<a href="#">Représentation graphique</a>	125
<a href="#">4.3.2</a>	<a href="#">Processus de visualisation</a>	127
<a href="#">4.3.3</a>	<a href="#">Tâches utilisateur</a>	128
<a href="#">4.4</a>	<a href="#">Technique représentation tabulaire</a>	129
<a href="#">4.4.1</a>	<a href="#">Représentation graphique</a>	129
<a href="#">4.4.2</a>	<a href="#">Processus de visualisation</a>	129
<a href="#">4.4.3</a>	<a href="#">Tâches utilisateur</a>	130
<a href="#">4.5</a>	<a href="#">Technique représentation en pixels</a>	132
<a href="#">4.5.1</a>	<a href="#">Représentation graphique</a>	132
<a href="#">4.5.2</a>	<a href="#">Processus de visualisation</a>	134
<a href="#">4.5.3</a>	<a href="#">Tâches utilisateur</a>	134
<a href="#">4.6</a>	<a href="#">Synthèse</a>	135
<a href="#">5</a>	<a href="#">Conclusion</a>	136

## 1 Introduction

Dans le chapitre précédent nous avons présenté des techniques de visualisation de données temporelles organisées selon une taxonomie que nous avons définie. En considérant une donnée temporelle comme une association d'une dimension temporelle et d'une dimension structurelle (Chapitre 2), nous classons les techniques de visualisation selon le processus de visualisation utilisé [Chi 00]. Un processus de visualisation d'une donnée temporelle correspond à une association entre un processus de visualisation de sa dimension temporelle et un processus de visualisation de sa dimension structurelle. Plusieurs modèles ont été définis variant entre processus complètement fusionnés et processus complètement séparés. Notre taxonomie fait apparaître qu'il y a deux processus, présentés à la Figure 67, peu couverts par les travaux de visualisation de données temporelles proposés dans la littérature. Comme le montre cette figure, dans ces deux processus, la dimension temporelle est visualisée indépendamment de la dimension structurelle. Nous retenons donc ces deux processus pour la visualisation de données temporelles. Nous présentons dans ce chapitre nos travaux sur la conception de techniques de visualisation de données temporelles dans le cas de ces deux processus de la Figure 67.

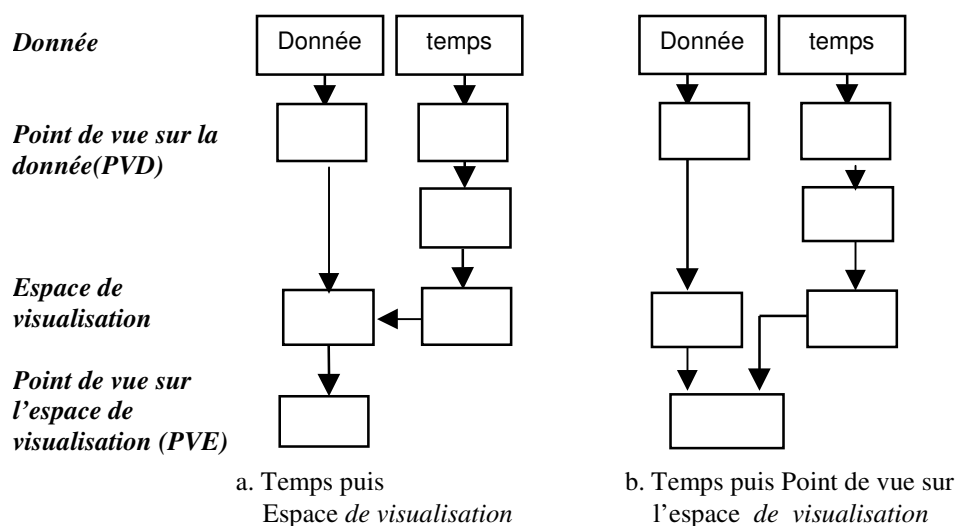


Figure 67 : Processus retenus pour la visualisation de données temporelles (Chapitre 3).

Le processus de visualisation proposé par Chi est structuré en quatre étapes (*donnée*, *point de vue sur la donnée*, *espace de visualisation* et *point de vue sur l'espace de visualisation*) que nous avons définies dans le chapitre précédent. Le rendu perceptible par l'utilisateur est défini par l'étape *point de vue sur l'espace de visualisation*. Cette étape consiste à traduire les

valeurs des données en représentations visuelles. Pour garantir l'utilisabilité des techniques de visualisation, nous adoptons une approche centrée tâches utilisateur pour réaliser l'étape *point de vue sur l'espace de visualisation*. Notre choix est dicté par des résultats issus de travaux antérieurs [Casner 91, Roth 94, Chuah 95] qui recommandent la conception de techniques de visualisation au regard des tâches utilisateur : "Graphics appear to succeed in practice when they have been designed to directly support a specific task, the success arising out of a judicious combination of tasks to be performed and particular graphic used." Extrait de [Casner 91]. Notre approche consiste à traduire chaque tâche utilisateur en un ensemble de propriétés ergonomiques que l'étape *point de vue sur l'espace de visualisation* doit respecter. Comme le montre la Figure 68, une technique de visualisation est conçue selon le processus de visualisation utilisé et selon les tâches utilisateur visées.

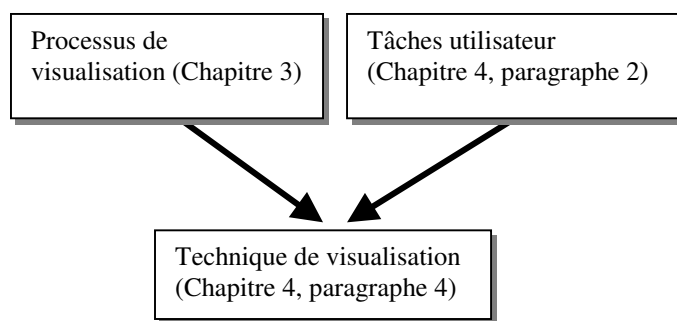


Figure 68 : Notre approche de travail.

Nous organisons ce chapitre comme suit : nous présentons au paragraphe 2 notre approche de conception des techniques de visualisation de données temporelles au regard des tâches utilisateur. Comme le montre la Figure 67, le processus de visualisation de la dimension temporelle est achevé indépendamment du processus de visualisation de la dimension structurelle. Nous présentons dans le paragraphe 3 le processus de visualisation du temps puis dans le paragraphe 4 les techniques de visualisation que nous avons conçues selon l'approche détaillée dans le paragraphe 2.

## **2 Approche centrée tâches utilisateur**

Afin de garantir l'utilisabilité du futur système, une approche de conception centrée utilisateur met l'utilisateur au cœur du cycle de conception et de développement du système [Norman 86]. Idéalement, l'utilisateur participe à l'accomplissement des différentes phases de la production du logiciel : commençant par la phase d'analyse des besoins jusqu'à la phase de

test. Les phases, où l'intervention de l'utilisateur est incontournable, sont celles d'analyses des besoins (tâches) et d'évaluation. Dans ce paragraphe, nous nous limitons à l'étape analyse des tâches utilisateur, l'étape évaluation est présentée dans le Chapitre 5.

L'objectif est de concevoir des techniques de visualisation adaptées aux tâches utilisateur. Nous distinguons deux types de tâches : des tâches d'extraction de données et des tâches de manipulation de données [Daassi 99b]. Les tâches d'extraction de données ne sont pas spécifiques aux données temporelles. Elles sont définies en termes d'opérateurs temporels au niveau des langages de requêtes pour bases de données temporelles (Chapitre 2). Dans nos travaux, nous n'exploitons pas ces opérateurs pour décider des choix de représentation ni pour identifier les tâches utilisateur. L'un des objectifs de notre travail est de garantir l'indépendance des techniques de visualisation développées vis-à-vis de la source de données. Or, les opérateurs temporels et donc la syntaxe du langage de requêtes utilisé, dépendent largement du modèle du temps (calendrier et types temporels) utilisé par la source de données.

Les tâches de manipulation de données concernent l'interaction de l'utilisateur avec les données et leur manipulation afin de favoriser leur analyse. Selon Norman [Norman 86], les utilisateurs élaborent des modèles conceptuels qui déterminent leurs comportements durant le processus d'analyse de données. Suivant le type d'applications dans lesquelles il est impliqué, l'utilisateur juge différemment la façon selon laquelle les données lui sont présentées.

De plus, les tâches utilisateur dépendent largement du type de données manipulées. La manipulation d'un flux vidéo est différente de la manipulation de données quantitatives. Pour ceci, nous nous limitons à un type particulier d'applications qui concernent l'analyse visuelle des données temporelles et quantitatives pour les applications géographiques. Une technique de visualisation de données temporelles est ainsi conçue pour favoriser l'analyse de ces données. Dans ce cadre, nous avons identifié une liste de tâches utilisateur en terme d'analyse de données temporelles. Pour se faire, nous avons mené des interviews avec des utilisateurs géographes du laboratoire SEIGAD (Systèmes Environnementaux, Information Géographique et Aide à la Décision) de l'Université Joseph Fourier. Nous les avons observé accomplir leurs tâches d'analyse de données réelles. Nous ne prétendons pas l'exhaustivité ou la complétude de la liste des tâches identifiées. Nous estimons néanmoins que ces tâches sont les plus

fréquentes et importantes pour l'analyse de données temporelles quantitatives. Tous nos travaux sur la conception et le développement des techniques de visualisation découlent donc de cette étude.

Les tâches utilisateur de manipulation de données temporelles ne peuvent pas directement servir de guide à la conception des techniques de visualisation. Une phase de traduction de ces tâches de manipulation de données en tâches concrètes d'interaction (sélectionner des éléments de l'espace de représentation, faire un zoom, etc.) est nécessaire. Des listes de tâches concrètes d'interaction ont été proposées dans [Shneiderman 96, Kerpedjiev 97]. Pour réaliser cette phase de traduction, nous nous sommes appuyés sur les propriétés ergonomiques. L'objectif de notre étude est d'utiliser ces propriétés comme guide dans la conception des techniques de visualisation de données temporelles. A chaque tâche utilisateur identifiée, nous associons un ensemble de propriétés ergonomiques [Daassi 99a]. Les tâches utilisateur se traduisent ainsi au niveau de l'interface par des propriétés ergonomiques à vérifier, qui elles-mêmes sont définies pour qualifier l'utilisabilité d'un système. L'utilisabilité d'un système caractérise sa capacité à permettre à l'utilisateur d'atteindre ses objectifs (obtenir un résultat ayant une qualité donnée) avec efficacité et confort et en toute sécurité. Plusieurs listes de propriétés ergonomiques ont été proposées dans la littérature [Bass 01, Bastien 93]. Dans ce manuscrit, nous utilisons les propriétés ergonomiques définies dans [Abowd 92, Gram 98], réparties en deux catégories : *les propriétés de souplesse* et *les propriétés de robustesse*. Ces propriétés ergonomiques sont détaillées dans l'Annexe A, page 189.

Dans la suite, nous définissons chaque tâche identifiée et nous présentons les propriétés ergonomiques correspondantes. Nous répartissons les tâches utilisateur de manipulation de données temporelles en deux catégories : des tâches liées à l'aspect interactif de l'interface et des tâches liées à l'analyse des données.

## **2.1 Tâches d'interaction avec la source de données**

**Commencer par une requête générale.** Dans la perspective de découvrir et même de connaître les données à analyser, les utilisateurs commencent par une requête générale qu'ils vont affiner progressivement en agissant sur le résultat de la requête. Ceci établit une boucle d'interaction entre l'utilisateur et la source de données (base de données, fichiers XML, fichiers textes, etc.). La visualisation du résultat de la requête n'est pas effectuée une fois pour

toute, elle se “ construit ” et se “ reconstruit ” jusqu’à ce que toutes les relations entre les aspects qu’elle représente soient perçues [Bertin 81]. Du point de vue du système, nous distinguons deux types d’interaction selon qu’elles concernent ou pas la source de données :

- Des manipulations sont gérées localement dans l’interface, sans faire appel à l’exécution d’une nouvelle requête sur la source. Les interactions consistent par exemple, à agir sur les données visualisées afin de les réarranger dans l’espace de représentation.
- A l’opposé, certaines manipulations aboutissent à la spécification d’une nouvelle requête. C’est par exemple le cas d’une mise à jour d’une donnée à partir de sa représentation, ou encore la soumission d’un calcul impliquant des données non visualisées.

**Spécifier une nouvelle requête.** L’interprétation du résultat d’une requête peut aussi impliquer la spécification d’une autre requête qui n’est pas forcément un affinement de la première. Ce comportement s’inscrit dans la démarche globale de travail du géographe qui part souvent d’une requête très générale pour ensuite cerner son sujet d’étude. Par exemple, pour analyser la pollution atmosphérique, une première requête est formulée : quelles sont les concentrations de polluants plus grandes que la moyenne durant les dix dernières années ? Puis, après une analyse des résultats obtenus, l’utilisateur se concentre sur un polluant particulier, le NO par exemple : quel est l’historique des valeurs de NO observées durant les trois dernières années à l’échelle du jour ?

Les deux tâches *commencer par une requête générale* et *spécifier une nouvelle requête*, se traduisent par les mêmes propriétés ergonomiques. Pour répondre à ces deux tâches, nous devons garantir la propriété ergonomique *réutilisation des données d’entrées et de sorties*. Le résultat (donnée de sortie) de la requête est utilisé comme donnée d’entrée pour affiner la requête. Ceci suppose que l’interface fournit tous les moyens pour assurer cette interaction, c’est-à-dire que l’utilisateur peut par exemple sélectionner un sous-ensemble des données présentées graphiquement au niveau de l’interface, et leur appliquer un opérateur temporel (agrégation, changement d’unité, etc.). L’utilisateur obtient alors le résultat de cette nouvelle requête selon une nouvelle présentation. Ce même résultat peut à son tour représenter une donnée d’entrée à une troisième requête plus précise.



Lors de l'interaction avec la source de données, l'utilisateur doit pouvoir anticiper ses actions de manipulation de l'interface pour affiner sa requête : *la prévisibilité*. L'utilisateur construit une représentation mentale de ce qu'il va obtenir comme résultat de ses actions. Le fait de garantir cette propriété permet de réduire le taux d'erreurs commises par l'utilisateur au cours de son interaction. De plus, il doit avoir la possibilité de corriger une situation non désirée suite à une erreur qu'il a commise au cours de son interaction, et revenir à un état antérieur : c'est la propriété de *curabilité*. La *curabilité* implique l'annulation de l'effet de l'action dans le but de corriger l'état obtenu. Par exemple, l'utilisateur peut sélectionner un sous-ensemble de données autre que celui qu'il voulait effectivement sélectionner initialement. Il doit avoir la possibilité de revenir à la présentation à partir de laquelle il a sélectionné les données non désirées.

## **2.2 Tâches d'analyse de données**

Les tâches d'analyse de données sont des tâches que l'utilisateur effectue pour extraire des informations pertinentes à partir des données. Nous classons ces tâches en quatre catégories : tâches de navigation, tâches d'observation, tâches de comparaison et tâches de manipulation.

### **2.2.1 Tâches de navigation**

La Figure 69 présente deux dimensions du modèle d'exploration d'un espace d'informations proposé par Waterworth et Chignell [Waterworth 91]. Dans l'axe vertical, ce modèle distingue les rôles respectifs de l'utilisateur et du système. Dans l'axe horizontal, il modélise l'objectif de la navigation en fonction de la cible que l'acteur (utilisateur ou système) a fixée. Nous retenons ici seulement l'utilisateur comme acteur de la tâche de navigation.

Dans le cadre de notre étude, nous considérons les deux objectifs (chercher et feuilleter) du modèle d'exploration d'un espace d'informations de la Figure 69. Nous associons les deux objectifs de la tâche de navigation aux deux dimensions d'une donnée temporelle. Ainsi, chercher (respectivement feuilleter) peut concerner une valeur temporelle ou une valeur structurelle selon que l'utilisateur s'intéresse à la dimension temporelle ou à la dimension structurelle d'une donnée.

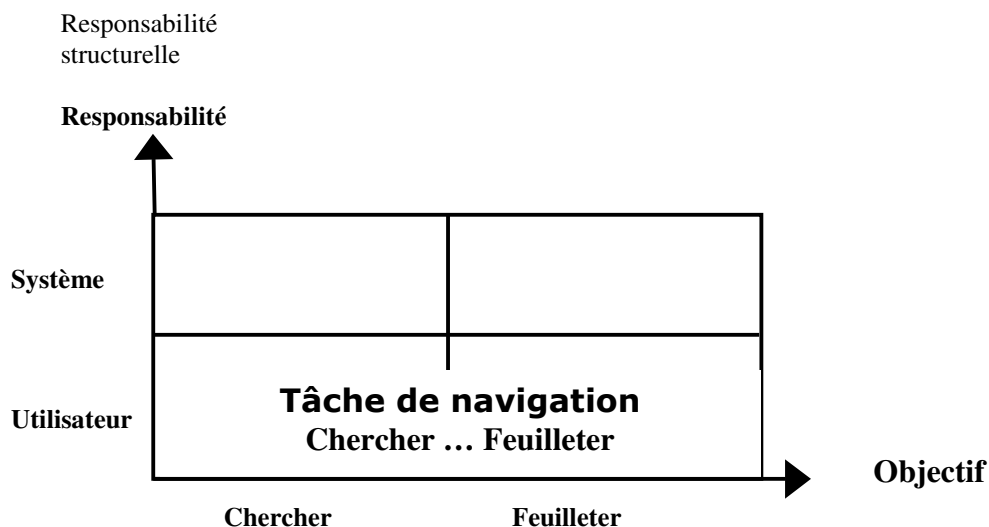


Figure 69 : Deux dimensions du modèle d'exploration d'un espace d'informations [Waterworth 91].

Nous estimons qu'il est important que l'utilisateur puisse anticiper ses actions de navigation dans l'espace d'informations afin de mieux le maîtriser : c'est la propriété de *prévisibilité*. Si l'utilisateur se trouve perdu, il faut lui permettre de revenir à un endroit connu de cet espace à partir duquel il peut reprendre la navigation pour atteindre son but. Il s'agit ici de garantir la propriété de *curabilité* qui caractérise la capacité pour l'utilisateur de corriger une situation non désirée et revenir à un état antérieur.

Nous considérons la navigation définie sous le nom de fortement couplé comme l'a défini Vernier [Vernier 01] : "Un outil de navigation est fortement couplé lorsque tous les événements en entrée sont répercutés sur la visualisation et que l'utilisateur utilise ces modifications dans la visualisation pour produire les événements en entrée." Extrait de [Vernier 01]. L'interface doit donc garantir en plus la propriété de *non-préemption*, c'est-à-dire permettre à l'utilisateur d'atteindre facilement son prochain but. Le but ici est de pouvoir atteindre n'importe quel point de l'espace de données avec un retour d'information immédiat sur les actions de l'utilisateur.

## 2.2.2 Tâches d'observation

**Observer les données dans le temps.** L'utilisateur est parfois intéressé par les valeurs exactes des données alors que dans d'autres situations, il souhaite observer les tendances de l'évolution de la donnée dans le temps. Par exemple, l'utilisateur veut anticiper le futur à partir de l'observation de l'historique de l'évolution des données dans le temps.

Nous devons distinguer ici l'observation de valeurs toutes perceptibles (car peu nombreuses) de la visualisation d'un grand espace d'informations. Dans les deux cas, l'interface doit vérifier les trois propriétés d'ergonomie suivantes : *observabilité*, *insistance* et *honnêteté*.

➤ **Observer les valeurs des données**

L'*observabilité* s'applique à :

- Une représentation des valeurs des données,
- Le type des valeurs visualisées (type ordonné ou non, valeurs mesurables ou non), et
- La taille du domaine temporel.

L'*insistance* consiste ici en la représentation des valeurs de la donnée de manière à donner à l'utilisateur une vue générale ou précise de l'évolution de la donnée dans le temps. La technique utilisée peut par exemple remplacer les valeurs des données mesurables par des objets graphiques ou des variables visuelles.

La propriété d'*honnêteté* caractérise les représentations qui permettent à l'utilisateur une perception correcte des données visualisées. Par exemple, assurer que la taille des objets graphiques soit proportionnelle aux valeurs qu'ils représentent. De plus, il ne faut pas confondre la variation de la représentation avec la variation des données dans la représentation, car ceci conduit à une ambiguïté [Tufte 83]. Les techniques utilisées pour visualiser l'évolution des données ne doivent pas utiliser de fonctions de déformation, car la déformation porte sur la taille des objets représentant les valeurs. En particulier de grandes valeurs peuvent être représentées par des objets de petite taille suite à la déformation, ce qui peut induire l'utilisateur en erreur et le conduire à une mauvaise interprétation de l'état du système.

➤ **Visualiser un grand espace d'informations**

L'*observabilité* s'applique :

- Au centre d'intérêt,
- A l'espace d'informations, et
- A la taille de l'espace d'informations.

Dans le cas de la visualisation d'une seule donnée, la taille de l'espace d'informations est égale à celle du domaine temporel. Dans le cas contraire, c'est-à-dire lorsqu'il s'agit d'un ensemble de données temporelles, la taille de l'espace d'informations est égale au nombre de ces données, et chacune possède un domaine temporel qui lui est propre. La taille du domaine temporel est égale au nombre d'instantanés estampillant les valeurs structurelles.

Dans ce contexte, l'*honnêteté* consiste à définir une représentation exacte de la taille de l'espace d'informations. Si par exemple, chaque élément du résultat est représenté par un objet graphique, le nombre d'objets graphiques représentés au niveau de l'interface doit être égal au nombre des éléments de l'espace de données.

**Etudier la régression.** La régression consiste en la prédiction de la valeur d'une donnée continue en se basant sur les valeurs d'autres données, indépendamment du fait que la dépendance entre ces données soit linéaire ou non. Il s'agit par exemple de prévoir la vitesse par laquelle tombe la neige en fonction de la température, de l'humidité et la pression atmosphérique.

L'événement à prédire est dit événement cible. Le problème de la prédiction est de connaître une procédure P qui prédit correctement l'événement cible. Donc P est une fonction qui à partir d'un ensemble d'événements E1...En prédit l'événement cible. Dans le cadre de notre travail, nous ne visons pas à définir la procédure P, mais à proposer un support visuel pour aider à la réalisation de la tâche. Nous considérons qu'à partir d'une représentation bien appropriée aux différentes données, l'utilisateur pourra identifier visuellement les relations entre elles. Par la suite, il pourra voir s'il peut prédire une valeur passée à partir des valeurs antérieures d'autres données. Si c'est le cas, il sera probablement possible de prédire les valeurs futures à partir des valeurs courantes.

Les propriétés ergonomiques à vérifier pour cette tâche sont les mêmes que pour la tâche d'observation des données dans le temps, c'est-à-dire *observabilité*, *insistance* et *honnêteté*.

**Rechercher des motifs de comportement des données.** Un motif ou un schéma de comportement est une séquence particulière de valeurs successives d'une donnée temporelle. La plupart des motifs peuvent être décrits en termes de deux classes : tendance et saisonnalité [StatSoft 03]. La première classe représente un composant général qui change dans le temps

et qui ne se répète pas. La seconde classe est de même nature mais se répète dans le temps. La Figure 70 donne une représentation d'une donnée temporelle sous la forme d'une courbe. Celle-ci permet la mise en évidence d'un motif pendant la période A, qui se répète pendant les périodes B et C. L'identification de ces motifs repose sur l'observation du comportement général des données et non pas sur les valeurs exactes des données.

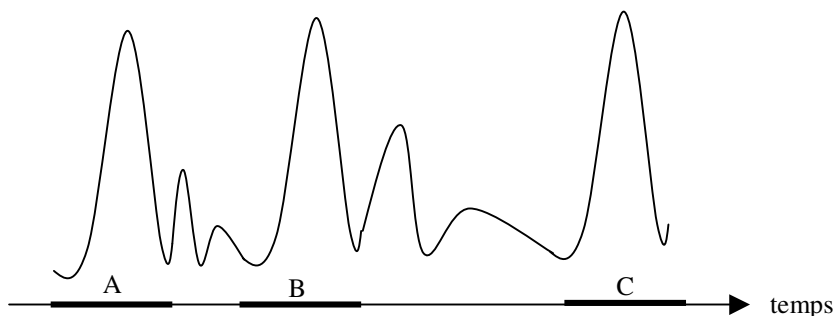


Figure 70 : Motif de comportement des données.

Des travaux de recherche en base de données temporelles [Agrawal 95b, Dumas 98, Dumas 00] ou en fouille de données [Srikant 96, Tawfik 00] ont été proposés pour identifier automatiquement des motifs. Par opposition à l'approche où le motif à chercher n'est pas reconnu à l'avance, dans Tempos [Dumas 00] par exemple, c'est l'utilisateur qui doit le spécifier, et l'outil en recherche les occurrences. Contrairement à ces deux approches, nous considérons ici la reconnaissance visuelle de motifs.

Nous caractérisons cette tâche avec les mêmes propriétés ergonomiques que celles associées à la tâche d'observation des données dans le temps, à savoir *observabilité, insistance et honnêteté*.

Pour identifier les motifs, l'utilisateur doit pouvoir faire une copie d'une partie de l'objet graphique (la courbe par exemple) qu'il pense être un motif de comportement. Il doit pouvoir par la suite la superposer à plusieurs niveaux de la représentation afin de la comparer à d'autres parties et décider s'il s'agit d'un motif de comportement ou non. L'interface doit donc garantir la propriété de *non-préemption*. Le but ici est de pouvoir faire une copie d'une partie de la représentation et la superposer à différents niveaux de cette représentation.

**Détecter des points de rupture.** Après avoir observé des séries de données similaires, un élément introduisant une distorsion du comportement normal des données est dit point de

rupture, exception, point de changement ou déviation. Les points de rupture amènent à une dissimilarité entre les données [Arning 96, Weiss 98]. L'une des tâches utilisateur est d'identifier ces points de rupture et les traiter. Il s'agira par exemple de supprimer les points de rupture ou de les modifier afin de traduire une évolution " souple " des données.

Nous ne nous intéressons qu'à la détection des points de rupture au niveau du comportement des données dans le temps. Pour se faire, l'utilisateur doit pouvoir correctement observer l'espace de données. Nous caractérisons cette tâche avec *l'observabilité, l'insistance et l'honnêteté*. Ces propriétés garantissent que les représentations reflètent le plus fidèlement les valeurs des données. *L'observabilité* est définie au travers d'une représentation des valeurs des données. L'utilisateur peut ainsi détecter les déviations s'il en existe.

**Etudier la répartition des données (identifier des points de concentrations)** . L'étude de la répartition des valeurs des données dans le temps permet d'identifier des concentrations de valeurs (ou de classes de valeurs) à des instants donnés. Cette tâche consiste à étudier les relations entre les instants ou périodes correspondant à ces points de concentration. L'utilisateur veut par exemple connaître la concentration du NO<sub>2</sub> par jour. La répartition des valeurs dans l'espace de représentation peut indiquer des points de concentration et révéler ainsi que chaque jour entre 7h et 10h, la concentration est la plus élevée.

L'étude de la répartition des données passe par l'observation de l'espace de données. Nous caractérisons cette tâche par les mêmes propriétés ergonomiques que la tâche d'observation des données dans le temps.

**Regrouper les données par catégories.** Il s'agit ici de regrouper des données similaires dans une même catégorie. Le regroupement facilite l'analyse et la comparaison entre les évolutions des données inter-catégories. L'utilisateur souhaite par exemple étudier l'évolution du salaire de deux employés ayant le même grade. Plus qu'un simple déplacement, l'utilisateur a donc besoin de réorganiser toute la représentation par catégorie, par exemple le grade de l'employé.

Pour répondre à cette tâche, nous caractérisons *l'observabilité* des variables suivantes :

- L'espace d'informations : l'ensemble des catégories,
- La taille de l'espace d'informations : le nombre de catégories, et

- Le centre d'intérêt : une catégorie particulière.

Pour pouvoir regrouper les données en se basant sur leurs représentations graphiques, il faut garantir la propriété d'*honnêteté*. L'*insistance* consiste ici en l'utilisation de variables visuelles telle que la couleur pour distinguer les catégories les unes des autres.

### 2.2.3 Tâches de comparaison

**Comparer des éléments de l'espace de données.** La comparaison de deux éléments est une tâche fréquemment observée. Pour cela, l'utilisateur souhaite pouvoir déplacer ou permuter des éléments visuels de l'espace de données afin de les rapprocher dans l'espace de représentation pour faciliter leur comparaison. Il veut par exemple permuter deux courbes ou les superposer pour comparer les évolutions qu'elles représentent.

La permutation est un réarrangement des éléments de l'espace de données dans un nouvel ordre. Une approche d'analyse de données basée sur la permutation des éléments a été développée dans [Bertin 81] : la permutation permet de créer des regroupements d'éléments visuels qui peuvent être considérés comme classes ou modèles de comportement des données.

Pour obtenir des conclusions correctes suite à une comparaison des éléments de l'espace de données, il faut que la représentation de ces éléments soit cohérente avec l'état interne du système, et qu'elle puisse engendrer chez l'utilisateur une interprétation conforme à cet état : la propriété d'*honnêteté*. Pour le cas de la représentation des données par des objets graphiques, nous pouvons mesurer l'honnêteté par le calcul du facteur de distorsion défini par Tufte [Tufte 83] :

$$\text{Facteur de distorsion} = \frac{\text{taille de l'effet présenté dans le graphique}}{\text{taille de l'effet au niveau des données}}$$

Si le facteur de distorsion est égal à 1, la représentation graphique est honnête. Lorsque ce facteur est supérieur à 1.05 ou inférieur à 0.95, la représentation graphique est malhonnête et risque de conduire à une interprétation fautive de la part de l'utilisateur.

Pour mener à bien cette comparaison, l'interface doit forcer la perception de l'utilisateur de cet état : la propriété d'*insistance* doit donc aussi être vérifiée. Ceci peut se faire à l'aide de

variables visuelles. De plus, la comparaison est plus efficace si elle est faite entre deux éléments situés l'un à côté de l'autre. Pour ceci, les utilisateurs souhaitent pouvoir permuter ou déplacer des éléments afin d'établir des rapprochements. La propriété de *non-préemption* caractérise cette possibilité

Pour répondre à cette tâche, les techniques utilisées doivent permettre une indépendance des éléments au niveau de leurs représentations, c'est-à-dire que l'utilisateur doit facilement parvenir à distinguer les différents éléments du résultat de la requête.

**Etudier la corrélation de l'évolution des données.** La corrélation consiste en l'étude du comportement de plusieurs données aux mêmes moments. L'utilisateur veut par exemple étudier la corrélation de l'évolution du NO<sub>2</sub> avec l'évolution du NO relevées en Janvier 1999.

Nous caractérisons la tâche de corrélation de données par les propriétés *observabilité*, *insistance* et *honnêteté*. L'espace de données peut être formé de plusieurs données temporelles. L'utilisation de deux techniques de visualisation pour observer deux données n'est pas satisfaisante dans la perspective d'étudier la corrélation entre ces deux données. Dans ce cas, nous caractérisons l'*observabilité* au travers des variables suivantes :

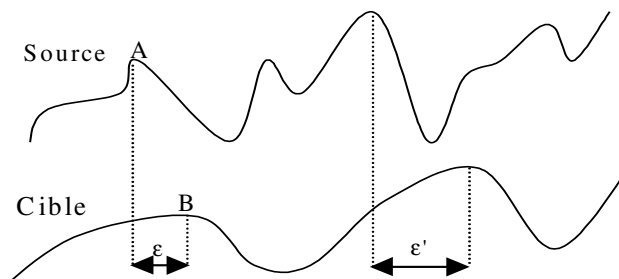
- L'axe temporel, et
- Les valeurs des données ou leurs représentations.

L'*insistance* consiste ici en la mise en correspondance des valeurs des données par rapport aux mêmes valeurs temporelles. La technique utilisée peut par exemple exploiter un seul axe temporel autour duquel sont placées les valeurs des deux données. L'*honnêteté* consiste par exemple en l'utilisation d'une même échelle pour représenter les valeurs des deux données.

**Rechercher des effets de causalité entre données.** L'utilisateur étudie ici l'influence du comportement d'une donnée particulière sur une autre donnée. Il s'agit de mettre en évidence les relations de causalité entre le changement de valeur (un événement) d'une donnée (la source) et le changement de valeur (une réaction) d'une autre donnée (la cible). Dans le cas où un effet de causalité est identifié, il convient alors d'étudier le temps (le délai  $\epsilon$ ) de réaction de la donnée cible suite à un événement de la donnée source. Comme le montre la Figure 71, un événement B apparaît en réaction à un événement A (source) avec un délai  $\epsilon$ . Il convient de



noter, que deux données peuvent être associées sans avoir une relation de causalité qui les relie.



*Figure 71 : Effet de causalité entre données.*

Nous caractérisons cette tâche par les mêmes propriétés ergonomiques que celles associées à la tâche "étude de la corrélation" puisqu'il s'agit de représenter deux données par rapport aux mêmes valeurs temporelles. L'*observabilité* sera caractérisée de plus par le délai  $\epsilon$ . Pour observer le délai  $\epsilon$  de réaction de la cible suite à un changement de valeur de la source, l'utilisateur a besoin d'un moyen qui lui permette de faire une projection de l'instant de changement de la cible sur la représentation de la source et sur l'axe temps. Ceci peut être une ligne en pointillés comme indiqué à la Figure 71. La perception du délai  $\epsilon$  sera donc renforcée au niveau de la représentation : cela contribue à assurer la propriété d'*insistance*.

**Mesurer la similarité entre données temporelles.** Comme son nom l'indique, il s'agit d'étudier la mesure de similarité entre deux ou plusieurs données temporelles. Pour le cas des données continues, les mesures de similarité sont basées sur des distances euclidiennes. Ceci augmente la difficulté de cette tâche (étude de similarité) du fait de la différence d'échelle d'observation des données temporelles. De point de vue fouille de données, deux données temporelles sont dites similaires si elles ont suffisamment de sous séquences ordonnées dans le temps qui sont similaires [Agrawal 95a, Antunes 01]. Une représentation appropriée de plusieurs données temporelles facilite l'identification visuelle des sous-séquences similaires.

Nous caractérisons cette tâche par les mêmes propriétés ergonomiques que la tâche de "comparaison des éléments de l'espace de données". En effet, cette tâche est équivalente à la tâche de comparaison mais appliquée à deux ou plusieurs données au lieu des éléments d'un espace d'une même donnée. L'*observabilité* de l'espace d'informations est ainsi nécessaire pour réaliser cette tâche.

**Disposer d'un référentiel.** L'utilisateur souhaite parfois disposer d'un référentiel explicite dans la représentation, lui permettant de comparer l'évolution de la donnée par rapport à une valeur spécifique. S'il s'agit par exemple de comparer l'évolution de la concentration du polluant NO<sub>2</sub>, un référentiel pertinent est la moyenne de ces concentrations sur la période considérée. Le référentiel peut représenter la valeur considérée comme "normale" et donc pouvoir situer les valeurs observées par rapport à cette normale.

Nous traduisons cette tâche en terme *d'observabilité et d'insistance*. Ces deux propriétés sont définies à travers le référentiel lui-même. Ce dernier doit être observable et différentiable de toute autre représentation dans l'espace de représentation du résultat.

#### **2.2.4 Tâches de manipulation**

**Obtenir le détail des valeurs agrégées.** Il faut pouvoir passer des valeurs agrégées à l'ensemble des valeurs qui ont permis de les obtenir. Par exemple, à partir de la visualisation des données à l'échelle du mois, quelle est la concentration du polluant NO<sub>2</sub> chaque jour ? L'utilisateur peut sélectionner un mois donné et obtenir le détail de tous les jours de ce mois.

Pour obtenir le détail des valeurs agrégées, l'utilisateur doit pouvoir sélectionner l'objet graphique traduisant la valeur d'une donnée observée à une unité d'observation donnée, l'unité mois par exemple, et pouvoir par la suite avoir le détail des valeurs observées dans une unité plus fine, l'unité jour par exemple. L'interface doit donc garantir la propriété de *non-préemption*. Le but ici correspond à la tâche elle-même qui consiste obtenir le détail des valeurs agrégées.

**Segmenter les données temporelles.** La segmentation d'une donnée temporelle consiste en sa partition en des segments linéaires, chacun ayant un poids représentant son importance [Keogh 98]. Nous considérons ici l'identification visuelle des partitions, leur nombre et les positions des points de changements.

Pour pouvoir segmenter une donnée temporelle, l'utilisateur a besoin d'une vision complète de l'espace d'informations pour détecter les sous-ensembles de données. La propriété *d'observabilité* doit donc être garantie. De plus, *l'honnêteté et l'instance* doivent être garanties pour renforcer la perception de l'utilisateur des sous-séquences homogènes de l'espace de données.

Pour segmenter une donnée temporelle, l'utilisateur doit pouvoir sélectionner des parties des séquences de la représentation graphique (la courbe par exemple) qu'il pense être une sous-séquence homogène de l'espace de données. L'interface doit donc garantir la propriété de *non-préemption*. Le but ici est de pouvoir sélectionner les différentes parties de la représentation.

## **2.3 Synthèse**

Dans ce paragraphe, nous avons présenté notre approche de conception des techniques de visualisation de données temporelles centrée sur les tâches utilisateur. Les tâches de manipulation de données temporelles ne peuvent pas directement servir de guide à la conception des techniques de visualisation. Une phase de traduction de ces tâches en tâches concrètes d'interaction (sélectionner des éléments de l'espace de représentation, faire un zoom, etc.) est nécessaire. Notre approche consiste à traduire les tâches utilisateur en propriétés ergonomiques que l'interface doit respecter. Les résultats obtenus tout au long de ce travail ont été expérimentés avec des utilisateurs géographes.

Dans la suite, nous appliquons cette approche à la conception des techniques de visualisation de données temporelles. Comme le montre la Figure 67 présentée à la page 91, le processus de visualisation de la dimension temporelle est achevé indépendamment du processus de visualisation de la dimension structurelle. Nous présentons dans un premier temps le composant logiciel conçu pour la visualisation de la dimension temporelle. Dans un deuxième temps, nous présentons cinq techniques de visualisation de données temporelles. Nous les situons par rapport aux tâches utilisateur en terme de manipulation de données temporelles.

## **3 Processus de conception de l'espace temps**

Nous consacrons ce paragraphe à l'étude et la conception de l'espace de représentation de la dimension temporelle d'une donnée. Nous présentons tout d'abord une synthèse des travaux développés dans la littérature sur la conception des lignes temps. Par la suite, nous présentons la ligne temps que nous proposons pour gérer la dimension temporelle d'une donnée. Nous présentons une évaluation de cette ligne temps au Chapitre 5.

### 3.1 Lignes temps proposées dans la littérature

Une revue des travaux sur la conception des lignes temps, montre que le problème de gestion d'un grand espace d'informations temporelles est le plus souvent considéré pour choisir la forme de représentation de la ligne temps. En effet, le domaine temporel (nombre d'instant d'observation) d'une donnée temporelle peut être d'une grande taille. Ainsi, plusieurs approches ont été développées pour résoudre le problème de gestion d'un grand espace de données. Pour le cas du temps, nous distinguons quatre cas :

**Distribution des valeurs temporelles.** La plupart des lignes temps développées dans la littérature utilisent une distribution linéaire des valeurs temporelles. La ligne temps de la Figure 72 utilise le principe d'une représentation de vue en œil de poisson. Elle utilise deux types de distribution des valeurs temporelles. Les valeurs sont linéairement visualisées au niveau de la partie centrale de la ligne temps : l'échelle d'avancement dans le temps est égale à une unité (un Jour dans la Figure 72). Aux deux extrémités, la distribution est non-linéaire : l'échelle d'avancement est égale à deux unités dans la Figure 72. La distribution non-linéaire permet de visualiser un grand nombre de valeurs temporelles, mais ne résout pas le problème de gestion d'un grand espace d'informations temporelles. Deux flèches sont ainsi utilisées pour naviguer dans le temps.

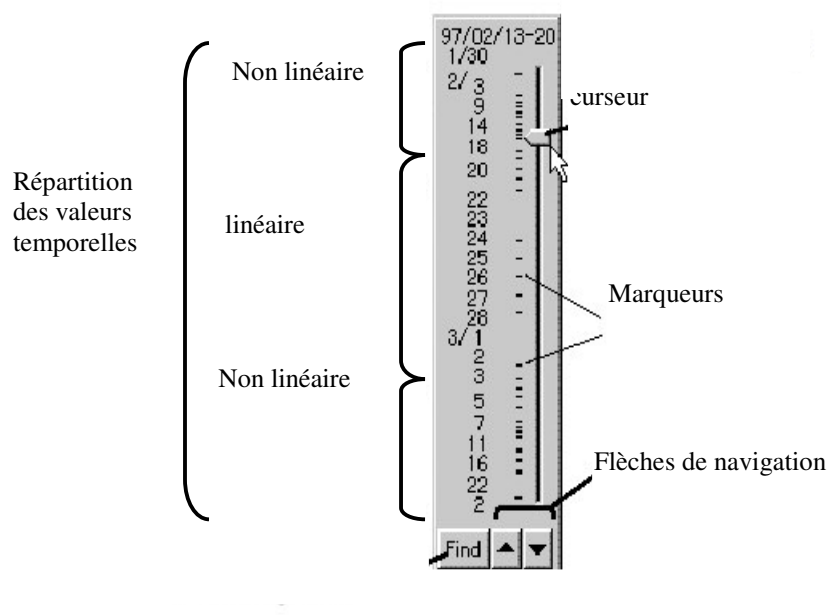


Figure 72 : Timeslider : une interface pour spécifier les points temps [Koike 97].

**Utilisation de la relation de composition entre unités temporelles.** L'utilisation de plusieurs axes chacun dédié à gérer les valeurs d'une unité temporelle permet de résoudre le problème de gestion d'un grand espace d'informations temporelles. Dumas [Dumas 00] a proposé cette approche pour représenter la dimension temporelle des données. La Figure 73 présente une ligne temps avec deux représentations complémentaires des valeurs temporelles. Trois axes sont utilisés pour représenter respectivement les années, les mois et les jours. La position du curseur au niveau de chacun de ces axes permet de fixer la valeur de l'unité d'observation correspondante. Une deuxième représentation du temps est utilisée. Elle sert à donner une représentation globale de l'espace temps en positionnant la date fixée par les valeurs définies au niveau des trois autres axes.

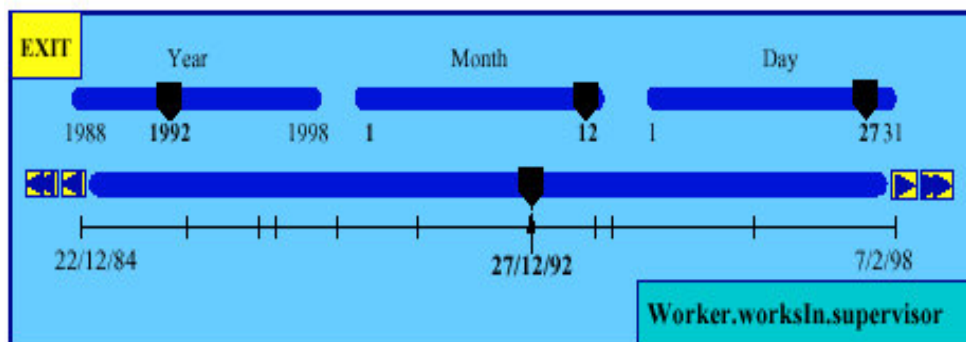


Figure 73 : Utilisation de plusieurs axes pour représenter le temps [Dumas 00].

**Représentation des instants pertinents.** La valeur structurelle d'une donnée temporelle peut être la même sur un ensemble de valeurs temporelles successives. Le salaire d'un employé, observé à l'échelle du mois, peut rester le même sur plusieurs années (historisation des valeurs des données étudiée au Chapitre 2). En se basant sur cette information, certaines lignes temps réduisent l'espace nécessaire pour représenter les valeurs temporelles en utilisant des signes graphiques indiquant que les valeurs des données restent sans modification pour tous les instants remplacés par le signe graphique. La ligne temps de la Figure 74 utilise la forme pointée par la flèche pour remplacer l'espace temps nécessaire pour représenter les valeurs temporelles observées à l'échelle du jour et définies entre le 15 Mars 1999 et le 19 Mars 2000.

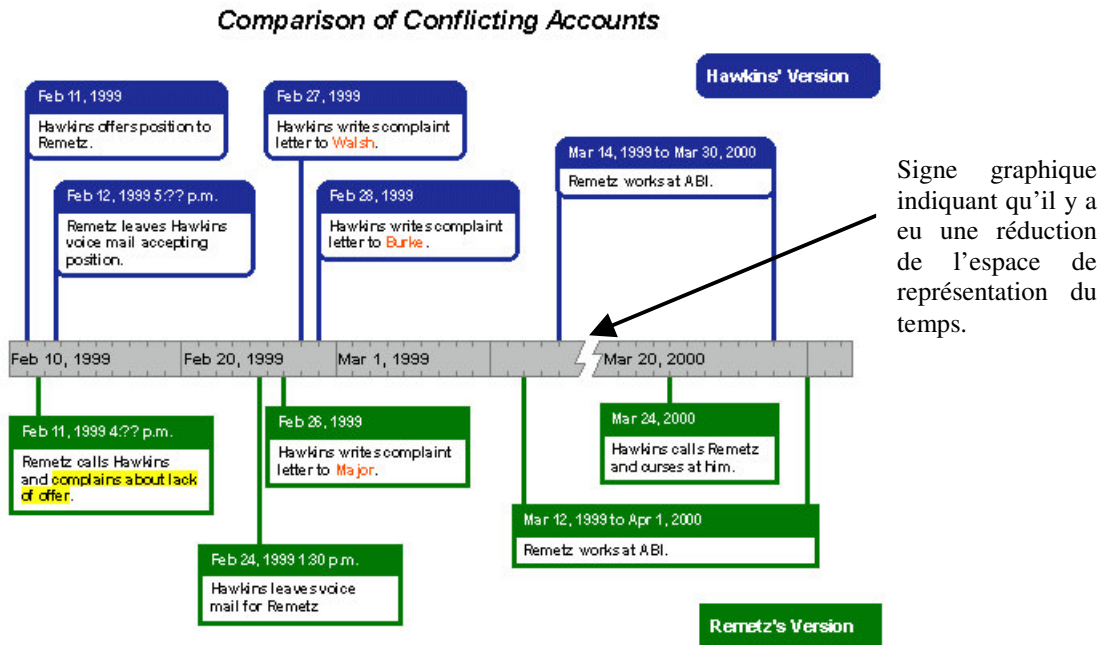


Figure 74 : Réduction de la représentation de l'espace temps [TimeMap 03].

Cette représentation du temps nécessite une connaissance préalable de la manière avec laquelle les valeurs de la dimension structurelle évoluent dans le temps. Ainsi, le processus de visualisation des valeurs de la dimension temporelle est fortement couplé à celui des valeurs de la dimension structurelle.

**Représentation par intervalles.** D'autres lignes temps utilisent une représentation par intervalles afin de ne représenter que les intervalles de temps pertinents par rapport à l'évolution de la donnée. Le temps est représenté en tant que succession d'intervalles de temps séparés par des points temporels ou des intervalles vides. La ligne temps de la Figure 75 utilise une représentation par intervalles. Ce type de représentation ne résout pas le problème de gestion d'un grand espace d'informations temporelles. Par exemple, dans le cas des intervalles de grande taille, ce type de représentation nécessite une gestion du grand espace d'informations intra-intervalles.

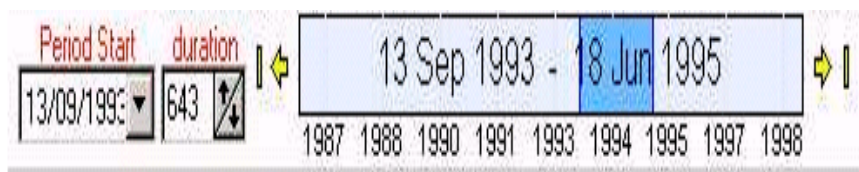


Figure 75 : Représentation par intervalles [Browser 03].

## 3.2 Vue en mur en perspective pour représenter le temps

Dans ce paragraphe, nous présentons la technique de visualisation conçue pour la dimension temporelle. Comme nous l'avons indiqué à l'introduction de ce chapitre, nous utilisons le processus de visualisation de Chi [Chi 00]. Concrètement le résultat est un composant logiciel réutilisable (JavaBean) présenté à la Figure 76. Nous nous limitons à l'étude de l'étape *point de vue sur l'espace temps* parce qu'elle représente le rendu perceptible par l'utilisateur. Nous l'avons souligné au paragraphe précédent, les lignes temps proposées dans la littérature sont conçues seulement en considérant le problème de gestion d'un grand espace d'informations temporelles. Pour réaliser l'étape *point de vue sur l'espace temps*, nous considérons les trois points suivants : le problème de gestion d'un grand espace d'informations temporelles, les caractéristiques du temps étudiées au Chapitre 2, et la tâche de navigation étudiée au paragraphe 2 de ce chapitre. Nous détaillons dans la suite chacun de ces points tout en justifiant nos choix de conception.

### 3.2.1 Gestion d'un grand espace d'informations temporelles

Pour traiter le problème de la grande taille de la dimension temporelle, nous avons choisi de représenter le temps sous la forme d'un mur en perspective [Mackinlay 91]. Comme le montre la Figure 76, la forme d'un mur en perspective distingue trois faces : une face en avant pour représenter les données d'une manière détaillée, et deux faces de côtés pour représenter l'effet en perspective. L'effet en perspective est utilisé pour souligner le fait que les données ne se limitent pas à celles représentées au niveau de la face avant. Les deux bornes de la ligne temps modélisent les points temporels de début et de fin (bornes du domaine temporel), l'utilisateur peut ainsi déduire le nombre d'éléments cachés par l'effet en perspective. L'instant courant correspond à celui défini par la position du curseur.

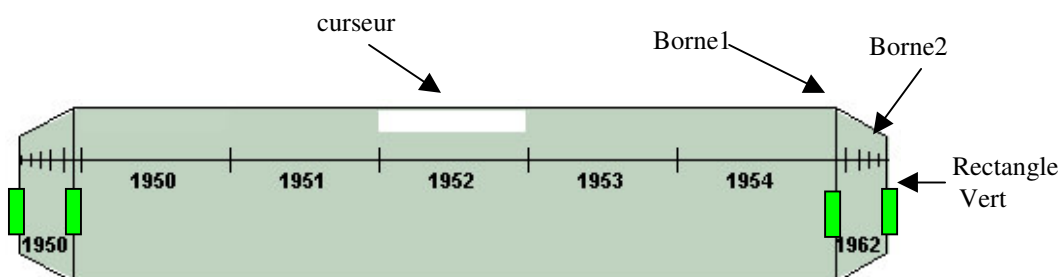


Figure 76 : Vue en mur en perspective pour représenter le temps.

L'utilisateur peut changer la taille de la ligne temps ainsi que le nombre d'éléments temporels visualisés dans la face avant. La forme en mur en perspective (Figure 76) est définie par deux types de bornes selon que ces derniers délimitent la face avant par rapport aux faces en perspectives (Borne1) ou encore délimitent le composant lui même (Borne2). En étirant la ligne temps de la borne Borne1, l'utilisateur peut augmenter ou réduire le nombre des éléments temporels affichés au niveau de la face avant en direction du sens de l'étirement. Si le nombre des éléments affichés est égal au nombre des instants à visualiser, seule la face avant du mur est affichée. De même, en étirant la borne Borne2, l'utilisateur peut augmenter ou diminuer la taille de la ligne temps. Ce dernier mode permet d'augmenter la visibilité d'affichage des valeurs temporelles au niveau de la ligne temps.

### **3.2.2 Caractéristiques du temps**

Nous avons conçu cette ligne temps en respectant les caractéristiques du temps étudiées au Chapitre 2, c'est-à-dire la relation d'ordre, la structure de l'espace temps et la granularité des éléments du temps.

**Relation d'ordre.** Le temps est ordonné. La ligne temps en mur en perspective respecte cette caractéristique des valeurs temporelles. Elle peut ainsi visualiser et gérer un espace de données ordonnées selon le prédicat d'ordre  $<$ . Comme nous le verrons avec la technique diagrammes superposés présentée au paragraphe 4.3, ce composant peut servir à visualiser les valeurs des données quantitatives ordonnées.

**Structure de l'espace temps.** Le temps est perçu selon différents modèles, en particulier le modèle linéaire et cyclique (Chapitre 3). L'utilisation de la forme linéaire pour la conception des lignes temps (composant réutilisable et indépendant de l'espace de représentation des valeurs structurelles) est la seule utilisée. C'est par exemple le cas des lignes temps présentée aux Figure 72, 73, 74 et 75.

En psychologie de la perception, le modèle linéaire et le modèle logarithmique sont combinés pour représenter le temps (Chapitre 2). La forme d'un mur en perspective peut être considérée comme une représentation visuelle du modèle logarithmique et linéaire du temps. Le temps est linéaire au niveau de la face avant du mur et il est logarithmique au niveau des deux faces en perspectives.



En synthèse, nous choisissons d'utiliser la forme de mur en perspective pour la conception de la ligne temps parce qu'elle cumule plusieurs avantages : elle combine à la fois le modèle linéaire et le modèle logarithmique de perception du temps, et elle gère mieux un grand espace de données.

**Granularité des éléments du temps.** Comme nous l'avons présenté dans le Chapitre 2, le temps peut être considéré comme continu ou discret (discontinu). Dans une vision continue, entre deux instants il existe toujours un autre instant. A cette perception s'oppose une autre, qui considère le temps comme discret, modélisé par des grains séparables.

La ligne temps que nous avons conçue implémente la vision discrète du temps. En effet, dès que nous nous intéressons à la représentation de phénomènes dans le monde informatique, la discrétisation est incontournable.

### **3.2.3 Tâches utilisateur se rapportant à la dimension temporelle**

La ligne temps implémente deux tâches utilisateur : la sélection de valeurs temporelles et la navigation dans l'espace temps.

**Sélection de valeurs temporelles.** L'utilisateur peut sélectionner une valeur temporelle par un simple double clique de la souris au niveau du point temporel correspondant. Les instants temporels sélectionnés sont représentés en bas de la ligne temps. Il peut les supprimer en double cliquant dessus.

**Navigation dans le temps.** La ligne temps offre deux modes de navigation pour explorer l'espace des valeurs temporelles. Dans le premier mode, l'utilisateur peut naviguer en déplaçant la souris dans l'espace de représentation de la ligne temps, et suivant la borne, il fait défiler les valeurs temporelles dans la direction indiquée par la borne en question. Ce mode est généralement utilisé lorsque l'utilisateur cherche une valeur temporelle lointaine car il permet des grands sauts temporels. Le second mode est la navigation point par point dans le temps. Ce mode correspond à un avancement pas par pas. Ceci est réalisé par un simple clique au niveau de la face en perspective selon la direction de navigation que l'utilisateur désire.

### **3.3 Dispositif physique pour l'interaction avec l'espace temps**

Le résultat du processus de visualisation de la dimension temporelle est la ligne temps, un composant Java Bean réutilisable, que nous avons présentée au paragraphe précédent. Bien que le processus de visualisation de la dimension temporelle soit indépendant de celui de la dimension structurelle, la réutilisation de la ligne temps reste encore limitée par la nature même du composant logiciel. En effet, le composant Java Bean est fortement dépendant de la plateforme de développement, ici le langage de programmation Java. L'objectif est de pouvoir réutiliser l'espace de représentation du temps indépendamment du processus de visualisation de la dimension structurelle et de la plate-forme de développement utilisée. Nous proposons de concrétiser l'idée d'un dispositif physique pour l'interaction avec l'espace des valeurs temporelles. La conception du dispositif physique lui-même sort du cadre de nos compétences. Ayant réalisé l'étape de conception et de développement du composant ligne temps, nous nous intéressons à l'étape suivante, c'est-à-dire la simulation d'un dispositif physique pour l'interaction avec l'espace temps.

La Figure 77 présente notre dispositif physique qui correspond à la ligne temps, définie au paragraphe précédent, associée à un ordinateur de poche de type iPaq. Comme le montre la Figure 78, un « pilote » joue le rôle de médiateur et assure la communication entre le dispositif physique et l'application qui l'utilise, ici une technique de visualisation de données temporelles. Le dispositif et le « pilote » communiquent à travers des sockets ce qui garantit l'indépendance du dispositif temps vis-à-vis de la plate-forme de développement des techniques de visualisation. En effet, les sockets sont utilisées par la plupart des langages de programmation (Java, C, C++, Ada, etc.). Le « pilote » permet d'interpréter les données provenant du dispositif temps et les traduire en événements utilisables par les techniques de visualisation. Ces événements correspondent par exemple aux cas où un nouvel élément temporel est ajouté au niveau du dispositif temps, le nombre des éléments temporels affichés au niveau de la face avant de la ligne temps a augmenté ou diminué, etc. Aussi, le « pilote » reçoit des données provenant des techniques de visualisation pour les transmettre au dispositif temps.

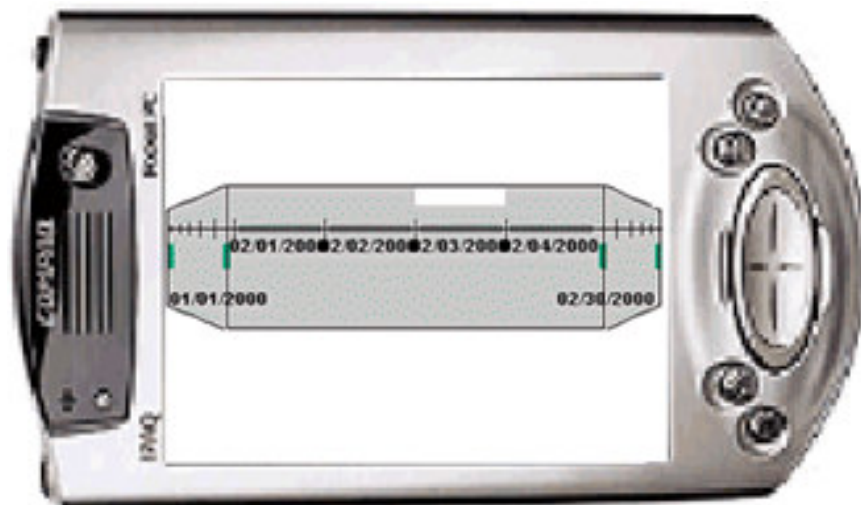


Figure 77 : Dispositif physique pour l'interaction avec l'espace temps.

En synthèse, la réalisation du dispositif physique pour l'interaction avec les valeurs temporelles est devenue possible puisque le composant ligne temps est conçu selon un processus complètement indépendant du processus de visualisation des valeurs structurales. Dans l'état actuel de notre travail, nous n'avons pas évalué l'apport de l'utilisation du dispositif par rapport à l'utilisation d'un composant logiciel. L'évaluation et l'utilisation de ce dispositif pour concevoir des techniques de visualisation sont des perspectives de cette thèse.

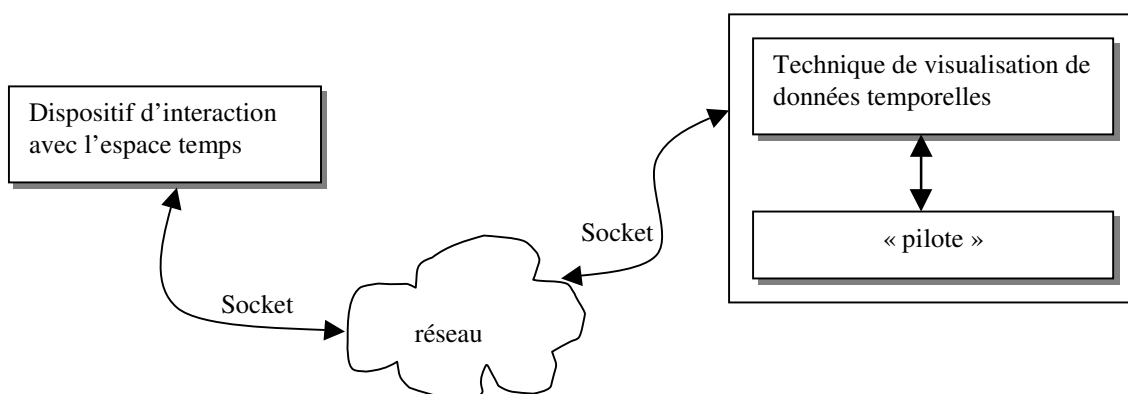


Figure 78 : Communication entre dispositif temps et une technique de visualisation.

Ayant présenté notre technique de visualisation de la dimension temporelle, nous présentons maintenant cinq techniques de visualisation de données temporelles que nous avons conçues.

## **4 Techniques de visualisation de données temporelles**

Nous rappelons que nous nous limitons à la visualisation de données temporelles quantitatives. Il s'agit de produire une ou plusieurs représentations graphiques traduisant l'évolution des données dans le temps par des objets graphiques comme un cercle, un segment ou un rectangle. Les techniques sont conçues pour assister les utilisateurs à réaliser leurs tâches de manipulation de données temporelles. Nous considérons les tâches utilisateur étudiées au paragraphe 2 de ce chapitre. Nous soulignons ici qu'une technique de visualisation ne peut pas répondre à toutes les tâches utilisateur. Plus elle répond à un grand nombre de tâches, certes plus elle est générale mais plus elle perd en performance. L'objectif est donc de concevoir plusieurs techniques de visualisation complémentaires et adaptées à un ensemble restreint de tâches utilisateur.

Dans la suite, nous présentons les cinq techniques de visualisation que nous avons conçues pour visualiser des données temporelles périodiques. Une période est une séquence d'instantants de taille fixe. L'unité d'observation de la dimension temporelle est donc régulière puisque les grains de la partition qui la définissent ont tous la même taille (paragraphe 3.2.2 du Chapitre 2). Chaque valeur de la dimension structurelle est ainsi définie par rapport à deux unités d'observations : une unité grossière qui correspond à la période elle-même et une unité fine qui correspond aux grains définissant la période. Par exemple, un jour est une période de 24 heures : l'unité grossière est le Jour et l'unité fine est l'Heure.

Pour chaque technique nous décrivons sa représentation graphique c'est-à-dire son apparence et son mode de fonctionnement, puis nous présentons le processus de visualisation par lequel elle a été conçue et les tâches utilisateur qu'elle satisfait. Notons que pour toutes ces techniques, le processus de visualisation de la dimension structurelle est indépendant de celui de la dimension temporelle comme indiqué à la Figure 67 (page 91).

### **4.1 Technique cercles concentriques**

#### **4.1.1 Représentation graphique**

La technique « *cercles concentriques* » (*TCC*) [Daassi 02] consiste en un ensemble de cercles de même centre. Chaque cercle correspond à une période particulière (un jour dans la Figure 79) et chaque rayon représenté sur les cercles correspond à un instant de cette période

(une heure dans l'exemple de la Figure 79). Les valeurs relatives à cette période sont représentées par des rectangles positionnés sur les cercles. La *TCC* a été conçue pour manipuler une ou deux données temporelles dont les valeurs structurelles sont quantitatives. Nous associons une couleur et une position (intérieure ou extérieure au cercle pour distinguer les données) aux rectangles correspondants aux valeurs.

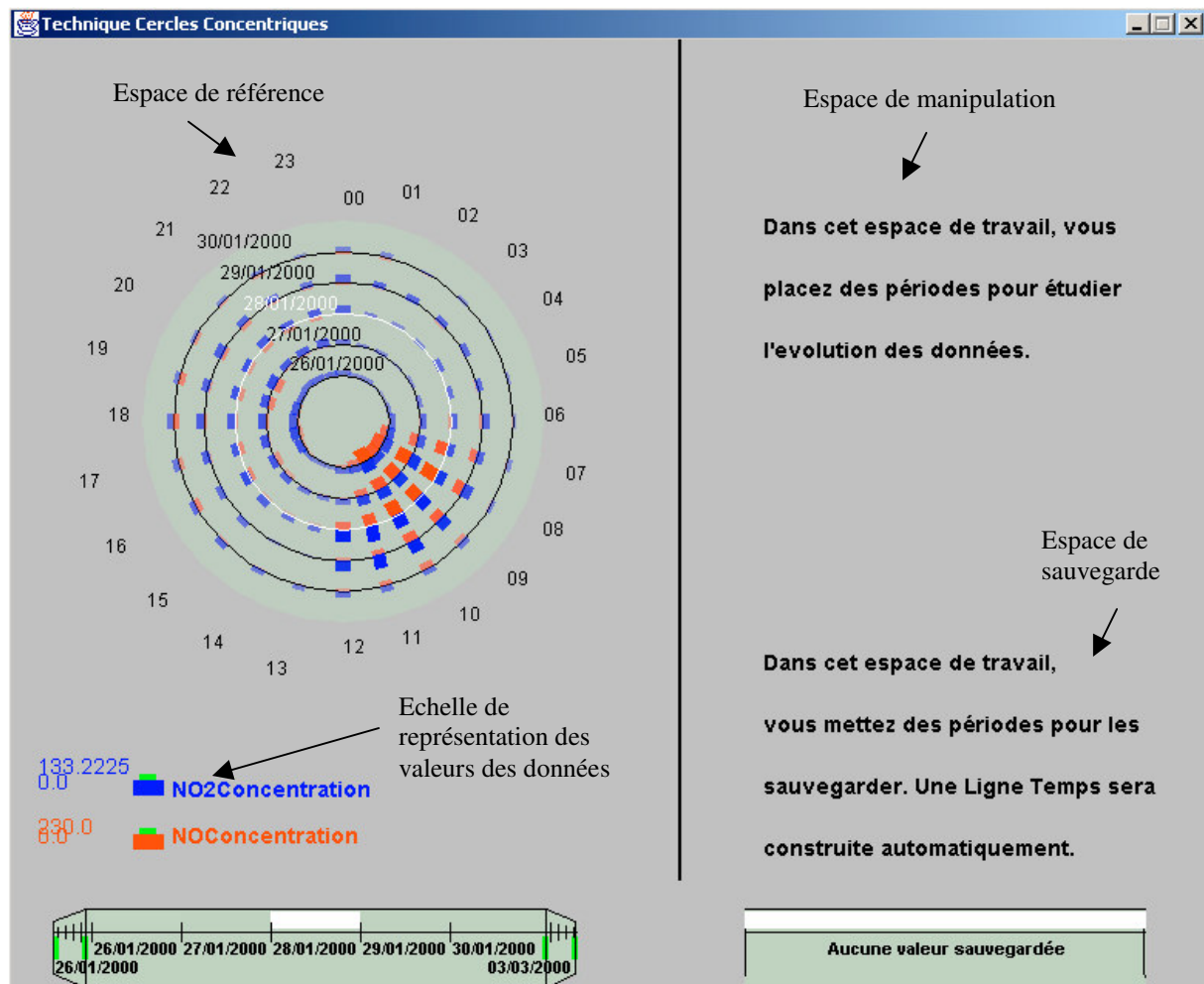


Figure 79 : Technique cercles concentriques (*TCC*) [Daassi 02].

La technique *cercles concentriques* dispose de deux espaces. A gauche dans la Figure 79, un espace de référence sert à visualiser les données en respectant l'ordre entre valeurs temporelles. Les données à visualiser avec la *TCC* peuvent représenter un grand espace d'informations. Une ligne temps permet donc de naviguer dans ce grand espace selon le temps. Les périodes visualisées dans l'espace de référence correspondent aux périodes affichées au niveau de la ligne temps. Ainsi l'utilisateur ne peut que naviguer dans cette ligne temps pour explorer l'espace de données. Le cercle correspondant à la période courante dans

la ligne temps (position du curseur) est affichée en blanc. De même, la sélection d'un nouveau cercle dans cet espace déplace le curseur dans la ligne temps à la position correspondante. Il s'agit ici d'un lien direct entre valeurs temporelles et valeurs structurelles.

La visualisation de données quantitatives nécessite l'utilisation d'une échelle de représentation. Les valeurs des données sont traduites en objets graphiques. L'utilisateur interprète ces objets pour en déduire (ou estimer) les valeurs correspondantes. Cette interprétation est réalisée en comparant les objets graphiques à l'échelle utilisée. Avec la technique *cercles concentriques*, les valeurs des données sont traduites en terme de rectangles dont la hauteur et/ ou la couleur sont proportionnelles aux valeurs correspondantes. La largeur des rectangles est fixe pour tous les rectangles. Elle est calculée en fonction du nombre de rectangles à afficher par cercle (la taille d'une période) et du périmètre du plus petit cercle. Comme le montre la Figure 80, les deux rectangles font références aux deux données visualisées en bleu et en rouge. Le nom de chaque donnée est affiché en face du rectangle correspondant. De même, la plus grande et la plus petite valeur de chacune des deux données sont représentées à gauche du rectangle correspondant. La hauteur du rectangle indique l'échelle de représentation des valeurs des données. Elle représente la plus grande valeur de la donnée.

A droite dans la Figure 80, l'espace de travail est subdivisé en deux parties. La première en haut correspond à un espace de manipulation de données. L'utilisateur ajoute à cet espace les périodes (cercles) sélectionnées à partir de l'espace de référence par un simple sélectionner-glisser-déposer. Il peut permuter ou déplacer des cercles pour faciliter leur comparaison. Il peut refaire ces actions de permutation ou annuler leur effet en utilisant les touches-clavier flèches haut et bas. Pour supprimer un cercle il convient de le sélectionner et cliquer sur le bouton supprimer.

La deuxième partie de cet espace de travail est un espace de sauvegarde des périodes sélectionnées par l'utilisateur. Pour se faire, il suffit de faire un sélectionner-glisser-déposer d'un cercle (période) à partir de l'espace de référence ou de l'espace de manipulation, et la placer dans l'espace de sauvegarde. Une ligne temps dans cette zone de sauvegarde est créée automatiquement pour permettre la navigation dans l'ensemble des périodes sauvegardées. Les données correspondantes sont sauvegardées dans un fichier.

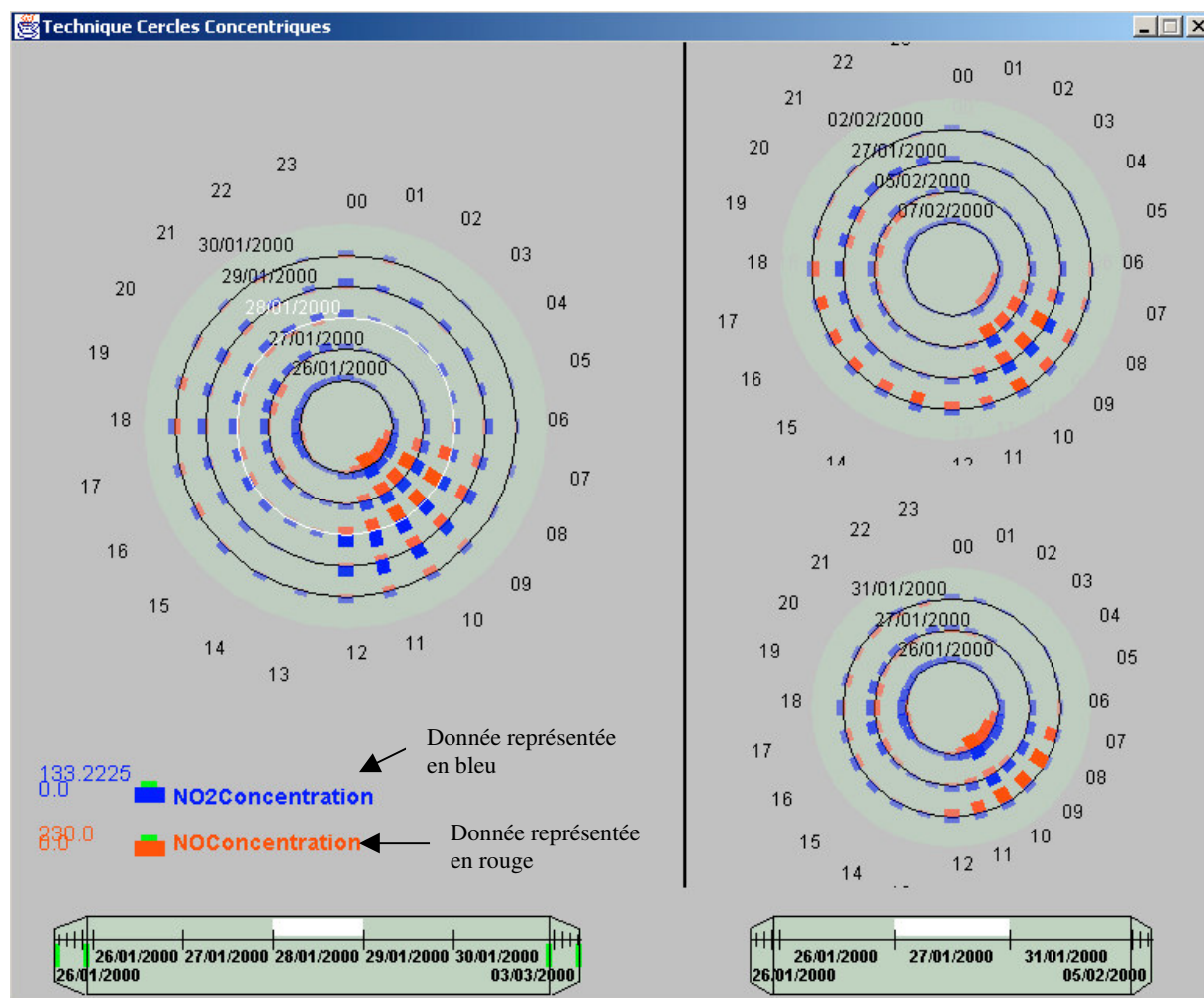


Figure 80 : Espace de travail dans la technique *cercles concentriques*.

#### 4.1.2 Processus de visualisation

La technique *cercles concentriques* a été développée selon le processus de visualisation de la Figure 67b (page 91) c'est-à-dire selon le processus *temps* puis l'étape *point de vue sur l'espace de visualisation des valeurs structurelles*. En effet, les deux espaces de représentation sont séparés. Le résultat du processus de visualisation de la dimension temporelle est l'espace temps représenté par la ligne temps. Deux points de vue sur le temps ont été considérés dans cette technique. Le premier est un point de vue linéaire qui a été utilisé par le processus de visualisation des valeurs temporelles (ligne temps). Le deuxième point de vue sur le temps est cyclique (les formes circulaires) utilisé pour organiser l'espace de représentation des valeurs structurelles. Les deux étapes *espace de visualisation des valeurs structurelles* et *espace de visualisation des valeurs temporelles* sont complètement indépendantes. Rappelons que le nombre de dimensions de l'espace géométrique est fixé au

niveau de l'étape *espace de visualisation* de la dimension correspondante. Dans le cas de la technique *cercles concentriques* le temps est représenté dans un espace en deux dimensions, les valeurs structurelles pouvaient être représentées dans un espace en deux dimensions comme pour le cas actuel de cette technique mais aussi dans un espace en trois dimensions.

### **4.1.3 Tâches utilisateur**

La *TCC* permet de répondre aux tâches utilisateur suivantes :

**Comparer des données dans le temps.** La comparaison peut porter sur deux points :

- Comparer l'évolution d'une ou plusieurs données sur des périodes différentes : une même période (séquence de valeurs correspondant à une période) est perçue différemment en fonction du périmètre du cercle sur lequel elle est représentée. La concentration des valeurs augmente sur un petit périmètre. Ceci rend difficile la comparaison des valeurs de deux périodes représentées sur deux cercles définies par deux périmètres différents. Pour résoudre ce problème, la *TCC* dispose d'un espace de travail dans lequel l'utilisateur déplace des cercles afin de les comparer. Dans cet espace, l'utilisateur peut permuter des cercles et les rapprocher dans l'espace pour pouvoir correctement et facilement les comparer. La *TCC* ne permet pas de résoudre complètement le problème de différence de périmètres entre cercles, elle réduit seulement son effet. La permutation des cercles au niveau de l'espace référence n'est pas possible, car les cercles doivent toujours être affichés en respectant l'ordre des valeurs temporelles,
- Comparer l'évolution de deux données : il s'agit de découvrir si une donnée évolue différemment qu'une seconde donnée.

**Connaître la valeur d'une donnée à un instant particulier.** En déplaçant la souris sur les rectangles représentant les valeurs des données au niveau de l'espace référence, la *TCC* visualise la valeur de la donnée à l'instant correspondant. La valeur est affichée au niveau de l'échelle de représentation en se référant à la donnée correspondante.

**Observer les données dans le temps.** L'utilisateur observe la tendance de l'évolution des données dans le temps à travers l'observation de l'évolution de la taille des objets graphiques



(rectangles) traduisant les valeurs. La hauteur et/ou la coloration de l'objet graphique sont proportionnelles à la valeur de la donnée représentée.

**Identifier les points de concentration de certaines valeurs.** Comme nous l'avons déjà indiqué, les valeurs des données sont traduites en terme de rectangles dont les hauteurs et/ ou les couleurs sont proportionnelles aux valeurs correspondantes. Faire varier à la fois la couleur et la hauteur des rectangles en fonction des valeurs des données fait disparaître les petits rectangles (représentant les petites valeurs). En effet, les plus grandes valeurs sont représentées avec de grands rectangles et en couleur foncée, alors que les petites valeurs sont représentées par des petits rectangles en couleur pâle. La disparition des petits rectangles fait apparaître les points de concentration des grandes valeurs.

**Etudier la corrélation de l'évolution de deux données dans le temps.** L'étude de la corrélation implique une représentation de l'espace temps qui soit commune aux deux données considérées. Bien que la dimension temporelle soit représentée plusieurs fois (ligne temps et cercles concentriques), chaque représentation est commune aux deux données. Cette technique représente les deux données à corrélérer à l'intérieur et à l'extérieur du cercle et avec deux couleurs différentes (rouge et bleu) pour les différencier. La corrélation repose sur l'observation simultanée dans le temps de la taille des objets graphiques relatifs aux deux données.

**Rechercher des effets de causalité entre données.** L'utilisateur étudie l'influence du comportement d'une donnée particulière sur une autre donnée. Il s'agit ici de mettre en avant les relations de causalité entre le changement de valeur (un événement) d'une donnée (la source) et le changement de valeur (une réaction) d'une autre donnée (la cible). L'utilisateur observe l'évolution des valeurs d'une donnée par rapport à celle de la seconde donnée.

**Changer l'échelle d'observation de l'évolution des données.** L'espace de référence est de taille limité. Le nombre de périodes (chacune représentée par un cercle) qui peuvent être visualisées est ainsi limité. Dans certains cas, l'utilisateur souhaite pouvoir réduire l'échelle d'observation des valeurs afin de gagner de l'espace et y ajouter d'autres cercles. Dans d'autres cas, il souhaite réduire le nombre de cercles affichés et augmenter l'échelle d'observation des données. L'utilisateur peut augmenter (respectivement réduire) la taille du

rectangle rouge ou bleu au niveau de l'échelle de représentation ce qui permet d'augmenter (respectivement réduire) les diamètres des cercles.

Les périodes affichées correspondent à celles affichées au niveau de la facette avant de la ligne temps. Ainsi, pour augmenter (respectivement réduire) le nombre de périodes affichées au niveau de l'espace référence, l'utilisateur doit augmenter (respectivement réduire) le nombre de périodes affichées au niveau de la ligne temps (voir conception de la ligne temps étudiée au paragraphe 3.2).

**Naviguer dans un grand espace de données.** La navigation concerne seulement l'espace temps. Elle s'effectue grâce à la ligne temps de l'espace de référence.

## **4.2 Technique de représentation en étoiles**

### **4.2.1 Représentation graphique**

La technique « *représentation en étoiles* » (à rapprocher avec la technique Radar d'Excel) consiste à représenter les données sous la forme d'une étoile (radar). Comme le montre la Figure 81, cette technique dispose de deux espaces. Un espace de référence, à gauche, dispose d'une ligne temps pour naviguer dans le temps. La période visualisée dans l'espace de référence correspond à la position du curseur dans la ligne temps. Cette technique permet de visualiser séquentiellement une ou deux données. Ainsi, une seule donnée peut être visualisée à la fois au niveau de l'espace de référence. Pour visualiser la seconde donnée, l'utilisateur doit sélectionner la case correspondante affichée en dessous de la ligne temps. Chacune des deux données est visualisée en une couleur différente, bleu ou rouge.

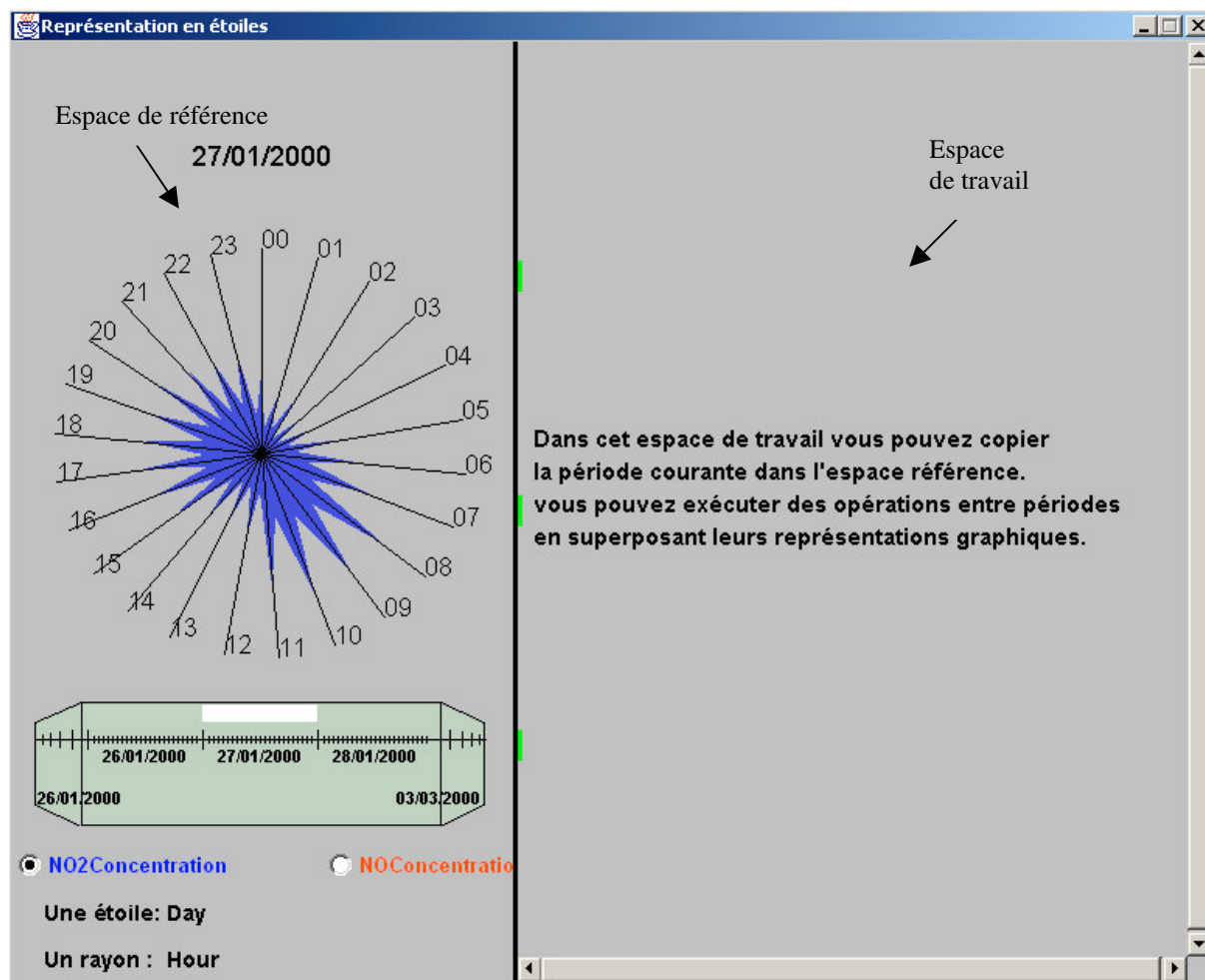


Figure 81 : Technique représentation étoiles.

Un espace de travail, à droite dans la Figure 82, sert à placer les périodes (étoiles) sélectionnées par l'utilisateur à partir de l'espace de référence. Pour se faire, l'utilisateur doit sélectionner une période par un clic souris sur l'étoile en cours dans l'espace de référence, et cliquer la souris au niveau de l'espace de travail. Dans cet espace, l'utilisateur peut déplacer des périodes pour les rapprocher dans l'espace et faciliter leur comparaison. En plaçant deux périodes de manière à ce qu'elles se chevauchent, une liste d'opérations entre périodes sera affichée. De telles opérations concernent par exemple la soustraction des valeurs d'une période de celles d'une autre période, la somme des valeurs de deux périodes, etc. Le résultat d'une opération est aussi visualisé sous la forme d'une étoile en bleu "pâle". L'utilisateur peut supprimer une période de l'espace de travail en sélectionnant l'étoile correspondante et en cliquant sur la touche-clavier supprimer.

## 4.2.2 Processus de visualisation

La technique *représentation en étoiles* est conçue selon le même processus de visualisation de la technique *cercles concentriques*, c'est-à-dire le processus *temps* puis l'étape *point de vue sur l'espace de visualisation des valeurs structurelles* (Figure 67b, page 91). Deux points de vue sur le temps ont été considérés dans cette technique : linéaire et cyclique. La différence entre les deux techniques provient du fait qu'elles s'appuient sur différentes organisations de l'espace de représentation des valeurs structurelles. Le même point de vue cyclique du temps est exploité différemment dans les deux techniques.

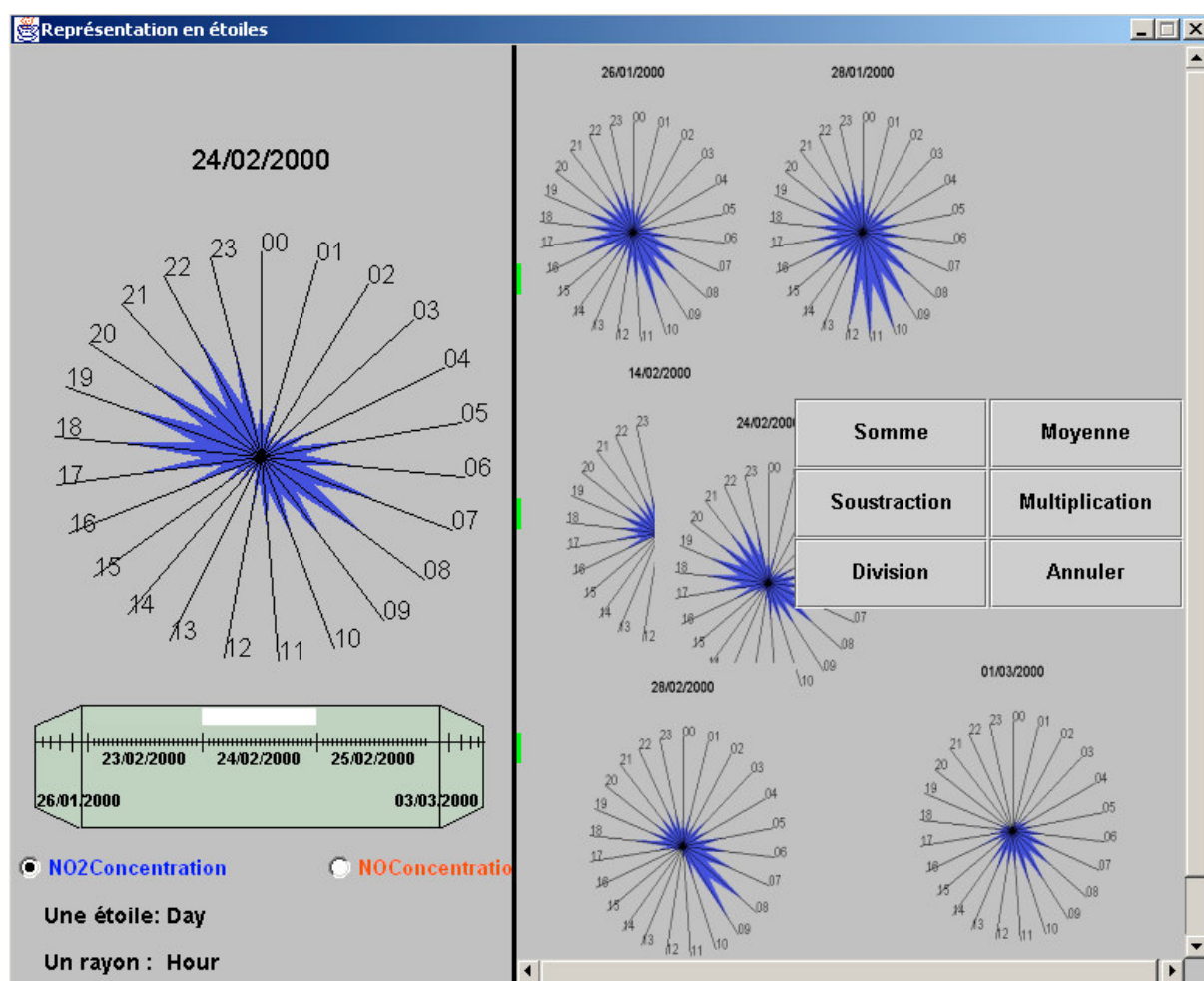


Figure 82 : Espace de travail de la technique représentation en étoiles.

## 4.2.3 Tâches utilisateur

La technique représentation en étoiles permet de répondre aux tâches utilisateur suivantes :

**Observer les données dans le temps.** L'utilisateur observe la tendance de l'évolution des données dans le temps à travers l'observation de l'évolution de l'espace occupé au niveau des étoiles. Cette tâche implique nécessairement la navigation selon la ligne temps.

**Comparer des périodes entre elles.** L'utilisateur peut placer dans l'espace de travail différentes périodes (des jours dans la Figure 82) afin de les comparer. Deux manières sont possibles pour comparer les périodes :

- La comparaison peut se faire sur la base de l'espace occupé ce qui permet d'avoir une vue globale de la période qui a les plus faibles valeurs, celle qui a les plus grandes valeurs et celles qui sont similaires. L'utilisateur rapproche ainsi les étoiles (périodes) dans l'espace de travail pour faciliter leur comparaison,
- Comme pour les nombres, la comparaison peut être faite au travers d'un calcul sur les valeurs. L'utilisateur peut visualiser le résultat de l'opération sur les deux périodes à comparer.

**Identifier des points de rupture dans le temps.** Cette technique permet d'observer les points de rupture dans le temps. Cette tâche se fait en observant l'espace de valeurs nulles (ou très petites) dans le temps. Il s'agit d'identifier les moments où il y a eu une rupture de l'évolution normale des données. La rupture concerne les périodes, en observant la forme de l'étoile.

**Identifier les points de concentration.** En observant la variation de l'espace occupé, l'utilisateur peut identifier les points temporels pour lesquels il y a eu plus de grandes valeurs ou de petites valeurs. Dans l'exemple de la Figure 82, nous remarquons que les valeurs les plus élevées de la concentration du polluant NO<sub>2</sub> au 24/02/2000 sont observées entre 5h et 10h, et entre 17h et 23h.

**Naviguer dans un grand espace de données.** Dans l'espace de référence, la navigation est réalisée à travers une navigation dans le temps. Cette navigation est assurée par la ligne temps. Pour l'espace de travail, la navigation dans l'espace de représentation est assurée par deux barres de défilement.

### 4.3 Représentation en diagrammes superposés

#### 4.3.1 Représentation graphique

La technique « *représentation en diagrammes superposés* » [Daassi 02] utilise explicitement la relation de dépendance entre unités temporelles, en considérant les données en deux dimensions. De ce point de vue, chaque valeur structurelle est définie par rapport à deux valeurs de deux unités temporelles régulières. Rappelons qu'une unité temporelle U1 est dite régulière par rapport à une autre unité U2, si chaque valeur de U1 contient un nombre fixe de U2 (Chapitre 2). Un exemple concret d'unités régulières est Jour et Heure. Chaque jour contient 24 heures. Dans l'exemple de la Figure 83, les unités Jour et Heure ont été transformées en deux axes.

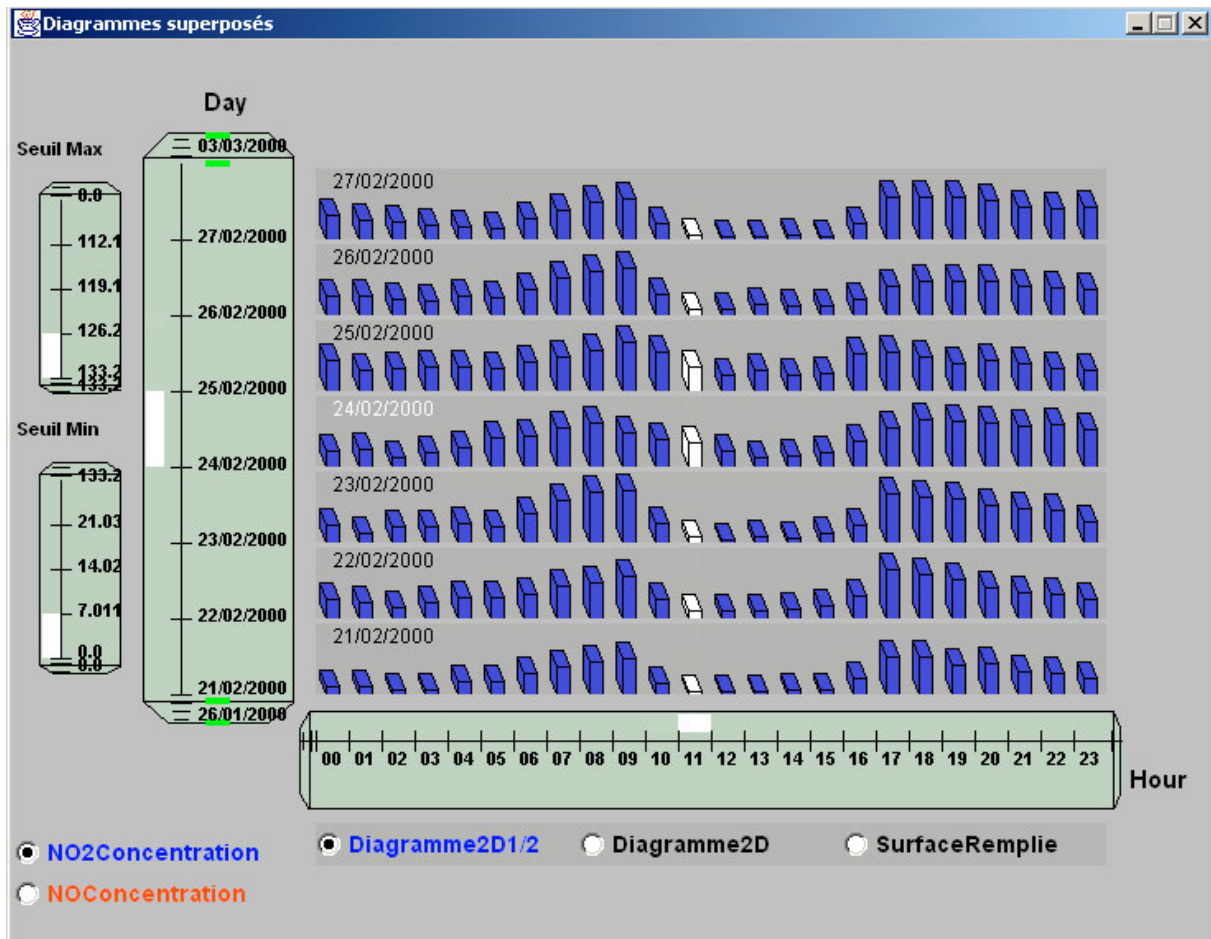
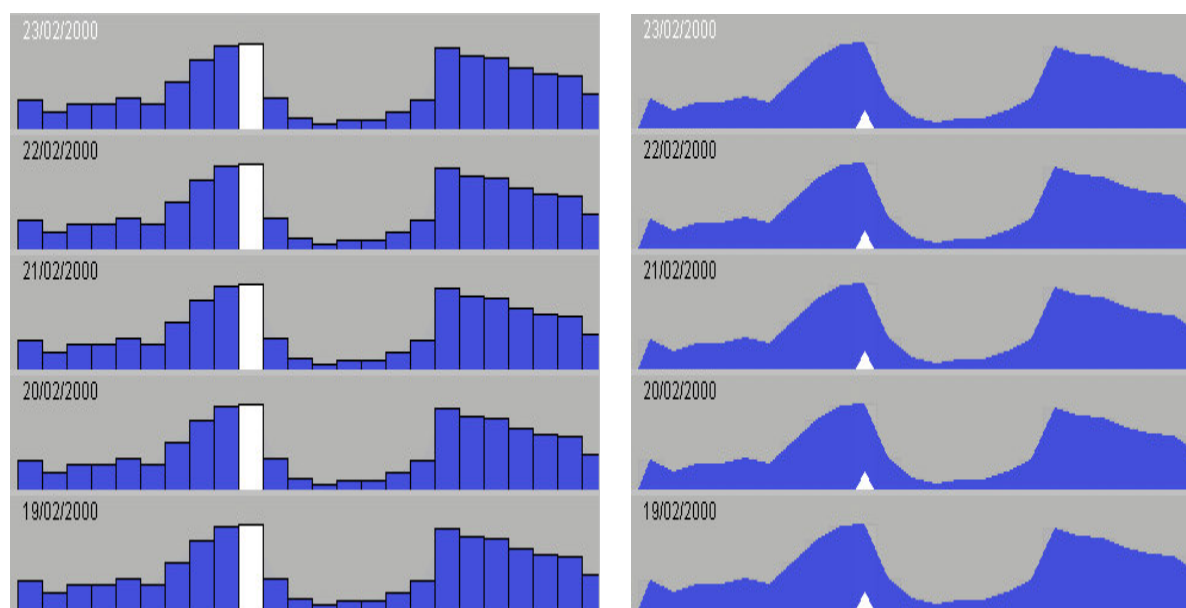


Figure 83 : Technique *diagrammes superposés* [Daassi 02].

La technique des *diagrammes superposés* permet de visualiser séquentiellement une ou deux données. Ainsi, une seule donnée peut être visualisée à la fois. Pour visualiser la seconde

donnée, l'utilisateur doit sélectionner la case correspondante affichée en bas et à droite au niveau de l'espace de représentation. Chacune des deux données est visualisée en une couleur différente, bleu ou rouge.

Cette technique consiste à représenter les périodes selon trois types de diagrammes de manière superposée. Les données peuvent être en diagramme en bâtons en représentation 2D1/2 (Figure 83), en représentation 2D (Figure 84a) ou en surface remplie (Figure 84b).



a. Représentation en diagramme 2D

b. Représentation en surface remplie.

*Figure 84* : Modes de représentation des données dans la technique diagrammes superposés. Deux lignes temps sont utilisées pour gérer la dimension temporelle des données, l'une pour gérer les jours et l'autre pour gérer les heures dans la Figure 83. Le curseur au niveau de la ligne temps horizontale permet de fixer le centre d'intérêt de l'utilisateur. La ligne temps verticale (respectivement horizontale) peut être redimensionnée. Ainsi l'utilisateur peut augmenter ou réduire le nombre de périodes (respectivement le nombre d'éléments par période) affichées. Il peut par exemple réduire le nombre de périodes et augmenter l'espace de représentation par période, ou inversement. Par exemple, deux périodes (jour) sont visualisées dans la Figure 85.

### 4.3.2 Processus de visualisation

La technique *diagrammes superposés* est conçue selon le processus de visualisation de la Figure 67a (page 91), c'est-à-dire le processus *temps* puis l'étape *espace de visualisation des valeurs structurelles*. En effet, l'espace de représentation des valeurs structurelles dépend de l'espace de représentation des valeurs temporelles. Les deux lignes temps (verticale et horizontale) définissent les caractéristiques de l'espace de représentation des valeurs structurelles. Le temps est représenté dans un espace en deux dimensions, l'espace des valeurs structurelles l'est obligatoirement. Un seul point de vue sur le temps est utilisé avec cette technique, c'est-à-dire linéaire et multigranulaire.

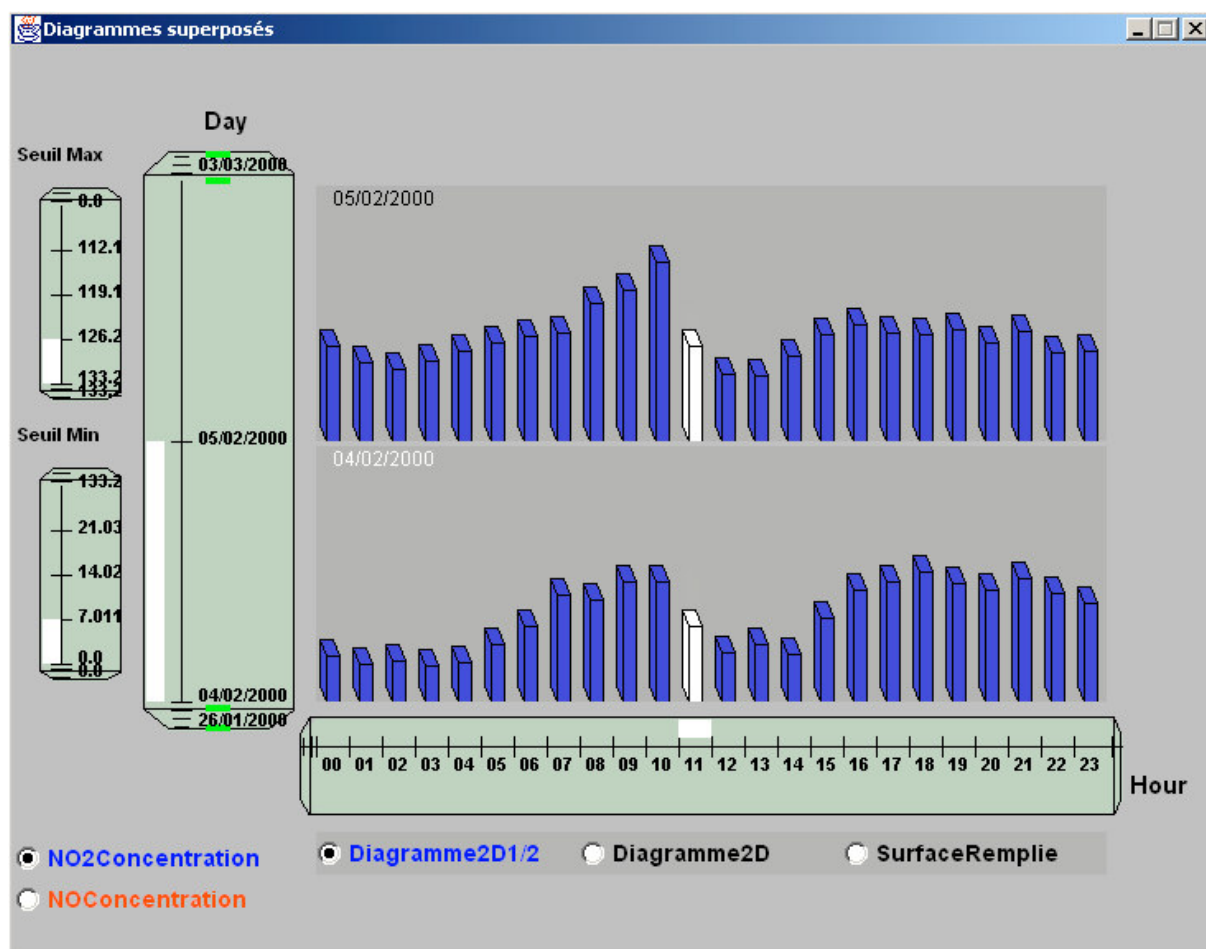


Figure 85 : Réduction du nombre de périodes affichées.



### **4.3.3 Tâches utilisateur**

La technique *représentation en diagrammes superposés* permet de visualiser une ou deux données séquentiellement comme la technique *représentation en étoiles*. Elle permet de répondre aux tâches utilisateur suivantes :

**Observer l'évolution des données.** L'utilisateur observe la tendance de l'évolution des données dans le temps à travers l'observation de l'évolution de la taille des objets graphiques représentant les valeurs.

**Comparer des données dans le temps.** La comparaison peut porter sur deux points :

- Comparer deux valeurs de la même donnée,
- Comparer des périodes entre elles : les périodes sont représentées sous la forme de diagrammes superposés. Ce type de représentation est essentiellement guidé par la remarque de Bertin [Bertin 67] : si la tâche de l'utilisateur est de comparer des périodes, il est conseillé de les représenter d'une façon superposée ou en forme de spirale (comme la conception de la technique *cercles concentriques*). La superposition des périodes facilite leur comparaison par rapport au mêmes instants.

**Disposer d'un référentiel.** Il s'agit ici d'observer l'évolution des données par rapport à une valeur référence. La technique diagrammes superposés dispose de deux référentiels pour fixer des seuils d'observation des valeurs des données. Un premier référentiel (partie gauche de la Figure 85, barre de défilement du bas) permet à l'utilisateur de fixer un seuil pour lequel seule la proportion de valeurs supérieures à ce seuil sera visualisée. Un second référentiel permet à l'utilisateur de fixer un seuil pour lequel seule la proportion de valeurs inférieures à ce seuil sera visualisée. L'utilisation des deux référentiels permet de fixer un intervalle pour observer les données. Seules les valeurs comprises dans cet intervalle sont visualisées.

Comme le montre la Figure 85, nous avons réutilisé le composant ligne temps pour représenter les deux référentiels. En effet, les valeurs des données quantitatives sont ordonnées et peuvent représenter un grand espace de données. Aussi, ces données peuvent être visualisées avec une représentation en mur en perspective. La représentation des deux référentiels pourrait être combinée en un seul composant avec deux curseurs l'un pour la

borne supérieure et l'autre pour la borne inférieure de l'intervalle. Nous avons préféré réutiliser le composant en mur en perspective pour réduire l'effort de développement.

**Naviguer dans un grand espace de données.** Avec cette technique, deux niveaux de navigation sont possibles : l'utilisateur peut naviguer dans l'espace des valeurs de l'une et l'autre des deux unités de temps (Jour et Heure dans la Figure 85).

## **4.4 Technique représentation tabulaire**

### **4.4.1 Représentation graphique**

La technique « *représentation tabulaire* » [Daassi 00] consiste à représenter une donnée dans un espace à deux dimensions. Elle peut cependant permettre de visualiser deux données et de passer de l'une à l'autre (Bouton Radio en bas de la Figure 86 : soit la concentration en NO<sub>2</sub>, soit la concentration en NO). Les données sont représentées par des objets graphiques (rectangles remplis) dont l'intensité de la couleur (bleu ou rouge) est définie proportionnellement aux valeurs des données correspondantes. Les deux axes de cet espace correspondent à deux lignes temps. La ligne temps verticale permet de naviguer dans l'ensemble des périodes, c'est-à-dire l'ensemble des jours dans la Figure 86. La ligne temps horizontale permet de naviguer au sein d'une période c'est-à-dire un jour, contenant 24 heures.

### **4.4.2 Processus de visualisation**

La technique tabulaire est conçue selon le même processus de visualisation que celui de la technique *diagrammes superposés*, c'est-à-dire le processus *temps* puis l'étape *espace de visualisation des valeurs structurelles* (Figure 67a, page 91). La différence entre ces deux techniques provient du fait qu'elles s'appuient sur différentes visualisations des valeurs structurelles.

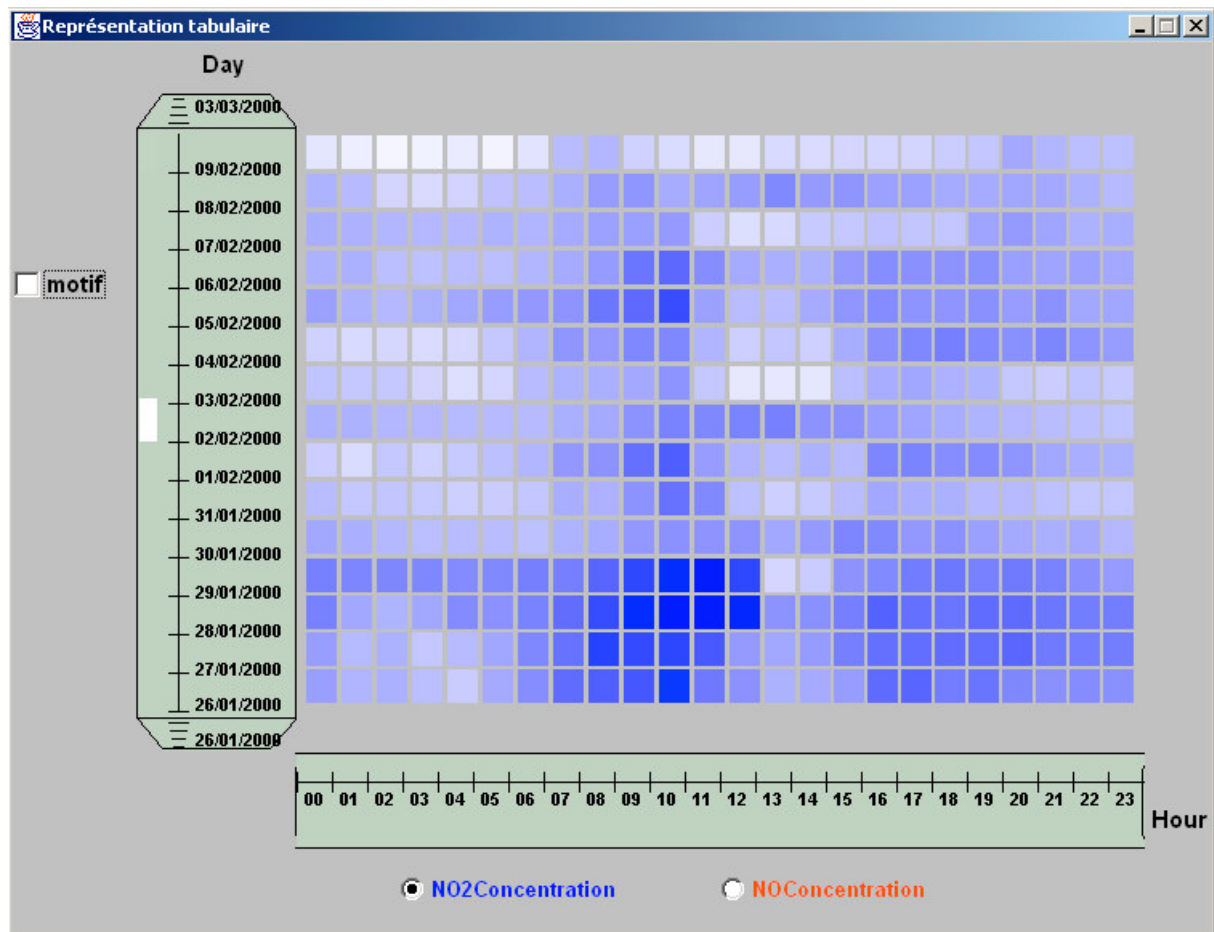


Figure 86 : Technique *représentation tabulaire* [Daassi 00].

#### 4.4.3 Tâches utilisateur

La technique *représentation tabulaire* permet de répondre aux tâches utilisateur qui sont :

**Observer les données dans le temps.** L'utilisateur observe la tendance de l'évolution des données dans le temps à travers l'observation de l'évolution de la coloration des objets graphiques traduisant les valeurs des données.

**Comparer des données dans le temps.** La comparaison peut porter sur trois points :

- ❑ Comparer deux valeurs de la même donnée,
- ❑ Comparer les valeurs observées sur deux périodes,
- ❑ Comparer l'évolution de deux données : il s'agit ici de déterminer si une donnée évolue différemment d'une autre donnée.

**Identifier les points de concentration.** Il s'agit d'identifier des zones de concentration de valeurs ou classes de valeurs particulières et les instants correspondants. Ces valeurs sont représentées par une intensité de couleur particulière.

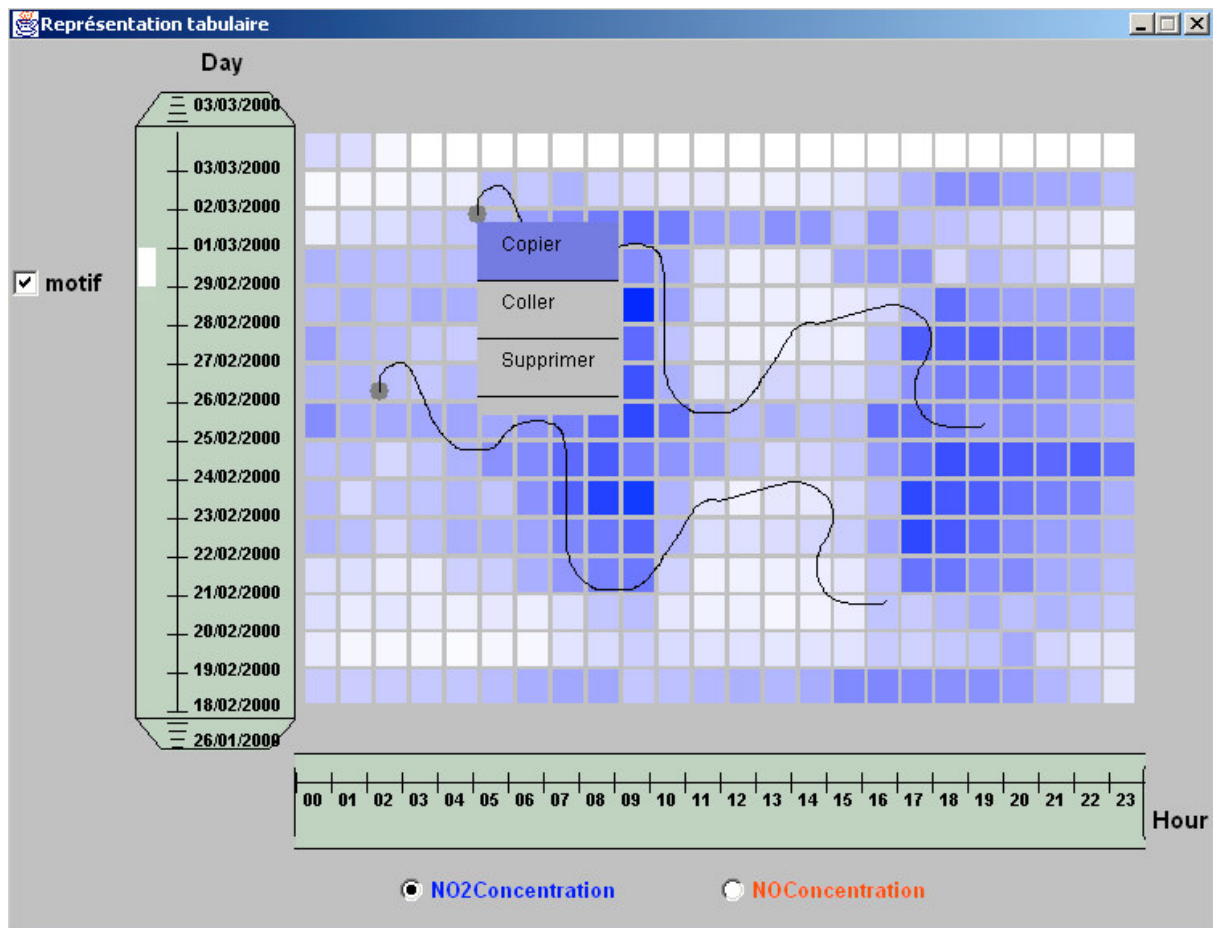


Figure 87 : Identification de motifs de comportement avec la technique *représentation tabulaire*.

**Identifier des motifs de comportement des données dans le temps.** Il s'agit d'observer le comportement des données dans le temps et d'identifier une représentation particulière définie par une association de coloration d'objets dans la visualisation. La case intitulée « motif », à gauche dans la Figure 87, permet de passer d'un type de tâche à un autre. Ainsi, si cette case est cochée (le mode « motif » est activé), l'utilisateur peut dessiner un motif (une forme libre dessinée avec la souris, par dessus la zone de visualisation). Elle/Il peut déplacer cette forme en positionnant la souris dans le petit cercle gris (marqueur du motif présenté à la Figure 87) affiché au début du motif, et en déplaçant le bouton gauche pressé. Presser le bouton droit au niveau du marqueur d'un motif permet d'obtenir un menu. Ce menu offre les options

suivantes : copier, coller ou supprimer un motif. En désactivant la case « motif », le système sera mis en mode sélection et l'utilisateur peut ainsi sélectionner un ensemble dans l'espace de représentation. En pressant le bouton droit de la souris dans la zone sélectionnée, des informations (la plus petite valeur, la plus grande valeur et la moyenne dans la zone sélectionnée) sont affichées. Ces informations seront supprimées par un clique sur le bouton gauche de la souris.

**Naviguer dans un grand espace de données.** Avec cette technique, deux niveaux de navigation sont possibles : l'utilisateur peut naviguer dans l'espace des valeurs de l'une et l'autre des deux unités de temps (Jour et Heure dans la Figure 87).

## **4.5 Technique représentation en pixels**

### **4.5.1 Représentation graphique**

Toutes les techniques que nous avons présentées dans ce chapitre sont conçues et développées à base d'objets graphiques. A l'opposé, la technique « *représentation en pixels* » de la Figure 88 est une technique à base de pixels. Elle consiste à traduire les valeurs des données en coloration de pixels. Le fait qu'une valeur soit représentée par un seul pixel permet de représenter un grand espace de données dans un espace réduit au niveau de l'écran. Cette technique permet de visualiser une ou deux données en parallèle. Elle dispose de deux espaces. Un premier espace à gauche, dit espace de référence, est utilisé pour visualiser la totalité de l'espace de données. Dans cet espace, l'utilisateur peut redimensionner l'espace de données en étirant à partir d'un côté choisi. Ainsi, elle/il peut redimensionner la longueur, la largeur de l'espace de données ou les deux. Une indication sur l'agrandissement effectué ainsi que la plus grande et plus petite valeur de la donnée sont visualisées à gauche au niveau de l'espace de référence. Par un clique du bouton ré-initialiser, l'espace de données sera ramené à son état initial.

La sélection d'une zone de l'espace de données peut avoir deux interprétations différentes selon que la case Focus (à gauche dans l'espace de référence) est cochée ou non. Si cette case est cochée, une représentation de la zone sélectionnée sera projetée dans l'espace de référence. Un seul espace projeté est disponible à un instant donné. La zone sélectionnée peut être utilisée comme une loupe. L'utilisateur peut la déplacer dans l'espace de données et les

valeurs correspondantes seront affichées dans la représentation projetée. La représentation projetée peut elle-même être redimensionnée. En désactivant l'option Focus (décocher la case), la représentation projetée disparaît.

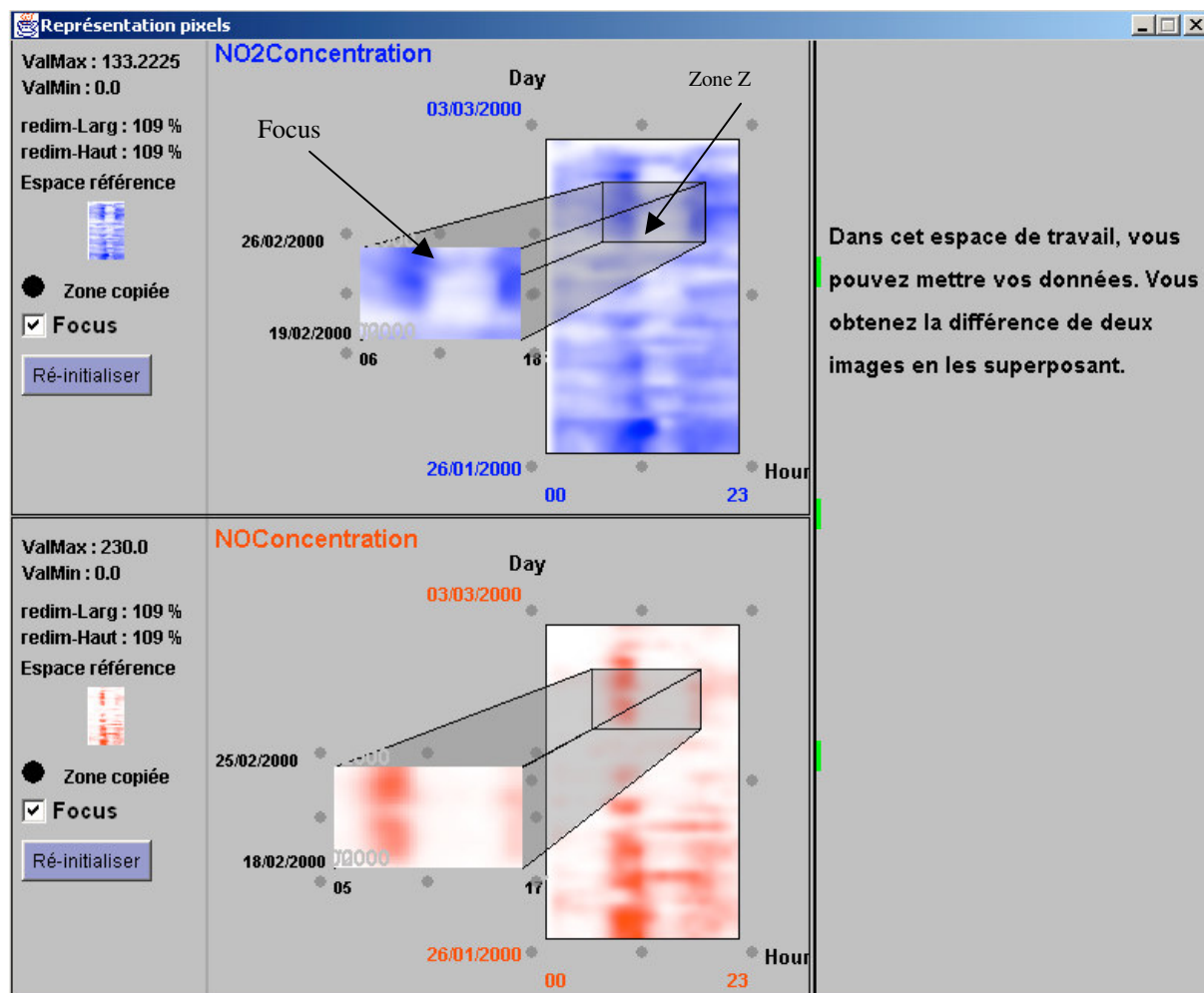


Figure 88 : Technique représentation en pixels.

Si la case Focus est désactivée ou que l'utilisateur a déjà sélectionné une zone (sa représentation est déjà projetée dans l'espace de référence), la sélection d'une nouvelle zone dans l'espace de données sera interprétée comme une mémorisation de cette zone (la pastille marquant le label « Zone copiée » sera colorée en vert, sinon en noir). Pour supprimer cette zone, il suffit de cliquer sur la pastille. Pour coller la zone mémorisée dans l'espace de travail (à droite), l'utilisateur clique dans l'espace de travail. Dans cet espace de travail, l'utilisateur peut déplacer les zones pour les comparer. En plaçant deux zones (images) l'une sur l'autre, une image différence sera calculée et affichée (Figure 89). Cette dernière contient un niveau de gris du blanc vers le noir. Un pixel noir signifie que les valeurs des colorations des deux

pixels relatifs aux deux images intervenant dans son calcul sont égales (ou presque). Il ne s'agit pas d'une égalité stricte entre les valeurs des données correspondantes. Pour V1 et V2 deux valeurs respectives des données à traiter D1 et D2 et, M1 et M2 les deux plus grandes valeurs de D1 et D2, le pixel noir signifie que  $V1/M1$  est presque égale à  $V2/M2$ . Pour supprimer une image de cet espace de travail, il convient de la sélectionner avec la souris (clique souris) puis de cliquer la touche-clavier supprimer.

### **4.5.2 Processus de visualisation**

La technique *représentation en pixels* est conçue comme pour les deux techniques précédentes (la technique *diagrammes superposés* et la technique *représentation tabulaire*) selon le processus de la Figure 67a (page 91), c'est-à-dire le processus *temps* puis l'étape *espace de visualisation des valeurs structurelles*.

### **4.5.3 Tâches utilisateur**

La technique *représentation en pixels* permet l'accomplissement des tâches suivantes :

**Observer les données dans le temps.** L'utilisateur observe la tendance de l'évolution des données dans le temps à travers l'observation de l'évolution de l'intensité de la couleur au niveau de l'espace de représentation.

**Etudier la corrélation de l'évolution de deux ou plusieurs données dans le temps.** L'étude de corrélation consiste en l'étude du comportement de plusieurs données aux mêmes moments. Cette étude est réalisée en observant la coloration de l'image différence correspondante aux données à étudier.

**Comparer des données.** Cette technique permet de comparer l'évolution de deux ou plusieurs données. Elle n'est pas adaptée à la comparaison de périodes ou des séquences périodiques. En effet, il est difficile de comparer la coloration de deux pixels traduisant deux valeurs d'une donnée ou de deux données.

**Identifier des points de concentration.** Il s'agit ici d'identifier des moments particuliers où des valeurs ou classes de valeurs (par exemple les grosses valeurs) particulières sont

observées. Les points de concentration correspondent à des zones avec des couleurs similaires (homogènes) au niveau de l'espace de représentation.

**Naviguer dans un grand espace de données.** La navigation dans l'espace de données est réalisée à travers une navigation dans l'espace de représentation défini par les colorations de pixels. L'utilisateur peut sélectionner une zone de l'espace de représentation et la déplacer afin d'observer d'autres parties de cet espace. De plus, l'utilisateur peut faire un zoom sur la zone sélectionnée en la redimensionnant avec la souris.

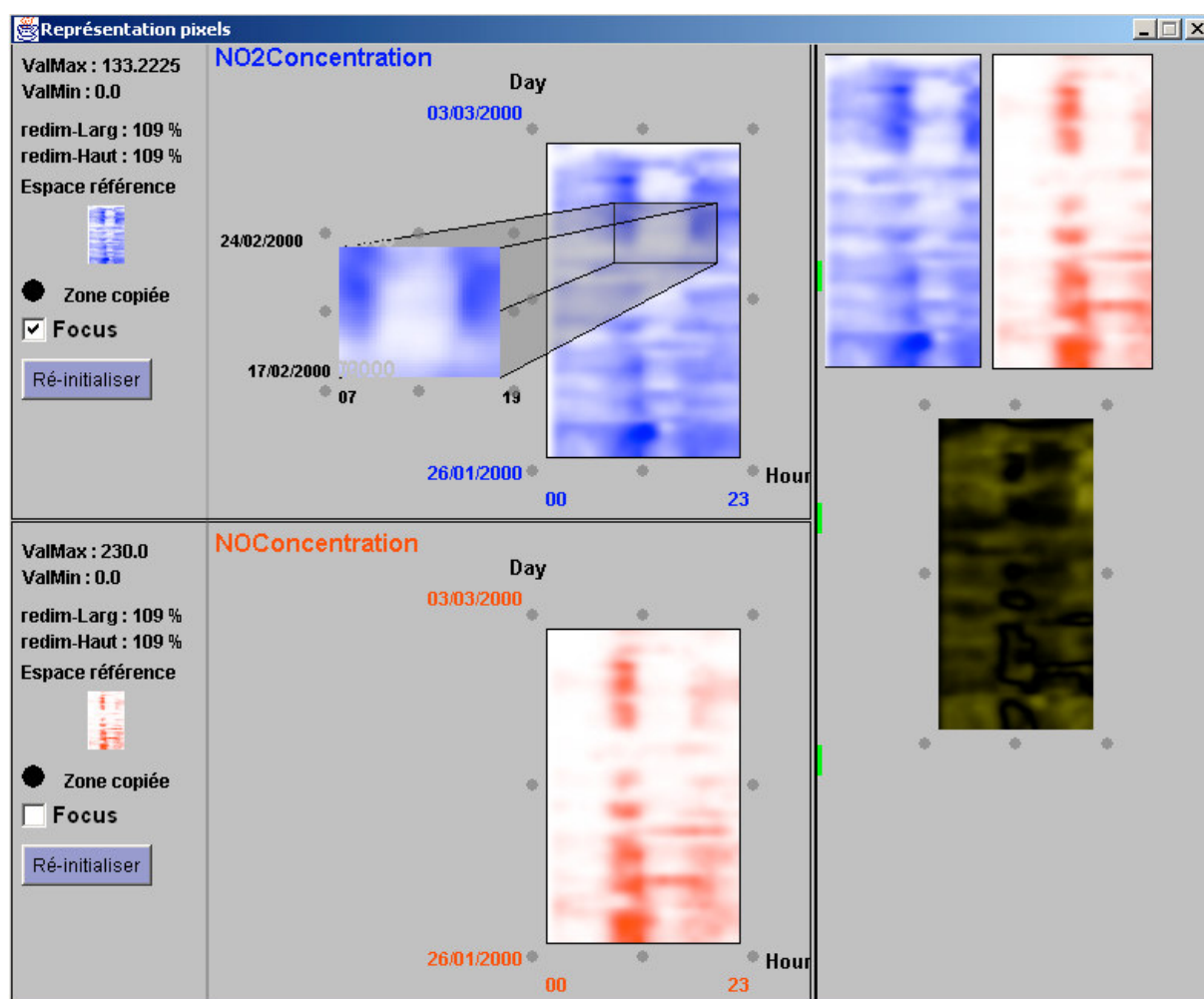


Figure 89 : Espace de travail dans la technique représentation pixels.

## 4.6 Synthèse

Dans ce paragraphe, nous avons présenté nos travaux sur la conception des techniques de visualisation de données temporelles. Nous avons présenté cinq techniques complémentaires nommées *cercles concentriques*, *représentation en étoiles*, *représentation en diagrammes*



*superposés, représentation tabulaire et représentation en pixels.* Nous avons présenté chacune de ces techniques par rapport à ses composantes graphiques, le processus de visualisation utilisé, et les tâches utilisateur pour lesquelles elle a été conçue.

## **5 Conclusion**

Dans ce chapitre, nous avons présenté notre approche centrée utilisateur pour la conception de techniques de visualisation de données temporelles. Nous avons conçu ces techniques selon les deux processus de visualisation de données temporelles présentés à la Figure 67, page 91. Dans l'objectif de garantir l'utilisabilité des techniques développées, nous cernons la réalisation de l'étape *point de vue sur l'espace de visualisation* du processus proposé par Chi [Chi 00] aux choix motivés par des propriétés ergonomiques. Ces propriétés ergonomiques traduisent les tâches utilisateur en terme d'analyse de données temporelles. Nous avons présenté au paragraphe 2 comment les tâches utilisateur sont traduites en propriétés ergonomiques.

Dans les deux processus de la Figure 67 (page 91) que nous considérons dans ce chapitre, le processus de visualisation de la dimension temporelle est conduit indépendamment du processus de visualisation de la dimension structurelle. Nous avons consacré le paragraphe 3 de ce chapitre à la présentation de notre travail de conception d'un composant logiciel réutilisable pour gérer la dimension temporelle au niveau d'une technique de visualisation. Ce composant est conçu en respectant les caractéristiques du temps étudiées au Chapitre 2. Le temps est représenté sous la forme d'un mur en perspective pour concrétiser les perceptions linéaire et logarithmique étudiées au Chapitre 2. La forme d'un mur en perspective est aussi adaptée pour gérer un grand espace d'informations temporelles.

Enfin, nous avons présenté cinq techniques de visualisation de données temporelles. Ces techniques représentent différentes vues du même espace de données. Chaque vue est conçue au regard d'un ensemble restreint de tâches utilisateur. Nous consacrons le chapitre suivant à notre système INVEST qui est une plateforme pour la gestion de plusieurs vues d'un même espace de données temporelles. En particulier, nous montrons comment INVEST assure le lien entre les cinq techniques de visualisation présentées dans ce paragraphe.

## *Chapitre 5*

### *INVEST : une plateforme pour l'analyse visuelle de données temporelles*

“What information consumes is rather obvious: it consumes the attention of its recipients. Hence a wealth of information creates a poverty of attention, and a need to allocate that attention efficiently among the overabundance of information sources that might consume it.”

~Herb Simon

Table des matières

**Chapitre 5.** INVEST : une plateforme pour l'analyse visuelle de données temporelles.

<a href="#">1</a>	<a href="#">Introduction</a>	139
<a href="#">2</a>	<a href="#">Expérimentation</a>	140
<a href="#">2.1</a>	<a href="#">Expérimentation 1</a>	141
<a href="#">2.2</a>	<a href="#">Expérimentation 2</a>	142
<a href="#">2.2.1</a>	<a href="#">Actualiser les valeurs de la fonction d'utilisabilité</a>	143
<a href="#">2.2.2</a>	<a href="#">Tester les hypothèses de travail</a>	146
<a href="#">2.3</a>	<a href="#">Synthèse</a>	148
<a href="#">3</a>	<a href="#">INVEST</a>	149
<a href="#">4</a>	<a href="#">Conclusion</a>	157

## 1 Introduction

Dans le chapitre précédent, nous avons présenté notre travail de conception des techniques centrées sur les tâches utilisateur pour la visualisation de données temporelles. Dans ce chapitre, nous présentons INVEST (Interactive Visualization and Explorative System of Temporal data) [Daassi 02], une plateforme pour l'analyse visuelle de données temporelles qui s'appuie sur ces techniques de visualisation.

La Figure 90 présente les étapes du processus de conception de INVEST. Les cinq techniques de visualisation (*cercles concentriques*, *représentation en étoiles*, *diagrammes superposés*, *représentation tabulaire*, et *représentation en pixels*) présentées précédemment sont implantées. Rappelons que ces techniques sont conçues au regard des caractéristiques des données manipulées étudiées au Chapitre 2, des processus de visualisation à utiliser présentés au Chapitre 3 et des tâches utilisateur étudiées au Chapitre 4. Le processus de conception de INVEST consiste en premier lieu à évaluer l'utilisabilité des techniques de visualisation développées, et ceci indépendamment les unes des autres. Les résultats de cette évaluation sont des mesures empiriques du degré d'utilisabilité des techniques de visualisation. Les valeurs de ces mesures sont par la suite utilisées lors de l'étape d'intégration qui conclut la conception de INVEST.

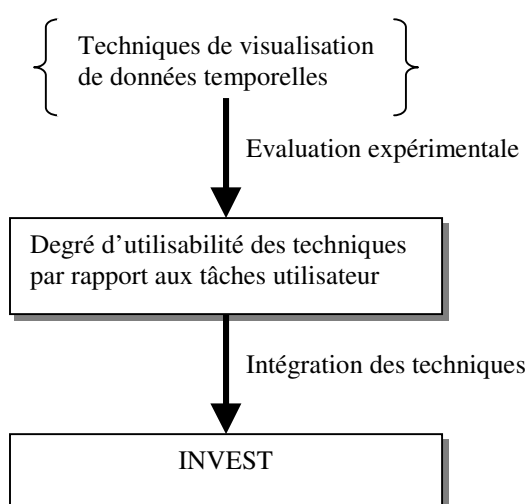


Figure 90 : Etapes du processus de conception de INVEST.

Ce chapitre est organisé comme suit : nous présentons au paragraphe 2 les résultats de deux expérimentations que nous avons menées pour évaluer les différentes techniques de

visualisation. Les résultats de ces évaluations nous ont servi dans un premier temps à améliorer l'utilisabilité des différentes techniques de visualisation, puis de les intégrer dans un seul environnement qui offre ainsi plusieurs vues d'un même espace de données. Cette intégration s'appuie sur le modèle d'architecture logicielle PAC-Amodeus [Nigay 94]. Nous présentons au paragraphe 3 la modélisation logicielle de INVEST selon ce modèle d'architecture.

## **2 Expérimentation**

Dans ce paragraphe, nous décrivons les deux expérimentations que nous avons menées pour évaluer les cinq techniques de visualisation de données temporelles intégrées dans le système INVEST.

Nous nous sommes appuyés sur plusieurs travaux de recherche qui proposent des méthodes pour l'évaluation de techniques de visualisation de données. Nous classons ces travaux selon quatre approches expérimentales :

(1) Certains travaux procèdent à la comparaison d'une version visuelle de la technique ou du système de visualisation à une version non visuelle. Comme dans [Turo 92], où Turo et Brian proposent une évaluation de la technique *Treemaps* par rapport à la représentation UNIX (tosh shell) pour la tâche de navigation dans une hiérarchie de répertoires.

(2) D'autres approches consistent à évaluer les techniques de visualisation en les confrontant les unes aux autres. La comparaison se fait par rapport à des tâches utilisateur [Holmquist 98, Liston 00, Kobsa 01]. Les tâches utilisées pour évaluer la technique *Treemaps* et la technique *Sunburst* [Stasko 00] sont par exemple : l'identification du répertoire contenant des fichiers d'un certain type, la comparaison de la taille de deux fichiers et l'identification du plus grand.

(3) Dans [Cockburn 00] la technique *Cone Trees* est évaluée seulement par rapport aux tâches utilisateur indépendamment d'autres techniques de visualisation.

(4) Alors que les trois premières approches se basent sur des évaluations empiriques, dans [Hascoët 98], une méthode analytique est proposée pour l'analyse de deux visualisations : représentation en spirale et une représentation en ressort (spring display). Les critères analytiques retenus sont : AM (Average Matching) et WM (Worst Mismatches). Le critère

AM représente la différence moyenne entre la matrice euclidienne et la matrice de dissimilitude normalisées. Pour chaque couple d'éléments affichés sur l'écran, la matrice euclidienne et la matrice de dissimilitude définissent respectivement la distance euclidienne et la dissimilitude des sémantiques des éléments du couple. Le critère WM est la proportion, parmi les couples les plus similaires, des couples dont la similarité et la distance euclidienne entre leurs deux éléments ne sont pas corrélés.

Nous visons à mesurer l'adéquation d'une technique de visualisation à une ou plusieurs des tâches utilisateur que nous avons identifiées pour la manipulation de données temporelles. Plusieurs travaux ont déjà souligné l'importance d'une évaluation par rapport aux tâches utilisateur afin de garantir leur utilisabilité : "Clearly, further work evaluating information visualization is necessary. New visualization techniques, no matter how innovative, are not valuable unless they provide true utility and assist people with real tasks." Extrait de [Stasko 00]. Nous définissons l'utilisabilité d'une technique de visualisation  $V_i$  pour une tâche utilisateur  $T_j$  par la valeur d'une fonction  $F$ . La valeur  $F(V_i, T_j)$ , définie dans l'intervalle  $[0, 1]$ , représente l'utilisabilité de la technique de visualisation  $V_i$  par rapport à la tâche  $T_j$ . Les valeurs de la fonction  $F$  sont obtenues empiriquement à partir des deux expérimentations que nous avons menées.

Les techniques de INVEST ont été évaluées par deux expérimentations séparées dans le temps par un délai d'une année et demi. Pour chacune d'entre elles, six doctorants et étudiants en DESS du laboratoire de géographie SEIGAD y ont participé. Les participants sont tous familiers avec la manipulation et l'analyse de données temporelles ainsi que l'utilisation d'outils tels que des tableurs et des Systèmes d'Information Géographique (SIG). Nous décrivons dans la suite la démarche d'évaluation ainsi que les résultats des deux expérimentations.

## **2.1 Expérimentation 1**

Pour mettre en œuvre cette expérimentation, nous avons utilisé des données réelles qui représentent des mesures de polluants atmosphériques (NO, NO<sub>2</sub> et Nox). Cette expérimentation a un double objectif :

- *Identifier les problèmes lexicaux et syntaxiques existants.* Ces problèmes concernent les modes de représentation des données, les difficultés d'interaction

avec la représentation graphique, ou encore les incohérences entre les différentes techniques utilisées. Ces incohérences concernent par exemple une utilisation non homogène des couleurs dans les différentes techniques, ou encore des réalisations concrètes de la même tâche implantées différemment dans plusieurs techniques. A partir des remarques des participants, de nos observations de leur comportement dans la réalisation de certaines tâches, et des problèmes qu'ils rencontrent pour atteindre leurs objectifs, des améliorations ont été réalisées sur les différentes techniques de visualisation de INVEST.

- **Définir une fonction d'utilisabilité.** L'évaluation des différentes techniques vise à identifier les valeurs de la fonction d'utilisabilité  $F$ . Pour se faire, nous nous sommes basés sur l'utilisation de formulaires. Le participant disposant d'un formulaire pour chaque technique de visualisation, répond à des questions en relation avec les tâches utilisateur. Nous avons demandé aux participants de spécifier le résultat de la tâche, les avantages de l'utilisation de cette technique pour la réalisation de la tâche ainsi que les difficultés rencontrées. L'utilisateur peut aussi faire des propositions ou recommandations pour améliorer l'utilisabilité de la technique par rapport à la tâche en question. Enfin, il peut indiquer si la technique peut éventuellement répondre à d'autres tâches utilisateur. Ceci nous permet de vérifier l'exhaustivité des tâches par rapport à chaque technique de visualisation.

Chaque question, à choix multiple, est évaluée par une note comprise entre 0 et 20. La moyenne de notes données par les participants pour une technique  $V_i$  par rapport à une tâche  $T_j$  représente la valeur de la fonction  $F$  pour  $V_i$  et  $T_j$ . Ainsi, à chaque technique de visualisation est associé son degré d'utilisabilité par rapport à chacune des tâches utilisateur.

## **2.2 Expérimentation 2**

Nous avons procédé à cette deuxième évaluation en visant deux objectifs : 1) affiner la fonction d'utilisabilité  $F$  et 2) valider des hypothèses de travail que nous avons considérées durant le processus de conception des différentes techniques de visualisation. Nous détaillons dans la suite chacun de ces deux objectifs.

### 2.2.1 Actualiser les valeurs de la fonction d'utilisabilité

Les résultats obtenus lors de cette deuxième expérimentation sont représentés dans les [Figure 91, 92, 93, 94 et 95](#). Ces figures présentent respectivement les valeurs de la fonction d'utilisabilité  $F$  pour les techniques *cercles concentriques*, *représentation en pixels*, *représentation tabulaire*, *représentation en étoiles* et *diagrammes superposés*. Dans chacune de ces figures, les valeurs de la fonction d'utilisabilité sont représentées en histogrammes. L'axe horizontal est utilisé pour représenter les tâches utilisateur que la technique en question peut réaliser. L'axe vertical représente l'échelle utilisée pour représenter les degrés d'utilisabilité. Pour une tâche utilisateur donnée, la valeur 0 sur l'axe vertical signifie que la technique de visualisation considérée ne permet pas la réalisation de la tâche en question. Par contre, la valeur 1 sur l'axe vertical signifie que la technique de visualisation est très adéquate à la tâche utilisateur. Nous présentons dans le paragraphe 3 comment ces résultats ont été intégrés dans le système INVEST.

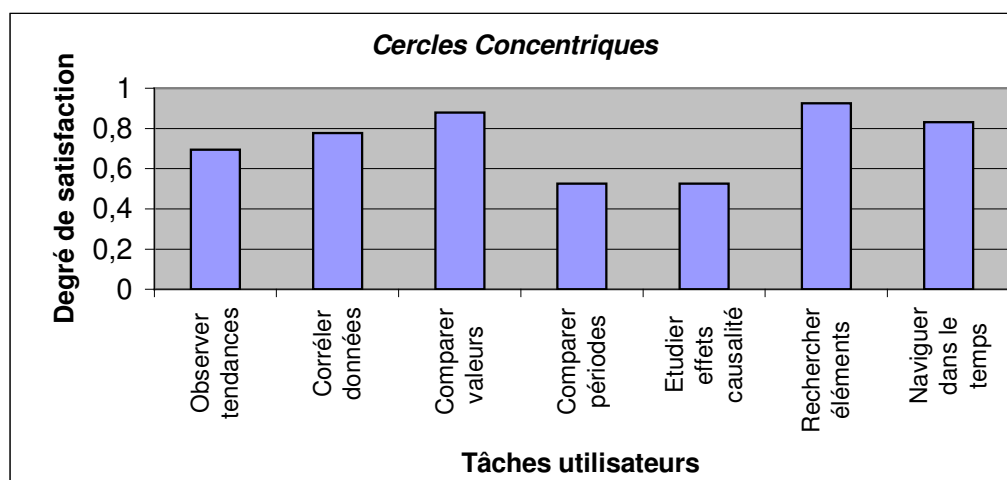


Figure 91 : Valeurs de la fonction d'utilisabilité de la technique *cercles concentriques*.



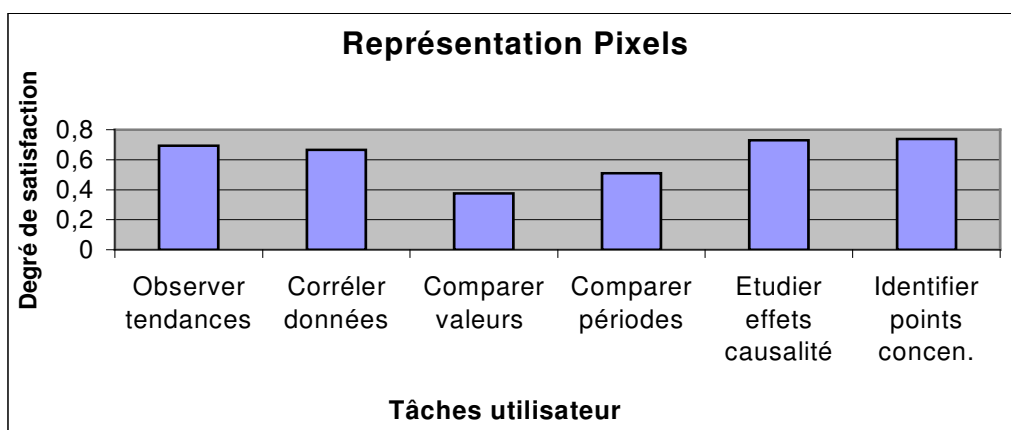


Figure 92 : Valeurs de la fonction d'utilisabilité de la technique *représentation en pixels*.

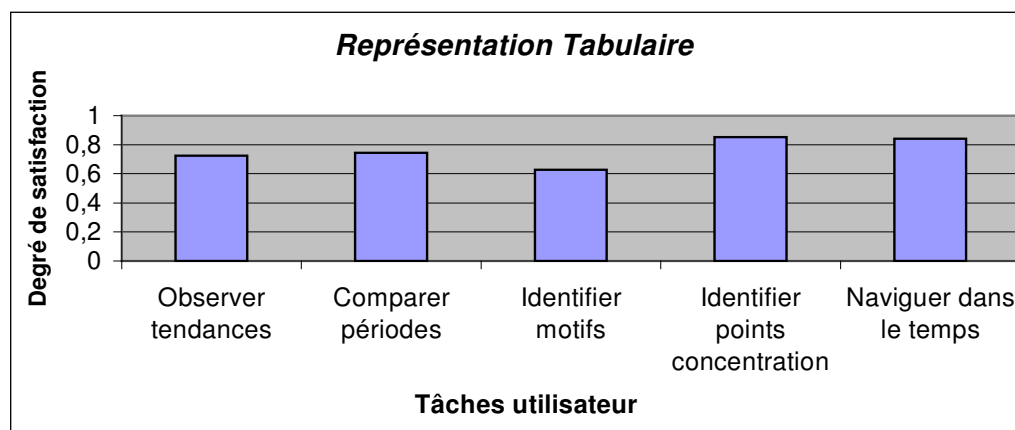


Figure 93 : Valeurs de la fonction d'utilisabilité de la technique *représentation tabulaire*.

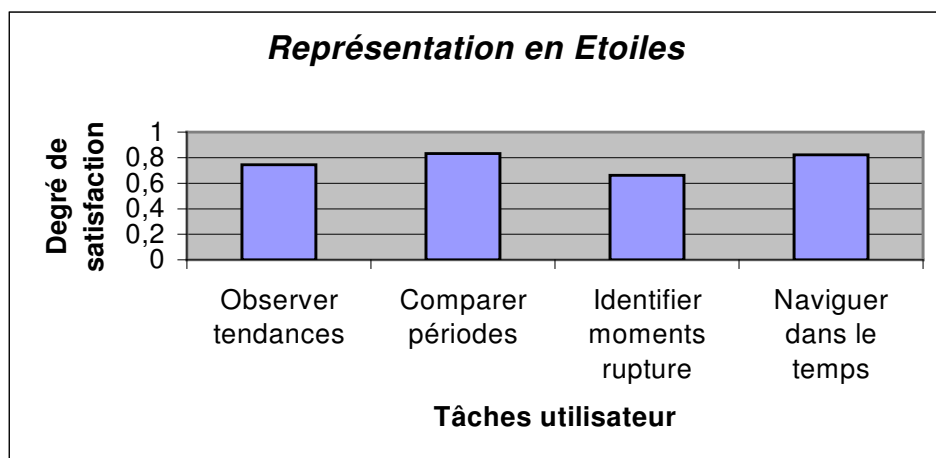


Figure 94 : Valeurs de la fonction d'utilisabilité de la technique *représentation en étoiles*.

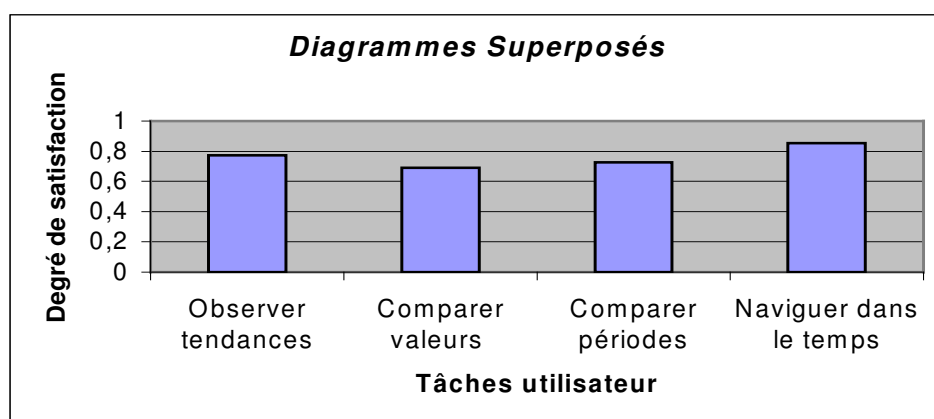


Figure 95 : Valeurs de la fonction d'utilisabilité de la technique *diagrammes superposés*.

Les résultats de cette expérimentation confirment le fait qu'une technique de visualisation ne peut pas répondre à toutes les tâches utilisateur. Le Tableau 3 récapitule les résultats obtenus. Les deux axes horizontal et vertical de ce tableau définissent respectivement les techniques de visualisation et les tâches utilisateur. La valeur contenue dans la case à l'intersection d'une valeur sur l'axe horizontal (une technique de visualisation) et d'une valeur sur l'axe vertical (une tâche) dénote le degré d'utilisabilité de la technique de visualisation par rapport à la tâche spécifiée. Comme le montre ce tableau, contrairement à la technique *cercles concentriques*, la technique *représentation tabulaire* est adaptée à l'identification des motifs de comportement ou à l'identification des points de concentration de certaines valeurs. Par contre la *TCC* est bien adaptée pour étudier la corrélation et l'effet de causalité entre deux données. De même, la technique *représentation en étoiles* est adaptée à l'identification des points de rupture dans le temps alors que la technique *diagrammes superposés* ne l'est pas. Par contre si la tâche est de comparer les valeurs des données, la technique *diagrammes superposés* est mieux adaptée que la technique *représentation en étoiles*. De plus, la technique *représentation en pixels* est mieux adaptée que les techniques *cercles concentriques*, *représentation en étoiles* ou *diagrammes superposés* pour la tâche d'identification des points de concentration de valeurs particulières dans l'espace de données.

Ces deux expérimentations confirment les résultats de la littérature soulignant la nécessité d'utiliser plusieurs vues (c'est-à-dire plusieurs techniques de visualisation) [Jonathan 00, Wang 00, North 01] d'un même espace de données temporelles. Ceci permet de répondre à la diversité des tâches utilisateur en terme de manipulation de données temporelles. Chaque vue doit être conçue au regard d'un nombre limité de tâches utilisateur.

Techniques Tâches utilisateur	Cercles Concentrique	Diagrammes Superposés	Représ. Etoile	Représ. Tabulaire	Représ. Pixels
Observer tendances	0,694117	0,77450917	0,745097	0,725489	0,694111
Rechercher des éléments	0,92483617	0	0	0	0
Comparer valeurs	0,878431	0,690195	0	0	0,376470
Comparer périodes	0,525489	0,725489	0,833333	0,745097	0,509787
Naviguer dans le temps	0,831699	0,85293	0,823528	0,839868	0
Corréler données	0,77647	0	0	0	0,666652
Etudier les effets de causalité	0,525489	0	0	0	0,729397
Identifier moments de ruptures	0	0	0,660130	0	0
Identifier motifs	0	0	0	0,627450	0
Identifier des points de concentration	0	0	0	0,852940	0,738551

Tableau 3 : Tableau comparatif des résultats expérimentaux.

### 2.2.2 Tester les hypothèses de travail

Outre l'actualisation des valeurs de la fonction d'utilisabilité des différentes techniques de visualisation par rapport aux tâches utilisateur, cette deuxième expérimentation vise à valider ou rejeter deux hypothèses de travail que nous avons prises en considération. Pour se faire, nous avons implémenter des observateurs au niveau des techniques de visualisation pour capturer des informations sur l'activité de l'utilisateur. Nous présentons dans la suite les deux hypothèses et les résultats obtenus suite à l'expérimentation.

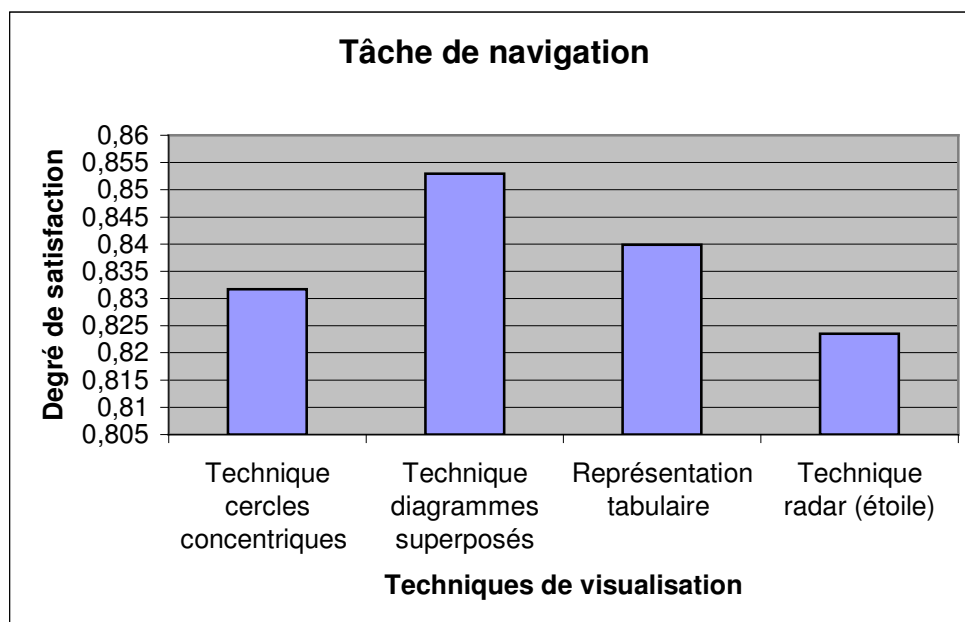
**H1** : l'utilisation d'un espace de travail au niveau d'une technique de visualisation de données temporelles est nécessaire.

Un espace de travail permet à l'utilisateur d'isoler des données à traiter, de pouvoir les manipuler et surmonter la contrainte de relation d'ordre entre valeurs temporelles. Pour évaluer cette hypothèse, nous avons procédé à la capture des cliques souris au niveau des deux espaces (référence et travail) de chacune des trois techniques de visualisation disposant d'un espace de travail (*cercles concentriques*, *représentation en pixels* et *représentation en étoiles*). Outre la capture des cliques souris, nous avons calculé le temps passé par l'utilisateur au niveau de chacun des deux espaces. L'expérimentation montre que les utilisateurs ont passé environ 39% de leurs temps de travail au niveau de l'espace de travail. Bien que cette proportion soit inférieure à celle du temps passé au niveau de l'espace de référence (61%), elle reste cependant très significative. La proportion 39/61 est sans doute sous évaluée car l'espace de référence est un passage obligé, ce qui amplifie le temps passé sur cet espace. Il serait intéressant de compléter cette expérimentation par un système de suivi du regard [Masuda 00] qui permettrait une autre mesure complémentaire à la capture des cliques souris pour évaluer l'utilisation des deux espaces.

**H2** : les caractéristiques de l'espace de représentation de la dimension structurelle influent sur la réalisation de la tâche de navigation selon la dimension temporelle.

Pour les quatre techniques de visualisation : *cercles concentriques*, *diagrammes superposés*, *représentation en étoiles* et *représentation tabulaire*, nous avons demandé aux participants d'évaluer la tâche de navigation dans le temps. Dans ces techniques, la navigation dans le temps est réalisée au travers du même composant "ligne temps". Les résultats pour les quatre techniques sont très proches les unes des autres.

Comme le montre le graphique de la Figure 96, les valeurs fournies par les utilisateurs pour les quatre techniques de visualisation pour la tâche de navigation dans le temps varient entre 0.823 (technique *représentation en étoiles*) et 0.852 (technique *diagrammes superposés*). Nous considérons que les quatre valeurs de degré de satisfaction de la Figure 96 représentent la même valeur d'utilisabilité des différentes techniques pour la tâche de navigation. La variation entre ces valeurs est inférieure à 3%.



*Figure 96* : Valeurs de la fonction d'utilisabilité de quatre techniques de visualisation pour la tâche de navigation dans le temps.

Nous concluons que les modes de représentation et les caractéristiques de l'espace des valeurs structurelles n'influencent pas la tâche de navigation selon la dimension temporelle. Nous rejetons donc cette hypothèse de travail.

## 2.3 Synthèse

A partir des deux expérimentations nous retenons trois recommandations pour la conception d'un système interactif pour l'analyse visuelle de données temporelles. Nous recommandons :

- ❑ D'utiliser plusieurs vues complémentaires (techniques de visualisation) du même espace de données temporelles. Ceci permet de répondre à la diversité des tâches utilisateur en terme de manipulation de données temporelles. Chaque vue doit être conçue au regard d'un nombre restreint de tâches utilisateur,
- ❑ D'utiliser un espace de travail en parallèle à un espace de référence. Ceci permet à l'utilisateur d'isoler des données, et se concentrer sur l'analyse des données ainsi isolées, et
- ❑ De concevoir l'espace de la dimension temporelle indépendamment de la dimension structurelle pour favoriser sa réutilisation. Le mode de représentation de

l'espace valeurs structurelles n'affecte pas la réalisation des tâches (la navigation dans le temps en particulier) au niveau de l'espace des valeurs temporelles.

Ayant présenté les résultats de l'évaluation des techniques de visualisation de données temporelles du système INVEST, nous présentons dans la suite l'étape Intégration de ces techniques de la Figure 90 (page 139). Pour cette étape d'intégration, nous présentons le modèle d'architecture logicielle de INVEST, puis nous détaillons comment les valeurs de la fonction d'utilisabilité sont exploitées au niveau de l'implantation.

### 3 INVEST

Pour la conception logicielle de INVEST, nous avons utilisé le modèle d'architecture logicielle PAC-Amodeus [Nigay 94, Nigay 97] permettant l'organisation des composants d'un système interactif.

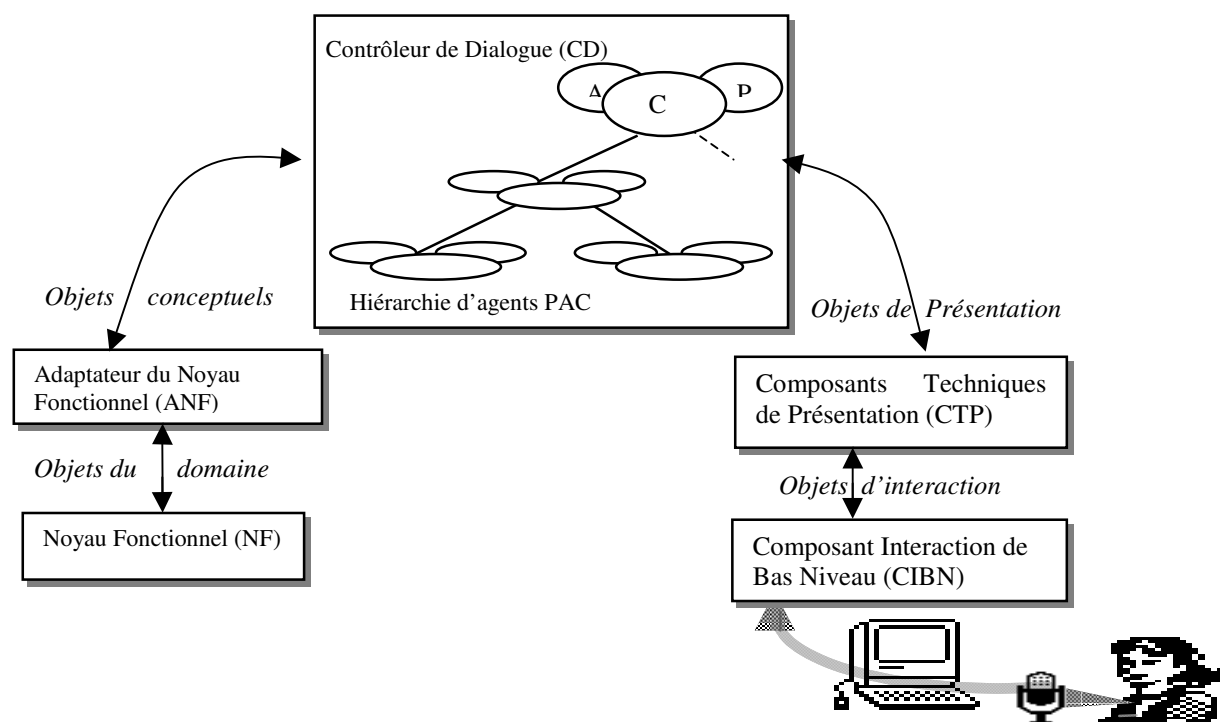


Figure 97 : Modèle PAC-Amodeus [Nigay 94].

La structuration du système selon le modèle PAC-Amodeus préconise une décomposition en cinq composants logiciels comme le modèle Arch [UIMS 92]. Dans la Figure 97, aux deux extrêmes du modèle nous trouvons le Noyau Fonctionnel (NF) et le Composant Interaction de Bas Niveau (CIBN). Le NF implémente les concepts spécifiques au domaine. Il maintient ainsi les informations (source de données) à rendre perceptibles à l'utilisateur. Dans notre cas,

il s'agit de la source de données temporelles (par exemple un SGBDT ou une collection de documents XML ou encore des fichiers textes). Le CIBN correspond à la plateforme matérielle et logicielle. Pour réaliser les différentes techniques de visualisation et implémenter INVEST, nous avons choisi d'utiliser l'environnement JDK [JDK 03] de SUN. Cet environnement fournit le langage JAVA et des boîtes à outils abstraites comme AWT et Swing indépendantes du système d'exploitation utilisé. Le CIBN correspond aux éléments des boîtes à outils AWT et Swing utilisées. Nous y trouvons par exemple, les boutons, les labels, etc. Les Composants Techniques de Présentation (CTP) sont utilisés comme des modules réutilisables pour construire les différentes techniques de visualisation de INVEST. Nous avons développé un ensemble de composants Java Beans réutilisables et opérationnels selon le principe "Plug and Play". Un exemple de ces composants est la ligne temps que nous avons présentée au paragraphe 3 du Chapitre 4.

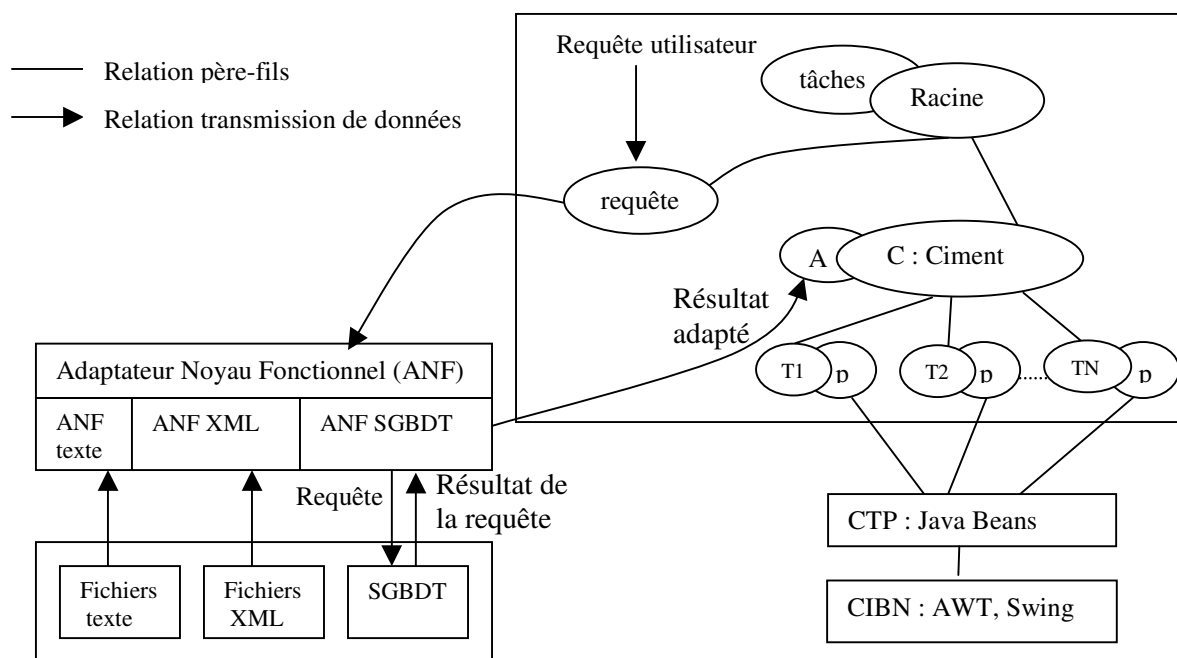


Figure 98 : Architecture d'INVEST selon PAC-Amodeus.

L'Adaptateur du Noyau Fonctionnel (ANF) est une interface conçue pour absorber les changements entre les voisins directs. En particulier, cette interface transforme les données à visualiser en un format unifié compréhensible par le code de l'interface utilisateur. L'ANF garantit une indépendance du système développé vis-à-vis du Noyau Fonctionnel utilisé. Il sera donc facile de changer le NF moyennant une modification ou une redéfinition de cette

interface. Comme le montre la Figure 98, nous utilisons un adaptateur par type de source de données. Les trois adaptateurs (fichiers texte, fichiers XML et SGBDT) sont liés à un adaptateur père qui permet de sélectionner l'adaptateur fils adéquat pour gérer la requête de l'utilisateur. La connexion à une base de données temporelles nécessite seulement d'implémenter l'adaptateur correspondant, de même pour les fichiers XML. L'ajout d'une autre source de données nécessite seulement l'implémentation de son adaptateur et de le connecter à l'adaptateur père. Ce dernier permet de communiquer les données au Contrôleur de Dialogue (CD).

Le Contrôleur de Dialogue (CD) est le cœur du modèle PAC-Amodeus. Le CD, structuré en une hiérarchie d'agents<sup>8</sup> PAC [Coutaz 97], est responsable de la planification des tâches. Un agent PAC est défini par trois facettes : (1) La **Présentation** implémente le comportement perceptible de l'agent. Dans PAC-Amodeus, cette facette est en relation directe avec le Composant Techniques de Présentation (CTP). (2) L'**Abstraction** définit la compétence de l'agent. Pour le cas de la visualisation de données, elle maintient les données à rendre perceptibles par la facette présentation. (3) La facette **Contrôle** relie les deux facettes présentation et abstraction. Aucune communication directe n'est possible entre ces deux facettes. La facette contrôle permet aussi la communication avec les contrôles d'autres agents PAC. Le raffinement du CD en une hiérarchie d'agents PAC offre beaucoup d'avantages en particulier la gestion du parallélisme et de la représentation multiple.

Concrètement, le CD est modélisé dans INVEST [Daassi 02] en une hiérarchie d'agents PAC à trois niveaux. La racine de cette hiérarchie (Figure 98) est un agent PAC permettant de relier l'agent Requête à l'agent Ciment. L'agent Racine a la responsabilité de gérer les tâches (besoins) utilisateur en terme de manipulation de données temporelles. L'agent requête maintient la valeur de la chaîne de caractères représentant la requête utilisateur (requête base de données, nom(s) et emplacement(s) de(s) fichier(s) texte ou fichier(s) XML, etc.). L'agent requête communique avec l'ANF pour envoyer la requête au NF afin d'être exécutée. La Figure 99 présente l'interface de l'agent requête qui représente le point d'entrée pour l'utilisation du système INVEST.

---

<sup>8</sup> Le terme "agent" est utilisé ici au sens d'objet comme défini en programmation par objet.



A partir de l'interface d'entrée, l'utilisateur spécifie sa source de données dans un format déterminé, et les tâches d'analyse de données qu'il désire réaliser. Une extension de INVEST pour gérer d'autres sources de données pose la question d'intégration des données provenant de différentes sources. En effet, il s'agit, par exemple, de permettre à l'utilisateur de corréler une donnée d'un fichier XML avec une autre donnée d'un fichier texte ou avec le résultat d'une requête base de données. Dans la version courante d'INVEST, nous ne traitons pas le problème d'intégration de plusieurs types de données.

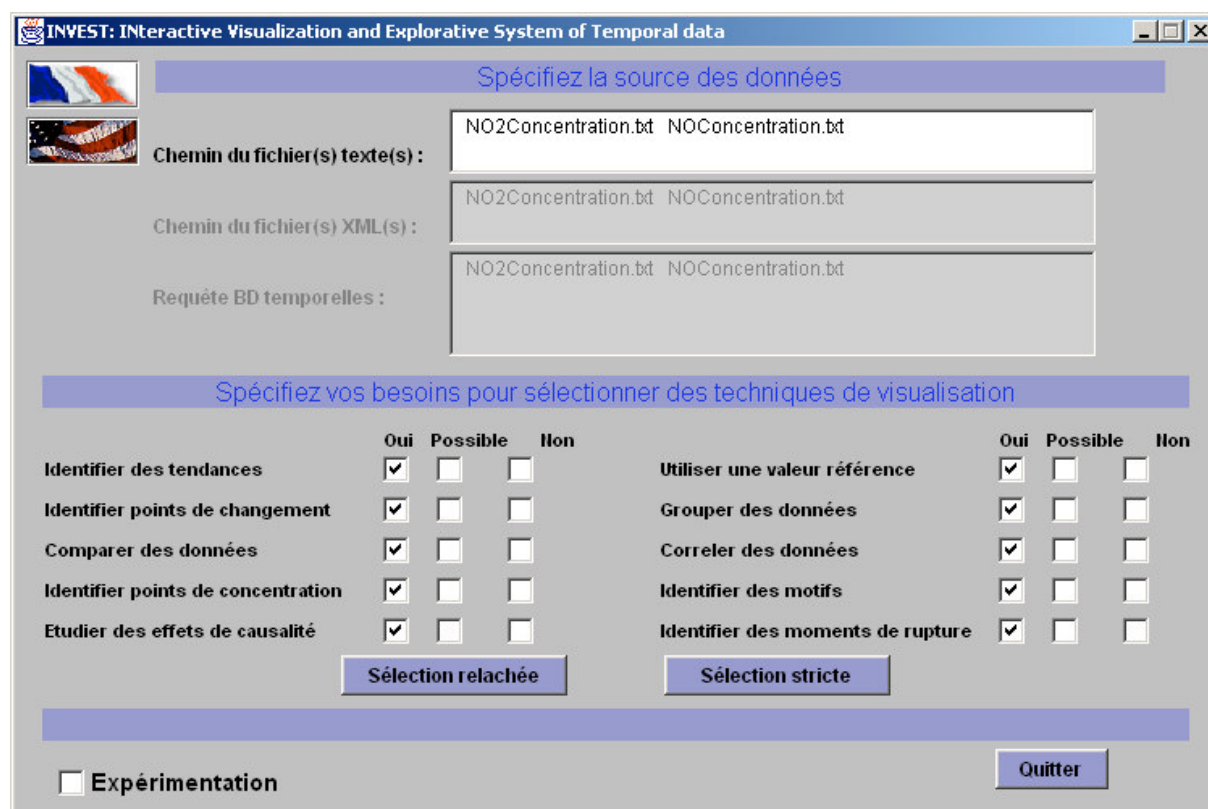


Figure 99 : Spécification explicite des tâches utilisateur et de la source de données.

Dans la hiérarchie des agents PAC du CD, nous utilisons un agent Ciment pour respecter les règles de composition du modèle PAC-Amodeus : un agent PAC doit avoir un père et un seul. Dans INVEST l'agent Ciment connaît les cinq techniques de visualisation que nous avons présentées dans le chapitre précédent et l'interface de sortie (voir Figure 101). L'agent ciment concrétise donc le patron de conception *vues multiples* défini par le modèle PAC-Amodeus : les vues multiples d'un même concept sont gérées par un agent *vues multiples* [Nigay 94]. L'agent Ciment joue un rôle essentiel au niveau de la hiérarchie des agents PAC : il implémente un moteur de sélection des techniques de visualisation par rapport aux tâches

utilisateur. Chaque technique de visualisation est modélisée par une hiérarchie d'agents PAC. La technique *représentation en pixels* par exemple, est définie par deux agents (voir Figure 100) : un agent pour gérer l'espace de référence (Es. Ref.) et un agent pour gérer l'espace de travail (Es. Tr.). La communication entre les deux agents fils, c'est-à-dire l'espace de référence et l'espace de travail est assurée par leur agent père, ici celui associé à la technique même.

L'agent Ciment sélectionne la ou les techniques les mieux adaptées aux tâches spécifiées par l'utilisateur. Les deux boutons, "sélection relâchée" et "sélection stricte" de l'interface d'entrée de la Figure 99 permettent à l'utilisateur d'appliquer un *seuil d'acceptabilité* des techniques de visualisation par rapport aux tâches. La sélection relâchée permet une sélection souple des techniques en fixant un seuil faible alors que la sélection stricte fixe un seuil relativement élevé pour garantir une pertinence des techniques sélectionnées par rapport aux tâches spécifiées par l'utilisateur. Dans l'état actuel, nous avons, de manière arbitraire, fixé le seuil de la sélection relâchée à 0.1 et celui de la sélection stricte à 0.5. Nous devons évaluer le fonctionnement du moteur de sélection pour affiner ces valeurs et identifier des seuils aussi pertinents que possible.

**Tabulaire** : Représentation tabulaire (Agenda)

**TCC** : *Technique cercles concentriques*

**Etoile** : *Représentation en étoiles (Radar)*

**Pixels** : *Représentation en pixels*

**DS** : *Diagrammes superposés*

**Es. Ref.** : Espace de référence

**Es. Tr.** : Espace de travail

**Int. Sor** : Interface de sortie

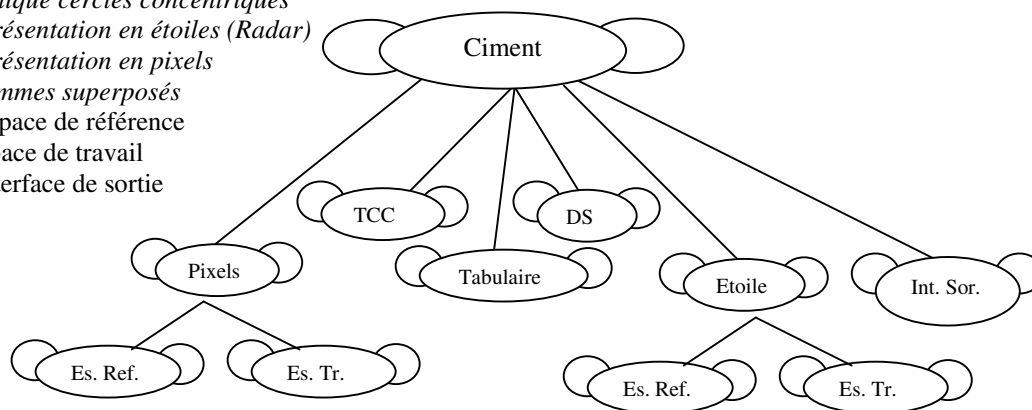


Figure 100 : Structure de l'agent ciment de INVEST.

INVEST utilise deux approches pour implémenter le moteur de sélection des techniques de visualisation au regard des tâches utilisateur. Soit  $V = \{v_1, v_2, \dots, v_m\}$  l'ensemble des techniques de visualisation de INVEST,  $S$  le seuil d'acceptabilité fixé par l'utilisateur,  $T = \{t_1, t_2, \dots, t_N\}$  l'ensemble des  $N$  tâches fixées par l'utilisateur, et  $F(V_i, T_j)$  l'utilisabilité

(degré de satisfaction) d'une technique de visualisation  $V_i$  pour une tâche  $T_j$ . Les deux approches de sélection implémentées dans INVEST dénotent une sélection absolue et une sélection basée sur les propriétés CARE. Nous présentons dans la suite chacune de ces deux approches de sélection.

➤ **Sélection absolue**

Le mode de sélection absolue permet de sélectionner une technique de visualisation  $V_i$  si et seulement si la moyenne des valeurs de la fonction d'utilisabilité de cette technique définies par rapport à l'ensemble de tâches  $T$  fixées par l'utilisateur est supérieure ou égale au seuil d'acceptabilité  $S$ . La fonction  $A$  sur laquelle est basé l'algorithme de sélection absolue est définie par :

$$A(V_i) \Leftrightarrow \forall t_j \in T, (\sum_{j=1..N}(F(V_i, t_j) / N) \geq S.$$

Comme l'ont confirmé les résultats des deux expérimentations décrites aux début de ce chapitre, une technique ne peut pas répondre à toutes les tâches utilisateur. Dans la plupart des cas, une technique de visualisation répond à un sous-ensemble des tâches spécifiées par l'utilisateur. Prenant par exemple le cas où l'utilisateur veut corréler deux données et identifier des points de concentration de certaines valeurs, il peut utiliser une première technique, *cercles concentriques* par exemple, pour corréler les deux données et une seconde technique, *représentation en pixels* par exemple, pour identifier des points de concentration. Les deux techniques sont complémentaires par rapport au besoin de l'utilisateur défini par ces deux tâches. Se limiter à l'utilisation du mode de sélection absolue n'est donc pas satisfaisant pour considérer en particulier la complémentarité (propriétés CARE) entre techniques.

➤ **Sélection basée sur CARE**

Les propriétés CARE (définies dans l'annexe A, page 189) sont les initiales des quatre propriétés pour l'usage des modalités en entrées : **C**omplémentarité, **A**ssignation, **R**edondance et **E**quivalence [Coutaz 95, Martin 93]. Nous utilisons ces propriétés pour modéliser les relations qui peuvent exister entre les différentes techniques de visualisation par rapport à un ensemble de tâches. Nous définissons ces propriétés puis nous expliquons leur implémentation au sein du moteur de sélection d'INVEST.

**Complémentarité entre techniques de visualisation de données temporelles.** Deux techniques  $V_1$  et  $V_2$  sont complémentaires par rapport à un ensemble de tâches  $T$  représentant le besoin de l'utilisateur, si et seulement s'il existe au moins une tâche  $t_i$  qui peut être satisfaite par  $V_1$  et non pas par  $V_2$  ou inversement. La fonction booléenne  $C$  qui teste la complémentarité entre deux techniques de visualisation est définie par :

$$C(V_1, V_2) \Leftrightarrow \forall V_1, V_2 \in V, V_1 \neq V_2, \\ \exists t_i \in T / (F(V_1, t_i) = 0 \text{ et } F(V_2, t_i) > 0) \text{ or } (F(V_1, t_i) > 0 \text{ et } F(V_2, t_i) = 0)$$

**Assignment d'une technique de visualisation à une tâche utilisateur.** Une technique de visualisation  $V$  est assignée à une tâche  $t_i$  uniquement lorsque cette technique peut répondre à cette tâche utilisateur. La technique *représentation tabulaire* présentée dans le chapitre précédent est assignée à la tâche d'identification de motif. En effet, dans le jeu de techniques proposée, seule cette technique peut être utilisée pour cette tâche utilisateur. La fonction  $A$  qui teste l'assignation d'une technique de visualisation à une tâche, est définie par :

$$A(V_k, t_i) \Leftrightarrow V_k \in V, \forall V_j \in V, V_j \neq V_k, t_i \in T, F(V_k, t_i) > 0 \text{ et } F(V_j, t_i) = 0$$

**Equivalence entre techniques de visualisation.** Deux techniques  $V_1$  et  $V_2$  sont équivalentes par rapport à un ensemble de tâches  $T = \{t_i\}$  représentant le besoin de l'utilisateur, si et seulement si  $V_1$  et  $V_2$  répondent à l'ensemble des tâches  $T$  avec le même degré de satisfaction. Nous distinguons deux cas d'équivalence entre techniques de visualisation de données temporelles :

- Equivalence tâche par tâche : l'équivalence dans ce cas porte sur chaque tâche de l'ensemble  $T$ . La fonction  $E$  représentant l'équivalence entre deux techniques de visualisation est dans ce cas définie par :

$$E(V_1, V_2) \Leftrightarrow \forall V_1, V_2 \in V, V_1 \neq V_2, \forall t_i \in T \\ (F(V_1, t_i) > 0 \text{ et } F(V_2, t_i) > 0) \text{ et } (|F(V_1, t_i) - F(V_2, t_i)| \leq \epsilon)$$

Le paramètre  $\epsilon$  désigne le facteur de tolérance de l'équivalence entre deux techniques de visualisation. Une valeur de  $\epsilon = 0$  indique une équivalence absolue, qui est une situation très difficile à atteindre du fait que les valeurs de la fonction  $F$  sont obtenues empiriquement. Le bon choix de la valeur de  $\epsilon$  influence largement sur la performance du moteur de sélection des techniques pour répondre au besoin de l'utilisateur. Une valeur réaliste de la valeur de  $\epsilon$  pour le cas de l'équivalence tâche par tâche est celle de la variation entre les valeurs d'utilisabilité

des techniques pour une tâche donnée. Cette valeur a été définie dans la section expérimentation pour la tâche de navigation. La valeur de  $\epsilon$  est donc égale à 0.03.

- Equivalence globale : l'équivalence dans ce cas porte sur les tâches de l'ensemble T prises globalement. L'ensemble T est défini par N tâches utilisateur. La fonction d'équivalence est dans ce cas définie par :

$$E(V_1, V_2) \Leftrightarrow \forall V_1, V_2 \in V, V_1 \neq V_2, \forall t_i \in T, \\ (|\sum_{i=1..N}(F(V_1, t_i) / N) - (\sum_{i=1..N}(F(V_2, t_i) / N)| \leq \epsilon))$$

**Redondance entre techniques de visualisation.** La redondance entre techniques de visualisation dénote l'utilisation, en séquence, ou en parallèle, de plusieurs techniques équivalentes pour répondre au besoin de l'utilisateur. Ainsi, deux types de redondance sont possibles en fonction du type d'équivalence considérée. La redondance est absolue dans le cas d'une équivalence tâche par tâche, et elle est partielle dans le cas d'une équivalence globale.

Bien que le moteur de sélection dispose de deux modes de sélection des techniques de visualisation adaptées aux tâches utilisateur, c'est le mode de sélection absolue qui est utilisé par défaut. L'utilisation des propriétés CARE pour la sélection des techniques de visualisation serait plus pertinente avec un plus grand nombre de techniques de visualisation de données temporelles. Il convient donc d'alimenter INVEST par de nouvelles techniques de visualisation. En effet, INVEST comporte aujourd'hui seulement cinq techniques de visualisation tout en considérant un grand nombre de tâches utilisateur qui dans certains cas ne sont pas couverts par les techniques de visualisation proposées. Pour cette raison, nous n'avons pas évalué le fonctionnement du moteur de sélection.

Comme l'illustre la Figure 101, les techniques sélectionnées par l'agent ciment sont envoyées à l'interface de sortie et sont classées selon leur niveau de satisfaction estimé par rapport aux tâches annoncées par l'utilisateur. L'interface de sortie active la technique pour laquelle la valeur de la fonction d'acceptabilité est la plus élevée. L'utilisateur peut ainsi manipuler les données, passer d'une technique à une autre pour observer les données selon différents modes de représentation. L'utilisateur peut recommencer une nouvelle session d'analyse de données en revenant à l'interface d'entrée (bouton accueil). Il peut ainsi spécifier de nouvelles sources de données ou encore de nouvelles tâches.

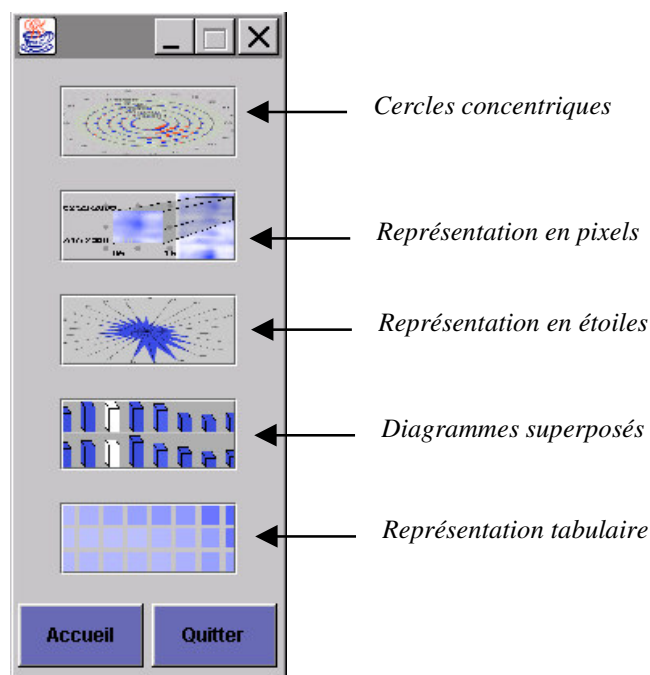


Figure 101: Palette de techniques de visualisation de INVEST.

## 4 Conclusion

Dans ce chapitre, nous avons présenté INVEST, la plateforme que nous avons développée pour la gestion de vues multiples et complémentaires dans la perspective de l'analyse visuelle de données temporelles. Nous avons tout d'abord présenté les résultats de deux évaluations des techniques de visualisation développées et présentées au chapitre précédent. Ces deux évaluations nous ont permis d'évaluer l'utilisabilité de chaque technique de visualisation au regard de chaque tâche utilisateur. Les résultats de ces évaluations sont exploités par l'algorithme de sélection des techniques de visualisation par rapport aux tâches utilisateur. De plus, ces deux expérimentations nous ont permis de vérifier deux hypothèses de travail que nous avons présentées sous la forme de recommandations pour la conception de systèmes pour l'analyse visuelle de données temporelles. En particulier, nous préconisons d'une part, d'utiliser plusieurs vues (techniques) du même espace de données pour répondre à la diversité des tâches utilisateur et d'autre part, d'utiliser un espace de travail pour manipuler les données en parallèle avec un espace de référence.

Nous avons par la suite présenté le modèle d'architecture logicielle d'INVEST, conçu selon le modèle PAC-Amodeus sur lequel nous nous sommes appuyés pour l'intégration logicielle. En particulier, nous avons présenté le moteur de sélection des techniques de visualisation

adaptées aux tâches utilisateur. L'implémentation de ce moteur de sélection se base sur les valeurs de la fonction d'utilisabilité issues des deux expérimentations. Nous notons que le système INVEST est de l'ordre de vingt cinq mille lignes de code Java.

Tandis que chaque technique a été expérimentalement évaluée par des utilisateurs géographes utilisant un vrai espace de données temporelles, il convient maintenant de mener une évaluation expérimentale d'INVEST et de son moteur de sélection. Il s'agit ici d'une des perspectives à nos travaux que nous présentons dans le chapitre suivant qui conclut ce mémoire.

## *Chapitre 6*

### *Conclusion*

“Would you tell me, please,  
which way I ought to go from here?”

“That depends a great deal on where you  
want to get to,” said the Cat

**Lewis Carroll**



Table des matières

**Chapitre 6.** Conclusion.

<a href="#">1</a>	<a href="#">Résumé des contributions</a>	161
<a href="#">2</a>	<a href="#">Limites et extensions</a>	163
<a href="#">2.1</a>	<a href="#">Réalisation logicielle</a>	164
<a href="#">2.2</a>	<a href="#">Evaluation</a>	164
<a href="#">3</a>	<a href="#">Prolongements</a>	165
<a href="#">3.1</a>	<a href="#">Prolongements en Interaction Homme-Machine</a>	165
<a href="#">3.2</a>	<a href="#">Prolongement en Base de Données</a>	167

Motivés par le constat que les données temporelles sont manipulées dans de très nombreuses applications, nos travaux contribuent à la conception et à la réalisation de techniques d'interaction pour l'analyse visuelle de données temporelles. En effet, le problème actuel n'est plus tant la disponibilité des données et en particulier des données temporelles mais la capacité de sélection d'une information qui réponde à la tâche utilisateur grâce à la mise en œuvre d'une ou de plusieurs techniques d'interaction. La multiplicité des techniques d'interaction dédiées aux données temporelles témoigne du dynamisme de cet axe d'étude.

Nous proposons ici, en conclusion, un résumé de nos contributions en soulignant leur originalité. Une prise de recul par une analyse critique des résultats permet d'envisager pour nos travaux de multiples perspectives, que nous organisons en deux parties : les extensions et les prolongements à plus long terme.

### **1 Résumé des contributions**

De très nombreuses techniques de visualisation et plus généralement d'interaction existent pour manipuler des données temporelles. Dans ce contexte, aux avancées brutales se traduisant par une profusion de techniques d'interaction, notre contribution concerne l'établissement de référentiels applicables à la conception afin de mieux cerner et comprendre l'espace de conception de techniques d'interaction dédiées à des données temporelles.

Notre démarche d'analyse s'appuie en premier lieu sur les caractéristiques des données manipulées, en l'occurrence des données temporelles (Chapitre 2). Or ces dernières sont d'une part manipulées par des utilisateurs et d'autre part gérées par un système informatique. L'originalité de notre analyse tient à sa capacité à prendre en compte à la fois l'utilisateur et le système : ainsi nous avons établi un ensemble de caractéristiques liées à la dimension temporelle de la donnée qui sont à la fois issues du domaine de la Psychologie de la Perception (composante utilisateur) et de celui des Bases de Données Temporelles (composante système).

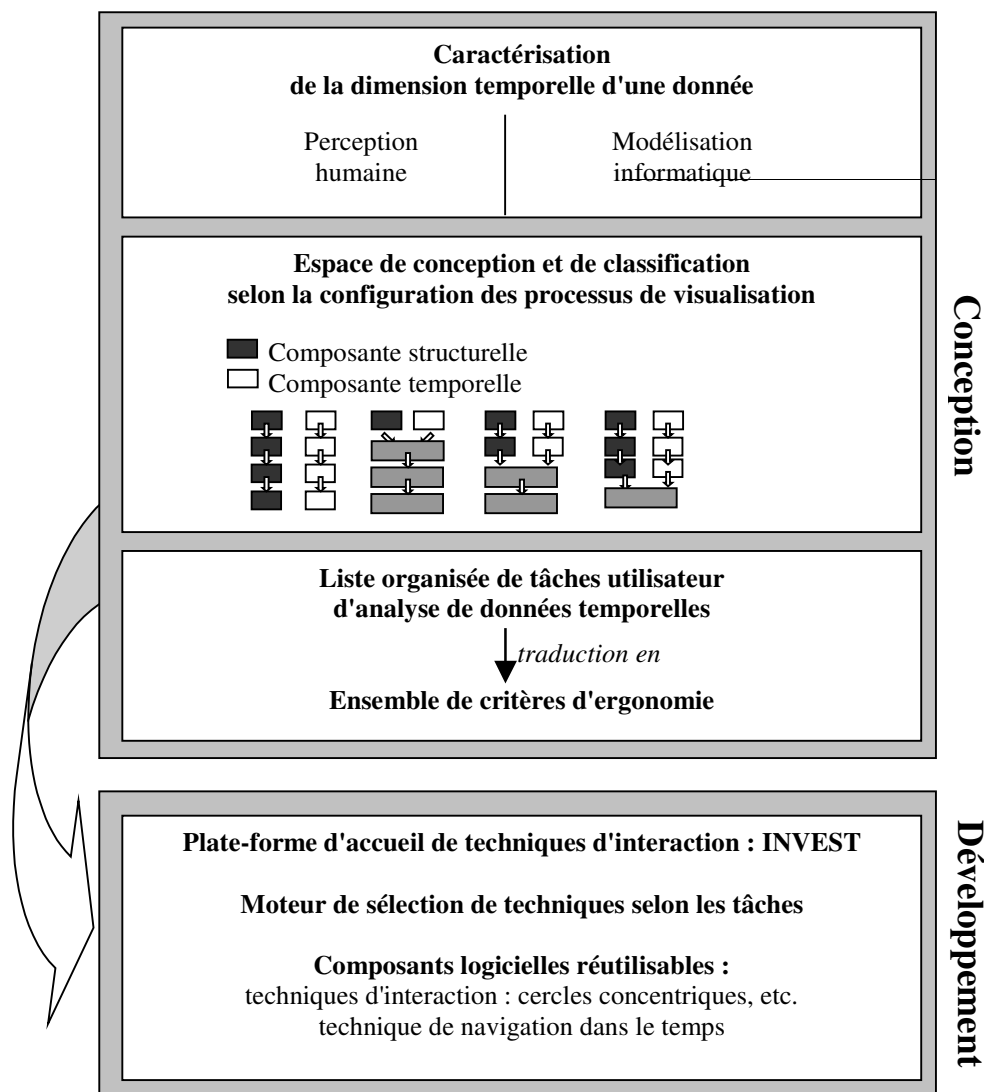
Notre étude des techniques d'interaction existantes repose sur les caractéristiques d'une donnée temporelle, ces dernières sont organisées en deux facettes : les caractéristiques structurelles et celles temporelles. Nous jetons alors un regard nouveau et structurant sur les techniques d'interaction existantes en considérant le processus de visualisation des deux

facettes d'une donnée temporelle (Chapitre 3). Nous cernons alors l'espace des possibilités et classons les techniques existantes en considérant de façon systématique les différentes configurations des deux processus de visualisation mis en jeu. Notre exploration méthodique de l'espace des possibilités nous a permis d'identifier deux classes de techniques d'interaction (deux configurations des processus de visualisation) qui ont peu été étudiées à ce jour.

L'espace des possibilités cerné, nous proposons une démarche de conception de techniques d'interaction dédiées à des données temporelles (Chapitre 4). Notre démarche est centrée sur la tâche de l'utilisateur, ses besoins et les caractéristiques des données temporelles. Elle consiste à traduire les besoins des utilisateurs en termes de propriétés ergonomiques. Les propriétés ergonomiques représentent des contraintes générales qui doivent être vérifiées par la technique d'interaction à concevoir. Nous illustrons la faisabilité de notre démarche de conception par la proposition de cinq nouvelles techniques d'interaction pour visualiser des données temporelles : *cercles concentriques*, *représentation en étoiles*, *représentation tabulaire*, *diagrammes superposés*, et *représentation en pixels*. Ces techniques ont été conçues à partir d'une liste organisée de tâches, que nous avons élaborée grâce à une collaboration étroite avec le laboratoire SEIGAD (Systèmes Environnementaux, Information Géographique et Aide à la Décision) de Géographie de l'Université de Grenoble.

Nous validons ces résultats de conception ergonomique avec la réalisation d'INVEST (Chapitre 5) qui inclut les techniques d'interaction conçues. Au lieu de nous limiter à développer et à évaluer expérimentalement les techniques d'interaction indépendamment, nous avons décidé de développer une plate-forme d'accueil pour la gestion de plusieurs vues et qui aujourd'hui contient nos cinq techniques d'interaction. L'originalité d'INVEST est aussi de contenir un moteur de sélection de la technique la plus adaptée à la tâche de l'utilisateur. Pour mettre en œuvre ce moteur de sélection, nous avons exploité les résultats expérimentaux des évaluations de chaque technique d'interaction considérée isolément. Enfin, de ce développement, selon le modèle d'architecture PAC-Amodeus, nous retenons aussi des composants logiciels réutilisables comme ceux des techniques d'interaction ou le composant (Java Bean) dédié à la représentation de la dimension temporelle de la donnée et aux tâches correspondantes de navigation dans le temps.

L'approche de travail adoptée et les résultats se résument par le schéma suivant. Il comporte deux parties structurantes qui sont corrélées et dédiées aux grandes étapes du cycle de vie d'un système interactif : la conception et le développement.



## 2 Limites et extensions

Le développement d'une plate-forme logicielle intégrant plusieurs techniques d'interaction différentes est certes une tâche ambitieuse dans le temps imparti d'une thèse. Aussi la plate-forme INVEST est incomplète. Nos perspectives à court terme visent donc la complétude de la plate-forme INVEST : réalisation logicielle et évaluation expérimentale.

### **2.1 Réalisation logicielle**

Nous identifions de nombreuses extensions logicielles à la plate-forme INVEST, ces dernières concernent différents composants du modèle d'architecture logicielle PAC-Amodeus. Par rapport au Noyau Fonctionnel (NF), nous envisageons d'étendre INVEST pour gérer d'autres sources de données, en particulier des fichiers XML et une base de données temporelles. Pour ceci, il nous faut implémenter les deux adaptateurs (ANF) correspondants. Par rapport au Contrôleur de Dialogue (CD), nous envisageons de concevoir et développer de nouvelles techniques de visualisation de données temporelles ou d'intégrer des techniques existantes. L'alimentation d'INVEST par de nouvelles techniques nous permettra de couvrir l'ensemble des tâches utilisateur. Il convient de noter que les cinq techniques développées sont des représentations en deux dimensions (2D). Nous envisageons de concevoir des techniques en trois dimensions (3D) et d'étudier l'apport d'une représentation 3D par rapport à une représentation 2D pour assister les utilisateurs dans leurs tâches d'analyse de données temporelles.

### **2.2 Evaluation**

Les données temporelles quantitatives sont fréquemment manipulées dans plusieurs domaines d'applications. Nous envisageons de conduire de nouvelles expérimentations dans le cadre d'autres domaines d'applications que celles géographiques, comme des applications cliniques et/ou financières avec des utilisateurs et des données du domaine en question. Nous pensons que l'interprétation des signes graphiques ainsi que la tolérance à l'erreur dépendent du domaine applicatif. L'objectif de l'expérimentation est d'identifier les seuils d'acceptabilité des différentes techniques de visualisation au regard des tâches utilisateur et ceci dans le cadre des applications cliniques et/ou financières. Pour cette raison, il nous faut tout d'abord actualiser la liste des tâches utilisateur définie dans le Chapitre 4 par rapport à ces deux domaines d'application. Un test ANOVA (analyse de la variance) nous permettra de savoir si l'utilisabilité des techniques de visualisation dépend du profil de l'utilisateur ou non. Il s'agit d'étudier si les seuils d'acceptabilité et le profil utilisateur sont deux variables dépendantes ou indépendantes.

L'alimentation du système INVEST par de nouvelles techniques de visualisation de données temporelles rend possible une évaluation plus détaillée du moteur de sélection. Sur la base de

cette évaluation, nous pourrons tout d'abord identifier les bonnes valeurs des seuils d'acceptabilité et par la suite comparer les deux approches de sélection : sélection d'une seule technique ou sélection de plusieurs techniques complémentaires.

### **3 Prolongements**

Nous retrouvons dans nos prolongements à plus long terme le caractère pluridisciplinaire de nos travaux. Aussi nous les présentons en deux parties : ceux concernant le domaine de l'Interaction Homme-Machine et ceux liés au domaine de la Base de Données.

#### **3.1 Prolongements en Interaction Homme-Machine**

Nous avons limité nos travaux à l'étude des techniques d'interaction *visuelle* (type du rendu) de données *temporelles quantitatives* (type de données) dans le contexte d'un travail *mono-utilisateur* (type de l'interaction). Par rapport à chacun des mots en gras-italique nous envisageons une extension.

##### **➤ *Sonification des données temporelles***

Peu de travaux de recherche dans la littérature se sont intéressés à la sonification des données temporelles et peu encore pour des fins d'analyse [Hermann 99, Flowers 01, Madhyastha 95]. Bien que la sonification présente un intérêt certain lorsque l'utilisateur ne peut porter son attention sur l'écran comme en situation de mobilité [Walker 00], nous envisageons plutôt de considérer l'usage synergique du son avec la visualisation. Notre approche est d'augmenter les techniques de visualisation d'INVEST par une représentation sonore des données pour faciliter la réalisation des tâches utilisateur. Par exemple, dans le cas où les valeurs des données à visualiser sont très proches, l'identification des points de concentration d'une classe de valeurs ou l'identification des maxima locaux dans l'espace de représentation devient plus facile si le rendu visuel est accompagné d'un rendu sonore. Nesbitt et Barrass [Nesbitt 02] ont adopté cette approche dans le cas de la visualisation de données financières temporelles sur des dispositifs dont l'espace d'affichage est limité, comme l'écran d'un ordinateur de poche.

##### **➤ *Extension d'INVEST à d'autres types de données temporelles***

Dans nos travaux, nous avons considéré des données temporelles quantitatives avec un domaine temporel ancré. Rappelons que le domaine temporel est l'ensemble des moments

temporels estampillant les valeurs des données. Il est dit ancré quand ces moments sont connus et fixes, dans le cas contraire il est dit non-ancré. Nous envisageons étudier d'autres types de données temporelles, en particulier celles dont le domaine temporel est non-ancré comme les données multimédias.

Plusieurs travaux [Harrison 94, Hibino 96, Hibino 98] se sont intéressés à la conception de techniques de visualisation de données multimédias. Une première étape consiste donc à étudier les tâches utilisateur pour l'analyse des données dont le domaine temporel est non-ancré. Nous étudierons par la suite la réutilisation du composant ligne temps ainsi que son utilisabilité pour ce type de données. Nous envisageons la vidéo comme type de donnée pour cette étude.

### ➤ *Utilisation d'un dispositif physique dédié pour l'interaction avec l'espace temps*

Nous avons conçu et développé un composant logiciel dédié à l'espace temps et à la navigation dans le temps. Ce composant logiciel (Java bean) est par exemple exploité par plusieurs des techniques que nous avons développées. Dans l'objectif de rendre ce composant logiciel encore plus indépendant, nous lui avons dédié un dispositif physique. En particulier, nous l'avons porté sur un ordinateur de poche de type iPaq. L'utilisateur a alors deux surfaces d'interaction, l'écran de la station de travail et celui de l'ordinateur de poche, cette deuxième surface étant dédiée à la navigation dans le temps.

Nous notons que dans le cas d'une technique de visualisation utilisant deux lignes temps, la technique *cercles concentriques* par exemple, l'utilisation de ce dispositif soulève le problème de l'identification de l'espace temps à représenter sur le dispositif physique à un moment donné. En d'autres termes, quel est le composant graphique concerné par les tâches effectuées au niveau du dispositif temps ? Ce problème est connu sous le nom "fenêtre active". Il est souvent rencontré dans le cadre d'une interaction multimodale qui exploite la parole pour manipuler des objets graphiques : quel est l'objet graphique concerné par la commande vocale courante ? Grasso [Grasso 96] décrit ce problème en ces termes : "Avec la manipulation directe, il n'y a pas d'ambiguïté de contexte de référence possible. Quand l'utilisateur sélectionne un objet, il n'y a aucune interprétation incorrecte possible quant à

l'objet sélectionné. Le revers est qu'il n'y a qu'un seul moyen de désigner un objet. La parole est tout l'inverse"<sup>9</sup> Extrait de [Grasso 96].

Nous envisageons tout d'abord d'apporter des solutions à ce problème dit de "fenêtre active" puis d'étudier les avantages et les inconvénients de l'utilisation d'un tel dispositif pour l'interaction avec l'espace des données temporelles. Ce prolongement concerne deux axes de recherche en IHM, l'interaction multisurface [Coutaz 03] et l'interaction bi-manuelle [Leganchuk 98].

### ➤ *Analyse visuelle multi-utilisateurs de données temporelles*

Notre approche de conception des techniques de visualisation est centrée sur les tâches utilisateur dans le contexte d'une session mono-utilisateur d'analyse visuelle de données temporelles. L'intérêt d'une analyse multi-utilisateurs (dite aussi collaborative) est la complémentarité de compétences et l'échange d'expériences. Dans ce contexte, la décision prise au regard d'une tâche donnée devient plus robuste et plus sûre.

La visualisation collaborative de données est un axe de recherche active dans le cadre des applications de Réalité Virtuelle [Brewer 00]. Nous envisageons de jeter un pont entre l'analyse collaborative de données et la visualisation collaborative sans autant travailler dans le domaine de la Réalité Virtuelle. En effet, nous visons à étendre INVEST à l'analyse visuelle multi-utilisateurs. Pour se faire, il convient d'étudier l'activité d'un groupe en situation d'analyse collaborative de données temporelles. L'étape suivante consiste alors à adapter l'architecture logicielle d'INVEST à une architecture de système collaboratif. La thèse de Laurillau [Laurillau 02] présente les fondements nécessaires pour réaliser cette étape.

## **3.2 Prolongement en Base de Données**

D'un point de vue base de données, nous envisageons de traduire les tâches utilisateur identifiées en termes d'opérateurs. Les langages de requêtes sont limités à spécifier les besoins utilisateur en termes de tâches d'interrogation (sélection, projection, etc.) de la base de données. Ayant établi une liste de tâches utilisateur d'analyse de données visuelles, nous

---

<sup>9</sup> Citation :

"With direct manipulation interfaces, there is no reference ambiguity. When the user selects an object, there is no way the computer will misinterpret this selection as some other object. The down side to this is that there is only one way to reference an object....Speech is just the opposite." Extrait de [Grasso 96].



envisageons d'étendre le langage de requête pour traiter ces tâches : cette extension consiste à permettre de spécifier la tâche d'analyse dans l'opérateur afin qu'en sortie, la technique d'interaction la mieux adaptée à la tâche en question soit sélectionnée. Nous envisageons ainsi de définir des opérateurs de type `correlate` ou `identify_concentrations`. Tout en interrogeant la base de données, l'utilisateur annonce donc sa tâche. A notre connaissance, aucun langage de requête ne permet de spécifier les tâches d'analyse de données. Des travaux similaires ont déjà été envisagés mais sans expliciter la tâche de l'utilisateur : c'est le cas des langages animés de requête [Rex 96]. Un exemple de requête est le suivant : "ANIMATE soil\_moisture FOR sim\_time = EXISTS IN interest\_area SPACE BEGIN 03-01-90:00.00 END 6-18-90:18.00 INCREMENT :12.00 DISPLAY 10." Extrait de [Rex 96]. L'opérateur ANIMATE de la requête désigne une technique animée pour visualiser le résultat de la requête sans que la tâche utilisateur soit spécifiée. Nous envisageons donc d'aller plus loin que de spécifier la forme de sortie pour le résultat d'une requête. Notre approche consiste à spécifier la tâche d'analyse de données au niveau de la requête et le système choisit alors automatiquement la technique de visualisation la plus adaptée.

## *Bibliographie*



Cette bibliographie est structurée en deux parties : des références papiers et des sites Web.

## **1 Références papiers**

[Abowd 92] Abowd G.D., Coutaz Joëlle, Nigay Laurence, Structuring the space of Interactive System Properties. Engineering For Human-Computer Interaction, IFIP 10-14 August 1992.

[Agrawal 95a] Agrawal Rakesh, Lin King-Ip, Sawhney Harpreet S. and Shim Kyuseok. Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases. Proceedings of the 21<sup>st</sup> VLDB Conference, Zürich, Switzerland, 1995.

[Agrawal 95b] Agrawal Rakesh, Psaila Giuseppe, Wimmers Edward L. and Zzit Mohamed. Querying Shapes of Histories. Proceedings of the 21<sup>st</sup> VLDB Conference, Zürich, Switzerland, 1995.

[Allen 91] Allen James F. Time and Time again : the Many Ways to Represent Time. International Journal of Intelligent Systems, 6(4), July 1991, P : 341-355.

[Ankerst 00] Ankerst Mihael, Visual Data Mining. Thèse de doctorat en Informatique de l'Université Ludwig-Maximilians, München.

[Ankerst 01] Ankerst Mihael, Visual Data Mining with pixel-oriented Visualization Techniques. ACM SIGKDD Workshop on Visual Data Mining, San Francisco, CA, 2001.

[Ankerst 96] Ankerst Mihael, Keim Daniel. A., Kriegel Hans-Peter: Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets , Visualization `96, Hot Topic Session, San Francisco, CA, 1996.

[Antunes 01] Antunes Claudia M. and Oliveira Arlindo L. Temporal Data Mining : an Overview. Workshop on Temporal Data Mining. 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. August 26, 2001, San Francisco, CA, USA

[Arning 96] Arning Andreas, Agrawal Rakesh, Raghavan Prabhakar. A Linear Method for Deviation Detection in Large Databases. KDD Conference Proceedings, 1996, P : 164-169.

[Bass 01] Bass Len, John Bonnie E. and Kates Jesse. Achieving Usability Through Software Architecture. Technical report, CMU/SEI, March 2003. Accessible à partir de : <http://www.sei.cmu.edu/publications/documents/01.reports/01tr005.html>.

[Bastien 93] Bastien J.M Christian et Scapin Dominique L. Ergonomic Criteria for the Evaluation of Human-Computer Interfaces. Rapport technique de l'Institut National de Recherche en Informatique et en Automatique (INRIA), N° 156, Juin 1993.

[Baudel 02] Baudel Thomas. Visualisations Compactes : Une Approche Déclarative pour la Visualisation d'Information. Actes de la conférence IHM, Poitiers, France, Novembre 26-29, 2002, P : 161-168.

[Becker 94] Becker Richard A., Clark Linda A. et Lambert Diane. Cave Plots: A Graphical Technique for Comparing Time Series. J. of Computational and Graph. Stat., 3:3 (1994), P : 277-283.

[Bertin 67] Bertin J. Sémiologie Graphique : les Diagrammes, les Réseaux, les Cartes. Editions Gauthier-Villars, Paris, 1967.

[Bertin 81] Bertin Jacques, Graphics and Graphic Information Processing, Walter de Gruyter, Berlin New York, 1981.

[Brewer 00] Brewer Isaac, MacEachren Alan M., Abdo Hadi, Gundrum Jack, George Otto. Collaborative Geographic Visualization: Enabling shared understanding of environmental processes. IEEE Information Visualization Proceedings, 9-10 October, 2000.

[Bruley 99] Bruley Christophe, Analyse des Représentations Graphiques de l'Information, Extension aux Représentation Tridimensionnelles. Thèse de doctorat en Informatique de l'Université Joseph-Fourier, Grenoble, France, 1999.

[Bui 98] Bui Alex A.T., Aberle Denise R., McNitt-Gray Michael F., Cardenas Alfonso F., Goldin Jonathan, et M.D. The Evolution of an Integrated TimeLine for Oncology Patient Healthcare. Annual symposium of Amia, November 10, 1998.

[Cailleteau 99] Cailleteau Laurent and Plaisant Catherine, Interfaces for Visualizing Multi-Valued Attributes: Design and Implementation Using Starfield Displays, Partial requirement for the Diplôme d'Etudes Approfondies, August 26, 1999.

[Callendar 01] Callendar Craig et Edney Ralph. *Introducing Time*. Publié par Icon Books Ltd., UK 2001.

[Canavaggio 97] Canavaggio Jean-François. *Tempos : un modèle d'historique pour un SGBD temporel*. Thèse de doctorat en Informatique à l'Université Joseph-Fourier, Grenoble France, 1997.

[Card 96] Card Stuart K., Robertson George G., York William. *The WebBook and the Web Forager: An Information Workspace for the World-Wide Web (1996)*. Proceedings of the Conference on Human Factors in Computing Systems CHI'96.

[Card 97] Card Stuart K. and Mackinlay Jock. *The Structure on the Information Visualization Design Space*. Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97).

[Carlis 98] Carlis J.V. and Konston Joseph A. *Interactive Visualization of Serial Periodic Data*, UIST'98, ACM, San Francisco, Ca, 1998.

[Casner 91] Casner S.. *A task-Analytic Approach to the Automated Design of Graphic Presentations*. ACM Transactions on Graphics, vol. 10, N° 2, April 1991, P : 111-151.

[Cattell 00] Cattell R.G.G et Barry D., éditeurs. *The Object Database Standard : ODMG 3.0*. Morgan Kaufmann. January 2000.

[Chandra 94] Chandra Rakesh and Segev Arie. *Implementing Calendars and Temporal Rules in Next Generation Databases*. IEEE International Conference on Data Engineering. 1994. P : 264-273.

[Chenet 00] Chenet François. *Le temps : Temps Cosmique, Temps Vécu*. (Collection U. Philosophie). Armand Colin, 2000.

[Chi 00] Chi Ed H. *A Taxonomy of Visualization Techniques using the Data State Reference Model*. Proceedings of InfoVis 2000 (Salt Lake City UT, October 2000), IEEE Computer Society, P : 69-75.

[Chi 02a] Chi Ed H. *Expressiveness of the Data Flow and Data State Models in Visualization Systems*. In Proc. of the Advanced Visual Interfaces Conference, ACM Press, May 2002, Italy.

[Chi 02b] Chi Ed H. A Framework for Visualizing Information. Human-Computer Interaction Series. Kluwer Academic Publishers, 2002.

[Chi 98] Chi Ed H., Pitkow James, Mackinlay Jock, Pirolli Petre, Gossweiler Rich, et Card Stuart K. Visualizing the Evolution of Web Ecologies. Proceedings of the Conference on Human Factors in Computing Systems CHI'98.

[Chomiki 94] Chomiki Jan. Temporal query languages : a survey. In Proceedings of the International Conference on Temporal Logic, Bonn, DE, 1994.

[Chuah 95] Chuah Mei C., Roth Steven F., Mattis J. and Kolojejchick J. SDM: Selective Dynamic Manipulation of Visualisations. In proceedings of UIST'95 Conference, Pittsburgh PA USA, P: 61-70.

[Church 99] Church Russell M. Evaluation of Quantitative Theories of Timing. Journal of the experimental analysis of behavior. Number 2, March 1999. P : 253-256.

[CNRS 01] Le temps. Le journal du CNRS, numéro 133, Janvier 2001, P : 14-24..

[Cockburn 00] Cockburn Andy et McKenzie Bruce. An evaluation of cone trees. In People and Computers XV. Proceedings of the 2000 British Computer Society Conference on Human-Computer Interaction, University of Sunderland, 4-8 September, 2000.

[Coutaz 03] Coutaz J., Lachenal. C. Ontology for Multi-surface Interaction. In Proc. of Interact. 2003, ACM Press.

[Coutaz 95] Coutaz Joëlle, Nigay Laurence, Salber Daniel, Blandford Ann, Jon May, M. Young Richard, Four Easy Pieces for Assessing the Usability of Multimodal Interaction : The CARE Properties, In proceeding of Interact'95, Edited by Knut Norby, Per H. Helmersen, David J. Gilmore and Svein A. Arnesen, P : 115-120.

[Coutaz 97] Coutaz Joëlle, PAC-ing the Architecture of Your User Interface, DSV-IS'97, 4<sup>th</sup> Eurographics Workshop on Design, Specification and Verification of Interactive Systems, Springer Verlag Publ., P: 15-32.

[Daassi 00] Daassi Chaouki, Dumas Marlon, Fauvet Marie-Christine, Nigay Laurence et Scholl Pierre-Claude. Visual exploration of temporal object databases. Actes de la conférence BDA 2000. (24-27 Octobre, Blois, France), P : 159-178

[Daassi 02] Dâassi Chaouki, Fauvet Marie-Christine, Nigay Laurence. Visual Mining: Representation Multiplicity of Temporal Data. Proceedings de DEXA, Septembre 2-6, 2002, Aix-en-Provence, France, 10 pages.

[Daassi 99a] Dâassi Chaouki. Visualisation de données temporelles. Dans les actes de la Conférence AFIHM sur l'Interaction Homme-Machine, 22-26 Novembre 1999, Montpellier, France. Article de rencontres doctorales. P : 129-131.

[Daassi 99b] Dâassi Chaouki. Techniques de visualisation de données temporelles. Rapport de DEA Système d'Information (MATIS) de l'Université Joseph-Fourier, 1999.

[Damasio 02] Damasio Antonio. La Conscience du Temps. Publié dans la revue Pour la science, Spécial Cerveau. Numéro 302, Décembre 2002.

[Dias 95] Dias Antonio Eduardo, Silva João Pedro et Câmara Antonio S. BITS: Browsing In Time and Space. CHI'95, May 7-11 1995, Denver Colorado, papier court, P: 248-249.

[Dragicevic 02] Dragicevic Pierre et Huot Stéphane. SpiraClock : A Continuous and Non-Intrusive Display for Upcoming Events. Proceedings de CHI'02. 2002.

[Dumas 00] Dumas Menjivar Marlon. Tempos : une plate-forme pour le développement d'applications temporelles au dessus de SGBD à objets. Thèse de doctorat en Informatique à l'Université Joseph-Fourier, Grenoble France, 2000.

[Dumas 98] Dumas Marlon, Fauvet Marie-Christine and Scholl Pierre-Claude, Handling temporal grouping and Pattern-matching queries in a temporal object model. Publié à CIKM'98 Washington DC, novembre 1998.

[Feiner 90] Feiner Steven et Beshers Clifford. Worlds within Worlds : Metaphors for Exploring n-Dimensional Virtual Worlds. Proceedings of UIST'90, Snowbird, UT, October 3-5, 1990, 76-83.

[Flowers 01] Flowers John H., Whitwer Laura E., Grafel Douglas C. et Kotan Cheryl A. Sonification of Daily Weather Records: Issues of Perception, Attention and Memory in Design Choices. Proceedings of the 2001 International Conference on Auditory Display, Espoo, Finland, July 29-August 1, 2001.



[Fredrikson 99] Fredrikson A., C. North, C. Plaisant, and B. Shneiderman. Temporal, Geographical and Categorical Aggregations Viewed through Coordinated Displays : A Case Study with Highway Incident Data. HCIL Technical Report No. 99-31, December 1999.

[Freeman 00] Freeman Walter J. Perception of Time and Causation through the Kinesthesia of intentional action. *Cognitive Processing* 1, P : 18-34.

[Gram 98] Gram and Cockton, Design Principles for Interactive Software, St Edmundsbury Press, 1996. UIST'98, San Francisco, November 1998.

[Grasso 96] Grasso Michael A. Speech Input in Multimodal Environments : A Proposal to Study the Effects of Reference Visibility, Reference Number, and Task Integration. Rapport de recherche de l'Université Maryland. TR CS-96-09, Juillet 1996.

[Hansen 01] Hansen Henning Sten. A Time-Series of Urban Growth in Copenhagen Area. ScanGIS'2001, The 8th Scandinavian Research Conference on Geographical Information Science, Ås, Norway, 25th-27th June 2001, P : 225-235.

[Hao 01] Hao Ming C., Ladisch Julian, Dayal Umeshwar, Hsu Meichun, et Krug Adrian. Visual Mining of E-Customer Behavior Using Pixel Bar Charts. HP Labs Technical Reports, 2001. Cessible à partir de <http://www.hpl.hp.com/techreports/2001/HPL-2001-144.html>.

[Harrington 99] Harrington Deborah L. and Haaland Kathleen Y. Neural Underpinnings of Temporal Processing : A review of Focal Lesion, Pharmacological, and Functional Imaging Research. *Reviews in the Neurosciences*, 10, P : 91-116 (1999).

[Harris 99] Harris, C., Allen, R. B., Plaisant, C., and Shneiderman, B. (June 1999) Temporal Visualization for Legal Case Histories. ASIS'99 Proceedings of the 62nd Annual Meeting of the American Society for Information Sciences, 1999, Vol. 36, P : 271-279.

[Harrison 94] Harrison Beverly L., Owen Russell et Baecker Ronald M. Timelines: An Interactive System for the Collection and Visualization of Temporal Data. Proceedings of Graphics Interface'94, 1994, P: 141-148.

[Hascoët 98] Hascoët Mountaz, Analytical Versus Empirical Evaluation of Spatial Displays. Proceedings of CHI'98 Extended Abstracts, 1998, P : 257-258.

- [Havre 00] Havre Susan, Hetzler Beth and Nowell Lucy. ThemeRiver: Visualizing Theme changes over Time. Proceedings of the IEEE Symposium on Information Visualization 2000.
- [Havre 99] Havre Susan, Hetzler Beth, et Nowell Lucy. ThemeRiver : In Search of Trends, Patterns, and Relationships. Présenté au IEEE Symposium on Information Visualization, InfoVis '99, October 25-26, San Francisco CA.
- [Hermann 99] Hermann T. and Ritter H. Listen to your Data: Model-Based Sonification for Data Analysis. In M. R. Syed, editor, Advances in intelligent computing and multimedia systems. Int. Inst. for Advanced Studies in System Research and Cybernetics, 1999.
- [Herz 01] Herz Andrea V. M. How is time represented in the brain ? Problems in Systems Neuroscience. J.L. Van Hemmen and T.J. Sejnowski (eds.), Oxford Press, Oxford, 2001, P : 1-17.
- [Hibino 96] Hibino S., Rundensteiner E.A. MMVIS: Design and Implementation of a Multimedia Visual Information Seeking Environment. ACM Multimedia'96 Conference Proceedings, NY: ACM Press, 1996, P : 75-86.
- [Hibino 98] Hibino S., Rundensteiner E.A. Comparing MMVIS to a timeline for temporal trend analysis of video data. Advanced Visual Interfaces, May 1998.
- [Hochheiser 02a] Hochheiser Harry, Interactive Querying of Time Series Data. Proceedings de la conférence CHI 2002, Avril 20-25, Minneapolis, Minnesota, USA.
- [Hochheiser 02b] Hochheiser Harry and Shneiderman Ben, A dynamic Query Interface for Finding Patterns in Time Series Data. Proceedings de la conférence CHI 2002, Avril 20-25, Minneapolis, Minnesota, USA.
- [Holmquist 98] Holmquist Lars Eric and Björk, Staffan. Formative Evaluation of Flip Zooming : Towards Effective Integration of Detail and Context on Small Displays'. In Viktoria Research Report VRR-98-1, Gotenburgh: Viktoria Institute.
- [Ivry 00] Ivry Richard B. and Richardson Thomas C. Temporal control and coordination : Multiple Timer Model. In Press, Brain and Cognition, Special issue on Timing. October 29, 2000.

[Jacquemin 02a] Jacquemin Christian and Jardino Michèle. Une interface 3D multi-échelle pour la visualisation et la navigation dans de grands documents XML. In Proceedings, IHM 2002, Poitiers, France, P : 263-266.

[Jacquemin 02b] Jacquemin Christian and Jardino Michèle. Multi-dimensional and Multi-scale Visualizer of Large XML Documents. In Proceedings, EUROGRAPHICS 2002, Saarbrücken, Germany.

[Jensen 98] Jensen C. S. et Dyreson C. E. (Eds). The consensus glossary of temporal database concepts – February 1998 version.

[Jerding 97] Jerding Dean F. et Stasko John T. The Information Mural : A Technique for Displaying and Navigating Large Information Spaces. Technical Report GIT-GVU-97-24, Graphics, Visualization, and Usability Center, Georgia Institute of Technology. Disponible à [http://www.cc.gatech.edu/gvu/softviz/infoviz/information\\_mural.html](http://www.cc.gatech.edu/gvu/softviz/infoviz/information_mural.html)

[Jog 95] Jog Ninad and Shneiderman Ben. Starfield Information Visualization with Interactive Smooth Zooming. . CS-TR-3286, CAR-TR-714, ISR-TR-94-46. IFIP 2.6 Visual Databases Systems Proceedings, P : 1-10, Lausanne, Switzerland. March 27-29,1995.

[Johnson 02] Johnson Ian et Wilson Andrew. The TimeMap Project: Developing Time-Based GIS Display for Cultural Data. Journal of GIS in Archaeology Vol 1. ESRI Inc., Redlands. 2002. Disponible à partir de : <http://www.archaeology.usyd.edu.au/>

[Jonathan 00] Jonathan C. Roberts. Multiple-View and Multiform Visualization. Proceedings of Visual Data Exploration and Analysis VII, Vol. 3960, January 2000, P : 176-185.

[Jueptner 01] Jueptner Markus, Krukenberg Michael, W. Müller Bernhard and Jentzen Walter. The role of the cerebellum and basal ganglia in time estimation. Lessons from PET Studies. Postère à 7th Annual Meeting: Human Brain Mapping - Brighton Neuroimage 13 (2), S987.2001. Disponible à [http://www.uni-essen.de/schizophrenia/Brighton\\_MJ-HBM2001-1.pdf](http://www.uni-essen.de/schizophrenia/Brighton_MJ-HBM2001-1.pdf)

[Keim 96a] Keim D. A.: Pixel-oriented Visualization Techniques for Exploring Very Large Databases, in: Journal of Computational and Graphical Statistics, March 1996, Vol. 5, No. 1, P : 58-77.

- [Keim 96b] Keim Daniel A. Pixel-Oriented Databases Visualization. SIGMOD Record, special issue on “information visualization”, Décembre 1996.
- [Keim 97] Keim D. A.: Visual Techniques for Exploring Databases, Invited Tutorial, Int. Conference on Knowledge Discovery in Databases (KDD'97), Newport Beach, CA, 1997.
- [Keogh 98] Keogh Eamonn J. and Pazzani Michael J. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. Proceedings of the 4th International Conference of Knowledge Discovery and Data Mining, AAAI Press. 1998. P : 239-241.
- [Kerpedjiev 97] Stephan Kerpedjiev, Carenini Giuseppe, Roth Steven F., Moore Johanna D. Integrating Planning and Task-based Design for Multimedia Presentation. In: Proceedings Intelligent User Interfaces, ACM, Orlando, FL, Jan. 6-9 1997, P : 145-152.
- [Kobsa 01] Kobsa Alfred. An Empirical Comparison of Three Commercial Information Visualization Systems. Proceedings of InfoVis'01, IEEE Symposium on Information Visualization, San Diego, CA. 2001.
- [Koike 97] Koike Yuichi, Sugiura Atsushi et Koseki Yoshiyuki. TimeSlider : An Interface to Specify Time Point. UIST 1997, Alberta Canada, P : 43-44
- [Kosara 00] Kosara Robert et Miksch Silvia. A Visualization of Medical Therapy Plans compared to Gantt and PERT Charts. Proceedings of the Seventh International Workshop on Temporal Representation and Reasoning (TIME-00), IEEE Computer Society, P : 153-181, 2000.
- [Kosara 99] Kosara Robert, Miksch Silvia. Visualization Techniques for Time-Oriented, Skeletal Plans Therapy Planning. Proceedings of the Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making (*AIMDM'99*), Springer, Berlin, P : 291-300, 1999.
- [Kullberg 95] Robin L. Kullberg, Dynamic Timelines: Visualizing Historical Information in Three Dimensions. Mémoire de Master à Media Lab. MIT, 1995. Accessible à partir de : <http://robin.www.media.mit.edu/people/robin/thesis/toc.html>

[Laurillau 02] Laurillau Yann. Conception et réalisation logicielles pour les collecticiels centrées sur l'activité de groupe : le modèle et la plate-forme Clover. Thèse de doctorat en Informatique à l'Université Joseph-Fourier, Grenoble, France. 2002.

[Leganchuk 98] Leganchuk Andrea, Zhai Shumin et Buxton William. Manual and Cognitive Benefits of Two-Handed Inputs: An Experimental Study. *ACM Transaction on Computer-Human Interaction*, Vol. 5, N° 4, December 1998, P : 326-359.

[Liston 00] Liston Kathleen, Fischer Martin et Kunz John. Designing and Evaluating Visualization Techniques for Construction Planning. The 8th International Conference on Computing in Civil and Building Engineering, August, 2000 Stanford University Silicon Valley, CA, USA.

[Mackinlay 86] Mackinlay Jock, Automating the Design of Graphical Presentation of Relational Information, *ACM Transaction on Graphics*, vol. 5, No 2, April 1986, P: 110-141.

[Mackinlay 91] Mackinlay Jock D., Robertson George G. and Card Stuart K., The Perspective Wall : Detail and Context Smoothly Integrated. *Proceedings of the CHI '91*, ACM Press, 173-179.

[Mackinlay 94] Mackinlay Jock D., Robertson George G., et DeLine Robert. Developing Calendar Visualizers for the Information Visualizer. Dans *proceedings de UIST 1994*.

[Madhyastha 95] Madhyastha Tara M. et Reed Daniel A. Data Sonification: Do You See What I Hear?, *IEEE Software*, Vol. 12, No. 2, March 1995, P : 45-56.

[Martin 93] Martin Jean-claude and Beroule D., Types et Buts de coopérations entre modalités, *IHM'93*, Cinquième Journées sur l'Ingénierie des Interfaces Homme-Machine, Lyon- 19-20 octobre 1993, P : 17-22.

[Masuda 00] Masuda Hisanori, Imamiya Atsumi, Go Kentaro et Mao Xiaoyang. An Evaluation of a Graphical History Tool with an Eye Tracker. Publié dans 6<sup>th</sup> ERCIM Workshop "User Interfaces for All". CNR-IROE, Florence, Italy. 25-26 Octobre 2000.

[Matković 02] Matković Kresimir, Hauser Helwing, Sainitzer, and Gröller M. Eduard. Process Visualization with Levels of Details. *Proceedings of the IEEE Symposium on Information Visualization 2002 (InfoVis 2002)*, Oct. 28-29, 2002, Boston, MA.

[Miksch 98] Miksch Silvia, Kosara Robert, Shahar Yuval et Johnson Peter. AsbruView : Visualization of Time-Oriented, Skeletal Plans. The fourth International Conference on Artificial Intelligence Planning Systems, juin 7-10, 1998.

[Moon 96] Moon Bongki, Jagadish H.V, Faloutsos Christos and Saltz Joel H. Analysis of the Clustering Properties Hilbert Space-filling Curve. CS-TR-361 and UMIACS-TR-96-20, 1996. <http://www.cs.umd.edu/TR/UMCP-CSD:CS-TR-3611> .

[Nair 99] Nair Dinesh G. Time Perception. Publié à 'Seminar in Cognition'; 1999. Disponible à <http://www.ccs.fau.edu/~nair/>

[Nesbitt 02] Nesbitt Keith V. et Barras Stephen. Evaluation of Multimodal Sonification and Visualization of Depth of Market Stock Data. Proceedings of the 2002 International Conference on Auditory Display, Kyoto, Japan, July, 2002.

[Nigay 94] Laurence Nigay, Conception et modélisation logicielles des systèmes interactifs : application aux interfaces multimodales. Thèse informatique à l'université Joseph-Fourier, 1994.

[Nigay 97] Nigay Laurence and Coutaz Joëlle. Software Architecture Modelling: Bridging Two Worlds Using Ergonomics and Software Properties, pp 49-73 in Formal Methods in Human-Computer Interaction, Palanque and Paterno (eds). Springer-Verlag, Berlin, 1997.

[Norman 86] D. A. Norman, Cognitive Engineering, User Centered System Design, New Perspectives on Computer Interaction, édité par D. A. Norman, S.W. Draper, Hillsdale, New Jersey : Lawrence Erlbaum Associates, (1986), P : 31-61.

[North 01] North Chris. Multiple Views and Tight Coupling in Visualization: A Language, Taxonomy, and System. Proc. CSREA 2001 Workshop of Fundamental Issues in Visualization, P: 626-632.

[petitRobert] Le Petit Robert. Dictionnaire Français.

[Plaisant 96] Plaisant Catherine, Milash Brett, Rose Anne, Widoff Seth and Shneiderman Ben. LifeLines : Visualizing Personal Histories. Dans Proceedings de CHI'96. Vancouver, BC, Canada, April 13-18, 1996.

- [Pouthas 99] Pouthas Viviane. Le temps de la conscience, la conscience du temps. Journal de la Société de Biologie, 193(4), P : 411-420.
- [Rekimoto 99] Rekimoto Jun. Time-Machine Computing : A Time-centric Approach for the Information Environment. Proceedings of UIST '99, ACM Symposium on User Interface Software and Technology, November 1999, P : 45-54
- [Renieris 99] Renieris Manos et Reiss Steven P. ALMOST : Exploring Program Traces. NPIVM' 99 Workshop, Kansas City, MO, November 6, 1999.
- [Rex 96] Rex, B. and Risch, J. Animation query language for the visualization of temporal data. Proc: Third International Conference on GIS and Environmental Modelling, NCGIA, Santa Barbara, USA, 1996.
- [Richter 99] Richter H., Brotherton J., Abowd G. D., et Truong K. A Multi-Scale Timeline Slider for Stream Visualization and Control. GVU Center, Georgia Institute of Technology, Technical Report GIT-GVU-99-30, June 1999.
- [Roth 94] Roth Steven F., Kolojejchick John, Mattis Joe and Goldstein Jade. Interactive Graphic Design Using Automatic Presentation Knowledge. In Proc. of CHI'94 conference, Boston USA, P : 112-118.
- [Schmidt 95] Schmidt Duri and Marti Robert. Time Series, a Neglected Issue in Temporal Database Research ? In Proceedings of the VLDB Workshop on Temporal Databases. Zürich, Switzerland, Springer-Verlag. 1995.
- [Scholl 98] Scholl Pierre-Claude, Fauvet Marie-Christine et Canavaggio Jean-Francois. Un modèle d'historique pour un SGBD temporel. Revue Technique et science informatique, pages 379 à 399, Volume 17-n° 3/1998, Edition Hermès.
- [Shneiderman 96] Shneiderman Ben, The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations , CS-TR-3665 July 1996, ISR-TR-96-66.
- [Snodgrass 95] Richard Snodgrass, Temporal Object-Oriented Databases : A critical Comparison. Modern Database Systems, Addison Wesley 1995.

- [Srikant 96] Srikant Ramakrishnan and Agrawal Rakesh. Mining sequential patterns: Generalizations and performance improvements. In Advances in Database Technology - EDBT'96, 5th International Conference on Extending Database Technology, P : 3-17, March 1996.
- [Stasko 00] Stasko John, Catrambone Richard, Guzdial Mark et McDonald Kevin, An Evaluation of Space-Filling Information Visualizations for Depicting Hierarchical Structures. Publié dans International Journal of Human-Computer Studies, vol. 53, 2000, P : 663-694.
- [Steiner 98] Steiner Andreas. A generalization Approach to Temporal Data Models and Their Implementations. Thèse de doctorat en Informatique à Swiss Federal Institute of Technology Zürich. 1998.
- [Stolte 99] Stolte Chris, Bosch Robert, Hanrahan Pat, et Rosenblum Mendel. Visualizing Application Behavior on Superscalar Processors. In proceedings of Information Visualization, San Francisco Airport Hyatt, San Francisco, CA. October 24 - October 29, 1999.
- [Tawfik 00] Tawfik Ahmed Y. and Strickland Krista. Mining Medical Data for Causal and Temporal Patterns. The Fourth European Conference on the Principles and Practice of Knowledge Discovery in Databases. Lyon, France, 2000.
- [Thalmann 94] Thalmann Nadia Magnenat et Thalmann Daniel. Computer Animation : a Key Issue for Time Visualization. Scientific Visualization, Academic Press, 1994, P : 201-222.
- [Theus 96] Theus Martin. Trellis Displays vs. Interactive Graphics. Advances in Statistical Software 5, Softstat '95, eds. Frank Faulbaum & Wolfgang Bandilla, Lucius & Lucius, Stuttgart, 1996.
- [Tufté 83] Tufté E. R., The Visual Display of Quantitative Information, Graphics press, 1983.
- [Turo 92] Turo David et Brian Johnson, Improving the visualization of hierarchies with Treemaps: Design issues and experimentation. Proceedings of the IEEE Conference on Visualization, 1992, P :124–131.
- [Tuzhilin 95] Tuzhilin Alexander and Clifford James. On Periodicity in Temporal Databases. Information Systems volume 20. N°08. Elsevier Science Ltd. 1995. P : 619-639.



- [UIMS 92] The UIMS Workshop Tool Developers : A Metamodel for the Runtime Architecture of an Interactive System, SIGCHI Bulletin, 1992.
- [Vernier 01] Vernier Frédéric. La multimodalité en sortie et son application à la visualisation de grandes quantités d'information. Thèse de doctorat en Informatique à l'Université Joseph-Fourier, Grenoble, France. 2001.
- [Walker 00] Walker, A. and Brewster, S.A. Spatial audio in small display screen devices. *Personal Technologies*, 2000, P : 144-154.
- [Wang 00] Wang Baldonado Michelle Q., Woodruff Allison et Kuchinsky Allan. Guidelines for Using Multiple Views in Information Visualisation. Proceedings of the Working Conference on Advanced Visual Interfaces, Palermo Italy, 24-26 May 2000. P: 110-119.
- [Waterworth 91] Waterworth John A. et Chignell Mark H. A model of Information Exploration. Publié dans le journal *Hypermedia*, Volume 3. 1991, P : 35-58.
- [Weber 01] Weber Marc, Alexa Marc et Müller Wolfgang. Visualizing Time-Series on Spirals. Proceedings on InfoVis 2001.
- [Weiss 98] Weiss Gary M. and Hirsh Haym. Learning to Predict Rare Events in Categorical Time-Series Data. AAAI Workshop, Predicting the Future : AI Approaches to Time-Series Problems. Technical Report WS-98-07, AAAI Press. 1998.
- [Wijk 99] Wijk Jarke J. Van et Selow Edward R. Van. Cluster and Calendar based Visualization of Time Series Data. IEEE Symposium on Information Visualization (InfoVis'99), San Francisco, October 25-26, 1999.
- [Young 97] Young Forrest W. et Bann Carla M. ViSta: A Visual Statistics System. Publié à Stine, R.A. & Fox, J. (Eds.) *Statistical Computing Environments for Social Research.*, Sage Publication, Inc. (1997), P : 207-235.
- [Zaniolo 97] Zaniolo Carlo and al. *Advanced Database Systems* Morgan Kaufmann 1997. Edition Hermès, Paris 1993, Publishers, inc, SAN FRANCISCO California.

## 2 Sites Web

Pour chaque site web, nous présentons son adresse et la dernière date à laquelle nous l'avons consulté.

[Brian 03] Brian Francis et John Pritchard. Visualization of historical events using Lexis Pencils. Disponible à [http://www.agocg.ac.uk/reports/visual/casestud/francis/compar\\_1.htm](http://www.agocg.ac.uk/reports/visual/casestud/francis/compar_1.htm). 30 Avril 2003.

[Browser 03] Browser : a tool for time-series exploration. <http://www.ncea.org.au/Browser/>. 30 Avril 2003.

[DataDesk 03] <http://www.datadesk.com/>. 30 Avril 2003.

[eviews 03] <http://www.eviews.com/>. 30 Avril 2003.

[Explorer 03] <http://opendx.npaci.edu/publications/environmental/mtwildfire/>. 30 Avril 2003.

[JDK 03] JDK, Java Development Kit. Développé par Sun et téléchargeable à : <http://java.sun.com>. 30 Avril 2003.

[jwave 03] <http://www.vnij.com/products/wpd/jwave/>. 30 Avril 2003.

[mathworks 03] <http://www.mathworks.com>. 30 Avril 2003.

[Microsoft 03] <http://www.microsoft.com/>. 30 Avril 2003.

[spotfire 03] <http://www.spotfire.com>. 30 Avril 2003.

[StatSoft 03] StatSoft, Time Series Analysis. <http://www.statsoftinc.com/textbook/sttimser.html>. 30 Avril 2003.

[TAO 03] Pacific Marine Environmental Laboratory, the Tropical Atmosphere Ocean Project <http://www.pmel.noaa.gov/tao/>. 30 Avril 2003.

[TimeLine 03] Small, Ishizki, and Cooper, TimeLine, MIT Media Lab. <http://acg.media.mit.edu/projects/timelines/index.html>. 30 Avril 2003.

[TimeMap 03] TimeMap : <http://www.timemap.net/>. 30 Avril 2003.

[TOOLS 03] <http://www.ess.co.at/TOOLS/timeseries.html>. 30 Avril 2003.

[Tulip 02] <http://cas.lancs.ac.uk/alcd/visual/>. 20 Décembre 2002.

[UML 03] <http://www.uml.org/>. 30 Avril 2003.

[Visual 02] <http://www.iath.virginia.edu/time/visual.html>. 20 Décembre 2002.

## *Annexe A*



## ***Propriétés ergonomiques***

Cette annexe présente la liste des propriétés ergonomiques proposées dans [Nigay 97, Gram 98]. Les propriétés ergonomiques sont définies pour qualifier l'utilisabilité d'un système. Nous adoptons l'approche qui consiste à classer ces propriétés en deux catégories : des propriétés de souplesse et des propriétés de robustesse. Dans la suite, nous présentons ces propriétés dans leur cadre général.

### **1 Propriétés de souplesse**

Les propriétés de souplesse caractérisent la flexibilité du système à permettre à l'utilisateur d'atteindre ses objectifs.

***Multimodalité*** : utilisation de plusieurs canaux de communication. L'utilisateur peut saisir ses données au travers de plusieurs canaux, tel que le clavier, la souris, la voix, etc. Les données de sortie d'un système peuvent être reçues par l'utilisateur à travers la vision, l'audio, etc. Chacun de ces canaux représente pour l'utilisateur une modalité d'interaction.

***Réutilisabilité des données d'entrée et de sortie*** : les données en sortie en provenance du système peuvent être utilisées comme données d'entrées, et inversement.

***Multiplicité de représentation*** : fournir plusieurs représentations d'un même concept. La multiplicité de représentation illustre la flexibilité pour rendre un concept perceptible. Par exemple, la température d'un objet physique à travers des périodes de temps peut être représentée par un thermomètre digital pour le cas où la valeur numérique de la température est assez importante, ou comme un graphe pour le cas où l'important est de tracer l'évolution. Il est parfois souhaitable pour l'utilisateur d'avoir simultanément ces différentes représentations. Chacune de ces représentations lui fournit une perception de l'état du système, et en plus, l'utilisateur est libre de choisir la représentation qui correspond le plus à la tâche courante.

***Multiplicité des rôles*** : c'est la capacité du système à connaître le rôle de l'utilisateur et à l'utiliser comme donnée dans l'interaction.

**Multifenêtrage ou encore interaction à plusieurs fils** : c'est la capacité du système à permettre à l'utilisateur de réaliser plusieurs tâches à la fois. Le multifenêtrage contribue à augmenter la flexibilité de l'interaction du moment où il permet à l'utilisateur d'exécuter simultanément plusieurs tâches.

**Non préemption** : c'est la capacité du système à permettre à l'utilisateur d'atteindre facilement son prochain but. La préemption définit l'espace de liberté de l'utilisateur. Généralement, on veut maximiser la probabilité que l'utilisateur augmente sa préemption du système et minimiser la probabilité de préemption du système pour l'utilisateur. De façon extrême, ceci veut dire que l'utilisateur soit capable d'exécuter n'importe quelle action d'entrée à n'importe quel instant. En réalité, ceci n'est pas souhaitable car l'utilisateur perdra le fils et n'arrivera pas à achever les tâches qu'il a déjà commencées.

**Atteignabilité** : c'est la capacité du système à permettre à l'utilisateur d'atteindre n'importe quel but à partir de n'importe quelle étape.

**Adaptabilité** : c'est la personnalisation du système suite à une intervention explicite de l'utilisateur. L'adaptabilité vise à ajuster les formes des entrées et sorties, telle est alors très limitée et l'utilisateur ne peut qu'ajuster la position des boutons sur l'écran, redéfinir les noms des commandes ou changer les couleurs. La personnalisation est alors limitée à la surface de l'interface et elle est qualifiée de personnalisation lexicale ou pragmatique.

Avec l'adaptabilité on risque de perdre la cohérence entre les systèmes d'une communauté d'utilisateurs sensés travailler ensemble.

**Adaptativité** : la personnalisation peut être faite par le système. La modification est automatique en se basant sur les connaissances requises sur l'utilisateur. Dans ce cas, les décisions sont prises par rapport au degré d'expertise atteint par l'utilisateur ou par rapport aux séquences de tâches répétitives observées à travers l'utilisation de l'interface. Elle s'appuie donc sur un modèle « embarqué » de l'utilisateur. Il est difficile de réaliser un tel système : il faut veiller à ce que le système ait un comportement prévisible et ne surprenne pas l'utilisateur. Il est à noter que l'adaptativité n'incluse pas une adaptation au changement de l'environnement matériel ou logiciel. Cette dernière est représenté par la propriété de plasticité.

**Plasticité** : c'est l'adaptation du système au changement de l'environnement de travail. Ceci inclut une adaptation à un changement au contexte d'interaction (environnement et dispositif, de logiciel ou matériel). Un exemple concret est de porter une interface d'un PC à un ordinateur de poche.

**Migrabilité de tâches** : c'est la capacité de délégation dynamique de tâches entre le système et l'utilisateur (c'est-à-dire changement de l'acteur responsable de la tâche). Par exemple, assurer une sauvegarde automatique, détection de tâches répétitives puis prise en charge de la réalisation de cette tâche, etc.

**CARE** : CARE sont les initiales des quatre propriétés pour l'usage des modalités en entrées : Complémentarité, Assignation, Redondance et Equivalence [Coutaz 95, Martin 93]. Une modalité est la technique d'interaction et la manière d'interagir avec le système. Nous choisissons de représenter une modalité par le couple <d, s> où d est le dispositif physique et s le système représentationnel qui est la façon de caractériser les échanges entre l'utilisateur et le système [Nigay 94]. Exemples : <microphone, langage naturel>, <stylo, geste 2D>, <clavier, langage naturel>, etc.

Dans ce qui suit, nous présentons ces propriétés tant pour les modalités en entrée qu'en sortie :

- **Complémentarité** : la complémentarité entre les modalités d'un ensemble M exprime le fait que pour passer d'un état à un autre, il faille utiliser toutes les modalités de l'ensemble M. Ceci consiste à faire transmettre différents sous-messages d'un même message selon les différentes modalités de l'ensemble M en associant à chacune d'elles un contenu particulier. Aucune modalité de l'ensemble M ne suffit à elle seule pour passer de l'état de départ à l'état d'arrivée mais au contraire il faut une fusion totale de toutes ces modalités.

Une commande vocale «met ça là» accompagnée de deux gestes, l'un pour identifier l'objet à déplacer et l'autre pour spécifier la localisation à laquelle l'utilisateur veut mettre l'objet, utilise une complémentarité entre les modalités d'entrées <caméra, geste de la main> et <microphone, langage naturel>. De même un retour d'information du système vers l'utilisateur par le biais d'un message vocal : «vous devez avoir une valeur supérieure à» accompagné d'un affichage



d'un message textuel sur l'écran «50», utilise une complémentarité entre deux modalités de sorties : <haut parleur, langage naturel> et <écran, langage naturel>.

- **Assignment ou spécialisation** : l'assignation consiste à choisir toujours la même modalité pour un type de message (en entrée ou en sortie) particulier et à ne pas l'utiliser pour d'autres types de message. Cette propriété exprime l'absence de choix pour réaliser une tâche donnée. Un exemple d'application de cette propriété à une modalité d'entrée consiste à affecter à la tâche de manipulation des représentations tridimensionnelles la modalité définie par <caméra, orientation de la main>. L'assignation d'une modalité à une tâche permet de garantir une stabilité de la forme des expressions (d'entrée et de sortie) du fait que la coopération entre les modalités par spécialisation ne doit pas provoquer la fusion.
- **Equivalence** : pour un état de départ et un état d'arrivée fixés, l'équivalence d'un ensemble de modalités est vérifiée si chaque modalité permet d'atteindre l'état d'arrivée depuis l'état de départ en une seule étape. Une équivalence pour deux modalités d'entrée peut concerner par exemple l'utilisation d'un menu ou une combinaison de touches de fonctions pour exécuter une commande. Pour les modalités de sorties, le système peut par exemple afficher le message d'erreur (<écran, langage naturel>) ou utiliser un message vocal (<haut parleur, langage naturel>).
- **Redondance** : la redondance dénote l'utilisation séquentielle ou parallèle de plusieurs modalités équivalentes pour la réalisation d'une tâche. La redondance d'informations en provenance de l'utilisateur peut impliquer la prise en compte d'une seule modalité que le système considère dominante par rapport aux autres modalités pour la réalisation de la tâche en cours. Ceci permet d'éviter tout conflit entre les différentes modalités utilisées. Il est à noter que la redondance peut être totale ou partielle selon que les deux modalités permettent d'atteindre le même objectif ou que l'une permet de faire une sous tâche que la première modalité est capable de faire.

La coopération par redondance entre modalités en sortie est illustrée par l'exemple suivant : afficher un message d'erreur accompagné par un message vocal. L'utilisation de la redondance peut être utile pour renforcer l'observabilité de

l'utilisateur. Pour les modalités en entrée, la redondance peut concerner la saisie d'une requête textuelle accompagnée d'une commande vocale.

## **2 Propriétés de robustesse**

Les propriétés de robustesse visent à prévenir les erreurs et augmenter les chances de succès pour que l'utilisateur réalise sa tâche et atteigne son objectif.

**Observabilité** : c'est la capacité pour l'utilisateur à évaluer l'état interne du système en se basant sur la représentation perçue de cet état. L'évaluation de l'état permet à l'utilisateur de comparer l'état courant avec l'état qu'il s'attendait à obtenir après avoir exécuté son plan d'actions pour pouvoir établir un plan de révision.

**Insistance** : c'est la capacité du système à forcer la perception par l'utilisateur de l'état du système.

**Honnêteté** : c'est la capacité du système à engendrer chez l'utilisateur une interprétation conforme à l'état interne du système. Dans les meilleures circonstances, cette notification peut être immédiate, ne demandant aucune interaction initialisée par l'utilisateur. Un exemple illustratif de la prise en compte de cette propriété dans la conception d'une interface est celui de la liste déroulante dans laquelle la barre de défilement doit être proportionnelle à la taille de la liste.

**Curabilité** : c'est la capacité pour l'utilisateur de corriger une situation non désirée et revenir à un état antérieur. L'utilisateur fait des erreurs qu'il doit pouvoir corriger. La curabilité représente la capacité d'atteindre un objectif après avoir fait des erreurs dans les interactions antérieures. Pour se faire, il s'agit soit d'accepter l'état courant et essayer à partir de cet état d'atteindre l'état désiré, soit d'annuler l'effet des actions antérieures.

**Prévisibilité** : c'est la capacité pour l'utilisateur de déterminer l'effet futur d'une action en se basant sur ses interactions passées et l'état courant du système. Il existe plusieurs degrés par lesquels la prévisibilité peut être satisfaite. La connaissance peut être limitée aux informations qui sont présentées, l'utilisateur n'a donc pas à se rappeler des informations qui ne lui sont pas directement observables.

***Tolérance au rythme*** : c'est la capacité du système à laisser l'utilisateur décider quand il peut agir.