
Conception de systèmes d'information spatio-temporelle adaptatifs avec ASTIS

Bogdan Moisuc, Jérôme Gensel, Marlène Villanova-Oliver, Hervé Martin

Laboratoire LIG, équipe STEAMER
681, rue de la Passerelle, Domaine Universitaire, BP 72
38 402 Saint-Martin d'Hères Cedex
{prenom.nom}@imag.fr

RÉSUMÉ. *Les développements conjugués du Web et de l'informatique ubiquitaire sont à l'origine de la diversité potentielle des utilisateurs, des dispositifs d'accès et des contextes d'utilisation des systèmes d'information spatio-temporelle (SIST). Adapter ces systèmes à l'utilisateur devient donc une nécessité, un gage d'utilisabilité et de pérennité. Cet article présente une approche générique pour la conception et la génération de systèmes d'information spatio-temporelle adaptés à l'utilisateur, appelé ASTIS. ASTIS offre des modalités générales de mise en œuvre de l'adaptation à l'utilisateur, visant tant le contenu que la présentation des applications. Elle permet aux concepteurs d'intégrer ces modalités d'adaptation dans des applications traitant des données spatio-temporelles. Afin de définir les besoins et types d'adaptation propres à leur application, il suffit aux concepteurs de créer des modèles conceptuels, par spécialisation et instanciation des modèles offerts par notre architecture.*

ABSTRACT. *The joint development of the Web and of ubiquitous computing increases constantly the potential diversity of the users, access devices and contexts of use of spatio-temporal information systems accessed via these means. Adapting these systems to their users becomes a necessity and a guarantee of usability and longevity. This paper presents a generic approach for the design and generation of adaptive spatio-temporal information systems. The presented framework integrates generic user adaptation mechanisms, aimed at adapting both the content and the presentation of the applications. It gives designers the possibility to integrate these mechanisms into adaptable applications. In order to define the needs and the adaptations of their applications, designers only have to create conceptual models, by instantiating generic models supplied by our framework.*

MOTS-CLÉS : *adaptation à l'utilisateur, adaptativité, conception de systèmes d'information spatio-temporelle, modélisation.*

KEYWORDS: *user adaptation, adaptivity, spatio-temporal information system design, conceptual modelling.*

1. Introduction

Les avancées technologiques récentes ont favorisé l'émergence de l'informatique mobile et ubiquitaire en tant que paradigme nouveau de diffusion d'informations. L'augmentation de l'autonomie des dispositifs mobiles, de leur taille mémoire et de leur puissance de calcul a donné la possibilité à la communauté de chercheurs en géomatique d'explorer de nouvelles applications des systèmes d'information géographique ([AKM02], [ZIP02]). De nouvelles perspectives s'ouvrent pour les utilisateurs qui peuvent accéder à des informations spatialisées n'importe quand, n'importe où, à travers des dispositifs d'accès très variés en termes de puissance de calcul et capacité d'affichage (par exemple les services localisés et les systèmes de navigation). Ces changements occasionnent également toute une série de problèmes liés tant aux limitations intrinsèques des dispositifs mobiles, qu'au problèmes cognitifs soulevés par l'interaction entre l'utilisateur et ces types de dispositif dans un environnement mobile.

Dans le domaine de la cartographie mobile et des systèmes d'information géographique mobiles, il est communément accepté depuis ces dernières années que les applications doivent prendre en considération une variété plus large d'influences et limitations que celles utilisées dans la cartographie conventionnelle. Après s'être focalisée sur le contournement des limitations techniques de tels dispositifs (bande passante [FBB05], capacité de calcul et mémoire [LCL03], taille d'affichage [SMB04]), l'attention dans les recherches récentes s'est portée vers les aspects cognitifs de l'adaptation à l'utilisateur et à son contexte ([ZIP02], [AKM02], [PRC06]). La conception de SIG mobile s'intéresse de plus en plus à la sensibilité au contexte et la personnalisation, afin de présenter à l'utilisateur une information adéquate, au moment opportun et au bon endroit, en fonction de son environnement ([ZIP05], [REI01]).

Un nombre de plus en plus important de travaux met en évidence l'importance de disposer d'architectures dédiées à la conception et la réalisation de systèmes d'information spatio-temporelle adaptatifs. Parmi les besoins les plus importants on peut citer la séparation entre contenu et présentation, la mise en œuvre de modalités d'adaptation tant pour le contenu que pour la présentation, le caractère générique des approches, le caractère dynamique de la génération du contenu et de la présentation de l'application et le support des standards existants. Zipf [ZIP05] montre qu'il existe à présent des technologies pour mettre en œuvre ces fonctionnalités. Par exemple, il propose la possibilité de générer dynamiquement du code SVG adapté à partir de fichiers GML¹ et SLD², en utilisant des transformations XSLT. Malheureusement, les approches existantes, même si de plus en plus nombreuses, n'arrivent pas à satisfaire les contraintes décrites ci-dessus. Les applications mélangent la gestion du contenu et de la présentation, la plupart d'entre elles

¹ Geography Markup Language <http://www.opengeospatial.org/standards/gml>

² Styled Layer Descriptor <http://www.opengeospatial.org/standards/sld>

n'offrant que des mécanismes d'adaptation de la présentation ([ZIP02], [AKM02], [WLC01]). D'autres prennent en compte le contenu mais ne traitent pas l'adaptation de la présentation [PRC06]. A notre connaissance, une seule approche combine certaines des aspects d'adaptation de la présentation avec un type simple d'adaptation de contenu sous la forme de cas d'utilisation [NIS04]. De plus, dans toutes les approches citées, les mécanismes d'adaptation sont codés en dur et se confondent avec la logique de l'application, ce qui nuit gravement à leur extensibilité et à leur généricité [SGP05].

Cet article présente un *framework*, appelé ASTIS, dédié à la conception et la réalisation de systèmes d'information spatio-temporelle adaptatifs et interactifs. ASTIS adopte une approche de conception et de mise en œuvre d'applications basée sur des modèles. Nous avons préalablement proposé une architecture modulaire appelée GenGHIS ([MDG05]), séparant contenu et présentation et permettant aux concepteurs de SIST de générer des applications à partir de modèles de données conçus par eux. GenGHIS est composé de deux modules principaux : l'un assure la gestion du contenu, l'autre la gestion de la présentation, en s'appuyant sur un système de représentation de connaissances par objets spatio-temporel appelé AROM-ST [MGM04]. ASTIS étend GenGHIS avec des mécanismes d'adaptation de contenu (un système de création de points de vue sur les données) et d'adaptation de la présentation (un système de styles graphiques et composants de présentation). Avec ASTIS, après avoir conçu le modèle de données et le modèle de présentation de base pour l'application, les concepteurs peuvent spécifier différentes adaptations de contenu et de présentation, en fonction des caractéristiques des utilisateurs et du contexte. Au démarrage de l'application, un module d'adaptation est chargé de gérer les utilisateurs, de choisir les adaptations nécessaires et de les mettre en œuvre dynamiquement.

L'article est organisé de la manière suivante : la section 2 présente l'architecture générale du *framework* ASTIS. Les sections 3 et 4 détaillent les mécanismes adaptatifs utilisés au sein d'ASTIS pour réaliser l'adaptation aux points de vue et aux niveaux d'expertise des utilisateurs. La section 5 conclut l'article en présentant les perspectives de développement.

2. L'architecture générale de la plate-forme

Notre approche pour la conception et le développement d'applications spatio-temporelles adaptées est basée sur la génération de code à partir de modèles. L'idée est de permettre aux concepteurs de créer des applications adaptatives en se basant sur une approche déclarative, de conception de modèles orientés objet. L'architecture est formée de composants paramétrables qui opèrent une suite de transformations et de conversions, depuis des formats et sources externes de données (aux formats relationnels les plus répandus : mif/mid, excel, mdb, etc.) jusqu'au format final, affichable à l'écran (documents SVG Tiny 1.2 et XHTML avec une

structure particulière). Le framework contient également des composants visuels génériques, capables d'afficher de manière interactive les documents structurés résultants du processus de transformation.

L'architecture est composée de trois modules principaux : un module de données, un module de présentation et un module d'adaptation. Le module de données est chargé du stockage, de l'interrogation et de l'analyse des données en conformité avec les modèles spécifiés par les concepteurs. Il permet aux concepteurs de décrire leurs modèles de données et de les instancier. Le module de présentation permet aux concepteurs de créer des modèles de présentation adaptés et de générer des présentations conformes à ces modèles. Il contient aussi les composants visuels nécessaires pour afficher ces présentations. Le module d'adaptation est chargé de la gestion du contexte, des utilisateurs, et de l'application dynamique des mécanismes d'adaptation choisis en fonction de ces cibles.

Le fonctionnement de l'architecture est décrit de manière simplifiée dans la Figure 1. La partie gauche de la figure illustre la partie statique de l'architecture, c'est-à-dire les composants impliqués dans les processus de conception. La partie droite contient la partie dynamique de l'architecture, c'est-à-dire les composants chargés de la génération à la volée et de l'affichage de contenus SVG adaptés. Les composants représentés par des rectangles à angles droits sont des composants paramétrisables via des modèles. Les composants représentés sous forme de rectangles à coins arrondis sont es composants chargés des conversions de format.

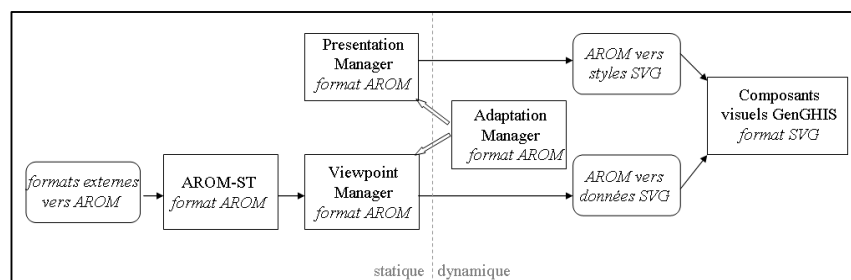


Figure 1. Schéma du fonctionnement d'ASTIS.

Pour créer une nouvelle application, les concepteurs doivent d'abord spécifier le schéma de données de l'application. Ensuite, ils doivent spécifier différentes transformations de ce schéma afin de l'adapter aux caractéristiques des utilisateurs et du contexte. De la même manière, la troisième étape consiste à spécifier des adaptations de présentation, c'est-à-dire différentes options de style et d'affichage applicables. La dernière étape de conception permet de définir et faire la liaison entre les différentes caractéristiques du contexte de l'utilisateur et les adaptations correspondantes. Cela revient à spécifier un modèle d'adaptation comme celui de la Figure 2. A présent, il n'existe pas un consensus sur une terminologie de

l'adaptation. Nous avons adopté la terminologie de Reichenbacher [REI01]. Ainsi, la cible de l'adaptation (*target*) définit les caractéristiques significatives de l'utilisateur ou de son contexte auxquelles l'application doit s'adapter (par exemple, le niveau d'expertise de l'utilisateur ou sa localisation). Une composante à adapter (*adaptee*) représente une caractéristique du contenu ou de la présentation qui peut être ajustée en adéquation avec le contexte. Le mécanisme d'adaptation (*adapter*) est la fonctionnalité du système qui permet d'ajuster les caractéristiques des composantes à adapter en fonction des caractéristiques connues des cibles.

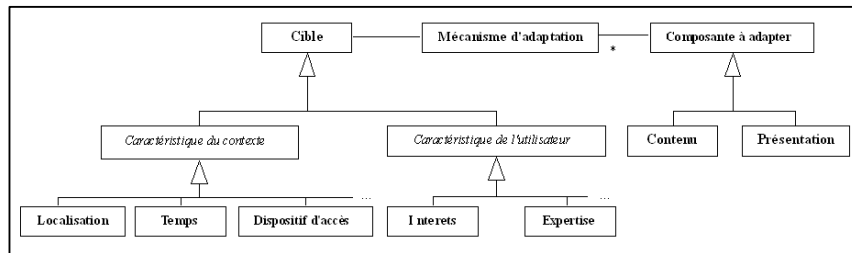


Figure 2. Modèle d'adaptation d'ASTIS.

Afin de finaliser la réalisation d'une nouvelle application les concepteurs doivent procéder à l'acquisition de données, c'est-à-dire peupler avec des instances la base de connaissances AROM-ST de l'application, en utilisant le composant de conversion de formats externes vers AROM-ST. Notre approche de conception s'inscrit donc dans le cadre de l'*adaptativité* [MAC98], les concepteurs sont amenés à décrire des adaptations en fonction des caractéristiques de l'utilisateur et de son contexte, qui seront appliquées automatiquement, sans l'intervention des utilisateurs, au moment de l'exécution de l'application. Il est important de noter que, du point de vue technique il est possible de mettre en œuvre de l'*adaptabilité*, c'est à dire de permettre aux utilisateurs de décrire et appliquer eux-mêmes ces adaptations, de manière dynamique, pendant l'exécution de l'application. Nous considérons que, pour que notre approche soit utilisable, il est nécessaire de créer de nouvelles interfaces et des manières simplifiées de contrôler les applications, parce qu'il est raisonnable d'imaginer que les utilisateurs finaux sont moins avisés en ce qui concerne la modélisation de données et de présentations spatio-temporelles. Une des perspectives de notre travail vise la mise en œuvre d'interfaces et modalités de présentation simplifiées pour rendre notre architecture plus utilisable aussi dans une approche basée adaptabilité.

Au moment de l'exécution, le gestionnaire d'adaptation choisit, en fonction des caractéristiques concrètes du contexte, les modalités adéquates d'adaptation à appliquer, décrites par le modèle d'adaptation. Ces modalités sont appliquées au moment de la génération du code SVG de l'application, au niveau de la présentation (structure de la présentation, styles et options d'affichage), et au niveau du contenu

(le point de vue sur les données). Les modalités d'application de ces adaptations sont détaillées dans les 2 sections suivantes, et sont illustrées par des exemples d'adaptation visant deux caractéristiques de l'utilisateur : ses intérêts et son niveau d'expertise.

3. Adaptation de contenu

L'adaptation de contenu est réalisée à travers le concept de *point de vue sur les données*. Notre plate-forme utilise pour la modélisation et pour le stockage des données un système de représentation de connaissances par objets, étendu avec des types et opérateurs spatio-temporels [MGM04]. Opérationnalisation de UML, AROM [PGC01] modélise la réalité sous forme de *classes* et d'*associations*. Les classes peuvent avoir des propriétés appelées *attributs*. Une association exprime une relation existante entre au moins deux classes, appelées *rôles*. Les instances de classes et associations sont appelées *objets* et, respectivement, *tuples*. AROM dispose d'un langage de modélisation algébrique (LMA) qui permet de formuler des requêtes, d'exprimer des contraintes et de calculer des valeurs d'attributs. Notre plate-forme repose sur les capacités de modélisation d'AROM et du LMA pour la gestion de points de vue sur les données.

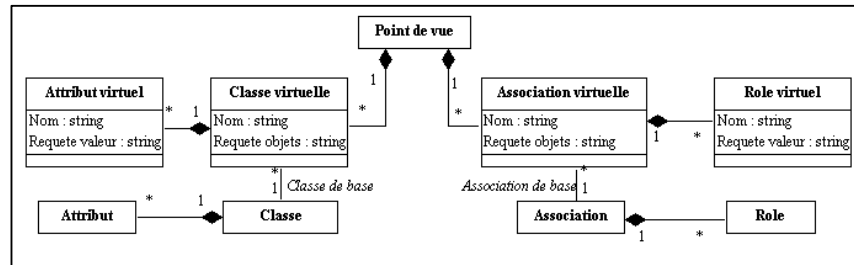


Figure 2 Structure de points de vue sur les données dans ASTIS.

En ASTIS, un point de vue sur les données est défini en dérivant, à partir du schéma de données initial d'une application, des schémas virtuels, représentant l'information sous différentes perspectives (voir la Figure 2). Ainsi, un point de vue de données est un schéma de données contenant les classes, associations, attributs et rôles virtuels (représentés respectivement par les classes *Classe virtuelle*, *Association virtuelle*, *Attribut virtuel* et *Rôle virtuel* dans la Figure 2), et un ensemble de règles de correspondance entre les entités du schéma dérivé et les entités du schéma de base (représentées par les associations *Classe de base* et *Association de base* dans la Figure 2). Notre approche est similaire à celle d'Abiteboul et Bonner [ABB91]. Elle est ici étendue pour le modèle d'AROM qui adopte une représentation explicite des associations et de leurs instances.

Le concepteur peut employer plusieurs mécanismes pour dériver des points de vue :

- Masquer des classes et des associations, des attributs et des rôles ;
- Masquer les objets et les tuples non pertinents, en employant des requêtes qui permettent de ne retenir que ceux qui correspondent à certains critères de pertinence (l'attribut *Requête objets* pour la classe *Classe virtuelle* et, respectivement, l'attribut *Requête tuples* pour la classe *Association virtuelle*) ;
- Renommer des classes, des associations, des attributs et des rôles ;
- Fusionner en une seule classe des classes qui sont liées par des associations (l'attribut *Requête valeur* de la classe *Attribut virtuel* permet de rechercher la valeur de l'attribut à partir d'attributs des classes liées à la classe de base), ou, au contraire, d'éclater une classe en plusieurs classes (en définissant plusieurs classes virtuelles à partir de la même classe de base) ;
- Définir de nouvelles classes, en utilisant des requêtes pour regrouper des objets du même type (en employant l'attribut *Requête objets* de la classe *Classe virtuelle*).

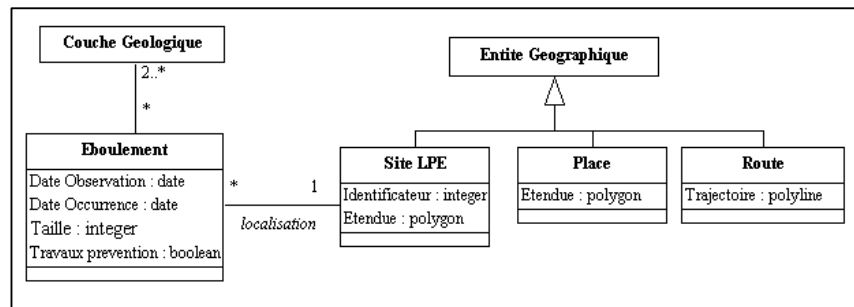


Figure 3. Exemple de schéma d'application de suivi de risques d'éboulement.

Nous illustrons la manière dont les mécanismes ci-dessus peuvent être employés par un court exemple issu du domaine des risques naturels. Supposons qu'un concepteur souhaite créer un SIST pour la gestion des risques d'éboulement, destiné à être utilisé par des cyndiniciens, des géologues et des aménageurs du territoire. À partir du schéma de base (voir Figure 3), le concepteur souhaite décrire un point de vue « aménagement du territoire » (voir Figure 4 à gauche). Etant donné que les aménageurs du territoire ne sont pas intéressés par les informations de nature géologique, le concepteur a intérêt à masquer pour eux des classes et des associations non pertinentes comme, par exemple, des informations géologiques sur des éboulements (la classe *Couche géologique* et l'association correspondante).

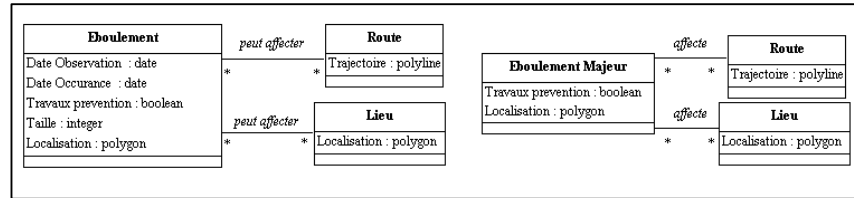


Figure 4. Deux exemples de schémas dérivés.

Supposons maintenant que, dans le schéma de base, les éboulements sont enregistrés par les cyndiniciens comme étant liés aux sites de localisation probable des éboulements (voir la classe *Site LPE* dans la Figure 3). Les cyndiniciens inspectent ces emplacements périodiquement pour vérifier si les nouvelles menaces d'éboulement apparaissent. Cependant, ces informations de propres aux cyndiniciens peuvent être rendues transparentes pour les responsables régionaux de l'aménagement du territoire. Le concepteur pourrait donc, pour le point de vue « aménageur », fusionner les deux classes (*Eboulement* et *Site LPE*), en ignorant l'attribut *Identificateur* et en définissant l'emplacement de l'éboulement comme l'emplacement du site LPE. Une requête de valeur en LMA permet de récupérer l'attribut à partir de l'attribut localisation de la classe *Site LPE* :

```
this.Localisation = this! localisation.Site LPE.Etendue
```

Supposons maintenant que le concepteur souhaite définir un point de vue simplifié pour des aménageurs du territoire moins habitué avec les outils informatique (voir Figure 4 à droite). Cela revient à éliminer les attributs moins pertinents (la taille et les différentes dates de recensement). Une autre transformation en ce sens consiste à diminuer le nombre d'instances d'éboulements. La classe *Eboulement* est alors renommée (à l'aide une classe virtuelle *Eboulement Majeur*), et on y inclut seulement les instances d'éboulements qui présentent un intérêt pour les aménageurs du territoire, c'est-à-dire les éboulements qui présentent un risque important pour la population, les bâtiments ou les infrastructures. La requête LMA permettant de récupérer les objets membres de la nouvelle classe est :

```
this.objects = set (e in Eboulement: (exists l in Lieu:
distance (e.Localisation, l.Etendue) < 500 or exists r in
Route: distance (e.Localisation, r.Trajectoire) < 100) and
e.Taille > 100)
```

Cette requête permet de présenter aux utilisateurs aménageurs seulement les éboulements d'un cubage supérieur à un certain seuil (100m^3) et qui, par leur localisation à proximité des endroits habités ou des routes (à moins de 500 m des endroits habités ou à moins de 100 m des routes), pourraient présenter un danger potentiel.

4. Adaptation de la présentation

Les points de vue peuvent influencer non seulement les données à présenter aux utilisateurs, mais également la manière dont les données sont présentées. La plateforme ASTIS permet de générer des interfaces de visualisation qui rendent compte des dimensions diverses de l'information (dimensions spatiale, temporelle et attributaire). Celles-ci sont affichées dans des fenêtres spatiales (cartes), temporelles (diagrammes temporels) ou attributaires (tableaux), dont les contenus sont synchronisés. Les mécanismes de synchronisation (*linking* [BMM91]) permettent de réaliser des requêtes visuelles et de visualiser leurs résultats sur toutes les fenêtres simultanément.

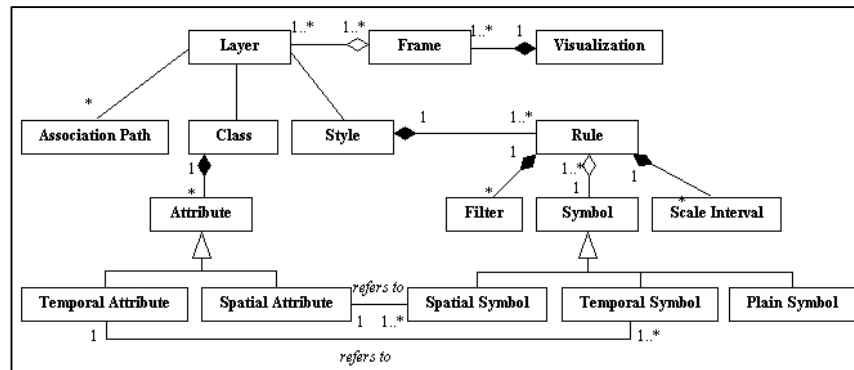


Figure 5. Modèle de présentation simplifié d'ASTIS.

ASTIS permet aux concepteurs de spécifier des adaptations de la présentation qui sont appliquées dynamiquement à l'exécution de l'application, en fonction des caractéristiques des cibles (l'utilisateur et son contexte). Un modèle de présentation, dont un schéma simplifié est présenté en Figure 5, permet de définir et de regrouper une série de transformations à appliquer sous la forme d'une visualisation. Le modèle de présentation d'ASTIS reprend la spécification SLD (Styled Layer Descriptor [OGC02]) de l'OGC et l'étend sur plusieurs aspects. SLD est un standard de présentation qui permet de décrire des transformations de style (classe *Style* en Figure 5) à appliquer à des couches de données (classe *Layer*). Une couche de données fait référence à une classe d'objets spatiaux (classe *Class* dans la Figure 5, appelée aussi *FeatureType* dans la terminologie OGC). Un style est composé d'un ensemble de règles, qui permettent de décrire le type exact de symbole graphique correspondant à chaque objet spatial, en fonction de ses valeurs d'attributs (classe *Filter*). Par exemple, pour une couche représentant les rivières dont le symbole graphique est un trait bleu, un ensemble de règles permet de spécifier différentes épaisseurs du trait en fonction du débit de chaque rivière. Une règle peut aussi

SAGEO'2007

spécifier l'intervalle d'échelles pour lequel l'ensemble d'objets contenus est visible sur la carte.

Le modèle de présentation d'ASTIS étend le standard SLD sur deux aspects : la prise en compte des présentations temporelles et attributaires, et non pas seulement spatiales, ainsi que la prise en compte des éléments dynamiques de présentation, afin de permettre des présentations interactives. ASTIS permet de définir des visualisations complexes et multidimensionnelles (classe *Visualisation*) contenant plusieurs fenêtres (classe *Frame*) de type spatial, temporel ou attributaire, chaque fenêtre affichant un certain nombre de couches (de type spatial, temporel ou attributaire). De nouveaux types de symboles graphiques temporels (1D, classe *Temporal Symbol*) et attributaires (0D, classe *Plain Symbol*) ont été ajoutés aux types existants en SLD (2D, classe *Spatial Symbol*). Nous pouvons apercevoir un exemple de visualisation multidimensionnelle en Figure 7 à gauche, contenant une fenêtre temporelle, une fenêtre attributaire et deux fenêtres spatiales, chacune des fenêtres étant peuplée de symboles de type correspondant : symboles spatiaux (polygones colorés an aplats, cercles proportionnels, signes conventionnels, texte ancré, etc.), symboles temporels (histogrammes), symboles attributaires (texte simple).

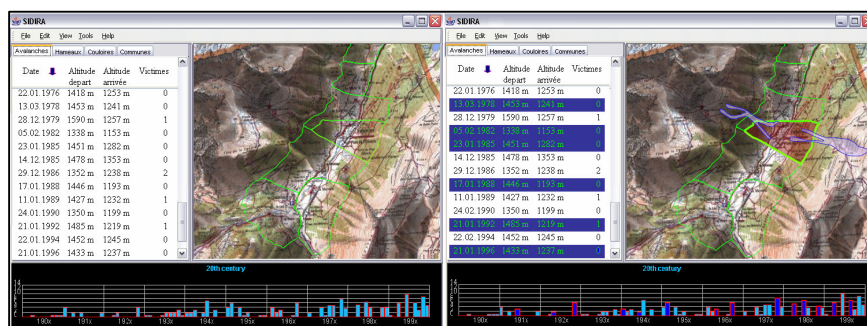


Figure 6. Exemple de synchronisme interactif dans une application de visualisation de risques avalanches.

Afin de permettre la génération de présentations dynamiques et interactives, le modèle de présentation d'ASTIS étend la notion de couche décrite en SLD en ajoutant des informations concernant les liens existants entre les différents types d'objet (classe *Association Path*). Par exemple, dans une application de gestion d'avalanches (voir Figure 6), en cliquant sur le contour d'une commune sur la carte (la commune avec un remplissage rouge) il est possible de visualiser (toujours sur la carte) quels sont les sites avalanches susceptibles de l'affecter (en violet), quels sont les caractéristiques temporelles des événements avalanches qui se sont produit sur ces sites (sur le diagramme temporel, en bas) ainsi que leurs caractéristiques attributaires (fenêtre attributaire, à gauche).

Nous donnons un exemple simple de la façon dont des présentations peuvent être adaptées aux besoins de l'utilisateur pour un certain point de vue. Considérons la conception des présentations selon un point de vue « aménagement du territoire » et selon un point de vue « touristique » dans une application dédiée à la diffusion d'informations sur des risques d'avalanche (voir la Figure 7). La visualisation pour le point de vue « aménagement du territoire » est conçue pour un groupe d'utilisateurs habitués aux outils de type SIST, alors que la visualisation selon le point de vue « touriste » s'adresse sans doute à des utilisateurs moins avisés.

Laissant de côté les différences de contenu entre les deux instances d'application (le point de vue « touristique » contient, par exemple, des informations sur des objectifs touristiques), la structure des présentations diffère également. Du point de vue des aménageurs du territoire, il est important de connaître les aspects temporels et les aspects attributaires liés aux éboulements, donc l'application doit inclure une fenêtre temporelle et une fenêtre attributaire permettant d'explorer ces aspects. Pour le point de vue « touriste » sur la même application, d'autres types de visualisations sont souhaitables. Etant donné que les touristes sont intéressés par la prévision à court terme (un ou deux jours) des risques d'avalanche, la fenêtre temporelle retraçant l'historique des événements n'est probablement pas nécessaire. De la même manière, une fenêtre attributaire permettant de connaître dans le détail les éboulements peut être omise.

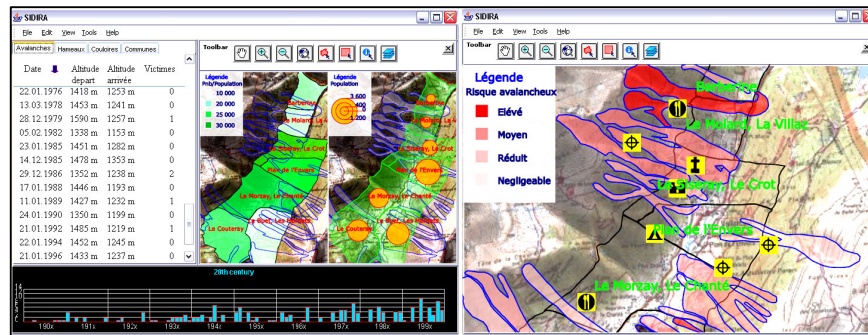


Figure 6. Exemple comparatif de deux points de vue au niveau de la présentation

En ce qui concerne les niveaux d'expertise, la visualisation s'adressant aux touristes est plus simple : elle ne contient qu'une seule fenêtre et la densité visuelle affichée est plus faible. Son style de présentation s'appuie davantage sur des symboles visuels que sur de l'information textuelle. Au contraire, la visualisation pour les aménageurs est plus dense et permet une analyse spatiale et d'observer la dépendance entre deux types d'indicateurs.

5. Conclusions et perspectives

Nous avons présenté dans cet article une plate-forme pour la conception et la génération de SIST adaptables à l'utilisateur. Cette architecture implémente une approche conceptuelle de l'adaptation à l'utilisateur qui permet aux concepteurs de SIST de générer des applications adaptables en décrivant simplement des modèles conceptuels. Ces modèles doivent décrire la transformation applicable à la composante à adapter (les éléments du système qui sont transformés afin de répondre aux besoins de l'utilisateur) et la cible de l'adaptation (les caractéristiques de l'utilisateur ou de son environnement que le système doit prendre en compte pour l'adaptation). Afin d'aider les concepteurs dans la modélisation, la plate-forme offre des modèles génériques pour chaque type de cible et de composante à adapter, qu'il suffit aux concepteurs d'instancier. Cela permet aux mécanismes génériques d'adaptation implémentés au sein de notre plate-forme de réaliser l'adéquation aux types d'utilisateurs au niveau de l'application finale. Notre approche met en œuvre tant l'adaptation du contenu (les données), que l'adaptation de la présentation (les interfaces de visualisation).

Plusieurs perspectives sont possibles, en vue d'améliorer notre proposition et de la rendre plus utile pour la conception et la réalisation d'applications pour des SIST mobiles. La première vise la réalisation d'un module d'acquisition du contexte. Ce module devrait utiliser un modèle compréhensif de contexte et devrait permettre aux applications d'acquies un maximum de paramètres du contexte (localisation, temps, paramètres du dispositif d'accès, historique de l'utilisateur, agenda, etc.) sans aucune intervention de la part de l'utilisateur ou du concepteur. La deuxième perspective est de compléter la plateforme avec un module d'adaptation dynamique, chargé de détecter les changements de contexte et de déclencher (en utilisant un mécanisme de type ECA – *Event Condition Action*) les adaptations nécessaires, c'est-à-dire régénérer de manière adéquate les parties de l'application.

6. Bibliographie

- [ABB91] Abiteboul, S., Bonner, A. Objects and Views, Proc. ACM-SIGMOD'91, 1991.
- [AKM02] Anegg H., Kunczier, H. Michlmayr, E. Pospischil G., Umlauf M. LoL@: Designing a Location Based UMTS Application, ÖVE-Verbandszeitschrift e&i, Springer, 2002.
- [BMM91] Buja, A., McDonald, J. A., Michalak, J., Stuetzle, W. *Interactive data visualization using focusing and linking*, Proc. of the 2nd Conference on Visualization '91, 1991.
- [FBB05] Follin, J.-M., Bouju, A., Bertrand, F. and Stockus, A., An Increment Based Model for Multi-resolution Geodata Management in a Mobile System, Proc. W2GIS 2005, LNCS, 2005.

Conception de systèmes d'information spatio-temporelle adaptatifs avec ASTIS

- [LCL03] Lee, C.-H., Chen, L., Lee, J.-D., Bae, H.-Y., Content Adaptation and Transmission Strategy of Spatial Information for WWW and Mobile Applications, HSI'03, LNCS, 2003.
- [MAC98] MacEachren, A.M.: VISUALIZATION - Cartography for the 21st century. 7th Annual Conference of Polish Spatial Information Association. Warsaw, Poland, 1998.
- [MDG05] Moisuc, B., Davoine, P.-A., Gensel, J., Martin, H., *Design of Spatio-Temporal Information Systems for Natural Risk Management with an Object-Based Knowledge Representation Approach*, Geomatica, Vol. 59, No. 4, 2005.
- [MGM04] Moisuc, B., Gensel, J., Martin, H., Représentation de connaissances par objets pour les SIG à représentations multiples, Proc. Cassini-SIGMA 2004.
- [NIS04] Nivala, A-M. and Sarjakoski, L.T. Preventing Interruptions in Mobile Map Reading Process by Personalisation. Proc. 3rd Workshop on "HCI in Mobile Guides", in adjunction to: MobileHCT04, 2004.
- [OGC02] OGC, Styled Layer Descriptor Implementation Specification, version 1.0.0, 2002.
- [PGC01] Page, M., Gensel, J., Capponi, C., Bruley, C., Genoud, P., Ziebelin, D., Bardou, D., Dupierris, V. *A New Approach to Object-Based Knowledge Representation: the AROM System*, LNAI, 2001.
- [PRC06] Petit, M., Ray, C., Claramunt, C. A contextual approach for the development of GIS: Application to maritime navigation. Proc W2GIS'06, LNCS, 2006.
- [REI01] Reichenbacher, T. *Adaptive Concepts for a Mobile Cartography*, Supplement Journal of Geographical Sciences, December, 43-53, 2001.
- [REI03] Reichenbacher, T., Adaptive Methods for Mobile Cartography. In: Proceedings of the 21st International Cartographic Conference (ICC) 2003, Durban, 2003.
- [SGP05] Schwinger, W., Grün, Ch., Pröll, B., Retschitzegger, W. and Schauerhuber, A., Context-awareness in Mobile Tourism Guides - A Comprehensive Survey, Technical Report, July 2005.
- [SMB04] Sester, M. and Brenner, C.: Continuous Generalization for Fast and Smooth Visualization on Small Displays, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, XXXV (B4:IV): 1293-1298, 2004.
- [WLC01] Wang, Y., Liu, Y., Chen X., Chen Y. & Meng, L., *Adaptive Geovisualization - an approach towards the design of intelligent geovisualization systems*, Journal of Geographical Sciences, Vol. 11, 1-8, 2001.
- [ZIP02] Zipf, A., User-Adaptive Maps for Location-Based Services (LBS) for Tourism, Proc. ENTER 2002, Innsbruck, 2002.
- [ZIP05] Zipf, A. Using Styled Layer Descriptor (SLD) for the dynamic generation of user- and context-adaptive mobile maps - a technical framework, In Proc. W2GIS'05, LNCS, 2005.