



HAL
open science

Gérer les informations temporelles dans et par les modèles informatiques

Frédéric Bertrand, Cyril Faucher, Jean-Yves Lafaye, Guillaume Moreau, Myriam Servières, Chamseddine Zaki

► **To cite this version:**

Frédéric Bertrand, Cyril Faucher, Jean-Yves Lafaye, Guillaume Moreau, Myriam Servières, et al.. Gérer les informations temporelles dans et par les modèles informatiques. *Revue Internationale de Géomatique*, 2015, MODÉLISER LES DYNAMIQUES SPATIALES, 25 (3), pp.415-436. 10.3166/RIG.25.415-436 . hal-01259330

HAL Id: hal-01259330

<https://hal.science/hal-01259330>

Submitted on 22 Jan 2016

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.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

Titre → TNR 18
Gras, bordure ¼ à
20pts au-dessus
18 après s'il y a un
sous-titre,
24 après s'il n'y a
pas de sous-titre

Gérer les informations temporelles dans et par les modèles informatiques

**Frédéric Bertrand¹, Cyril Faucher¹, Jean-Yves Lafaye¹,
Guillaume Moreau², Myriam Servières², Chamseddine Zaki³**

1. Laboratoire L3i, Univ. La Rochelle
2. Laboratoire CERMA-IRSTV, EC Nantes
3. Apside Technologies. Paris

RESUME. Les informations temporelles jouent un rôle important dans une grande majorité de problèmes scientifiques. En géographie, l'interaction avec les aspects spatiaux et la référence à des calendriers sur de longues périodes est fréquente. Les approches théoriques ou expérimentales croisent nécessairement différents points de vue de plusieurs communautés scientifiques. Pour préciser et unifier les langages et les pratiques, l'utilisation systématique de modèles plus ou moins formels est une garantie d'efficacité (cohérence, correction, réutilisation, etc.). L'accès à des masses toujours plus importantes d'informations complexes et hétérogènes, la volonté de capitalisation des résultats nécessitent le peuplement et l'interrogation d'entrepôts de données et de bases de connaissance. L'informatique produit des modèles qui permettent de formuler et de résoudre un grand nombre de questions en la matière. Le présent article donne un état de l'art sur le sujet ainsi que des exemples de tels modèles assortis de références à des travaux applicatifs.

ABSTRACT. Temporal information proves to be prominent in many scientific problems. As regards Geography, temporal concerns are often in long or mean term interaction with spatial aspects and calendar issues. Practical and theoretical approaches necessarily cross-reference various viewpoints from several scientific communities. The use of formal models is a means to make languages and practices more accurate and efficient, hence leading to more efficiency (consistency, correction, reusability...). The will to access larger, more and more complex and heterogeneous pieces of information, leads to master the way to populate and query data warehouses and knowledge bases. With several paradigms, computer science, provides models which can express and help solving many current and forthcoming problems. This paper gives an overview of the state of art in the matter of temporal modelling, accompanied with some meta-model instances and some insight into practical applications.

MOTS-CLES : Modélisation de l'information temporelle, Evénements répétitifs, Contraintes temporelles, Calendriers. Raisonnement temporel.

KEYWORDS: Temporal Information Modelling, Repeated Events, Temporal Constraints, Calendars, Temporal Reasoning

1. Introduction

Au-delà du périmètre des études qui leurs sont spécifiquement dédiées, les phénomènes temporels, avec leurs propriétés sémantiques, structurelles, géométriques, topologiques et calendaires sont devenus incontournables dans la gestion des systèmes d'information. La spécification conjointe cohérente mais cependant orthogonale des aspects précédents est la manière de rendre compte de façon opérationnelle de la réalité – cycle de vie, comportement – des objets manipulés à travers leurs modèles. Ces différents points de vue sont à l'évidence corrélés dans leurs propriétés et leur sémantique ; l'orthogonalité mentionnée concerne les activités de spécification de chacun d'entre eux qui doivent pouvoir se mener de manière indépendante.

Avec l'avènement du paradigme objet, il est clairement apparu que la portée des informations temporelles devait englober non seulement les objets eux-mêmes mais également leurs propriétés, attributs et associations, soulevant des questions d'intégrité attachées aux cardinalités, aux structures hiérarchiques et aux relations de compositions ou d'agrégations. Ces aspects sont pris en charge dans différents modèles dédiés ou dans des langages spécifiques. De manière générale, les aspects temporels sont liés à l'occurrence d'événements ; ils traduisent des caractéristiques qui peuvent s'implémenter dans le cadre de différents référentiels et calendriers à partir d'éléments plus ou moins complexes fondés sur les notions élémentaires d'instant et d'intervalles (iCalendar¹, ISO19108², OWLTime³, SQL_DATE⁴, Java Date & Time API). La sémantique temporelle des calendriers permet en quantifiant le temps, un raisonnement temporel quantitatif à travers la mise en place de systèmes de règles et des moteurs de raisonnement à des fins de vérification, de validation ou de recherche d'information. Les modèles de relations temporelles (comme ceux d'Allen) permettent d'utiliser la séquence et le chevauchement des événements. Cela permet un raisonnement temporel qualitatif utile à la construction d'historiques, de cycles de vie, ou encore d'une manière plus générale pour étudier la dynamique des interrelations entre objets.

Afin de mieux cerner le périmètre de notre propos, l'article s'organise en cinq sections. La section 2 expose brièvement des cas d'utilisation représentatifs de l'objet de notre contribution. Puis la section 3 présente un état de l'art en matière de modélisation temporelle. Elle est suivie de la section 4 détaillant le modèle MADS et ses limites avant d'en proposer des extensions qui constituent l'une des contributions du présent article. La section 5 introduit les modèles originaux et expose un cas d'utilisation pratique qui met en évidence la nécessité de produire des modèles objets pivots à la fois pour les événements complexes et pour leurs propriétés temporelles. Une réflexion sur le codage en intension ou en extension des informations temporelles est également menée.

¹ RFC 2445 - iCalendar : <http://www.ietf.org/rfc/rfc2445.txt>

² http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=26013

³ W3C : OwlTime Specification – <http://www.w3.org/TR/owl-time/>

⁴ http://docs.oracle.com/cd/B19306_01/server.102/b14225/ch4datetime.htm

Une discussion des propositions et de leurs applications amène à la conclusion en section 6. Il ne s'agit pas de clore le sujet, mais bien de montrer des exemples qui éclairent à la fois la problématique traitée et les manières de contribuer à apporter des solutions. Dans tous les cas, la contribution consiste en la spécification de modèles adaptés dont on précise les usages et les limites ; ceci répond effectivement à la thématique de ce numéro spécial : « Quels usages, quelle pertinence pour les modèles informatiques confrontés à l'évolution de l'information temporelle ? »

2. Cas d'utilisation types

Un premier objectif concerne la volonté de tracer le cycle de vie des objets dans les modèles. L'usage habituel, d'annoter les éléments de modélisation par des estampilles temporelles est une solution insuffisante si l'on souhaite typer les événements déclencheurs des changements d'états des objets et également des associations du modèle. Il convient alors de considérer les événements comme des objets à part entière dans la modélisation afin de permettre qu'ils se voient attribuer des propriétés et des comportements. Les solutions proposées étendent les modèles et environnements existants en spécifiant de nouveaux concepts et de nouvelles fonctionnalités.

Ainsi enrichis, les modèles peuvent alors faire l'objet de vérifications syntaxiques et sémantiques. Ils peuvent également servir de base à la construction de structures persistantes pour leurs instances qui pourront ensuite être interrogées via des requêtes exprimées selon les cas en XQuery, SQL ou SPARQL.

Un second objectif concerne la modélisation des expressions temporelles nécessaires pour décrire des occurrences d'événements complexes, par exemple lors de l'ordonnancement d'actions dans des processus système ou métier. Les expressions sont initialement énoncées en langage naturel et peuvent aussi bien être des énumérations de dates concrètes (calendaires) que des descriptions abstraites (ex. : « *le second lundi de chaque mois entre avril 2013 et mars 2014* »). La modélisation conjointe de la sémantique du calendrier de référence et d'un système de règles, permet de vérifier la correction et la cohérence des expressions saisies.

Divers contextes applicatifs correspondent à cette problématique. Nous donnons ci-après trois exemples qui explorent différentes formes de dialogue : homme/machine, machine/machine ou {machine, expert}/machine. Le concepteur du système d'information (SI) a pour responsabilité de satisfaire les exigences de ces cas d'utilisation.

- i) Un utilisateur interroge un SI pour accéder à des informations temporelles concernant des objets d'intérêt. Les requêtes peuvent être initialement exprimées en langage naturel avant d'être traduites conformément au modèle. Les informations attendues seront également restituées en langage naturel contrôlé :
 - Sur un plan pratique, ce cas peut concerner un spécialiste (historien, géographe, sociologue, etc.) qui cherche à mettre en évidence et à analyser des structures régulières dans les occurrences d'événements. Cela nécessite d'explorer et de nourrir une base de données intégrant explicitement la connaissance sur la périodicité – exacte ou approchée – des occurrences d'événements répétitifs.

- Un second exemple applicatif du même ordre, concerne l'implémentation de règles administratives sous forme d'une base de connaissance. Par exemple pour l'attribution municipale de places de marché hebdomadaire (horaires d'occupation, vente, stationnement, etc.). Dans le même ordre d'idée on peut penser aux législations sur la chasse ou la pêche.
- ii) Il est demandé de concevoir un ensemble d'applications échangeant des informations temporelles. Les modèles d'expressions temporelles que nous proposons joueront le rôle de modèles pivot en la matière, encapsulant la vue globale partagée par chaque application et offrant une gamme de services de base (accesseurs, calculs, transformations, etc.).
- iii) Un logiciel d'analyse automatique de contenu, couplé à un outil de traitement automatique de la langue, détecte des expressions temporelles dans des documents numérisés. Après annotation experte, phase nécessaire pour lever les inévitables ambiguïtés, les informations collectées iront peupler le modèle d'expressions temporelles.

Dans chacun de ces contextes, l'existence d'un modèle d'expressions temporelles permet d'assister l'utilisateur et de garantir la correction des paramètres d'entrée/sortie, ainsi que des informations enregistrées ou échangées. Les contrôles sont syntaxiques et sémantiques ; ils intègrent en conséquence la connaissance propre aux calendriers et les contraintes temporelles liées au métier. La chose est complexe car, outre la correction interne des différents éléments, il s'agit également d'assurer la cohérence de l'ensemble. Les modèles proposés sont spécialement conçus pour traiter les événements complexes et répétitifs et rendre opératoires leurs règles de périodicité.

3. État de l'art

Dans cette section, dédiée à l'étude des modèles conceptuels pour gérer et représenter les informations temporelles, nous présentons cet axe de recherche et nous exposons les exigences en terme de représentation des données ayant des caractéristiques temporelles, les principaux travaux en ce domaine ainsi que notre contribution.

3.1 Démarche générale de modélisation des objets à caractère spatio-temporel

Une des motivations fondamentales de la modélisation des données est de permettre d'organiser, de stocker et d'effectuer des requêtes sur ces dernières. La plupart des données actuelles représentent des objets et des phénomènes localisés dans l'espace terrestre (on admet communément que les données présentant un caractère géographique représentent plus de 75% des données sur le web). Ces données ont des caractéristiques et des contraintes à la fois spatiales et attributaires et subissent des modifications à travers le temps.

Traditionnellement, la démarche de modélisation comporte une étape de perception du monde réel (incluant l'analyse des problèmes des utilisateurs et de leurs besoins), suivie par l'enchaînement des étapes de conception et d'implémentation. La conception facilite l'échange d'informations ; elle est

indépendante des technologies informatiques de codage et de stockage des données. Elle s'effectue à l'aide des schémas conceptuels qui sont une schématisation de données qui inclut des descriptions détaillées des types de données, des relations et des contraintes. En d'autres termes, le schéma conceptuel représente le cas d'étude selon un modèle formel sur des spécifications précises et facilite la compréhension et l'utilisation de l'application. Conformément aux données traditionnelles (alphanumériques) et afin de bien modéliser les données qui ont des caractéristiques temporelles (spatio-temporelles) il faut se fonder sur un modèle conceptuel spécifique à ces types de données. Ce modèle doit représenter d'une manière claire et compréhensible la temporalité des objets et des événements.

En effet, des exigences, outre celles liées à la modélisation des données traditionnelles, doivent être prises en compte lors de la phase de conception et le modèle conceptuel doit, entre autres, permettre d'aborder les points suivants (Parent *et al.*, 2009 ; Moisuc *et al.*, 2005).

- La modélisation de la dimension temporelle ainsi que de l'évolution temporelle des objets (suivi historique) ;
- L'orthogonalité entre les dimensions structurelle, spatiale et temporelle, c'est-à-dire qu'un choix d'une représentation dans l'une des dimensions ne doit pas limiter les possibilités de choix dans une autre dimension ;
- La modélisation d'une manière réaliste des objets et des événements qui se produisent dans un terrain d'étude, ainsi que leurs interactions ;

Une fois les informations temporelles collectées et stockées, l'interrogation et l'analyse constituent les fonctionnalités à maîtriser. Les données temporelles font l'objet d'une très vaste littérature qui multiplie les points de vue transverses aux disciplines scientifiques : physique, économie, philosophie, statistique et naturellement informatique. Le traitement du signal et les automates temporisés développent des théories, langages et environnements applicatifs élaborés tels que LUSTRE (Halbachs *et al.* 1992) ou MARTE (Rivas et Harbour, 2001 ; André *et al.* 2007). Ces langages dédiés sont naturellement efficaces dans le domaine du raisonnement temporel qui leur est propre, mais ciblent des spécialistes de ces technologies et non des utilisateurs métier. En revanche, ils peuvent, dans un second temps, faire office de « langages-ressources » : l'utilisateur interagit de façon ergonomique et conviviale, dans un langage proche du langage naturel, avec des modèles pivots⁵ qui masquent la complexité sans nuire à la pertinence ni à la qualité des résultats. Ainsi, les questions nécessitant un raisonnement complexe sont automatiquement traduites *via* l'Ingénierie Dirigée par les Modèles (IDM, Jézéquel *et al.*, 2012) vers de tels langages ressources, directement appropriés pour apporter une réponse, elle-même automatiquement reformulée en retour dans les termes du système propre à l'utilisateur final.

⁵ De tels modèles pivots sont proposés en Section 5.

3.2 Conception de la dimension temporelle

En modélisation temporelle, on distingue un concept de base, à savoir « l'instant », considéré comme un élément de dimension zéro sur une droite réelle et un concept construit, « l'intervalle temporel », élément de dimension un, limité par deux instants respectivement identifiés comme début et fin. Ces définitions géométriques élémentaires se combinent de façon plus ou moins complexe pour former des éléments structurés plus élaborés, ce qui soulève des questions difficiles lorsque l'on mêle temps discret et temps continu, et qu'il convient de se préoccuper de granularité temporelle. Les difficultés apparaissent également lors de l'ancrage du temps abstrait dans un référentiel et un calendrier. Enfin, la topologie des instants et intervalles conduit à une combinatoire largement étudiée elle aussi (Allen, 1983 ; Chittaro et Combi, 2002).

Que l'on considère les normes (ISO19108), les standards (OWL-Time, iCalendar), les modèles de conception, les bibliothèques dédiées aux aspects temporels dans les langages de programmation, ou enfin les implémentations dans les systèmes de bases de données (relationnel, XML, objet), les concepts et opérateurs sont voisins même si les terminologies peuvent varier.

En ce qui concerne les langages de conception, le métamodèle d'UML connaît les événements, les diagrammes de machines à états et diagrammes de séquences, dans lesquels l'annotation temporelle se fait *via* des éléments de types *timer* et *TimeEvent*. Cependant, les annotations sont formulées en texte libre, ce qui interdit un contrôle automatique ou tout autre traitement numérique de ces informations. Enfin, les événements, possiblement porteurs d'information sont des déclencheurs de transitions et des activateurs d'opérations, mais ne peuvent être manipulés comme des objets de premier rang dans les modèles métier.

Plusieurs modèles de conception de données spatio-temporelles cherchent à pallier ces faiblesses en étendant le métamodèle d'UML ou les définitions du modèle E/R. Parmi ces extensions, nous citons STUML « *Spatio-Temporal UML* » (Price *et al.*, 2000), *Perceptory* (Bedard, 1999), « Profil UML-SIG » (Miralles, 2005), STER (*Spatio-Temporal Entity-Relationship*) (Tryfona et Jensen 1999) et MADS (Parent *et al.*, 1997).

Perceptory, le plus connu des modèles spatio-temporels orientés objet, est fondé sur UML (*Unified Modeling Language*) (Fowler, 1998), il propose un support (sous forme d'icônes) pour les propriétés spatiales et temporelles. Ces dernières peuvent être utilisées simultanément, attribuées à la fois aux classes et à leurs attributs, mais ne peuvent pas concerner les associations entre les objets des schémas conceptuels. STER donne la possibilité de créer des relations spatiales et temporelles entre de telles entités.

Quant à MADS, détaillé dans la section 4.1, il permet de représenter les entités du monde réel sous forme d'objets et d'associations. Il assure une orthogonalité entre les dimensions structurelle, spatiale et temporelle. MADS possède des notations visuelles intuitives et lisibles pour représenter des concepts spatiaux et temporels, il dispose aussi de son propre langage pour interroger le modèle de

données (Parent *et al.*, 2006). Par ailleurs, il est considéré comme le meilleur modèle de conception dans diverses études (Parent *et al.*, 2006 ; Laplanche, 2002 ; Zaki *et al.*, 2011 ; Pelekis *et al.*, 2004).

En revanche, MADS ne spécifie pas nativement une distinction claire entre les objets et les événements et ne représente pas la séquence des événements qui peuvent se produire dans un terrain d'étude.

4. Proposition d'extension du modèle MADS

Notre travail s'inscrit dans le contexte de modélisation générique des données spatio-temporelles. Il peut être vu comme une extension de MADS améliorant et complétant la représentation des événements et l'évolution temporelle des objets. Grâce à une distinction entre les entités physiques (les continuants) et les événements (les occurrents dans le temps), notre approche traite les événements comme des éléments de premier rang.

4.1 La conception des données temporelles dans MADS

En MADS, la temporalité permet de définir le cycle de vie des attributs, des objets ou des associations. En effet c'est l'estampillage de ces concepts qui permet de localiser dans le temps la pertinence d'une entité.

MADS définit des types temporels applicables sur les objets, les associations et les attributs, et les organise en une hiérarchie qui part des types abstraits généraux (*Time*, *SimpleTime* et *ComplexTime*) vers les types précis simples (*Instant*, *Interval*) et complexe (*Instant Bag*, *Interval Bag*). La Figure 1 présente cette hiérarchie générique.

La temporalité associée à une classe d'objet ou à une association permet de garder une trace du cycle de vie de ces derniers : date de création, de suspension, de réactivation ou de suppression. Elle est attribuée en ajoutant l'un des pictogrammes temporels (cf. Figure 1) au niveau de la classe d'objets ou l'association. Au niveau de l'attribut, la temporalité exprime la période de validité, c'est-à-dire la période pendant laquelle l'information portée par l'attribut est valide dans le monde réel.

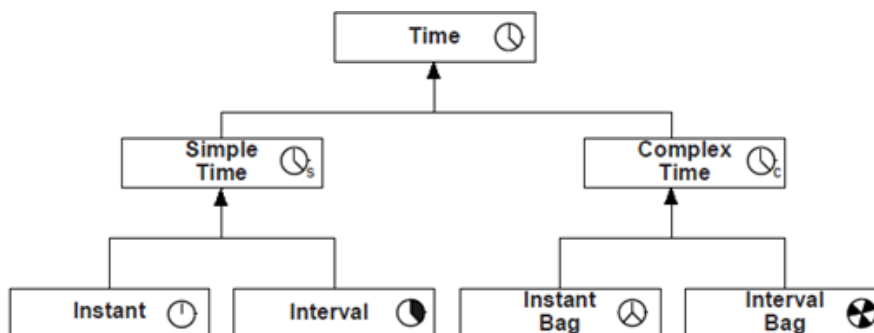


Figure 1. Hiérarchie des types temporels dans MADS (Parent *et al.*, 1997).

MADS permet aussi d'attribuer à tous les attributs, quel que soit leur type, une fonctionnalité de dépendance temporelle (*Time-varying attribute*), et dans ce cas on enregistre l'évolution de ses valeurs (Pour plus d'informations le lecteur peut consulter (Parent *et al.*, 2007)).

Il est également possible d'attribuer aux associations des contraintes temporelles sous forme de pictogrammes. Il s'agit alors des contraintes de synchronisations qui décrivent une relation temporelle entre des classes temporelles reliées par la relation en question.

4.2 Le schéma des continuants et le schéma des occurrents

Outre la modélisation générique des objets spatio-temporels, nous estimons nécessaire et utile de modéliser les processus exploitant des objets spatio-temporels dans un milieu urbain. Ces processus gèrent des ensembles (ou une succession) d'événements, qui peuvent posséder des caractéristiques spatio-temporelles et qui modifient et/ou utilisent les caractéristiques des objets physiques. En effet, dans notre modèle conceptuel, nous donnons la possibilité de représenter les événements et d'en étudier les causes, les conséquences, et les corrélations mutuelles. Autrement dit, la modélisation des événements (et la temporalité elle-même) peut être l'objectif d'un schéma conceptuel.

Notre approche de conception met en œuvre une distinction entre les entités physiques (les continuants) et les événements (les occurrents temporels). MADS ne permet pas nativement de faire cette distinction. Ainsi, nous avons différencié conceptuellement les entités de type physique (comme par exemple les bâtiments, les parcelles, les routes, etc.) de celles de type événement (comme par exemple la démolition des bâtiments, l'analyse des caractéristiques des bâtiments, la construction d'une route, l'effet des avalanches sur les bâtiments, etc.). Cette distinction se traduit ensuite graphiquement dans le schéma conceptuel.

Notre proposition va au-delà de la simple distinction graphique entre un objet physique (continuant) et un événement (occurrent temporel). En effet, nous proposons de créer des schémas conceptuels séparés pour modéliser soit les objets physiques d'un terrain d'étude soit les applications spatio-temporelles. Ainsi, un cas d'étude sera finalement représenté par un ou plusieurs schémas conceptuels qui interagissent. Nous imposons de subdiviser, lors de la conception, le schéma global d'un cas d'étude en « schéma des continuants comportant les objets physiques et leurs relations » et « schéma des occurrents décrivant les événements (influant sur les objets) et leurs relations ». Les schémas des continuants donnent une vue descriptive des terrains d'étude, tandis que les schémas des occurrents donnent une vue explicative des phénomènes (en suivant la succession des événements) qui touchent aux objets physiques.

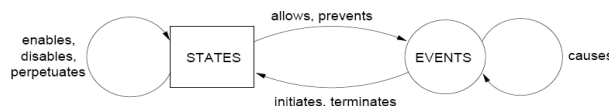


Figure 2. Relations entre les événements et les états (Galton et Worboys, 2005).

Pour organiser d'une manière rigoureuse ces relations et leur ajouter de la sémantique, nous nous appuyons sur le schéma proposé par Galton et Worboys (Galton et Worboys, 2005) qui propose un schéma (cf. Figure 2) spécifiant formellement la manière selon laquelle les événements interagissent, d'une part avec d'autres événements (relation de causalité) et d'autre part avec des objets (un événement peut changer l'état d'un objet et l'état d'un objet peut déclencher ou arrêter un événement). Le « statut » présenté en Figure 2 désigne, dans notre approche, un état d'un objet physique (en d'autres termes, il s'agit d'un attribut d'un objet physique dans le schéma des continuants).

Nous nous sommes inspirés des termes (*allows, prevents, causes, initiates, terminates*) utilisés dans cette étude pour imposer des nouvelles contraintes sur les relations qui peuvent exister entre des événements ou pour qualifier l'effet d'un événement sur le statut d'un objet physique.

Dans les modèles du domaine ainsi spécifiés, certaines classes peuvent se voir attribuer le statut d'événements. Les attributs de ces classes et leurs associations avec d'autres classes ayant ou non ce même statut dans le modèle, permettent de mettre en évidence des propriétés jugées importantes, qu'elles soient temporelles (période de validité) ou autres (structure, influence sur les continuants, etc.). Le modèle décrira les liens internes entre ces classes et des liens externes avec des classes non événementielles. Les instances des classes étiquetées/stéréotypées « événement » restent des instances des classes précitées, mais bénéficient de propriétés particulières propres aux événements (estampilles temporelles). Cette approche développe dans le cadre du modèle d'un domaine étendu, une vue sur les éléments de modélisation à caractère événementiel.

Si l'on veut tirer un profit maximum du nouveau statut d'événement il est nécessaire d'aller plus loin et de se dégager du modèle du domaine pour fournir un point de vue externe ; il convient de proposer à la fois un métamodèle qui traite des événements de manière générique, et un langage formel qui spécifie les propriétés temporelles qui leur sont attachées. Il faudra en particulier pouvoir traiter de différentes granularités, et adresser à la fois les classes et leurs instances.

Dans ces conditions, l'utilisateur dispose de deux vues cohérentes : d'un côté de la vue « Domaine » incluant classes étiquetées « événement », conformément à ce qui est exposé en section 4, et de l'autre, d'un modèle dédié à la description structurelle et temporelle de ces événements, présenté dans la section suivante, dans la perspective de mener des raisonnements formels et de répondre à des requêtes.

5. Spécifier des expressions temporelles en intension

Cette section décrit l'essentiel du métamodèle d'événement annoncé en section 4. On y précise également les fondements du langage formel permettant d'exprimer des propriétés temporelles. Un intérêt majeur de ce langage est de représenter des expressions temporelles exprimées en intension ou en extension sous une forme unique : la forme intensionnelle.

Une expression temporelle exprimée en intension permet de décrire un ensemble d'instant sans les énumérer. Ainsi l'expression « *Tous les jeudis du mois de*

novembre 2014 » correspond respectivement aux jeudis 6-11-2014, 13-11-2014, 20-11-2014 et 27-11-2014. A contrario, l'ensemble de ces dates forment une expression exprimée en extension. Grâce à l'expressivité en intension nous pouvons spécifier une récurrence temporelle d'un événement de manière concise.

Tout au long de cette section, nous nous appuyons sur un exemple concret tiré de la littérature dans le domaine de la Géographie. Nous présentons succinctement nos modèles avant de donner des indications techniques sur leur mise en œuvre.

Les cas d'utilisation que nous avons eu à traiter concernent des événements complexes, structurés de différentes manières et qui ont en général une durée non nulle. Ces caractéristiques sont plus larges que celles régissant usuellement les événements dans les systèmes à états. La complexité réside en particulier dans le fait que les événements peuvent être considérés à plusieurs niveaux de détail. Un événement peut lui-même se décomposer structurellement en sous-événements de différents types. Selon un point de vue temporel, on peut reconnaître des événements définis à différents degrés de granularité temporelle. Un travail complet et approfondi sur l'ensemble de ces questions est disponible sous la forme d'une thèse de doctorat (Faucher, 2012).

Concernant la modélisation spatiale, notre modèle s'appuie sur un modèle spatial existant (par exemple les modèles issus des normes ISO 19100) et notre contribution se focalise sur la spécification temporelle d'événements.

5.1 Modèle d'événement

Sur la base de ces remarques, nous pouvons produire le modèle correspondant pour spécifier de façon générique les événements selon un modèle de conception composite (Gamma *et al.*, 1995) enrichi, tel que donné en Figure 3.

La classe générique *Event* représente un événement quelconque ayant ou non des propriétés temporelles. Le modèle ne détaille pas la diversité des propriétés d'un tel événement et se contente de les regrouper dans la classe *NonTemporalMetadata*. Pour spécifier ces métadonnées, on se référera aux normes en vigueur qu'elles soient généralistes ou spécifiques à un métier. Nous faisons apparaître l'association *structural* avec les deux rôles *superEvent* et *subEvent*. Une telle architecture rend compte d'une hiérarchie sur les occurrences d'événements (e.g. : une des éditions annuelles des Fêtes de Gand (*superEvent*) et ses spectacles associés (*subEvent*)).

La classe *T_Event* spécialise la classe *Event* et représente un élément nommé qui est de fait un événement avec des propriétés temporelles. La trilogie *T_Event*, *T_AggregatedEvent* et *T_SimpleEvent* applique le modèle de conception composite où chacun des trois éléments joue respectivement le rôle de composite et de composant (intermédiaire ou final). La relation composite/composant structure le lien *temporal*. On représentera de cette manière l'association entre l'événement « FêtesDeGand » et ses éditions annuelles successives.

Le modèle d'information temporelle (*TemporalOccurrence*) sera décrit dans la section 5.3, mais le point d'ancrage est la classe *PeriodicTemporalOccurrence* à laquelle se rattachent les événements avec deux rôles spécifiques. L'un concerne les

feuilles du modèle composite à savoir : *tempElementPeriodicity*, et l'information temporelle décrit les propriétés des instances de la classe *T_SimpleEvent* ; l'autre concerne les nœuds composés. Il s'agit de *tempComponentPeriodicity* pour laquelle l'information temporelle décrit les propriétés des occurrences des nœuds composants selon le lien *temporal* et le rôle *temporalElement*.

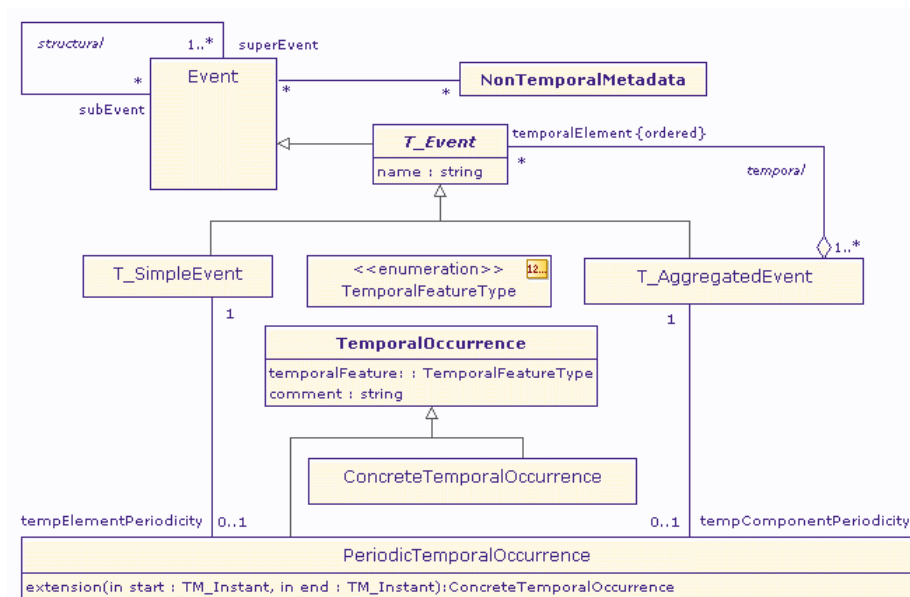


Figure 3. Modèle d'événement composite.

5.2 Cas d'utilisation

Pour nous référer à des exemples et des cas d'utilisation réels, nous aurions pu citer de nombreux travaux de fouille de données qui depuis ces vingt dernières années recherchent, exhibent et analysent des patrons d'occurrences d'événements répétitifs (Guyet et Quiniou, 2011 ; Srikani et Agrawal, 1996). Nous considérerons ici une étude particulière menée dans (Versichele, *et al.* 2012) qui traite de la fréquentation des Fêtes de Gand (Belgique) à des fins de gestion prévisionnelle des ressources (biens de consommation, moyens de contrôle et d'assistance) ou de maîtrise de la pollution. Les Fêtes de Gand constituent un événement qui se déroule annuellement au mois de juillet depuis un siècle et demi. Cette information est globale.

Plus précisément, les Fêtes de Gand ont lieu sur une période de dix jours durant la seconde quinzaine de juillet. Les dates exactes varient d'une année à l'autre. Certains sous-événements sont identifiés, tels que pour l'édition 2012, le festival des marionnettistes, le festival international de danse ou le festival du cirque pour la jeunesse, etc. Les programmes quotidiens, constituent une information encore plus

fine. Tous ces éléments sont des instances d'événements portant effectivement des informations temporelles plus ou moins complexes.

On distingue ainsi deux types d'associations, la Figure 4 illustre celles-ci dans le cadre du cas d'utilisation étudié.

En premier lieu, entre l'événement générique « LesFêtesDeGand » et l'ensemble des éditions annuelles de ces fêtes existe un lien temporel (*temporal*). Lorsque l'on affirme que « LesFêtesDeGand » ont lieu annuellement, on indique que l'on va trouver des occurrences : « FêtesDeGand_1985 », « FêtesDeGand_2006 », ..., « FêtesDeGand_2012 » dont les propriétés temporelles seront en accord avec la contrainte énoncée au niveau supérieur sur l'événement «LesFêtesDeGand» (une occurrence unique chaque année en juillet, etc.).

En second lieu, lorsque l'on reconnaît plusieurs sortes de festivals (Marionnettes, Danse, Cirque), on indique un lien structurel (*structural*) avec l'instance mère. Si les classes mère et filles sont porteuses d'informations temporelles, il doit exister une cohérence : les occurrences des filles doivent avoir lieu durant l'occurrence de la mère.

Pour les besoins de l'étude de fréquentation, chaque journée a été décomposée en tranches d'une heure, chaque tranche constituant de fait un événement. Il y a donc une succession de liens de type temporel (*temporal*) entre un sous-événement donné et chacune de ses occurrences ayant lieu chaque jour durant la période spécifiée (par exemple du 14 au 23 juillet 2012 pour les marionnettes).

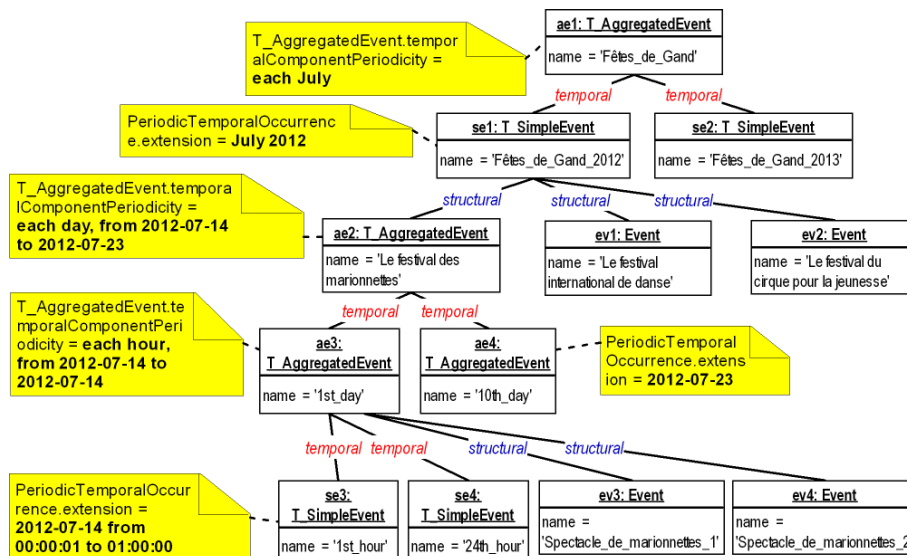


Figure 4. Diagramme d'objets illustrant les Fêtes de Gand.

Chaque jour se décompose lui-même (*temporal*) en 24 événements ayant respectivement lieu périodiquement toutes les heures. À chaque niveau de la décomposition hiérarchique correspond un événement ayant ses propres propriétés temporelles. La granularité de ces informations gagne en précision et perd en généralité selon le parcours descendant ou ascendant de la hiérarchie.

L'étude menée par Versichele *et al.* (2012), analyse les fréquentations absolues et relatives dans différents lieux et à différentes périodes avec des granularités diverses (heure, jour). Ses résultats font apparaître des périodicités et des relations entre occurrences d'événements éventuellement interprétées en termes de causalité. De manière générale, les pics de fréquentation ont lieu entre 22 h et 24 h sauf pour les dimanches et le jour de la fête nationale (21 juillet). Ces jours-là (dimanches et Fête Nationale), le pic de fréquentation est moins prononcé et s'étend entre 17 h et 20 h. *A contrario*, les faibles fréquentations ont lieu chaque jour de 2 h à 9 h.

Ainsi, les études menées sur l'ensemble des dates concrètes de présence de chaque visiteur⁶ apportent un ensemble d'informations de différents types :

- structurel : décomposition événements/sous-événements selon la catégorie et le lieu des spectacles,
- temporel : classes contiguës de fréquentation homogène avec leur motif (*pattern*) d'occurrences périodiques,
- relationnel : par exemple, l'interdépendance entre les Fêtes de Gand et la Fête Nationale en matière de fréquentation.

Pour rendre ces informations accessibles de façon systématique en les intégrant dans des bases de connaissance, il convient de les modéliser sous les trois aspects précédents. Plus précisément, les bases de connaissance ainsi constituées aident l'utilisateur (ici le géographe) à rechercher des événements aux propriétés voisines de celles mises en évidence afin de conforter ses hypothèses. En retour, les événements sur lesquels le géographe travaille et qu'il vient de contribuer à catégoriser, viendront enrichir la base de connaissance et participer *de facto* à la suite du processus.

Il est donc primordial de disposer d'un modèle et de langages adaptés à la description des propriétés périodiques ainsi que des catégories adéquates d'événements pour formuler des requêtes, interroger une base de connaissance et procéder à l'enregistrement de nouveaux résultats. Une suite de dates concrètes (l'extension des dates calendaires d'occurrence des événements) masque totalement la sémantique exprimée par des descriptions en intension avec des assertions du type : « *Lors des Fêtes de Gand, les pics de fréquentation ont lieu chaque jour à 23 h sauf les dimanches et pendant la Fête Nationale* ». Représenter de façon directe cette sémantique est précisément ce que nos modèles se donnent comme objectif.

⁶ Collectées en utilisant une technologie *Bluetooth*.

5.3 Modèle d'expression temporelle

La spécification du modèle d'expression temporelle vise deux objectifs. Le premier est de rester conforme aux normes reconnues dans le domaine, afin de garantir l'interopérabilité et faciliter la réutilisation. Le second, se rapprocher du langage naturel, est guidé par le processus d'interaction avec l'utilisateur qui préfère énoncer des règles de récurrence, plutôt que d'énumérer une liste (potentiellement non bornée) de dates concrètes. Exemple de règle synthétique, ici directement liée à la spécification d'un calendrier : « *Dimanche est un événement qui a lieu une fois tous les sept jours* ».

Notre modèle gère les dates concrètes calendaires exprimées en extension (*ConcreteTemporalOccurrence*), et aussi les expressions temporelles décrivant des ensembles en intension (*PeriodicTemporalOccurrence*). Les deux types d'information se correspondent, dans la mesure où – moyennant certaines conventions sémantiques – on peut associer à toute expression intensionnelle finie, la série des dates concrètes qui lui correspond.

Les attributs de la classe parente (*TemporalOccurrence*) permettent de décrire la nature et les caractéristiques du phénomène temporel considéré (cf. Figure 3). On peut ainsi pour un même événement (ex. : FêtesDeGand) spécifier plusieurs points de vue : période d'affluence, accès tout public, accès réservé aux professionnels, etc.

La Figure 5 donne un aperçu du modèle d'expression temporelle. Seul le cas des expressions périodiques est présenté ici, la modélisation des expressions concrètes étant en tout point conforme à la norme ISO19108.

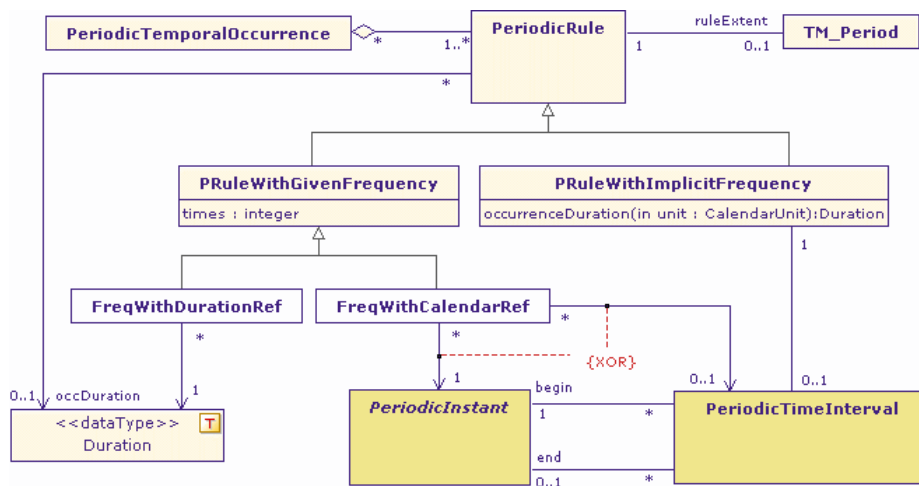


Figure 5. Extrait du modèle d'expression temporelle.

Une *PeriodicTemporalOccurrence* agrège plusieurs *PeriodicRule*. Chaque *PeriodicRule* représente un phénomène (pseudo) périodique⁷ simple (i.e. : mono fréquentiel). La sémantique de l'agrégation correspond à l'addition des signaux. Le rôle *ruleExtent* permet si nécessaire de limiter l'intervalle de temps durant lequel s'applique la *PeriodicRule*.

Notre modèle offre trois possibilités pour définir une *PeriodicRule*, qui correspondent respectivement aux spécialisations *FreqWithDurationRef*, *FreqWithCalendarRef* et *PRuleWithImplicitFrequency*. Contrairement au dernier cas, les deux premiers explicitent une fréquence (cf. attribut « *times* ») pour la *PeriodicRule*. On peut alors spécifier l'unité de temps en définissant une durée grâce au type *Duration* fourni par la norme ISO19108 :

- Pour *FreqWithDurationRef*, on spécifie un nombre d'unités calendaires (année, mois, jour, seconde, etc.). On dira par exemple : l'événement a lieu « *trois fois par période de deux mois* », l'élément *ruleExtent* permet de spécifier la première et/ou la dernière occurrence(s) si elles existent.
- Pour *FreqWithCalendarRef*, on spécifie un instant périodique en se référant au calendrier, comme dans des expressions telles que « *deux fois durant la troisième semaine de chaque mois* ». Les expressions dans lesquelles un événement calendaire est enrichi d'un rang sont spécifiées par des *PeriodicInstant* dont la structure n'est pas développée ici.
- Pour *PRuleWithImplicitFrequency*, on spécifie un intervalle périodique *PeriodicTimeInterval*, en spécifiant son début et sa fin par référence à deux *PeriodicInstant*. On dira par exemple : « *du 1^{er} jour au 14^{ième} jour du mois de janvier de chaque année* ». Pour des raisons d'intégrité, la définition de la classe *PeriodicTimeInterval* assure naturellement (entre autre) que les deux instants périodiques de début et de fin ont même fréquence.

Des modes de spécification supplémentaires intègrent des aspects non présentés ici. Il s'agit notamment d'exceptions interdisant certaines occurrences. Les exceptions ont la même syntaxe que les *PeriodicRule*, sauf qu'elles ne peuvent elles-mêmes supporter des exceptions. On peut également définir des positions relatives utilisant des relations topologiques d'Allen pour spécifier des *Periodic Rules* par rapport à d'autres (ex. : « *3 jours avant la pleine lune* », « *pendant la Fête Nationale* »).

5.4 Mise en œuvre conjointe des modèles d'événements et d'expressions temporelles

La Figure 3 montre comment s'articulent les événements et leurs propriétés temporelles (cf. les liens entre chacune des deux spécialisations de la classe *T_Event* et la classe *TemporalPeriodicOccurrence*). Dans la pratique, les approches métier

⁷ Des expressions telles que « *le second mardi de chaque mois* » réfèrent un phénomène non strictement périodique (i.e. : pseudo-périodique).

s'appuient sur des modèles du domaine reconnus comme des bases de connaissances fondamentales et stables, qui encapsulent les concepts, la terminologie et les contraintes. La question se pose alors de préciser comment le modèle d'événements peut être utilisé dans un tel contexte métier.

Deux points de vue sont envisageables.

Le premier importe le modèle d'événement et l'intègre tel quel lors de la phase de conception du modèle du domaine. Cela est naturel lorsque la notion d'événement est centrale dans le métier considéré. Cela est avéré notamment dans le contexte du tourisme, de la culture ou des médias.

Dans le second point de vue, on suppose que le modèle du domaine préexiste, et qu'il a déjà été instancié. Il s'agit alors d'apporter une dimension temporelle complémentaire, sur les instances existantes. Pour ce faire, on annote le modèle du domaine afin de distinguer les classes portant des informations temporelles, les associations de type *temporal* et de type *structural*. Ces éléments annotés sont mis en correspondance avec des classes et associations du modèle d'événement, ce qui permet d'ajouter un type « événement » aux instances du modèle du domaine et, en conséquence, d'y rattacher des informations temporelles.

Ce processus n'est pas trivial et nécessite de recourir à l'ingénierie dirigée par les modèles pour assurer à la fois un caractère générique et une automatisation. À titre d'exemple, annoter comme *T_Event* la classe « LesFêtesDeGand » du modèle du domaine aura pour effet de créer une instance d'événement temporel « LesFêtesDeGand » qui portera une *PeriodicRule* indiquant de façon formelle, pour l'événement, son caractère annuel, en juillet, depuis 1840. Cette nouvelle instance sera rattachée par un lien *temporal* aux instances du modèle du domaine correspondant à chacune des différentes éditions des Fêtes de Gand.

5.5 Outillage des modèles

Nous venons d'indiquer que l'usage pratique des deux modèles proposés, nécessite le recours à des techniques propres à l'Ingénierie Dirigée par les Modèles (IDM, Jézéquel *et al.*, 2012) comme la transformation et le tissage de modèles. Les règles de transformation utilisent conjointement en entrée le modèle du domaine annoté, le modèle d'événements et le modèle d'expressions temporelles pour produire automatiquement un modèle intégré qui permet soit d'interroger les événements pour obtenir leurs propriétés temporelles, soit de tester les propriétés temporelles pour filtrer des événements.

Dans le cas plus simple où le modèle d'événement se suffit à lui-même, la question n'est pas de réorganiser un corpus d'instances existantes, mais de peupler le modèle en question. Les solutions que nous avons retenues sont classiques : des adaptateurs spécifiques permettent de traduire des données sources XML, RDF ou relationnelles pour les rendre compatibles avec le modèle d'expressions temporelles (cf. Figure 6).

Ce processus nécessite naturellement de définir une correspondance (*mapping*) entre la structure entrante et notre modèle. On dispose pour cela d'un outil graphique

interactif ergonomique. Une fois ce *mapping* réalisé, vérifié et validé, le traducteur peut être généré automatiquement.

Si le modèle temporel doit être peuplé « manuellement », et non à partir de données numériques, nous utilisons une grammaire formelle proche du langage naturel, qui est la contrepartie rigoureuse de la structure objet des *PeriodicRules*. Un éditeur contextuel et syntaxique, aide l'utilisateur à saisir les données. Celles-ci lui sont restituées en langage pseudo naturel normalisé (c'est-à-dire reformulées selon notre grammaire) pour validation, avant d'être effectivement enregistrées.

Enfin, la structure mémoire (objet) du modèle temporel avec ses instances peut être sérialisée pour assurer la persistance des données.

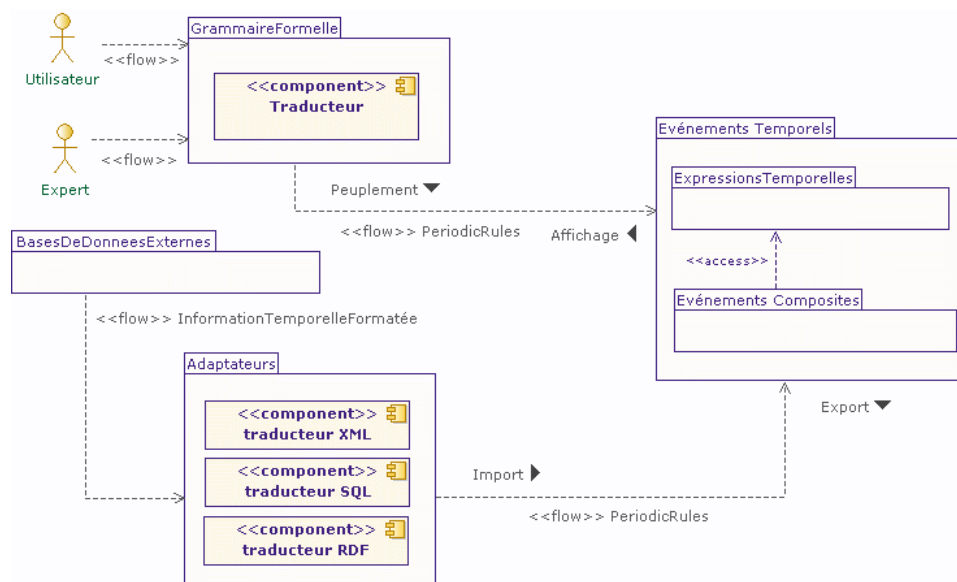


Figure 6. Architecture générale de la plate-forme.

6. Conclusion

Bien que les différents modèles présentés aient été élaborés de façon indépendante et répondent à des besoins différents, ils valent par la complémentarité comme des réponses apportées. Leur cohérence s'explique par le fait que le premier étend un modèle général (MADS) adapté à la spécification conceptuelle des systèmes d'information à caractère spatio-temporel, et que le second développe une approche objet et IDM qui se place dans le même paradigme à des fins voisines.

La modélisation conceptuelle dans MADS étendu, permet d'identifier dans un modèle métier les éléments de modélisation évolutifs (classes ou associations) dont on souhaite spécifier la dynamique. On raffine la notion de dynamique en précisant – dès l'analyse – le type de propriété temporelle : instant, intervalle, sac (*bag*)

d'instants, sac d'intervalles (cf. Figure 1). Un apport fondamental du présent travail est la promotion des événements en tant qu'éléments de modélisation de premier rang reconnus comme *occurents*.

Parallèlement, le modèle d'expressions temporelles traite des deux mêmes préoccupations à un niveau plus concret et apporte des moyens rigoureux d'implémentation des concepts spécifiés dans MADS étendu :

- En premier lieu, un métamodèle d'événement composite est fourni, il permet aussi bien de modéliser un événement simple qu'un événement structurellement ou temporellement complexe (périodicités et/ou granularités multiples), tout en prenant en compte sa dimension spatiale. Il a été présenté en section 5, de quelle manière ce métamodèle d'événement pouvait s'articuler avec un modèle métier. La même démarche permet de s'interfacer avec MADS. Dans un autre ordre d'idées, ce métamodèle peut être vu comme un type abstrait permettant d'implémenter directement les *occurents* identifiés avec leurs propriétés temporelles.
- La définition élaborée du métamodèle d'expressions temporelles permet la spécification extensionnelle ou intensionnelle des dates d'occurrences d'événements potentiellement répétés. Nous fournissons ainsi un moyen de décrire la dynamique grâce à un langage formel (grammaire) qui facilite la vérification et le raisonnement sémantique. C'est donc un moyen d'expression privilégié pour spécifier la dynamique et la synchronisation des *occurents* dans les systèmes d'information. Le métamodèle inclut la spécification du calendrier et de sa sémantique, ce qui assure de pouvoir ancrer dans le calendrier, les spécifications abstraites des *occurents*.

Ces modèles issus du paradigme objet ont vocation à servir de représentations pivot qui facilitent le développement d'applications en permettant l'interopérabilité. Grâce aux technologies de l'IDM, une partie importante du code peut être générée automatiquement par transformation de modèles. Les données gérées peuvent être traduites d'un espace technique vers un autre, facilitant ainsi l'édification de ponts par exemple entre le monde relationnel (RDBMS) et le monde ontologique (RDF) ou le raisonnement par exemple avec des langages à horloges (Halbwachs *et al.* 1992). On notera que l'exemple des Fêtes de Gand comporte à la fois une modélisation du phénomène (les fêtes, les différents festivals, les spectacles) dans leurs dimensions structurelle, spatiale et temporelle, et parallèlement, une modélisation du processus d'observation et d'analyse (événements périodiques de comptage des spectateurs, reconnaissance et positionnement spatio-temporel des instants ou intervalles de fréquentation extrême).

En conclusion, le modèle MADS et ses extensions apportent un contexte, des éléments de modélisation et un ensemble de concepts spatio-temporels fondamentaux dans le processus de conception d'un grand nombre de systèmes d'information. Nos métamodèles d'événements et d'expressions temporelles spécifient un ensemble cohérent de types abstraits organisés permettant de gérer ces éléments de modélisation, de vérifier leur intégrité sémantique et d'en faciliter l'implémentation ainsi que d'assurer la persistance des données.

MADS est un modèle connu et validé, les extensions proposées permettent d'étendre son champ d'application.

Le modèle d'événement temporel présenté s'appuie sur des standards et propose des extensions fonctionnelles. Des mises en œuvre concrètes séparées de chacune de ces approches ont été effectuées dans le cadre de MADS étendu (Zaki *et al.*, 2010) ainsi que dans celui du modèle temporel (Projet ANR⁸: RelaxMultimedia II – 2009/2012).

7. Bibliographie

- Allen J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, vol. 26, n° 11, p. 832-843.
- André C., Mallet F., De Simone R. (2007). Modeling Time(s). *MODELS'07*, LNCS, Springer Berlin / Heidelberg, vol. 4735, p. 559-573.
- Bedard Y (1999). Visual Modeling of Spatial Databases Towards Spatial Extensions of PVL and UML. *Geomatica*, vol. 53, n° 2, p. 169-186.
- Chittaro L., Combi C. (2002). Temporal granularity and indeterminacy in reasoning about actions and change: an approach based on the Event Calculus. *Annals of Mathematics and Artificial Intelligence* vol. 36, n° 1-2, p 81–119.
- Faucher C., Tissot C., Lafaye J.Y., Bertrand F., Brosset D., Rouan M, (2001a). Location- and temporal-based services for nature-society interaction regulation. *Journal of Location Based Services (JLBS)*, Taylor & Francis, vol 4/3, n°4, p 147-165.
- Faucher C., Teissède C., Lafaye J.Y., Bertrand F., (2001b). Temporal Knowledge Acquisition and Modeling. *EKAW 2010 - 17th International Conference on Knowledge Engineering and Knowledge Management*, Lisbon (Portugal), LNCS, Springer-Verlag, vol. 6317, p. 271-280.
- Faucher C., (2012). *Modélisation d'événements composites répétitifs, propriétés et relations temporelles*. Thèse en Informatique, Université de La Rochelle.
- Fowler M., Scott K. (1998). *UML Distilled - Applying the Standard Object Modeling Language*, Addison-Wesley, ISBN 0-201-65783-X.
- Galton A., Worboys M. (2005). Processes and events in dynamic geo-networks. *GeoSpatial Semantics. Proceedings of First International Conference GeoS 2005*, Mexico City, LNCS, vol. 3799, p. 45-59.
- Gamma, E., Helm R., Johnson R., Vlissides J. (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, ISBN 0-201-63361-2.
- Guyet T., Quiniou R. (2011). Extracting Temporal Patterns from Interval Based Sequences *Proc. IJCAI*, Barcelone (Espagne), p 1306-1311.
- Halbwachs N., Lagnier F., Ratel C. (1992). Programming and verifying critical systems by means of the synchronous data-flow programming language Lustre. *IEEE Transactions*

⁸ Projet Relaxmultimedias 2 : <http://www.rmm2.org>

on Software Engineering, Special Issue on the Specification and Analysis of Real-Time Systems. vol. 18, n°9, p. 785-793.

Hillier B., Hanson J. (1984). *The Social Logic of Space*, Cambridge University Press.

Jézéquel J.M., Combemale B., Vojtisek D. (2012). *Ingénierie Dirigée par les Modèles : des concepts à la pratique*, Ellipses, 144 pages.

Ladkin P.B. (1987a). Specification of Time Dependencies and Synthesis of Concurrent Processes, *International Conference on Software Engineering (ICSE)*, p. 106-115.

Ladkin P.B. (1987b). Models of Axioms for Time Intervals. *Proceedings AAAI-87 Sixth National Conference on Artificial Intelligence*, Seattle, WA., July 1987, p. 13-17.

Laplanche, F. (2002). Conception de projet SIG avec UML. *Bulletin de la Société géographique de Liège*, vol. 42, p. 19-25.

Leban B., McDonald D., Forster D. (1986). A Representation for Collections of Temporal Intervals. *Association for the advancement of artificial intelligence AAAI*, p. 367-371.

Miralles A. (2005). *Ingénierie des modèles pour les applications environnementales*. Thèse, Université de Montpellier.

Moisuc B., Davoine P., Gensel J., Martin H. (2005). GenGHIS : Un outil de modélisation spatio-temporelle pour le suivi historique des risques naturels. *Ingénierie des Systèmes d'Information*, vol. 10, n° 4, p. 35-58.

Olivé A., Raventós R., (2006). Modeling events as entities in object-oriented conceptual modeling languages. *Data Knowledge Engineering*, vol. 58, n° 3, p. 243-262.

Parent C., Spaccapietra S., Zimányi E., Donini P., Plazanet C. *et al.* (1997). MADS, modèle conceptuel spatio-temporel. *Revue Internationale de Géomatique*, vol. 7, n°3-4, p. 317-352.

Parent C., Spaccapietra S., Zimányi E. *et al.* (1998). Modeling Spatial Data in the MADS Conceptual Model. *Spatial Data Handling '98 Conference Proceedings*, Vancouver, BC, Canada, p. 138-150.

Parent C., Spaccapietra S., Zimányi, E. (2006). *Conceptual Modeling for Traditional and Spatio-Temporal Applications: The MADS Approach*, Springer, vol. 3882, p. 838-848.

Parent C., Spaccapietra S., Zimányi, E. (2009). Semantic Modeling for Geographic Information Systems. *Encyclopedia of Database Systems*, Springer Publishing Company. ISBN:0387355448 9780387355443, p. 2571-2576.

Pelekis N., Theodoulidis B., Kopanakis I., Theodoridis Y. (2004). Literature review of spatio-temporal database models. *The Knowledge Engineering Review*, vol. 19, n° 3, p. 235-274.

Price R.J., Tryfona N., Jensen C.S. (2000). Extended SpatioTemporal UML: Motivations, Requirements and Constructs. *Journal on Database Management*, Special Issue on UML, vol. 11, n°4, p. 14-27.

Rivas M.A., Harbourg M.G. (2001). MaRTE OS: An Ada Kernel for Real-Time Embedded Applications. *In Proceedings of the International Conference on Reliable Software Technologies, Ada-Europe-2001*, LNCS, vol. 2043, p. 305-316.

Srikant R., Agrawal R (1996). Mining Sequential Patterns: Generalizations and Performance Improvements. *Proc. 5th international conference on Extending DataBase Technologie (EDBT)*, p. 3-17.

- Terenziani P., (1997). Qualitative and Quantitative Temporal Constraints about Numerically Quantified Periodic Events. *Proc. of the 4th International Workshop on Temporal Representation and Reasoning. (TIME'97)* IEEE Computer Society, p. 94-101.
- Tryfona N, Jensen C.S. (1999). Conceptual data modeling for spatiotemporal applications. *GeoInformatica*, vol. 3, n° 3, p. 245-268.
- Versichele M, Neutens T., Delafontaine M., Van de Weghe N. (2012) The use of bluetooth for analysing spatiotemporal dynamics of human movement at mass events : A case study of the Ghent Festivities. *Applied Geography*, vol. 32, n° 2, p. 208-220.
- Zekri E., Hégron G., Legendre A. (2008). Environmental classification of urban pedestrian walkways. *9th International Conference on Design & Decision Support Systems in Architecture and Urban Planning*. Leende (The Netherlands).
- Zaki C., Zekri E., Servières M., Moreau G., Hégron G. (2010). Urban Spatiotemporal Data Modeling: Application to the Study of Pedestrian Walkways. *Proc. of the KES International Symposium IDT 2010. Smart Innovation, Systems and Technologies*, vol. 4, XI, p. 549-558.
- Zaki C., Zekri E., Servières M., Moreau G. (2011). Implementing Conceptual Spatiotemporal Model into Object DBMS with Semantic Preserving. *In ICSDM& BJ-IWGIS*, p. 281-290.