

Anticipating SMIL 2.0: The Developing Cooperative Infrastructure for Multimedia on the Web

Lloyd Rutledge, Jacco van Ossenbruggen, Lynda Hardman,
{Lloyd.Rutledge, Jacco.van.Ossenbruggen, Lynda.Hardman}@cwi.nl
CWI (Centrum voor Wiskunde en Informatica)
Kruislaan 413, 1098 SJ Amsterdam, Netherlands

Dick C.A. Bulterman
Dick.Bulterman@oratrix.com
Oratrix Development, B.V.
Valeriusplein 30, 1075 BJ Amsterdam, Netherlands

ABSTRACT

SMIL is the W3C recommendation for bringing synchronized multimedia to the Web. Version 1.0 of SMIL was accepted as a recommendation in June 1998. Work is expected to be soon underway for preparing the next version of SMIL, version 2.0. Issues that will need to be addressed in developing version 2.0 include not just adding new features but also establishing SMIL's relationship with various related existing and developing W3C efforts. In this paper we offer some suggestions for how to address these issues. Potential new constructs with additional features for SMIL 2.0 are presented. Other W3C efforts and their potential relationship with SMIL 2.0 are discussed. To provide a context for discussing these issues, this paper explores various approaches for integrating multimedia information with the World Wide Web. It focuses on the modeling issues on the document level and the consequences of the basic differences between text-oriented Web-pages and networked multimedia presentations.

Keywords: SMIL, hypermedia, multimedia, synchronization, linking

1. INTRODUCTION

From the early days of the Web, HTML has served as the Web's common, general purpose document language. Recently, both users and developers have recognized that a single document model can never meet all of the diverse requirements of the wide range of applications that make use of the Web's infrastructure. Subsequently, a plethora of special-purpose document languages has been developed. While most of these languages use XML as their common meta language, they are especially tailored for a specific range of applications. Examples include mathematical markup, chemical markup, and — the focus of this paper — synchronized multimedia.

For years the computer industry has had the joke "The nice thing about standards is that there are so many to choose from". This used to mean that for any one task there are multiple standards for performing it. More and more it is also coming to mean that for any one task there are several standards that are used together to perform it, adding a second dimension to the multiplicity. Over the past few years cooperative standards have emerged, and continue to emerge, that solve not individual problems but components of problems, and that solve them in concert with other standards. In order to operate in harmony these formats must develop in harmony — the

committees creating them must understand the development of related formats, on which their format will depend and which will depend on theirs.

Version 1.0 of SMIL (Synchronized Multimedia Integration Language) was accepted by W3C (The World Wide Web Consortium) as a recommendation in June 1998 [5]. It was developed by the SYMM (SYnchronized MultiMedia) Working Group of W3C. SMIL 1.0 consists of a declarative set of basic constructs for adaptive hypermedia presentations. SMIL 1.0 is intentionally basic, as it was decided that the concepts represented be readily understood and easily implemented. The requirement that each feature in the recommendation be implemented in at least two separate systems also limited the number of features included in the first version. These factors leave a number of features that are ripe for inclusion in future versions of SMIL.

Since the acceptance of SMIL as a W3C recommendation in June, several other related formats have become recommendations or working drafts, or are otherwise developing projects. Many of these formats can no longer be regarded as stand-alone, but are an integral part of the Web's cooperative document processing environment. The next version of SMIL will need to address its relation to these new formats as part of a larger distributed hypermedia environment.

In this paper we present our suggestions for some new SMIL constructs and for how SMIL could be incorporated into the developing multi-format Web processing infrastructure. Our suggestions for individual extensions to include in SMIL 2.0 are divided into several areas. First spatial layout constructs are discussed, along with their relationship with style specifications. Then hyperlinking components and related formats are described. Finally, we discuss time-related constructs, issues and languages.

2. SPATIAL LAYOUT AND STYLE

Defining where visual media objects appear on the screen is an important aspect of multimedia. The XML-related formats that specify layout currently do so mainly for text. However, processing spatial layout for multimedia involves different issues than for text. SMIL currently has its own internal format for specifying the basics of spatial layout. This section offers suggestions on how spatial layout specified for SMIL can be extended. Also discussed is SMIL's current and potential relationship with CSS for specifying spatial layout.

2.1 Relation between Spatial Layout and Lexical Flow for Text and for Multimedia

One important reason for developing a new document format specific to multimedia is that the needs for the multimedia spatial layout model are different than those for text-based documents. These differences are illustrated in Figure 1. For text-based documents such as HTML [11], the layout is based on one-dimensional text-flow which is displayed on a two-dimensional page. This transformation from one to two dimensions is typically rendered using wraparound and scrolling. Multimedia spatial layout, on the other hand, is inherently two dimensional, with little or no transformation from the document structure to the final display space, and typically without the use of scroll bars.

Furthermore, the relationship between layout and lexical flow for text is different than that for multimedia. The syntax of text formats such as HTML follows that of its one dimensional textual flow: what is defined first in an HTML document encoding typically appears at the top of the display, and what appears last in the encoding typically appears at the bottom. This matches the author's intuition of how the document progresses while it is being presented to the user.

For multimedia, on the other hand, the author's concept of how the document progresses during presentation is typically based on its timing, not on its spatial layout. Thus, the flow of the syntax of multimedia formats such as SMIL does not correspond directly with any aspect of the spatial layout but corresponds instead with the presentation's timeline. Multimedia documents orchestrate various media elements, arrange how these elements are to be synchronized, and on which portion of the screen they need to be displayed. The screen location associated with a media element is typically independent from the element's temporal, and thus lexical, position in the document.

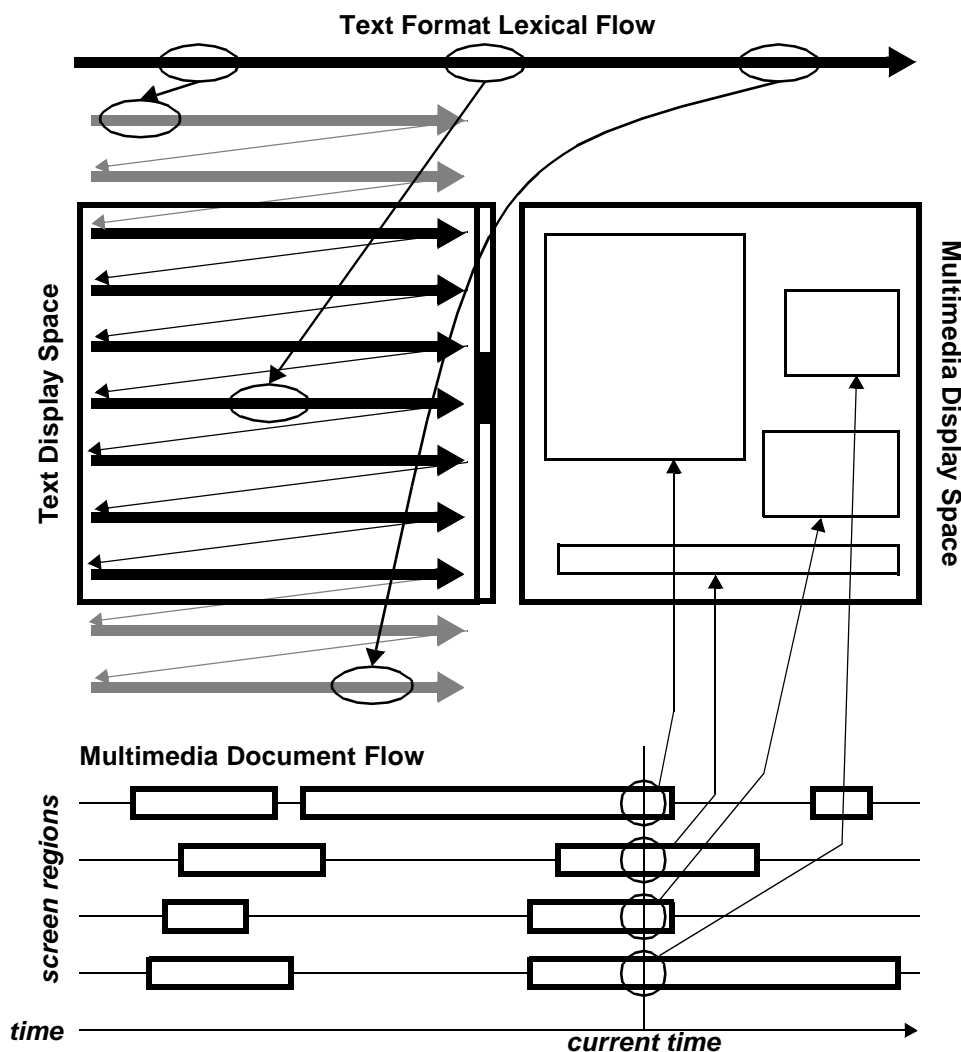


Figure 1: Document Flow and Spatial Layout for Text and for Multimedia

2.2 Relationship with CSS

CSS (Cascading Style Sheets) specify the appearance of HTML and XML documents [1]. This includes ways of assigning layout and style properties, such as font types and sizes, for different types of document components when they are presented. The output of processing a document with a CSS style sheet maintains the same basic lexical flow of the original document. CSS modifies the appearance of each object in this flow, and the appearance of global aspects of the document's presentation as a whole, but does not change the fundamental progression of this flow. This model works well within the realm of text documents because of text's correlation between lexical and spatial flow.

SMIL 1.0 already specifies a number of relationships with CSS2 [1]. First of all, since SMIL is a format that integrates objects of different media formats, CSS2 can apply to individual components of a SMIL presentation that are XML, such as HTML documents. Secondly, the SMIL 1.0 specification provides for the use of CSS2 as an alternative for SMIL's basic layout specification. It does not require browsers to be able to process such alternative layouts, and no browsers currently do so. However, it does set a foundation for upcoming SMIL browsers to use CSS2, and for upcoming versions of SMIL to extend upon this use of CSS2 as an alternative layout. Finally, the SMIL 1.0 rendering model for SMIL's basic layout was designed to be isomorphic to that of CSS2 [6]. This facilitates the use of existing CSS2 software for processing SMIL. It also eases future extended cooperation between the two formats.

The CSS2 facility that applies most to SMIL layout is *absolute positioning*. It determines that placement of document objects in terms of the *viewport*, which is typically the boundaries of the current browser window. Without absolute positioning, the positioning of objects on the screen is based on their location in the XML document lexical flow, as shown in Figure 1. Spatial layout in SMIL, as in most multimedia, is not derived from the lexical order which is temporally based. Therefore non-absolute positioning in CSS is not convenient for screen placement of objects in SMIL presentations.

2.3 Presenting SMIL without CSS

Those working in the computer field know it is possible for people to present themselves without style. Whether this possibility applies to documents is an issue discussed in this section. What is a document devoid of its style of presentation? Is there a default, "plain vanilla" means of presenting a document without style, or does a document without style have no manifestation in the presentable realm?

Key to discussion of this issue is the fact that distinguishing style from the rest of a document only provides a service if more than one style specification can be made for the document. If there is something about a document that remains consistent throughout all its presentations, then this is part of the core document and not an aspect of the style of its presentation. This can be used as a means of determining what are features of a document's style and what are intrinsic characteristics of the document. Once the core, intrinsic components of a document are determined, the question becomes: "Are these presentable in their own right?"

HTML is, for the most part, presentable without an accompanying style sheet, since for text-flow based layout there is a reasonable default which is incorporated within the browser. Early versions of HTML did not use CSS and many existing HTML documents do not use CSS. While more and more HTML documents are using CSS style sheets, they can still be processed in their absence. What gets presented in such cases is the browser's default for displaying the core components of the document: what its text is, what is the basic role of each text component, what its included images are, what its links are, and the order in which they appear.

A similar lack of dependence on style sheets exists with SMIL 1.0. For multimedia documents there exists no reasonable default spatial layout which can be incorporated in the browser or player since the lexical flow of the document is based on the temporal structure. In order to be presentable without a style sheet, a SMIL document needs to include the specification of the basics of its spatial layout. Another reason for not specifying spatial layout solely within a style sheet is that multimedia authors often convey information by a specific spatial arrangement of the media items that relate to each other. Authors consider this type of information to be a part of the core semantics of their document, semantics that typically should not be changed by a style sheet specification.

We feel that in SMIL 2.0, documents should remain basically presentable without needing external layout specifications in the style sheet. Style sheets, however, are needed to define typical style parameters such as background or anchor colors. Authors of SMIL should be able to readily and extensively use CSS to enhance the style of their multimedia documents, in the same manner that HTML authors currently enjoy. Presentations of SMIL without CSS may, and perhaps should, tend to be unappealing, lacking a lot of the polish that CSS could define. But they should be basically comprehensible to the user.

2.4 Extending the Ability to Specify Spatial Layout for SMIL

This section describes some features of spatial layout that we suggest for inclusion in the representational environment for SMIL 2.0. For a feature to be included in the representation environment for SMIL 2.0 means either that constructs representing it are added to SMIL itself or that constructs in a related format are used with SMIL to represent it. Having these constructs be in another format involves specifying how SMIL constructs work with these foreign constructs if such cooperative use is not already obvious from the format specification. It may also involved adding such constructs to the related format. This would occur if the constructs are more appropriately included in the other format than in SMIL, or if the constructs are useful not just for SMIL but also for other formats which use the related format with the new constructs.

For refining SMIL's basic spatial layout, CSS serves as such a related format. For the spatial layout features proposed below we discuss the possibilities for extending it both through SMIL 2.0 and through CSS. We also discuss the issues involved in deciding which format should encode the extension. Typically for each case the choice of which format to use depends on whether the represented semantic is an intrinsic component of the document or an aspect of the style of its presentation.

Intrinsic components of spatial layout should be represented as SMIL basic layout. Refinements of the basic layout should be encoded in CSS.

Some aspects of SMIL layout are appropriate for encoding in a separate style specification like CSS but have no equivalent construct in CSS2. This is often because CSS2 was specified in the context of text-based presentation and thus often does not address issues that are specific to multimedia. The facilities presented below serve not only to broaden the possibilities of SMIL processing but also to extend CSS from the domain of text presentation more fully into the domain of multimedia presentation.

Relative Positioning: In SMIL 1.0 the positions of all screen regions are specified with respect to a browser window. Authors often position visual objects in relation to each other. The addition of relative positioning of regions to SMIL would make it easier for authors to maintain these relationships as the presentation is modified. If the location of one region is changed, then the effective locations of the media objects placed in relation to it would also automatically change. The author would only need to change the location of the first region without explicitly modifying the locations of the related regions.

Similar constructs exist in SMIL 1.0 for relative timing. The starting and ending of document components can be defined as relative to that of other components. This provides a similar efficiency in the authoring of timing. Editing the timing of one object automatically effects the timing of related objects. Relative timing in SMIL 1.0 uses references to the ID of the media object a timing event is being related to. Relative positioning could use ID referencing in a similar fashion. The <region> element type of SMIL 1.0 basic layout already has an ID attribute, which allows it to be referred to by media objects in specifying their screen positioning. The attributes of <region> for positioning, top, left, width and height, could also use the IDs of regions in a similar manner to provide SMIL 2.0 basic layout with relative positioning.

We use the term *relative positioning* here as it is typically used in the context of multimedia. CSS2 has a facility called *relative positioning*, but this is used in the context of text. With CSS2 relative positioning, an object is placed relative to its default screen position, which in turn is determined by the object's location in the document's lexical flow. CSS2 absolute positioning provides the functional equivalent of SMIL regions. It also has the equivalent limitation: the CSS2 absolute positioning of an object can not be defined as relative to the absolute positioning of another object. In order for CSS to provide the same convenience for SMIL authors that the relative positioning of SMIL basic layout regions would, such a mechanism would need to be introduced. Since the term relative positioning in CSS2 is taken, we propose the term *absolute co-positioning* (to avoid the term "relative absolute positioning") for this potential new facility for future versions of CSS.

Centering: In SMIL 1.0, screen positions and areas for visual media object display are defined by <region> elements. Media objects, when defined in the SMIL body, refer to the regions in which they are displayed with a region attribute. It is possible that the intrinsic dimensions of the visual objects do not match the dimensions of the region space assigned to them. How to handle such situations is determined in SMIL 1.0 by

the fit attribute of the region element. The allowed values of this attribute describe different solutions, such as cropping, scaling and stretching. Specification of such behavior is included in SMIL 1.0 basic layout, primarily because in its absence it would be possible to have important parts of images be obscured during some presentations.

When fitting a visual object in a region with different dimensions, the object is placed against the top left corner of the region. For example, if an image is smaller than its region and the attribute value of hidden is used, then the image's smaller size is maintained and it is placed in the top left corner of the region, leaving a margin along the right and bottom between the image's edge and the region's border. Often in multimedia it would be more aesthetically pleasing to *center* the image in the region. SMIL 1.0 basic layout has no construct for specifying this.

It is tempting to conclude that a SMIL extension should provide this behavior. However, this behavior could be considered as style. The fit settings as they currently stand provide enough declarative power to make sure important parts of images remain not obscured. Furthermore, CSS2 has the ability to encode this behavior. The CSS2 center property can be applied to media object specifications to center them in the screen areas in which they are placed.

3. LINKING

Navigational behavior is another key aspect of hypermedia presentations. SMIL 1.0 specifies a basic set of navigation hyperlink semantics. SMIL hyperlink constructs are based on the HTML hyperlink constructs, both semantically and syntactically. The hypermedia research community has established a much wider collection of hyperlink navigational behavior, providing a rich set of potential extensions to SMIL. We discuss a few of these in this section. We begin by discussing XLink, a developing W3C format that could provide the definition of hyperlinks in SMIL.

3.1 XLink

XLink (XML Linking Language) is a format being developed for defining hyperlinks within XML documents [9]. It is intended to be able to represent the existing hyperlinks in HTML as well as the potentially more complex links of future XML document sets. SMIL 1.0 hyperlinking constructs were defined to be consistent with the working draft of XLink. As XLink progresses to a recommendation, it is expected to maintain its ability to represent hyperlinks in existing XML formats such as SMIL 1.0. Similarly, since it is a design goal of SMIL to work well in an XML-based environment, it is expected that future versions of SMIL will keep its hyperlinks defined in accordance with XLink as it develops into a W3C recommendation. In this section we discuss XLink, how SMIL 1.0 uses the XLink working draft, how SMIL 2.0 can use more of XLink, and what additional facilities we propose for inclusion in the XLink recommendation beyond those in the working draft.

XLink defines attributes that describe hyperlinking characteristics for the XML elements they are assigned to. XLink is not defined by a DTD (Document Type Definition) like HTML and SMIL are. Neither does XLink define element types nor a

document content model. Instead, multiple DTDs can be defined that use XLink constructs. XLink also provides DTD creators with more flexibility in tailoring the DTD to their specific needs of their application domain. Thus, both the HTML and SMIL DTDs, along with potentially many other DTDs, are capable of using the same XLink constructs. This syntactic technique is based on the specification of SGML architectures as defined by HyTime [7].

The link constructs that SMIL shares with HTML are representable with XLink. Both SMIL and HTML have the `a` element type for specifying hyperlinks. This element type name is not determined by XLink; SMIL's use of HTML constructs was decided upon to keep SMIL as similar to HTML as possible. The `href` attribute of this element in both formats is, on the other hand, specifically defined in XLink. In both formats, as in any XLink-conforming format, this attribute refers to destination of the link that the element defines.

3.2 Link Scope

Each SMIL 1.0 document defines a single temporal flow. Under ideal playback circumstances and without user interaction it can be determined what exactly the screen display and audio components should be at any given point along the timeline. SMIL 1.0 hyperlinks enable the user to jump back and forth along the timeline, but no matter how the user navigates to a point in the timeline, it is specified by SMIL as appearing the same.

One consequence of this is that it is not possible to cleanly specify optional material that is only displayed if the user requests it. This is a common characteristic of documents with hyperlinks. SMIL 1.0 documents exist on the Web that encode this behavior by specifying the optional portions as members of a sequence that are each infinite in length and have hyperlinks between them [8]. It would be better to have constructs in SMIL that explicitly encode this behavior.

Another desirable feature of hypermedia link-based navigation that we have presented in earlier work is *linking in context* [2][4]. This is the maintaining of one part of a presentation while another part is changed due to hyperlink activation by the user. This makes the author's task easier, makes documents more easily maintained and prevents unnecessary reuse of information. Similar behavior occurs in framed HTML documents, though frames lack many of the features enabled by linking in context.

However, hypermedia linking in context has issues that hypertext does not. One is that within a hypermedia presentation, multiple media streams can be active at the same time. During hyperlink navigation, those streams that are affected by link traversal are part of the *source context* of the link. All other streams should continue along their timeline while the part of the target presentation (the *destination context*) starts up. With this model, there are parallel timelines whose synchronization is determined by when the user activates the hyperlink. This results in different syntactic needs than hypertext has.

A solution provided for the easy specification of linking in context is the *choice node* [2][4]. We propose the inclusion of the choice node in SMIL 2.0. This would be a composite that can be used in the same places as the current par (parallel), seq (sequence) and switch composite elements. With this choice composite, the child of the composite that is to be considered active at a point in the presentation is the child that was most recently the endpoint of a hyperlink activated by the user. This has been demonstrated in the CMIFed hypermedia environment to be an effective authoring and representational construct [4].

SMIL's <switch> element allows inclusion of alternative media objects, of which one is chosen by the user's browser at runtime based on system and personal preferences—the user is not given the option of navigating among the children of the switch. While the choice element may seem similar to the switch element, its semantics are rather different. While a switch does not change the linear structure of the presentation, the choice composite allows for truly non-linear, but synchronized hypermedia presentations. The differences between the switch and choice elements are discussed in more detail in [3].

Since the choice construct is useful for other XML documents as well, a way of allowing reuse is to define it within the XLink specification, not within SMIL itself. In this way, text-based formats such as HTML can also use the choice construct to include material (e.g. footnotes) that is not presented as part of the main text flow, but only after explicit selection by means of link traversal.

4. TIME

Along with spatial layout and navigation hyperlinks, timing is typically considered a primary feature of hypermedia. However, although the existing infrastructure for hypertext on the Web provides useful constructs for layout and linking, it provides very little for specifying the timing of presentations because timing is typically not a feature of text documents. Because timing is not extensively specified in Web formats, it is a focus of SMIL, and SMIL 1.0 has a rich collection of timing constructs. As such, we do not propose new timing constructs here. Instead, we discuss the potential use of SMIL timing constructs by other formats. As an example, this section discusses the use of SMIL timing in HTML, but most of the issues discussed are also applicable to formats other than HTML. We compare three different approaches to the integration of time into HTML based on extending HTML, CSS and SMIL, respectively.

4.1 HTML-based Integration

HTML and its browsers do not provide support for defining temporal relations as required by multimedia presentations. An obvious approach to integration of multimedia and HTML is therefore extending HTML so that the timing of the various HTML elements can be defined within the HTML document itself. HTML+TIME takes this approach [12].

An HTML-based integration could potentially leverage existing standards and tools supporting HTML authoring and display, which has the following advantages:

- authors can apply the same authoring model as they are currently using;
- users can keep using the same browser-based interface;
- integration of other Web-based formats and techniques, such as style sheets and scripting, are relatively easy.

In practice, these advantages are hard to achieve, especially for documents with complex synchronization requirements. Even for documents with relatively simple timing requirements, this approach depends on a non-trivial extension of HTML. Additionally, by controlling the temporal aspects of the presentation from the HTML document itself, it breaks the separation between document structure and presentation.

4.2 CSS-based Integration

Several of the disadvantages of the approach described in the previous section can be overcome by moving the specification of the timing the presentation of the HTML elements from the HTML document to an associated stylesheet, such as one encoded in CSS. This is the approach taken by the Broadcast HTML (BHTML) proposal [13]. The use of BHTML has the advantage that it keeps HTML independent of all problems related to scheduling and synchronization. Additionally, from the perspective of separating structure from presentation, it is more appropriate to locate functionality controlling the presentation in the stylesheet.

Time extensions require the addition of a temporal dimension to the spatial dimensions of the rendering model of current style languages such as CSS. New style properties need to be introduced that attach timing information to HTML elements. Preferably, these properties provide for absolute and relative timing, and temporal composition mechanisms. For example, BHTML introduces several new properties based on the SMIL timing model to CSS. Consequently, this approach has a disadvantage similar to the HTML-based approach described, only here it is not a matter of overloading HTML, but overloading CSS.

Both the extensions based on HTML and CSS make use of HTML's hierarchical document structure. The most fundamental disadvantages of these extensions relate to the underlying text-flow document format of HTML versus the intrinsic time-based document format of multimedia, as illustrated in Figure 1. HTML is designed for basic hypertext pages — not for modeling multimedia presentations. While HTML's document structure may allow relatively simple timing constraints to be added, in general the timing relationships of multimedia presentations cannot be easily added to a text-flow based document structure. Only in cases where the temporal ordering and the text-flow based ordering are similar, as is the case in, for example, delayed display of bulleted lists, a simple extension may be practical. In the more general case, the text-flow based document structure will not be suited for attaching timing semantics. In BHTML, for example, there is, even for simple examples, often a need to introduce spurious div tags to provide a grouping mechanism that suits the temporal relations within the document. In more complex

examples, the timing and text-flow hierarchies are often orthogonal and hard to describe within a text-based document format.

4.3 SMIL-based Integration

Within SMIL, HTML content can be integrated within a SMIL presentation in the same way images or video fragments and other media are integrated. SMIL can, for example, be used to define the synchronized display of a number of individual HTML pages by specifying the timing and screen location of each HTML page. The hierarchical structure of such a SMIL document would reflect the temporal structure of the overall presentation instead of the textual flow of the various HTML pages. Within the current version of SMIL, it is not, however, possible to control the timing of individual elements of an HTML page. SMIL only works with a two dimensional layout and does not handle the one dimensional wraparound layout that HTML browsers work with. In the following, we present a possible solution which combines temporal schedules defined by SMIL with text-flow based layout as used in HTML.

This solution is based on introducing a new type of region specifically for text-flow based documents, such as HTML. Figure 2 shows a SMIL document on the left, which is used to control the timing of a number of HTML document elements. The SMIL document employs a new type of region element, called `linearregion` in the figure, in which only HTML content is shown. This region is divided into linear sub-regions (called `sublin` in the figure) that are referred to using IDs in the same way SMIL 1.0 regions are. These sub-regions have implicit spatial placement, which is dynamically derived from the HTML content as it is displayed by the browser. The display order of the sub-regions is determined by the lexical ordering of their definitions within the SMIL document header.

```
<smil>
  <head>
    <layout>
      <linearregion>
        <sublin id='s1' />
        <sublin id='s2' />
        <sublin id='s3' />
      </linearregion>
    </layout>
  </head>

  <body>
    <par>
      <text src='doc1.html#p3' region='s1' />
      <seq>
        <text src='doc1.html#p2a' region='s2' dur='5s' />
        <text src='doc1.html#p2b' region='s2' dur='5s' />
      </seq>
      <text src='doc1.html#p1' region='s3' />
    </par>
  </body>
</smil>
```

This is the third sentence in the source HTML document. Initially, the text looks like this, with the second sentence coming from paragraph 2a. The paragraph is concluded with a sentence which is the first sentence in the

This is the third sentence in the source HTML document. Later, the text looks like this, where the second sentence is taken from paragraph 2b in the same document. The paragraph is concluded with a sentence which is the first sentence in the

Figure 2: Linear subregions in text-flow based documents.

In the SMIL body, portions of an HTML document are included in the temporal composition hierarchy just like any other media type, as shown in the text elements in the figure. Each element is assigned a sub-region in which to appear for the duration specified by the SMIL temporal hierarchy. To determine the content of the virtual HTML document which is displayed by the browser at a given moment, the contents of the linear region's sub-regions should be concatenated. The author is responsible for ensuring that, at all times, the combined fragments form a valid HTML document so that it can be displayed by a browser. The virtual document displayed in the linear region is dynamically updated every time a new fragment is scheduled in one of its sub-regions. In the figure, the second sentence of the initial document is replaced by scheduling another HTML fragment in the same sub-region. Sub-regions with no HTML content are simply ignored during rendering.

The advantage of a solution based on regions, such as that presented above, is that it allows all SMIL timing functionality to be applied to HTML content. It also shows how integrating HTML into SMIL can be done in the manner SMIL has for laying out other types of included media. Furthermore, it allows the HTML to be shown on the screen with other media, including other HTML rendered in a separate area on the screen. The disadvantage of this proposal is that the lexical flow maintains SMIL's temporal-only nature and does not provide the linear text-based lexical flow that authors of more text-based documents may want.

Another disadvantage of this solution is that the elements in the HTML file need to be individually addressable. In the figure, we assumed the various sentences were addressable by using a name or id attribute. If the HTML source document did not define these attributes, the HTML document needs to be modified, or a more powerful addressing mechanism, such as XPointer [10], is required.

5. SUMMARY AND CONCLUSION

To be able to fully exploit the Web's hypermedia potential, interoperability among its various standards and protocols is required. Support for synchronized multimedia should be an integral part of the Web's basic infrastructure. This paper explores several issues in multimedia integration by discussing the relationships between SMIL and other relevant Web formats, such as HTML, CSS, XLink and XPointer. It explains the differences between text-based and synchronized multimedia documents, especially in the areas of spatial layout and style, hyperlinking, and temporal behavior. Additionally, it contains several suggestions for the next version of SMIL. It is our hope that these suggestions stimulate discussion in the Web community of how best to guide the development of SMIL and other formats into a better infrastructure for Web-based multimedia.

ACKNOWLEDGEMENTS

The ideas discussed in the paper come from many contributions by many people. These ideas include concepts that have been explored and understood within the hypertext and multimedia research communities for quite some time — we merely present here some suggestions for how these concepts can be represented in the syntax of SMIL 2.0 and related formats.

REFERENCES

- [1] Bos, B., Lie, H.W., Lilley, C. and Jacobs, I. (ed.). *Cascading Style Sheets, level 2: CSS2 Specification*, World Wide Web Consortium Recommendation. May 1998. <http://www.w3.org/TR/REC-CSS2/>.
- [2] Hardman, L. *Modelling and Authoring Hypermedia Documents*, Dissertation for Doctor of Philosophy, University of Amsterdam, 1998. <http://www.cwi.nl/~lynda/thesis>
- [3] Hardman, L., van Ossenbruggen, J., Rutledge, L. and Bulterman, D.C.A. "Do You Have the Time? Composition and Linking in Time-based Hypermedia", *Proceedings of ACM Hypertext 99*, Darmstadt, Germany, February, 1999.
- [4] Hardman, L., Bulterman, D.C.A., and van Rossum, G. "The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model". *Communications of the ACM*. Vol. 37, No. 2, February 1994.
- [5] Hoschka, P. (ed.). *Synchronized Multimedia Integration Language*, World Wide Web Consortium Recommendation. June 1998. <http://www.w3.org/TR/REC-smil/>.
- [6] Hoschka, P. and Lilley, C., *Displaying SMIL Basic Layout with a CSS2 Rendering Engine*. Note submitted to the Wide Web Consortium. July 1998. <http://www.w3.org/TR/NOTE-CSS-smil/>. (note for discussion)
- [7] International Standards Organization. *Hypermedia/Time-based Structuring Language (HyTime), Second Edition*, ISO/IEC IS 10744:1997. International Standards Organization, 1997.
- [8] Kuzmanovic, M., and Rutledge, L. *Fiets: A Tour of Historic Amsterdam Buildings*, <http://www.cwi.nl/SMIL/fiets.html>.
- [9] Maler, E. and DeRose, S. (ed.). *XML Linking Language (XLink)*, World Wide Web Consortium Working Draft. March 1998. <http://www.w3.org/TR/WD-xlink>. (work in progress)
- [10] Maler, E. and DeRose, S. (ed.). *XML Pointer Language (XPointer)*, World Wide Web Consortium Working Draft. March 1998. <http://www.w3.org/TR/WD-xptr>. (work in progress)
- [11] Raggett, D., Le Hors, A. and Jacobs, I. (ed.). *HTML 4.0 Specification (revised)*, World Wide Web Consortium Recommendation. April 1998. <http://www.w3.org/TR/REC-html40/>.
- [12] Schmidt, P., Yu, J. and Santangeli, P. *Timed Interactive Multimedia Extensions for HTML (HTML+TIME): Extending SMIL into the Web Browser*, Note submitted to the Wide Web Consortium. September 1998. <http://www.w3.org/TR/NOTE-HTMLplusTIME/>. (note for discussion)
- [13] Wugofski, T. *A Modular Hypertext Markup Language for Broadcast Applications*, October 1998. http://toocan.philabs.research.philips.com/misc/atsc/bhtml/bhtml_4.html.

VITAE

Lloyd Rutledge is a Post-Doc at CWI specializing in the study of hypermedia architectures. He holds a Ph.D. in computerscience from the University of Massachusetts in Lowell (1996) and an undergraduate degree in computer science from the University of Massachusetts in Amherst. He has published widely on the topic of structured multimedia systems, with a special interest in HyTime and MHEG document architectures. At CWI, he has a particular interest in the support of adaptive projections of applications in multiple document formats.

Jacco van Ossenbruggen has worked for four years as a PhD student at the Vrije Universiteit in Amsterdam, where his research topics included object-orientation and the (formal) modeling of hypermedia documents. He is currently working at CWI's Multimedia and Human-Computer Interaction group, where his interests include synchronized multimedia on the WWW and the automatic generation of user-tailored hypermedia presentations.

Lynda Hardman is head of the Multimedia and Human-Computer Interaction group at CWI in Amsterdam, where she has worked since 1991. Prior to joining CWI, she held research and development positions at a variety of universities and companies in the United Kingdom, most recently at Office Workstations, Ltd. (OWL), where she became interested in Hypertext architectures. She holds a Ph.D. from the University of Amsterdam (1998) and an undergraduate degree from Glasgow University in Scotland. She has been involved in several national and European research projects.

Dick Bulterman is President/CTO of Oratrix Development BV, which produces and markets authoring tools for creating and maintaining complex Internet-based multimedia presentations. He founded and, for several years, headed the Multimedia and Human-Computer Interaction group at CWI. Prior to joining CWI, he was assistant professor of engineering at Brown University in Providence, RI. He holds Masters and Ph.D. degrees in computer science (1977, 1981) from Brown and a degree in political economics from Hope College (1973). In addition to CWI and Brown, he has held visiting positions at Delft, Utrecht and Leiden Universities in Holland, and has consulted frequently on operating systems and multimedia architectures. He has been involved in several European Community projects.