
Outils pour la production de présentations hypermédias

Cécile Roisin

*Inria Rhône-Alpes (équipe Wam) & Université Pierre Mendès-France
ZIRST - 655 avenue de l'Europe - Montbonnot
F-38334 ST ISMIER CEDEX
Cecile.Roisin@inrialpes.fr*

RÉSUMÉ. La production et l'accès de plus en plus aisé aux contenus multimédias numériques suscitent naturellement l'envie de les intégrer pour réaliser des « documents multimédias » dans lesquels le sens est apporté non seulement par le contenu de chacun des éléments de la présentation mais aussi par la synchronisation (spatiale et/ou temporelle) et les liens de navigation entre ces éléments. Depuis plusieurs années, un effort important de recherche et de standardisation de ces formats d'intégration a été apporté (SMIL, MPEG4). Les éléments de contenu eux-mêmes font l'objet de standard pour enrichir leur niveau de description (MPEG7). Ces différents niveaux de description permettent de réaliser des présentations multimédias dans lesquelles non seulement il est possible de spécifier l'intégration et la navigation entre les éléments de contenu mais des synchronisations fines peuvent également être définies entre des fragments internes de ces éléments (plans d'une vidéo, parties de texte ou d'image, etc).

Tirant parti de ces possibilités, nous étudierons les besoins des utilisateurs à différents niveaux de la chaîne de production des présentations et verrons comment les outils répondent à ces attentes, en particulier en matière de pouvoir d'expression, de réutilisation et de facilité d'édition.

ABSTRACT. Even if multimedia contents are widely used and standard formats exist both for media items and presentation structures, there is still a gap between user requirements and existing tools. In this chapter we examine what are the user needs for specification, management and edition of multimedia documents. We illustrate our discussion with the SMIL language and its players and authoring tools.

MOTS-CLÉS : Edition multimédia, présentation multimédia, synchronisation, outil auteur, SMIL.

KEYWORDS: Multimedia composition, multimedia presentation, synchronization, authoring tool, SMIL.

1. Introduction

Cet article aborde le problème de la création de documents hypermédias et plus précisément des moyens mis à la disposition des utilisateurs pour concevoir et réaliser leurs documents. C'est un axe à la fois déjà bien balisé de par l'existence de standards et d'outils largement diffusés, mais encore en pleine évolution du fait de l'apparition de nouvelles possibilités de communication et de traitement de l'information. Par ailleurs, il n'existe pas d'outils vraiment conviviaux qui permettent la production aisée de contenus et de présentations multimédias. Ainsi, il est le plus souvent nécessaire de faire appel à des compétences informatiques pour réaliser les scénarios multimédias. C'est pourquoi, parmi les trois grands challenges identifiés récemment par l'ACM Multimedia Special Interest Group (Rowe *et al.*, 2003), le premier thème de recherche concerne la conception d'outils auteur « *to make authoring complex multimedia titles as easy as using a word processor or drawing program* ». Les directions de travail de cet axe englobent non seulement la définition de modèles répondant aux besoins de spécification, mais il s'agit aussi de trouver les bons paradigmes d'interface homme-machine permettant aux auteurs d'acquiescer à tout moment le bon niveau d'abstraction nécessaire à la réalisation des opérations d'édition voulues.

La section 2 de cet article fixe le cadre de travail dans lequel nous définirons plus précisément l'objet de notre étude : « le document multimédia ». La partie suivante est consacrée aux besoins d'expression des auteurs et à l'étude de cas du langage SMIL. Enfin, la quatrième partie analyse les besoins en termes d'édition multimédia et illustre les solutions possibles à travers les outils d'édition du langage SMIL.

2. Notion de document multimédia

Devant multimédia, il y a document, notion qui fait l'objet actuellement d'une vaste réflexion pluridisciplinaire dans le cadre du groupe de travail RTPdoc. Le texte signé R. T. Pédaque (Pédaque, 2003), qui présente une synthèse de ces travaux sur l'étude des conséquences de l'évolution du document papier vers le document électronique, a identifié trois axes pour appréhender ce qu'est un document :

- la forme : l'objet matériel (ou immatériel),
- le signe : le sens de l'objet,
- et le médium : le vecteur de transmission.

Dans le cadre de notre étude, c'est le premier axe qui nous intéresse en priorité puisque nous considérons le problème de la composition de l'objet « document ». Plus précisément, un auteur crée un document par le regroupement d'un ensemble d'éléments de contenu. Le terme de multimédia ajoute à cette définition l'intégration de contenus de différentes natures, discrètes comme le texte ou l'image et continues

comme le son ou la vidéo. La conséquence est l'introduction de la dimension « temps » puisque la synchronisation entre les éléments de contenu fait partie de la définition de la forme du document.

De fait, on constate une importance grandissante prise par cette dimension du temps dans les documents. En effet, les premiers documents multimédias sont apparus dans les années 60 et consistaient à intégrer des images dans du texte, donc des contenus statiques ne demandant pas de synchronisation temporelle. Puis on a cherché à intégrer des données visuelles continues (vidéo, animations) et des données non visuelles (son). Plusieurs approches ont été prises pour assurer la synchronisation entre ces données :

- À travers l'identification d'un élément maître sur lesquels les autres éléments se synchronisent. C'est donc une approche « média-centrique » qui est souvent utilisée pour constituer des présentations à partir d'un flux audiovisuel enrichi par des annotations textuelles ou des liens de navigation. Les synchronisations restent limitées et faciles à maîtrisées grâce au fil conducteur que constitue le contenu maître.
- À travers un modèle de temps qui s'appuie sur l'existence d'un axe temporel absolu (équivalent à un contenu maître vide), les éléments et événements de la présentation se synchronisant sur cet axe. Ce modèle a été largement utilisé pour la construction de présentations multimédias sur cédérom avec des logiciels comme Director, et il est à la base du placement des animations dans Powerpoint par exemple.
- À travers un modèle de temps à base d'opérateurs temporels et/ou de relations entre les éléments modélisés sous forme d'intervalles temporels. La différence principale avec les approches précédentes réside dans le fait qu'il n'y a pas de référent temporel absolu utilisé par l'auteur pour placer les éléments, c'est le système de présentation qui calcule le placement effectif des éléments de la présentation. Différents modèles ont été proposés (CMIF, ZyX, Madeus, SMIL) et offrent des pouvoirs d'expression variés qui ne sont pas développés ici (voir (Blakowski *et al.*, 1996), (Wahl *et al.*, 1994) pour une étude détaillée).

On peut également distinguer deux catégories de spécification : celles qui sont orientées données (déclaratives) et celles qui sont orientées traitements (impératives comme celles fournies par les langages de script). Bien évidemment, plus le langage déclaratif est de haut niveau, (l'auteur déclare le résultat qu'il attend et non comment l'obtenir), plus le système de formatage peut être amené à effectuer des calculs complexes comme résoudre l'ensemble des contraintes de placement spécifiées par l'auteur. Pour concrétiser cette étude, nous verrons en section 3 de cet article comment le langage SMIL, de type déclaratif, couvre les besoins d'expression des auteurs.

Les « documents multimédias » se trouvent au coeur de nombreuses classes d'applications, comme la réalisation de cédéroms interactifs, de sites web multimédia, ou les systèmes de télévision interactive et de visioconférences. Ces

applications nécessitent toutes (à des degrés divers) de prendre en compte les problèmes d'intégration d'éléments multimédia avec leur synchronisation temporelle (ordonnancement temporel), de stockage de données volumineuses nécessitant des techniques de compression, et d'adaptation de flux en fonction des contraintes d'accès. Par exemple, pour diffuser des séminaires, on peut utiliser ces technologies pour assurer la synchronisation des transparents avec le film vidéo et la navigation dans ce flux à l'aide d'une table des matières (Inria, 2004), (Nicoll *et al.*, 2003). L'identification d'une structure générique dans de tels documents permet d'envisager l'automatisation de leur production. Pour aller plus loin et réaliser des applications dans lesquelles les contenus sont adaptés aux utilisateurs (un exemple typique est la réalisation d'applications pédagogiques), il faut compléter ces processus de production avec des mécanismes d'adaptation (par filtrage et/ou restructuration) et enrichir les modalités d'interaction. Un autre domaine d'application est la mise en valeur et l'exploitation de fonds documentaires multimédia, qu'ils soient issus des musées, des entreprises (documentation technique par exemple), des laboratoires de recherche ou même de l'accumulation des contenus numériques de la famille.

Bien évidemment, en amont de ces applications se pose la question de comment concevoir et éditer ces documents. C'est ce que nous étudierons dans la section 4.

3. Besoins d'expression des documents multimédia

3.1. Formats de représentation

L'introduction de la dimension temporelle dans les documents a induit nécessairement de nouveaux besoins d'expression : la durée des objets, leur placement temporel et leur synchronisation qui constituent le scénario temporel. Ces informations temporelles doivent s'intégrer dans l'ensemble des informations attachées aux documents. Ainsi, la définition d'un *format de représentation* des documents multimédia doit permettre d'exprimer les caractéristiques relatives à chacune des différentes dimensions de ces documents : dimension logique, c'est-à-dire l'organisation des informations en composants hiérarchiques, dimension spatiale, dimension hypermédia et dimension temporelle. Par exemple, pour spécifier le déplacement d'un objet sur l'écran, il est nécessaire d'exprimer une information spatiale pour la trajectoire, comme les positions initiale et finale correspondant au déplacement, ainsi qu'une information temporelle comme la durée du déplacement et sa synchronisation avec d'autres objets.

Les formats de représentation standard s'appuient tous sur XML comme format de structuration et de description :

1. Les éléments source peuvent être organisés selon des structures et un vocabulaire « métier ». La plupart des grands domaines d'activité ont fait l'objet d'un travail de définition de leurs vocabulaires sous forme de

DTD (*Document Type Definition*) ou de *schema XML*. Le site *XML Applications and Initiatives* (Cover, 2004) référence ainsi plus de 700 définitions, comme par exemple *TransportationXML (tXML)*, *Election Markup Language (EML)*, *Trading Partner Agreement Markup Language (tpaML)* encore *XML System for Textual and Archaeological Research (XSTAR)*. De même, XML est utilisé pour décrire les contenus selon différents niveaux de granularité en fonction des applications visées. Par exemple, MPEG7 permet d'annoter les contenus audiovisuels à l'aide des descripteurs de différents niveaux structurels ou sémantiques.

2. La définition de la présentation : XML est également utilisé pour définir des structures de présentation, que ce soit pour les présentations statiques, XHTML est l'exemple le plus répandu, ou dynamiques comme le langage d'intégration de média SMIL. D'autres langages permettent de définir l'apparence des documents comme les langages de style CSS, XSL-FO, et de liens Xlink. Là encore, ces différents standards offrent un bon niveau d'expression. Les besoins non couverts concernent la possibilité de définir de façon plus fine et surtout plus abstraite des comportements spécifiques comme des animations.
3. La spécification des transformations : Dès lors qu'on est amené à décrire l'information avec plusieurs langages XML, le langage XSLT est l'outil nécessaire pour spécifier la transformation des structures d'un modèle à l'autre. En particulier, un schéma de production largement répandu consiste à définir et structurer les contenus selon un schéma « métier » et à les transformer vers un ou plusieurs langages de présentation cible (XHTML, XSL-FO, SMIL, etc.) à l'aide de feuilles de style écrites en XSLT.

3.2. Description des documents multimédia avec le langage SMIL

Le langage SMIL, Synchronized Multimedia Integration Language, comme son nom l'indique, est un langage d'*intégration* de média (au sens « d'élément de contenu ») pour réaliser des présentations audiovisuelles interactives accessibles sur le web. Le terme « intégration » concerne non seulement le placement spatial des objets dans l'écran mais surtout leur agencement temporel défini sous forme d'un scénario qui exprime leurs synchronisations et enchaînements. A ces deux fonctions de base, placement spatial et placement temporel, est associé un ensemble d'éléments du langage qui couvrent toute une palette de besoins complémentaires : spécification de transitions, de liens hypermédia, d'animations et de contrôle fin du temps. SMIL est un langage XML qui vise à permettre aux auteurs d'utiliser une approche déclarative pour décrire leurs documents multimédias. Cependant, la complexité de la syntaxe proposée, liée en partie à la richesse de son pouvoir d'expression, en fait un langage réservé à l'usage des informaticiens. D'où la

nécessité de proposer des environnements auteur conviviaux comme décrits en section 4.

Le modèle de synchronisation comporte trois types d'éléments syntaxiques :

1. des opérateurs temporels : par, seq et excl
2. des structures de contrôle : switch et repeat
3. des attributs de synchronisation : begin, end, dur et endsync.

Le modèle de placement spatial utilise deux types d'éléments syntaxiques :

1. des régions hiérarchiques,
2. des attributs de placement : top, width, etc.

La structure d'un document SMIL (voir figure 1) reflète ce besoin de prendre en compte deux structures indépendantes : la structure spatiale (par la hiérarchie d'éléments « region ») et la structure temporelle (par la hiérarchie d'éléments « seq », « par », « excl »). Cependant, comme l'accent est mis sur la spécification des synchronisations entre éléments, c'est la structure temporelle qui est la structure maîtresse : ses feuilles contiennent ou référencent les éléments de contenu de type « text », « img », « video », « audio », « a » (ancrage), etc. qui eux-mêmes référencent la région d'affichage dans laquelle ils doivent être placés (attribut « region »).

```
<smil>
<head>
<layout>
  <root-layout width="250" height="500"
    background-color="blue" />
  <region id="logo" left="25" top="30" width="32"... />
  <region id="pres" left="50" top="70" width="100"... >
    <region id="sous-part".../>  </region>
  ...
</layout>
</head>
<body>
<par>
  <seq>
    
    
  </seq>
  <video src="..." region="pres" dur="20s" />
</par>
</body>
</smil>
```

Figure 1. Extrait d' une source SMIL

Si on s'intéresse maintenant à l'intégration fine des objets de contenu, par exemple pour synchroniser spatialement et temporellement des textes avec l'apparition de personnages dans le flot d'une vidéo, d'autres fonctions d'édition sont nécessaires : la spécification (automatique ou non) de la décomposition de la vidéo au niveau de granularité souhaité (scène, plan, objet), la sélection des fragments et l'édition de leur synchronisation. Pour le premier type de fonctions, de nombreux projets de recherche ou des outils offrent des solutions pour analyser automatiquement les flux vidéos et les segmenter en plans. Les découpages plus fins et/ou plus sémantiques (comme l'identification et le suivi de personnage ou encore le découpage en scènes) sont encore au stade de la recherche. Par contre, un effort important de standardisation est mené pour permettre de décrire ces structures internes de contenu. C'est l'objectif des groupes de travail MPEG, et en particulier de MPEG7. Avec de tels outils de description de contenu, il est possible d'annoter les flux temporels pour permettre aux applications d'exploiter plus finement le contenu. La standardisation des descriptions offre l'indépendance entre les systèmes d'annotation et les systèmes applicatifs qui les utilisent principalement pour la recherche et l'accès au contenu stocké dans des bases de données multimédia. Les descriptions de contenu ont été plus rarement exploitées dans le cadre des systèmes de traitement de documents multimédia. Les annotations qui définissent la structure temporelle (enchaînement des plans dans les scènes) et spatio-temporelle (déplacement d'un personnage ou d'un objet dans un plan) d'un objet vidéo, constituent pourtant la base nécessaire pour offrir le moyen de spécifier des synchronisations fines avec cet objet.

Un telle approche a été expérimentée dans (Tran Thuong *et al.*, 2002). Ce système s'appuie d'une part sur un sous-ensemble de MPEG7 pour la description de la vidéo et sur un modèle temporel riche Madeus (à base des relations de Allen) pour la synchronisation interobjets.

SMIL ne permet pas d'aller aussi loin dans la description des contenus, entre autres parce que les éléments de contenu sont uniquement déclarés comme les feuilles de la structure temporelle. Cependant, il est possible d'identifier des fragments de contenu au sein d'un média avec l'élément SMIL « area » dont les attributs permettent de définir l'extrait choisi en terme d'instant de début et de fin (voir Figure 2). Ainsi, le besoin d'expression est-il couvert, mais uniquement selon un mode de spécification absolu, ce qui le rend difficile à utiliser par les auteurs.

```

<par endsync="musique">
  <text id="démarrage" src="..." />
  <audio id="musique"
    begin="démarrage.activateEvent" src="..." />
  
  <video id="film"
    fill="freeze" src="..."
    <area id="héros" begin="22s" end="31s"
      href="http://www.films-héros.html"/>
  />
</par>

```

Figure 2. Extrait d' une source SMIL avec des attributs de synchronisation

3.3. De la spécification à la présentation

Les concepteurs de documents multimédia n'ont pas seulement besoin de moyens de description de leur documents, ils veulent également des moyens pour gérer et exploiter le plus efficacement possible l'ensemble des ressources des présentations multimédia qu'ils réalisent. Cette efficacité peut être obtenue aux deux bouts de la chaîne de réalisation, d'une part par la réutilisabilité des objets utilisés et d'autre part par l'adaptabilité des documents au moment de leur présentation. Ces deux fonctions sont offertes par le langage SMIL. En effet, la réutilisabilité des éléments de contenus est offerte par le principe même de description du langage : comme c'est un langage d'intégration, les éléments sont simplement référencés par le mécanisme standard d'URI. Ils peuvent donc être aisément partagés entre différentes présentations, voire avec d'autres applications. Il est également possible de réutiliser tout ou partie d'un document SMIL grâce à ses structures déclaratives hiérarchiques. Comme SMIL est un langage XML, toutes les techniques qui y sont issues sont exploitables : utilisation des espaces de noms pour intégrer des vocabulaires externes, mise en place de chaînes documentaires permettant la production de différents formats cible en utilisant des techniques de transformation avec XSLT.

La capacité d'un document à être joué dans différents contextes et sur différents supports est un enjeu majeur pour le déploiement des présentations multimédias. Par exemple, la même présentation doit pouvoir être jouée de façon satisfaisante même lorsque les conditions du réseau changent (les synchronisations doivent être maintenues, le format des contenus peut être changé sans nuire à la sémantique du document). De même, une présentation doit pouvoir être restituée dans différentes langues sans nécessiter de dupliquer les contenus et la structure qui ne dépendent pas de la langue. Le langage SMIL offre quelques éléments de réponse pour couvrir cet objectif d'adaptabilité :

1. Le standard ayant été conçu de façon modulaire, différents profils ont été définis, chacun correspondant à un contexte de restitution différent : le profil SMIL2.0 Basic est adapté au terminaux de faibles capacités comme les PDA tandis que le profil SMIL2.0 complet vise les PC.
2. De façon plus fine, il est possible d'utiliser des éléments syntaxiques SMIL pour permettre l'adaptation du document au moment de sa présentation. Le constructeur « switch » associé aux attributs de sélection comme « systemScreenSize » ou « systemBitrate » permet de combiner une structure et/ou un contenu partagé avec des fragments spécifiques d'un contexte. Le contexte peut être défini à l'aide d'attributs prédéfinis comme la taille de l'écran ou la langue, mais il peut être également défini de façon ad'hoc à l'application.
3. Un attribut de préchargement permet de demander l'envoi en avance du contenu pour assurer la bonne synchronisation entre éléments (comme le démarrage simultané d'une vidéo et un commentaire audio associé).

Pour être exploités de façon dynamique et flexible, ces mécanismes doivent être utilisés dans le contexte d'une architecture plus complexe que celle d'un simple client/serveur classique. Une étape de négociation entre le client et le serveur doit précéder l'envoi des éléments du document (source SMIL et éléments de contenu). Cette négociation, qui s'appuie sur des structures d'identification du contexte comme définies dans CC/PP, peut être mise en oeuvre au niveau du serveur ou par un proxy (Lemlouma *et al*, 2004).

Pour terminer cette section sur les possibilités offertes par le standard SMIL, nous listons ci-dessous quelques logiciels de présentation SMIL appelés aussi « players SMIL ». Ces outils de visualisation SMIL n'offrent pas tous le même service en termes de couverture du standard et d'interopérabilité. En effet, un des problèmes majeurs de SMIL est que si les logiciels prennent en compte de façon homogène tout ou partie des éléments de structuration de SMIL (placement dans les régions spatiales, synchronisation temporelle), ils offrent une grande variété de traitement des contenus (format des éléments et attributs de style). En particulier les éléments textuels sont traités de façon différente selon les logiciels : soit ce sont des éléments externes au source SMIL (au format standard html ou propriétaire comme RealText) soit ils peuvent être insérés dans le source SMIL.

<i>Nom du logiciel</i>	<i>Version/profil SMIL</i>	<i>lien</i>	<i>commentaires</i>
RealPlayer	SMIL1.0	http://www.real.com	Premier player SMIL
RealOne	SMIL2.0	http://www.real.com/realone/index.html	Logiciel le plus utilisé. Limitation dans les formats de contenu acceptés.
QuickTime	SMIL1.0	http://www.apple.com/quicktime	
GriNS	SMIL2.0	http://www.oratrix.com	Editeur et « player ». Le système n'est plus développé depuis 2002.

<i>Nom du logiciel</i>	<i>Version/profil SMIL</i>	<i>lien</i>	<i>commentaires</i>
Ambulant	SMIL2.0, SMIL 2.0Basic	http://www.cwi.nl/projects/Ambulant/distPlayer.html	Effort pour assurer une large couverture de formats.
Internet Explorer	xhtml+time (non standard)	http://www.microsoft.com/windows/ie/default.asp	Intégration de SMIL dans XHTML
Xsmiles	SMIL.2.0	Http://www.x-smiles.org/	Implémentation incomplète
PocketSMIL	SMIL 2.0Basic	http://wam.inrialpes.fr/software/pocketsmil/	Prototype
Helix	SMIL2.0	https://www.helixcommunity.org/	

Tableau 1. logiciels de présentation du langage SMIL

3.4. Evolutions et perspectives du langage SMIL

Malgré les limites évoquées ci-dessus, de nombreuses applications utilisent le format SMIL comme format cible de présentations multimédia. En particulier pour la mise en ligne de séminaires ou de cours. Ce langage se prête bien à la restitution synchronisée de la vidéo (ou de l'audio) de l'orateur synchronisée avec le défilement des transparents. De plus les fonctions d'hyperliens permettent au lecteur de parcourir la présentation directement à partir de la table des matières. Notons enfin que le site du W3C tient à jour tous les travaux autour de ce standard (voir <http://www.w3.org/AudioVideo/>).

D'autres domaines que celui de l'informatique « traditionnelle » et de l'internet s'intéressent à SMIL, en particulier la téléphonie mobile. Ainsi, 3GPP (Third Generation Partner Project), dans le cadre de la spécification PSS « Transparent end-to-end packet switched streaming service », a défini le langage 3GPP SMIL (3GPP 2003) pour mettre en oeuvre le système de messagerie multimédia « Multimedia Messaging System » MSS de la 3ème génération de téléphonie mobile. Ce langage est un profil SMIL qui reprend le profil SMIL Basic ainsi que les modules SMIL nécessaires pour l'accès et le contrôle des éléments média : MediaAccessibility, MediaDescription, MediaClipping, MetaInformation, PrefetchControl, EventTiming ainsi que le module BasicTransitions. Ce profil doit permettre aux utilisateurs de créer des messages multimédia (contenant non seulement de l'audio mais aussi de la vidéo, du texte, des graphiques, etc.) synchronisés en tenant compte des limites des terminaux mobiles (petits écrans, faible capacité de traitement, ressources limitées en mémoire).

Les travaux en cours du groupe SMIL portent justement sur des besoins issus de ce domaine ainsi que celui de la télévision numérique :

1. Une nouvelle version de SMIL « SMIL 2.1 » est prévue fin 2004 pour couvrir les besoins de l'industrie des mobiles en intégrant et complétant le « 3GPP

SMIL Language » pour les services de messagerie multimédia. Parmi les fonctions qui seront ajoutées, on peut citer celles concernant la spécification d'effets sur l'audio, l'extension des transitions à plusieurs objets ou encore la possibilité d'attacher des attributs de style aux objets de contenu.

2. Les applications de la télévision numérique comme les régies de montage de téont besoin d'accéder et de traiter les éléments média continus de façon plus riche et plus fine (Gonno, 2003). Les extensions envisagées par le groupe de travail W3C couvrent :
 - le traitement des contenus : définition et gestion des coupures selon différentes échelles, harmonisation de la gestion du contenu textuel ;
 - une meilleure définition des styles et de la présentation : placement spatial plus riche (centrage), enrichissement des effets visuels (déformations, couche alpha) et gestion du temps selon différentes échelles en fonction des besoins ;
 - et l'intégration des métadonnées de l'industrie audiovisuelle : MPEG-7, TV-Anytime, etc.

4. Besoins des auteurs

4.1. Introduction

Avec l'émergence des langages ci-dessus et en particulier de SMIL, les auteurs pourraient s'attendre à avoir à leur disposition des environnements d'édition adaptés à leurs besoins. En fait c'est loin d'être le cas et on pourrait même dire que c'est quasiment le désert dans ce domaine ! L'outil utilisé le plus souvent pour réaliser un document multimédia est un simple éditeur de texte ou éventuellement un éditeur XML ! Par exemple, il n'existe que peu ou pas d'environnement pour éditer des documents SMIL (à part *Grins* d'Oratrix (Oratrix 2002) et *LimSee* (Weck 2004) un logiciel développé à l'Inria).

Comment expliquer ce manque d'outils ? La première raison est que, jusqu'à présent, la réalisation de présentations multimédia est confiée à des spécialistes informaticiens qui savent se contenter d'environnements de programmation. Une autre raison provient de la difficulté à réaliser des environnements d'édition multimédia car il est nécessaire de donner à l'auteur une perception correcte de l'enchaînement temporel des éléments média de sa présentation. Il est nécessaire aussi de lui offrir des paradigmes d'édition simples pour spécifier la synchronisation et la navigation entre ces éléments.

Plus largement, si on analyse l'ensemble des besoins nécessaires pour réaliser des présentations multimédia, il faut considérer les objectifs (parfois contradictoires) suivants :

1. La variété des objets documentaires à éditer : les objets de contenu, les différentes structures du document (disposition spatiale, liens, synchronisation),

les feuilles de style et/ou de transformation et éventuellement le modèle même de document (DTD, XML Schemas). Le plus souvent ces différents objets nécessitent d'utiliser des outils séparés car il n'existe pas d'environnement de développement intégré.

2. Les services d'édition offerts qui doivent trouver le meilleur compromis entre simplicité et puissance.
3. La prise en compte de l'existant, car la création de documents est très souvent un processus itératif qui (re)exploite des réalisations disponibles. Dans le cadre des documents multimédia, il peut être utile de réutiliser les différents objets documentaires listés ci-dessus (éléments de contenu, scénario, structuration de document, feuilles de style). Bien évidemment, les limites à cette réutilisation sont nombreuses et la multiplicité des formats en est certainement la principale.

4.2. Fondements des techniques d'édition

Les outils d'édition multimédia s'appuient naturellement sur les techniques mises au point depuis longtemps pour l'édition de documents, et en particulier pour l'édition de documents structurés (Furuta *et al.*, 1988). On peut distinguer quatre grandes classes de fonctions dans ces systèmes :

1. Les techniques de visualisation : elles concernent les méthodes pour rendre perceptibles à l'auteur l'état de son document dans toutes ses composantes. Il s'agit d'une part de trouver les bons paradigmes pour projeter dans un espace spatial 2D (ou 3D) les structures du document (logiques, spatiales, temporelles et hypermédia) et d'autre part d'aider l'auteur à comprendre les dépendances entre ces structures.
2. Le modèle de présentation et le système de formatage : la qualité d'un outil d'édition dépend du modèle de placement (cf. section 2). Ainsi, avec un modèle de placement absolu, la mise à jour d'un document est plus fastidieuse qu'avec un modèle de placement relatif. Ce modèle, combiné avec la technique de formatage utilisée, doit assurer, après chaque opération d'édition, une mise à jour en temps réel des informations présentées à l'auteur. L'outil auteur doit également permettre de traiter avec la même efficacité de gros documents. Pour prendre en compte ce facteur d'échelle, des techniques incrémentales de formatage sont nécessaires.
3. Les modes d'édition : Le mode d'édition le plus répandu est le mode « wysiwyg » qui assure qu'à tout instant l'auteur voit sur son écran le document tel qu'il sera restitué pour les lecteurs (sur le papier ou sur l'écran). Ce mode n'est plus possible pour les systèmes documentaires dans lesquels d'une part plusieurs formats de restitution sont associés à un même document source, et d'autre part la nature temporelle de la présentation ne peut permettre une perception instantanée. D'autres techniques sont alors employées : l'édition à l'aide de canevas ou « template » prédéfinis, l'accès aux opérateurs du langage

par manipulation directe sur les éléments graphiques représentant les composants du document, et surtout une combinaison de ces moyens offerts au sein d'un même environnement.

4. Le maintien de la cohérence du document en cours d'édition : Cette fonction est nécessaire dès lors que l'instance du document en cours d'édition suit un modèle (DTD ou Schéma XML). Deux approches sont possibles : soit l'outil offre un mode d'édition libre, pendant lequel le document peut se trouver dans un état non conforme, puis l'auteur passe en phase de mise en conformité avec le risque d'avoir un long processus de débogage ; soit l'édition est toujours sous contrainte de conformité avec le modèle, ce qui implique de bloquer toute opération d'édition qui conduirait à un état du document non conforme.

Citons enfin le cas des architectures de production de documents à base de transformation qui posent d'autres problèmes à l'édition. En effet, cette étape intermédiaire entre les données source et la présentation rend plus difficile la perception pour l'auteur des conséquences des actions d'édition qu'il effectue. Faute d'outils adaptés, les utilisateurs séparent le plus souvent la phase d'édition des sources (par exemple avec un éditeur XML) de la phase d'édition des feuilles de transformation (avec un éditeur de transformation ou même un éditeur XML). Mais c'est un travail de programmation important qui prive les utilisateurs non informaticiens de la possibilité de produire eux-mêmes leurs documents.

Dans le reste de cette section, nous allons plus particulièrement étudier un de ces aspects seulement, mais c'est de loin le plus complexe du fait des problèmes de perception et d'exécution qu'il pose : il s'agit de l'édition des structures temporelles des documents multimédia. Nous illustrerons notre propos avec les outils d'édition de SMIL.

4.3. Edition des structures temporelles pour les documents multimédias

Comme on l'a vu, la dimension temporelle intrinsèque aux documents multimédias rend la tâche d'édition plus complexe. Nous avons classé ci-dessous les besoins d'« édition temporelle » en trois principaux axes et nous en montrons les solutions apportées.

Edition multi-structures

Un document multimédia est tout d'abord un objet informatique intégrant plusieurs structures. L'environnement d'édition doit donc permettre d'éditer ces différentes structures à travers des composants d'interface adaptés à chacune : boîtes graphiques hiérarchiques pour le spatial, graphe pour les liens, arbre pour la structure logique, ligne temporelle pour le temps. Les outils exploitent les possibilités du multifenêtrage pour offrir à l'auteur des « vues » de document qui sont autant de filtres sur la structure générale. Bien sûr le risque de « perdre » l'auteur par trop d'informations dispersées doit être limité par des services de synchronisation entre les vues et d'aide à la perception de l'interdépendance des

structures. Par exemple, la possibilité de simuler l'activation d'un lien dans n'importe laquelle des vues est une bonne aide pour comprendre les enchaînements définis dans une présentation. De même, l'intégration synchronisée avec les vues d'un « player » ou d'un simulateur de présentation évite de nombreux aller-retour entre l'outil d'édition et l'outil de présentation.

Perception de l'enchaînement temporel

C'est à travers une « vue temporelle » que l'outil d'édition apporte à l'auteur la perception du placement dans le temps des objets de contenus et de leurs conteneurs hiérarchiques (éléments par, seq, exc de SMIL par exemple). Cette approche est très répandue dans les outils d'édition de contenu audio et vidéo car elle permet d'effectuer des coupures, extractions, enchaînement ou même annotations sur une suite continue définie par la suite d'éléments de base (images) représentée sur la ligne de temps. Si une telle représentation 1D convient bien pour un flux linéaire, elle doit être enrichie pour prendre en compte la complexité des structures temporelles des documents multimédias : leur structure est hiérarchique et comporte également des relations temporelles qu'il faut pouvoir représenter et permettre d'éditer. Ainsi le principe de représentation est le suivant : chaque intervalle temporel associé à un élément de contenu est représenté par un trait dont la longueur est proportionnelle à la durée de l'élément et un ensemble de marqueurs graphiques permet de visualiser les dépendances entre intervalles (relation de démarrage et/ou de terminaison simultanée par exemple). Ce principe peut être complété en fonction du modèle sous-jacent par : des représentations d'emboîtement d'intervalles, des mécanismes de zoom, des représentations 3D pour la visualisation d'animations spatio-temporelles, etc.

Opérations d'édition sur les structures temporelles

Puisqu'il n'est pas envisageable d'offrir à l'auteur un mode d'édition « wysiwyg », les opérations d'édition sont proposées sous la forme de manipulations « directes » sur les objets graphiques associés aux éléments de la structure temporelle : déplacement (ou étirement) avec la souris d'un intervalle temporel pour réaliser un décalage (ou une extension) dans le temps, sélection d'intervalles et choix d'opérateurs de regroupement temporel (séquence ou parallèle) et de contraintes de synchronisation. Pour assurer que le document reste cohérent quelle que soit l'opération effectuée, il est nécessaire que seules les manipulations autorisées soient accessibles à l'auteur. Les cas simples sont facilement contrôlables par l'application, comme par exemple, bloquer le déplacement de l'intervalle représentant un objet sur le bord de l'intervalle de son container lorsque celui-ci a une durée imposée. Mais de nombreuses autres situations sont vite complexes, en particulier lorsqu'il y a de nombreuses dépendances entre éléments et lorsque la sous-spécification du scénario peut conduire à des ambiguïtés dans l'interprétation des intentions de l'auteur. Ces problèmes sont couverts en partie avec des langages

restreints et avec des techniques algorithmiques efficaces comme des solveurs de contraintes (Jourdan *et al.*, 2001).

4.4. Les outils d'édition pour SMIL

Si l'on considère maintenant les outils utilisés pour éditer des documents SMIL, on peut constater que les auteurs ont une grande diversité de méthodes mais cette diversité cache en fait le peu d'outils vraiment complets et satisfaisants. Le premier niveau d'outils disponibles offre le moyen d'éditer le code source SMIL avec plus ou moins de services avancés : contrôle de structure, visualisation améliorée de la structure et des attributs, formulaires d'édition. L'environnement peut assurer simplement la validité XML (document bien formé), la validité vis-à-vis de la DTD SMIL (cas des éditeurs XML comme Xerlin ou XmlSpy) ou peut s'appuyer sur la connaissance de la DTD pour offrir à l'auteur des menus et formulaires contextuels comme SMIL Gen de Real Networks. Certains outils généralistes comme GoLive CS d'Adobe ou Macromedia Flash intègrent des fonctions d'export vers des formats multimédia standard MMS, SMIL ou MPEG4. De même, Autometa RPXP est un générateur de présentations SMIL à partir de template qui produit des flux streamés enrichis avec des contenus SMIL (texte, liens) et adaptés aux spécificités des « players » actuels (RealPlayer, Quicktime).

Les deux environnements qui ont vraiment cherché à couvrir tous les besoins d'édition de SMIL évoqués plus haut sont GriNS Editor d'Oratrix et LimSee de l'Inria. Tous deux proposent des services d'édition riches à partir d'un ensemble de vues dont une vue temporelle. Le développement de GriNS Editor étant stoppé, nous décrivons ci-dessous l'éditeur LimSee.

La figure 3 ci-dessous (extraite du site de LimSee) illustre les différentes vues offertes par ce logiciel :

- la vue hiérarchique (en haut à gauche) qui peut être filtrée selon plusieurs critères ;
- la vue des attributs (en bas à gauche) ;
- la vue des régions spatiales (en haut au milieu) qui offre plusieurs modes d'affichage (avec ou sans le contenu des éléments médias) ;
- la vue temporelle (en bas au milieu) avec zoom et visualisation hiérarchique. Cette vue permet également de visualiser les portions de contenu d'éléments qui ne seront pas joués du fait des synchronisations définies par l'auteur (par exemple une vidéo sera coupée parce que son container englobant a une durée plus courte).

Les services d'édition accessibles grâce à ces vues sont complétés par des services complémentaires comme l'aspiration de présentations SMIL existantes avec un mode d'ouverture incrémentale des fichiers mal-formés, l'importation de

documents PowerPoint pour générer des « slideshow » en SMIL, et bien sûr, le contrôle de validité des documents en cours d'édition.

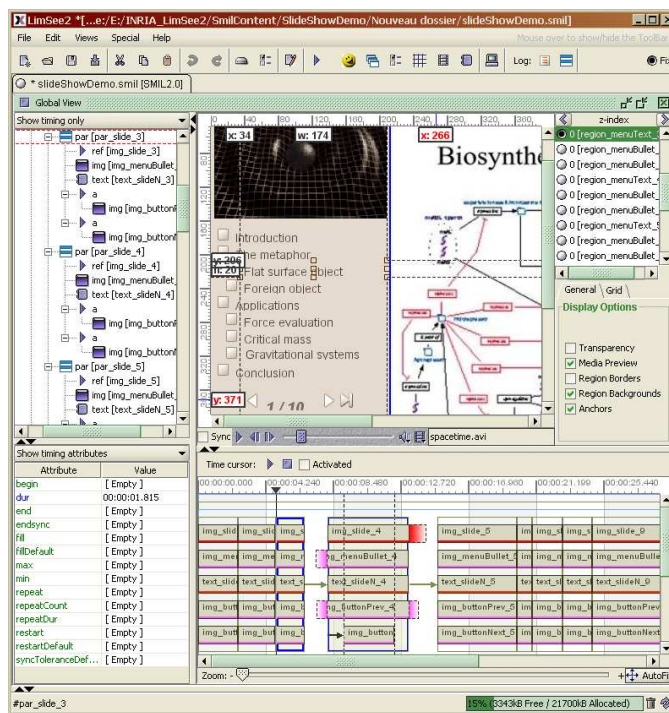


Figure 3. Les différentes vues de l' éditeur LimSee

L'auteur tient à remercier tous les membres des projets Opéra puis Wam qui ont contribué à développer les travaux de l'équipe sur les documents multimédias.

5. Bibliographie

- Blakowski G., Steinmetz R.A. Media Synchronisation Survey: Reference Model, Specification, and Case Studies. IEEE Journal Of Selected Areas In Communications, vol. 14, n° 1, janvier 1996, p.5-34.
- 3GPP. 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Transparent end-to-end packet switched streaming service (PSS); 3GPP SMIL Language Profile (Release 6), 650 Route des Lucioles - Sophia Antipolis Valbonne, <http://www.3gpp.org>, septembre 2003.

- Cover Pages, Contents Listing for XML Applications and Industry Initiatives
<http://xml.coverpages.org/xmlApplications.html>.
- Furuta R., Quint V., André J., Interactively Editing Structured Documents, *Electronic Publishing -- Origination, Dissemination and Design*, vol. 1, n° 1, avril 1988, p. 19-44.
- Gonno Y. SMIL 2.0 Extension for Professional Multimedia Authoring: Preliminary Investigation, SMIL Europe 2003 Conference, Paris, 12-14 février 2003.
- Inria, Les exposés scientifiques multimédia, <http://www.inria.fr/multimedia/Didactheque-fra.html>, 2004.
- Jourdan M., Roisin C., Tardif L., Constraints Techniques for Authoring Multimedia Documents, *Constraints Journal*, *Kluwer Academic Publishers*, vol. 6, n° 1, janvier 2001, p. 115-132.
- Lemlouma T., Layaïda N. Context-Aware Adaptation for Mobile Devices, IEEE International Conference on Mobile Data Management, Berkeley, 19-22 janvier 2004, p. 106-11.
- Nicoll S., Pirot V., Bonaventure O. eConf: a pragmatic tool to produce and distribute multimedia e-learning content, SMIL Europe 2003 Conference, Paris, 12-14 février 2003.
- MPEG Web Site, <http://www.csel.it/mpeg/standards.htm> for MPEG-1,2,4,7,21.
- Rowe L. A., Jain R., ACM SIGMM Retreat Report on Future Directions in Multimedia Research, www.acm.org/sigmm/main/events/sigmm_retreat/sigmm-retreat03-final.pdf, 4 mars 2004.
- Tran Thuong, Roisin C. A Multimedia Model Based on Structured Media and Sub-elements for Complex Multimedia Authoring And Presentation, *Int' l Journal of Software Engineering and Knowledge Engineering*, Special Issue on Image and Video Coding and Indexing, 2002.
- Wahl T., Rothermel K. Representing Time in Multimedia Systems. IEEE Proceedings of the International Conference on Multimedia Computing and Systems, IEEE Computer Society Press, Boston, Massachusetts, 14-19 mai 1994, p. 538-543.
- Weck D. LimSee2, A cross-platform SMIL2.0 authoring tool,
<http://wam.inrialpes.fr/software/limsee2/>, 2004