Dialectically sharing knowledge everywhere, every time and, above all, very fast – as it is nowadays possible - has revealed the existence and importance of a common, distributed intelligence growing with the contribution of many individuals (a community). The iterating refinement of any information resource by a wider and wider community of sharers leads to undoubted advantages on the final quality and effectiveness of use of this information. Following this path and focusing on the e-learning field, we present a novel system, termed LAUGH, for the cooperative creation and sharing of SMIL-based multimedia resources. Such a system allows users (typically learners and teachers) to enrich the didactical material made available to the learning community. Specifically, they can add captions/subtitles and alternative annotations to the original multimedia contents by resorting to a Wiki-type interface. This open process promotes students’ participation, data decentralization, assemblage from diverse sources, sharing of knowledge as well as an improvement to the efficacy of e-learning materials. Throughout the paper, we will meticulously show that LAUGH extends and improves the functionalities provided by the SMIL standard. This motivates the title of this article.
1. Introduction

The adoption of e-learning systems is nowadays a worldwide trend. Effective and consolidated online learning tools are available which offer a wide range of activities and resources (formally named as learning objects), organized to form lessons and modules, along with quizzes, tests and means for discussion [14]. WebCT, Blackboard, and Desire2Learn are examples of companies which provide e-learning systems, used in many universities and colleges [8, 12, 40]. In order to deliver the didactical material and promote discussions, these online tools exploit collaborative features such as e-mailing lists, newsgroups, whiteboards, bulletin boards, chats and online presentation tools [16]. Also technologies originally thought for entertainment purposes are now widely exploited for education, such as, for example, games, podcasting and video sharing [37, 39, 26]. The Stanford University, for instance, has teamed up with Apple to create the Stanford iTunes University [37]. This service offers learning contents that students can subscribe to, by using the well known Apple's iTunes application. A similar exemplar to distribute e-learning services is offered by MGSPodcast [25].

Following the Web revolution into its second wave, tools for online education are evolving into new technologies, often referred as e-learning 2.0 [14, 22]. In practice, new solutions are devised for the distribution of learning objects. According to the philosophy of open participation, modern e-learning systems allow data decentralization, collecting from diverse sources and freedom to share and re-use [30]. This new kind of software for education only needs to be equipped with a Web browser. Indeed, Web 2.0 shifts the process responsibility on the server side, according to its service oriented nature [15, 28]. From the client viewpoint, cooperation is the key to success to Web 2.0-based educational systems [14, 21, 23, 31]. Indeed, the wide diffusion of Wikis and blogs have demonstrated that if users are provided with easy-to-use editing mechanisms, they cooperate with great commitment to produce new contents and share knowledge.

With our work, we have tried to join the power of users’ cooperation, as envisaged in e-learning 2.0 concepts, together with the features offered by multimedia technologies and rich media contents. These media types (e.g., text, audio, video, images) may be collaboratively manipulated, synchronized and shared to present didactical contents in complex and varying forms, which can improve the clarity of the presentation and allow its personalization and adaptation [33, 34].

To this aim, we have developed a new collaborative annotation system for manipulating e-learning multimedia contents [16]. We named our system LAUGH (LABelling Unbound Grained Hypermedia). It utilizes (SCORM compliant [3]) learning contents which are structured as SMIL documents [42]. On LAUGH, users cooperate by actively adding textual information (such as captions and annotations) to SMIL multimedia lectures, thanks to a Wiki-like editing interface. Captions/subtitles can be associated to media objects (e.g., an audio stream) composing the presentation; a simple Wiki syntax allows to specify the temporal starting point of each caption and its duration and spatial
Why LAUGHing is Better Than SMILing

Further, also textual annotations can be added which can be thought of as alternative (or additional) descriptions summarizing the contents of a multimedia objects they refer to. Specifically, annotations are embedded in one or more separate HTML documents to be accompanied with the SMIL presentation [16].

The main advantage provided by such a system is given by the possibility of improving multimedia document accessibility by adding captions and alternative annotations [6, 7, 17, 35]. Moreover, having different versions of a given resource, encoded as different media types, allows a proper customization of the learning objects, thus opening the way for a more inclusive didactical approach. To manage the delivery of those different document versions, LAUGH is equipped with a dynamic adaptation module which is able to automatically transcode rich media contents based on the user profile and request [34]. Thus, for example, if the user is a hard of hearing one, the lecturer’s speech (stored as an audio file) may be automatically replaced by textual captions/subtitles (provided by some collaborative user). Also, a blind user can benefit from the presence of textual annotations for videos and images, when some assistive technology (e.g., Braille display, screen reader) is exploited. Finally, even mobile users can enjoy presentations using LAUGH, as they can receive data which are adapted to be played out on their handheld devices (e.g., automatically resized video files, multimedia files recoded based on available codecs, heavy-weight multimedia resources replaced with lighter ones).

It is important to mention that LAUGH follows the Web 2.0 philosophy, as it meets the following requirements. It:

i) is a service oriented system with a client-server architecture and an easy-to-use interface,

ii) enables open participation, since users are free to cooperate for the production of rich media didactical contents,

iii) allows data decentralization, since resources composing the SMIL presentation can be distributed over the network,

iv) lets the users to gather heterogeneous data, coming from different sources, and

v) facilitates knowledge sharing, due to its intrinsic collaborative nature.

There is a final issue to be discussed before concluding this section. It is as follows: anyone interested in using LAUGH should be enabled to retrieve a specific content under the form that better matches his/her own profile. To this aim, searching/adaptation methods are needed within LAUGH; they have to consider the content both in terms of its semantics (e.g., the subject matter taught in the lecture) and in terms of the media elements composing the corresponding rich media lecture. To facilitate searching features, LAUGH employs a resource classification method based on the ACCMD standard [19]. In essence, an ACCMD profile for a given resource exploits metadata and relative tags. A new problem arises here, unfortunately. As a lecture becomes richer in terms of added information and possible presentation alternatives, the more complex and branched the ACCMD profile arises accordingly. This is a good news for the user. Yet, the corresponding computational load to manage this profile increases. An extensive
experimentation we have carried out has shown that this computational load increases with a number of possible alternatives linked to a given resource. Hence, smart methods should be devised to keep under control this phenomenon [6]. This is still an open issue that demands further investigation.
To conclude, it is worth mentioning that LAUGH extends and improves the functionalities provided by the SMIL standard. This motivates the title of the article. The remainder of this paper is organized as follows. Section 2 discusses on the importance of having mechanisms for adding annotations and captions to rich media objects. The analysis goes through collaborative Wiki systems, rich media contents and systems to manage these contents, such as SMIL. Section 3 introduces the features of LAUGH and describes the Wiki-like syntax defined to allow users to add captions and annotations to a given multimedia lecture. In that section, we further present LAUGH from an architectural point of view. Section 4 reports on the results we have obtained to measure the impact of an increasing ACCM D on the performances of LAUGH. Finally, Section 5 concludes the work.

2. Background and Related Work

This section provides a discussion on the importance of having mechanisms able to cooperatively edit online rich didactical materials. In particular, we first survey on e-learning standards and Web 2.0 / multimedia technologies (Sections 2.1 – 2.3). Then, the focus will go on SMIL, the language we exploit in our system to manipulate multimedia presentations. Specifically, we will discuss in Section 2.4 how a “standard” SMIL presentation can be enriched with alternative and additional contents.

2.1. E-learning standards

Several interoperability standards have been developed by international organizations for e-learning. The main goal of the e-learning standards is to define metadata, data structures, communication protocols and interfaces that will make learning content work across different platforms, by providing specific guidelines to be used throughout the design, development and delivery of such contents. One of the more widely used e-learning standard is SCORM (Shareable Content Object Reference Model) [5] which has been defined by the ADL (Advanced Distributed Learning) initiative [1]. SCORM includes a de-facto method for defining a SCO (Sharable Content Object), i.e., a learning resource that can be presented in any SCORM compliant system, thought to track student progress. Each SCO is comprised of one or more assets (or resources), which are electronic representations of media (e.g. text, images, sound, video), Web pages or other types of data.
Another standard is the AccessForAll Meta-data (ACCMD) [19] developed by the IMS (Instructional Management System) Global Learning Consortium [18]. ACCMD allows one to describe which kind of content is being considered and whether there is an equivalent or alternative form for that content. Besides, ACCMD provides support to functional interoperability, i.e., any resource can be substituted or coupled with
alternative ones. To this aim, each media resource has an associated description of a set of additional resources, which are semantically equivalent to the primary one, but encoded according to different media formats.

2.2. Collaborative Wikis

As of today, Wiki engines are certainly the most widely utilized Web content management systems and represent one of the key technologies for enabling user collaboration on the Web. These systems embrace an approach ("the Wiki way") where contents are collaboratively edited by a multitude of users [11, 30].

In Wikis, a specific syntax must be employed to edit a new content; the source format that is produced to create and organize textual contents is called Wikitext. It is a simple markup language which exploits plain text with a few simple conventions for creating links and for giving some structure and/or style to the edited contents [16]. Then, the inserted source code is automatically converted into a final HTML document [11].

Each Wiki system has its own syntax, grammar, structure and keywords. No widely recognized standard has been provided yet. Hence, for instance, different Wiki engines typically have different syntax conventions to specify links.

While the very first Wiki systems were only able to produce spare HTML pages with fixed structure and only some simple graphic styling (e.g., bold, italic, acronyms), more recent versions add support for a more complex editing, which allows, for example, inserting text decorations, tables, images and formulas. For instance, MediaWiki, the software exploited in Wikipedia, uses a markup language able to produce text decorations which can be equivalently obtained through common HTML tags. OpenWiki allows to write mathematical formulas based on MathML [29]. In TWiki, users can include Latex commands [38]. The SnipSnap Wiki engine enables to write organigrams and diagrams [20].

To develop our system, we used DokuWiki, as it is a quite stable project, available under GNU General Public Licence [13].

2.3. Annotating Rich Media Content

The final result obtained from the use of a Wiki system is a Web page. As to rich media contents, no system has been provided yet, which allows a cooperative editing of complex multimedia contents, nor any way to enrich these contents with some additional information.

Rich media contents are composed of different media, like video, audio, text animations and other graphics. Alternative contents, such as captions/subtitles, may be associated to primary media types (e.g., audio or video). Hence, a collaborative editing system for such kinds of contents could represent a step forward along the path toward the so called “connected intelligence” [9, 23]. Furthermore, it would open up a wide range of new opportunities for the learning community. Think, for instance, about foreign students that deliberately add captions or notes to the didactical material with translations from the original language into their mother tongue. Also, think of students or experts who add
notes, with new schemes and images to give a lecture, thus helping the lecturer to better explain a particularly striking concept. Not to mention the possibility of adding news and updated information to have contents always up-to-date. Finally, think of the tremendous importance that these additional (mainly text-based) contents may have on the level of accessibility to users with disabilities.

Following the Wiki way, we have devised an easy-to-use system (discussed in Section 3) that allows one to add new contents to multimedia lectures so as to enrich the didactical material. This way, each student would be able to add and synchronize his/her own provided resources (e.g., audio, video, text, images) with other ones already available in the system.

In our work, we have put the focus on textual captions and annotations encoded as Web pages, as may represent alternative resources to primary contents.

Besides LAUGH, other captioning tools already exist which allow to associate captions/subtitles to media contents. We survey them below in isolation, for the sake of completeness of our presentation. Yet, is should be clear that none of them presents the advantage provided by LAUGH, i.e., they are not collaborative at all.

MAGpie 2 is an exemplar of a tool devised to create closed captions and audio/video descriptions for QuickTime, Real, Windows Media Player, or Flash presentations [24]. Similarly, Microsoft's Synchronized Accessible Media Interchange (SAMI) is an XML-based text language which allows to add captions to Windows Media, QuickTime and RealPlayer video contents [36]. To this aim, a SAMI file must be created which contains the captions and their how-and-when description. Finally, other exemplars exist such as the Described and Captioned Media Program (DCMP), which has been developed with accessibility issues in mind [10].

Apart from these proprietary proposals, a key idea to manage rich media contents amounts to exploit open standards as MPEG21 and SMIL [27, 42]. Developed by the Moving Picture Expert Group, MPEG-21 is the newest standard for multimedia delivery and consumption [27]. Each content (or digital item, in the MPEG jargon) is considered as a combination of resources, metadata, and structure. Resources are the individual assets or the real content being played out. The metadata describes the digital item and its individual resources. The structure relates to the relationships among the parts (i.e., resources and metadata) of the digital item. As it can be seen in most MPEG video streams being broadcast by TV channels, encoded in DVDs or made available online, captions can be easily activated by a user. As LAUGH has been developed based on SMIL, we devote the following section to discuss this standard.

2.4. Video lectures with SMIL

SMIL is a W3C XML-based markup language which is employed to create synchronized multimedia presentations [42]. SMIL allows to display heterogeneous media resources like text, video, and audio, and defines tags for managing timing specifications, layout, animations, visual transitions, media embedding and alternative presentations for each possible content. In the following, we provide a brief survey on SMIL. Indeed, reminding
how SMIL works may help the reader to better understand the technical innovations introduced by LAUGH.

A SMIL document arises with a XML-like structure, stating some XHTML tags like `<head>` section and `<body>` section. The `<head>` section contains layout and metadata information. The `<body>` section wraps all contents that must be presented together with spatial and temporal information. Timing information is generally a combination comprised of two main tags, i.e., parallel `<par>`, and sequential `<seq>`. Media objects may be associated to the presentation by referring to them with their URLs. This way, media objects can be shared among different presentations and stored on different servers for load balancing.

Figure 1: SMIL media synchronization

Figure 2: SMIL Video Lecture, courtesy of Prof. Alessandro Amoroso (University of Bologna)
Figure 1 shows a typical SMIL-based temporal synchronization scheme. In Figure 2 a SMIL video lecture is presented, comprised of a flow composed by a video clip (showing the lecturer), an audio stream (i.e., the lecturer talk), and a sequence of images (i.e., the slides shown during the lecture). In particular, each slide is synchronized to be played out at the proper instant during the talk.

The SMIL code in Figure 3 is a well formed description of such a presentation. It can be edited as a common text file (on the example for the sake of a cleaner code presentation, some syntactic trivia have been removed). Specifically, the video stream (line 2), the audio speech (line 3), and a sequence of images (lines 4-9) are played out in parallel (<par> block, lines 1-10).

Based on this kind of structure, it is also possible to represent the SMIL document in a tree-like form, typical of the Document Object Model (DOM) [41]. Figure 4 reports a
DOM representation of the SMIL code of Figure 3. Media played out in parallel (i.e., video file, audio flow and sequence of images) lie horizontally, while sequences of media (i.e., images composing the slideshow) are disposed vertically.

SMIL makes possible associating alternative media to the presentation. Features are offered for:

i) providing captioning of possible audio elements,
ii) providing an alternative content to media elements,
iii) showing SMIL presentations on different devices.

Alternative contents for any media objects can be shown by using the attributes: alt for alternative text, longdesc for a link to a longer description (a Web page) and title for a short meaningful textual description to be used in tooltips. In addition, a media object can have an abstract attribute, which is a brief description of the specific media content. Thus, for example, a video element can be coupled with an alternative textual description.

Let us consider the video file which the line 2 of Figure 3 is referred to. An example of an alternative text to that video file, according to the SMIL syntax, is shown in Figure 5.

```
<video src="video/lecturer.rm" region="region_video"
      alt="This video shows Prof. Amoroso that explains ..." />
```

Figure 5: SMIL code fragment with a description of a textual alternative

Alternatively, an example of a long description association (encoded as an HTML document) is depicted in Figure 6.

```
<video src="video/lecturer.rm" region="region_video"
      longdesc="lecturer_desc.html" />
```

Figure 6: SMIL code fragment with a pointer to an external file containing a textual alternative

As previously mentioned, also captions can be added to a SMIL document, to be shown in a given portion (region, in the SMIL jargon) of the visual presentation. To this aim, two solutions are possible. First, text can be added to be visualized in a given region by exploiting the <text> SMIL tag. As an example of this, the SMIL code reported in Figure 7 shows the text “Good morning, my name is ...”, in a region of the SMIL presentation named region_caption. The begin and dur attributes specify the timing properties of the caption, as shown in Figure 7.

```
<text begin="3s" dur="7s" region="region_caption"
      src="captions.txt">
```

Figure 7: SMIL code fragment with spatial and timing information for a caption specification

Other than a simple, linear text file as a caption, RealPlayer supports SMIL presentation captioned with an XML-based format named RealText [32]. It is a document (typically as
an .rt file) where all captions with their related timing properties are included in. RealText can be associated to a given region of the SMIL document. The <textstream> tag allows to add a stream of text as encoded by using the RealText to the SMIL presentation format. An example is the SMIL code on Figure 8. It points out how the text stream on the captions.rt file is shown inside a specific region (named region_caption) of the SMIL presentation.

```
<textstream src="captions.rt" region="region_caption" …/>
```

Figure 8: SMIL code fragment with a RealText-compliant caption specification

Figure 9: SMIL video lecture with captions

```
0 ... 1 <par>
2  <video src="video/lecturer.rm" region="region_video"
3      longdesc="lect_desc.html"/>
4  <audio src="audio/speech.mp3" longdesc="speech_desc.html"/>
5  <seq>
6      <img region="region_slide" src="img/1.jpg" dur="60s"
7      longdesc="slide1.html"/>
8      <img region="region_slide" src="img/2.jpg" dur="60s"
9      longdesc="slide2.html"/>
10     <img region="region_slide" src="img/3.jpg" dur="60s"
11      longdesc="slide3.html"/>
12     ... 9 </seq>
10  <textstream src="captions.rt" region="region_caption" …/>
11 </par>
12 ...
```

Figure 10: SMIL code fragment for a captions augmented multimedia presentation

Summing up, starting from the lecture shown in Figure 2, new elements can be added to obtain a different presentation with alternative media resources. Figure 9 shows an
example of the new presentation with activated captions. The related (simplified) SMIL code is reported in Figure 10.

In the example above, the multimedia lecture (comprised of a video stream, a sequence of images and alternative textual captions) is specifically distributed over of three different regions:

i) a region associated to the video showing the lecturer (i.e., region_video, line 2 of Figure 10),

ii) a region showing the slides (i.e., region_slide, lines 4-9),

iii) a region devoted to show captions (i.e., region_caption, line 10).

Finally, the audio stream playing out the lecturer speech is included in the SMIL document on line 3.

Moreover, each multimedia resource has been associated with an alternative HTML file containing a textual description summarizing the content of that given resource (annotations). To complete this example, Figure 11 reports a tree-like representation of the SMIL document as obtained when annotations and captions are added. Every media content has an alternative, i.e. an additional annotation encoded as an HTML document.

![Tree-like representation of the SMIL code fragment shown in Figure 10](image)

Those alternative descriptions represent important means to augment the level of presentation accessibility. Let us consider the case of a deaf student; he can replace the
audio flow with captions and the alternative description associated to the audio. A blind user, instead, may exploit the alternative textual data associated to the video and to the images. Indeed, these alternative textual descriptions can be easily managed by blind users through assistive technologies such as Braille displays and screen readers. It is worth noting that when dealing with multimedia presentations, the parallel play out of homogeneous different information sources can cause a cognitive overload. As an example, the audio speech of the lecturer cannot be played out while the screen reader is reading the textual description which is associated to a slide. Thus, these alternative media must be carefully managed so as to guarantee an effective content presentation.

Solutions exist to solve such a case, as described in [33, 34]. In the next section we will explain how LAUGH extends SMIL functionalities with new cooperative editing features.

3. LAUGH: A Collaborative Annotation System

From the discussion in the previous sections, it emerges that annotating rich media presentations can be of great help to let students/teachers customize and ameliorate their learning experiences and improve the accessibility of the didactical material. We also discussed on how the use of the SMIL markup language is effective to create multimedia presentations and allows to associate any content with alternative resources.

Yet, the use of a simple SMIL-based technology leaves all the multimedia editing work to the producer of the multimedia lecture. In other words, the lecturer is asked to produce contents in isolation, without the possibility of any cooperation with other partners (students and other teachers).

Lessons learned from the new Web 2.0 technologies suggest to let students becoming prosumers (i.e., producers and consumers) of the didactical material. In particular, we focus on providing them with the possibility of adding new annotations and captions to the multimedia lecture. Nevertheless, an important requirement of such a system is the easiness of use. It could not be easy to ask students to manipulate contents at the very beginning of the learning process. Hence our idea to allow an editing approach, within LAUGH, inspired by the Wiki way. Essentially, LAUGH provides prosumers with a Wiki-type editing interface to collaboratively manipulate SMIL presentations. The rest of this section is devoted to present LAUGH in detail, focusing on the user interface, the employed syntax and the architecture of the system.

3.1. Adding Captions

Generally, the idea of using captions and subtitles is meant to provide textual alternatives to figures and audio streams, respectively. Typically, subtitles are put into a specific region of the screen to avoid interferences with other images. To this aim, we have assumed that SMIL-based lectures include a default region, which is devoted to show the captions/subtitles. With the intention of allowing users to manage captions/subtitles, we have defined a specific Wikitext-like syntax which permits timing manipulation of captions/subtitles. In addition, we have extended the parser provided by the DokuWiki
tool, so that our specifications can run on it [13]. As our syntax is inspired by SMIL, the
typical timing properties of a SMIL element must be managed. As well known, they are:
i) \texttt{begin} which identifies the time for the explicit beginning of an element; ii) \texttt{end}
which identifies the explicit end of an element; iii) \texttt{dur} which identifies the explicit
duration of an element.

In our Wikitext syntax, a $ symbol corresponds to the \texttt{begin} SMIL attribute, the __ symbol corresponds to the \texttt{end} attribute, the :: symbol corresponds to the \texttt{dur} SMIL attribute. Hence, for example, to add a caption/subtitle, a user should exploit a syntax which is as follows:

\begin{verbatim}
$X__Y 'sentence'
\end{verbatim}
or
\begin{verbatim}
$X::Z 'sentence'
\end{verbatim}

where X is the starting point, specified in seconds, Y represents the time the caption/subtitle should disappear (time specified in seconds), Z represents the duration of the caption/subtitle, and ‘sentence’ is the specific caption/subtitle to be shown.

So, the very advantage of LAUGH against SMIL amounts to the fact that we make possible to add the caption/subtitle shown on Figure 9 by simply writing, à la Wiki, what follows:

\begin{verbatim}
$3::7 Good morning, my name is
Alessandro Amoroso, I am
professor of Computer Science
at the University of Bologna
\end{verbatim}

(or, alternatively, $3__10 Good morning, ...). As a result, the caption/subtitle will be automatically displayed in the proper region, starting from the third second of the presentation, for 7 second long duration.

Once the user has added a caption/subtitle by using our Wiki-like interface, the system automatically generates the SMIL markup code, which is needed to display that caption/subtitle for the specific time interval. Specifically, it adds the \texttt{<text>} element, reported in Figure 7, in the proper position of the SMIL document, without any need of any intervention of the user. Obviously, more than one caption/subtitle can be edit by a user in sequence. The system will automatically put such a list in sequence inside the SMIL document, by including it in a \texttt{<seq> ... </seq>} SMIL structure.

Moreover, as the RealText/SMIL technology represents an alternative for creating captions/subtitles based on the creation of given .rt files, we have extended our syntax to manage also the RealText technique. Basically, to use RealText within LAUGH, it is enough to specify the starting point of a caption/subtitle by means of the $$ symbol. The ending point of the caption/subtitle is indicated with the __$ symbol. Hence, with regard
to the previous example, if a user wishes to exploit RealText within LAUGH she/he should write what follows using our Wiki-type interface:

\[ \$3\texttrf{10}\text{ Good morning, my name is ...} \]

Consequently, LAUGH automatically adds the code fragment shown in Figure 8 to a SMIL document. Moreover, LAUGH creates (or updates) the captions.rt file and the caption/subtitle is added by using the RealText syntax shown below:

\[
\begin{align*}
\text{&<Time begin="0:00:03.0" end="0:00:10.0" /&<clear/>} \quad \text{Good morning, my name is ...} \\
\end{align*}
\]

To give the reader a clear impression of the results we obtained with LAUGH, we show in Figures 12 and 13 an example of use of our system. In particular, Figure 12 shows the attempt to add two captions using our Wiki-type interface. The result of this intervention is presented in Figure 13, where two different audio streams are subtitled accordingly.

![Figure 12: Adding captions/subtitles with a Wikitext interface](image)
3.2. Adding Annotations

Additional annotations typically represent important alternatives meant as textual descriptions summarizing the contents of a given audio/video/image element. Our idea, here, was to provide the user with the list of audio/video/image elements composing the lecture. On the basis of this list, annotations can be added using our Wikitext syntax. Such annotations are then translated and stored as a HTML document to be associated as an alternative for that specific audio/video/image element. Obviously, a link has to be established between the HTML document we have produced, and the SMIL document comprised of all the pointers to the audio/video/image resources for which LAUGH has...
created alternative descriptions. This link is established by exploiting a longdesc attribute to be added to the correspondent SMIL tag. This attribute specifies the URL of the HTML file containing the alternative annotations.
Figure 14 shows an example on how alternative textual descriptions can be added to two different images using our Wiki-type syntax. The result is shown in Figure 15.

---

**Figure 16**: The collaborative editing process

---

**Figure 17**: LAUGH: system architecture
3.3. **System Architecture**

On LAUGH, lecturers and learners cooperate together as follows. A lecturer produces a new SMIL-based, rich media lecture and then uploads it into LAUGH (step $p_1$). At the core of LAUGH there is an editing software component comprised of three main modules, as shown in Figure 16.

Basically, as soon as a lecturer uploads a SMIL-based presentation into LAUGH, a parser takes the SMIL document, parses its code, identifies all the multimedia elements and passes this information to a page producer, which in turn produces a Wiki-like Web page, on which users may now add annotations using the DokuWiki module.

As to captions, instead, an interpreter is employed to transform the Wiki-like code produced by the user into a SMIL code, as discussed at length in the previous section [13]. All these new contents are added to the SMIL presentation, so that to produce an updated multimedia lecture.

A big picture of the LAUGH architecture is shown in Figure 17. As a final consideration, it is worth noting that LAUGH incorporates also a dynamic adaptation system able to select, among all the possible alternatives, the one which is more adequate to a given user. The discussion on the mechanisms on which this component is based are out of the scope of this paper. Interested readers can find a detailed discussion of these issues in [33, 34].

4. **Experimental Evaluation**

LAUGH enables to enrich didactical contents, based on an open process which encourages the participation of students. Clearly, such a process has important consequences on the quality of the didactical material. The most prominent advantage is the presence of new alternatives and notes, which may improve the accessibility on the lecture.

To assess the efficacy of our approach, we simulate in Section 4.1 the collaborative process of manipulation of a multimedia lecture by means of a community of learners. We will observe how the SMIL code changes due to the enrichment process and how the corresponding ACCMD profile grows up accordingly. As anticipated, this collaborative process leads to a variety of alternative presentations, all stemming from the original lecture. The idea is that our system replies to a given user request delivering the most appropriate presentation, based on the user profile. This may be a complex process which requires not only the selection of the best version among different alternatives, but also a media transcoding activity needed, for example, to adapt a certain presentation to a given user device. With this in view, Section 4.2 reports some few experimental results obtained with LAUGH engaged to respond to hundreds of contemporary requests where some kind of transcoding activity is needed.
4.1. Case Study: Sharing a Video Lecture

To evaluate how a typical rich media content may change within LAUGH, we draw out a scenario where a teacher uploads his lecture into LAUGH and then the community of learners participate to the editing process by adding captions and annotations. Figure 18 shows this dynamics process for a given lecture. Specifically, the lecture is comprised of an audio track synchronized with a sequence of slides (without any caption). The original language is English (but some of the learners are Italians). Furthermore, slides 3 and 5 show images, while the others are purely textual.

We can summarize the activities of the community as follows:

1. The lecturer uploads the SMIL presentation.
2. Learners collaboratively enrich the didactical materials:
   a. adding all the necessary captions (in English) to the audio track;
   b. translating all the captions in Italian,
   c. adding an English annotation for images to the slides 3 and 5;
   d. adding an additional resource under the form of an image to the slide 4;
   e. adding an English annotation for this latter image.

Figure 18. Dynamics of collaborative editing on a lecture
As a result of such activities, not only SCORM-compliant metadata are added to the ACCMD specification of the original multimedia resource (i.e., the lecture), but also the corresponding SMIL document is enriched with additional information produced by the learners community [19].

All this leads to an increase in terms of size of both the ACCMD profile and SMIL documents. Tables 2 and 3 summarize these increments measured both in terms of additional elements (columns 3, 5) and in terms of Kilobytes, from the SMIL document and ACCMD profile, respectively.

### Table 2. SMIL growth

<table>
<thead>
<tr>
<th>Phases</th>
<th>KBs</th>
<th>Elements</th>
<th>KBs%</th>
<th>Elements%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2,21</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2a</td>
<td>2,73</td>
<td>32</td>
<td>23%</td>
<td>28%</td>
</tr>
<tr>
<td>2b</td>
<td>3,62</td>
<td>42</td>
<td>63%</td>
<td>68%</td>
</tr>
<tr>
<td>2c</td>
<td>3,7</td>
<td>42</td>
<td>67%</td>
<td>68%</td>
</tr>
<tr>
<td>2d</td>
<td>3,91</td>
<td>43</td>
<td>77%</td>
<td>72%</td>
</tr>
<tr>
<td>2e</td>
<td>3,95</td>
<td>43</td>
<td>79%</td>
<td>72%</td>
</tr>
</tbody>
</table>

### Table 3. ACCMD growth

<table>
<thead>
<tr>
<th>Phases</th>
<th>KBs</th>
<th>Elements</th>
<th>KBs%</th>
<th>Elements%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>48</td>
<td>49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2a</td>
<td>68</td>
<td>89</td>
<td>42%</td>
<td>82%</td>
</tr>
<tr>
<td>2b</td>
<td>88</td>
<td>134</td>
<td>83%</td>
<td>173%</td>
</tr>
<tr>
<td>2c</td>
<td>96</td>
<td>148</td>
<td>100%</td>
<td>202%</td>
</tr>
<tr>
<td>2d</td>
<td>100</td>
<td>152</td>
<td>108%</td>
<td>210%</td>
</tr>
<tr>
<td>2e</td>
<td>104</td>
<td>158</td>
<td>117%</td>
<td>222%</td>
</tr>
</tbody>
</table>

### 4.2. Adaptation subsystem evaluation

Additional metadata generated as a result of the collaborative process discussed before enable the production of a number of customization versions of the original contents to be delivered to given users in response to different specific requests. To assess whether LAUGH is able to respond to a number of several different requests, we carried out the following experiments. Several concurrent requests were randomly generated for dozens of rich media learning objects, collaboratively produced by resorting to LAUGH. For each request, a specific user profile was randomly selected, thus imposing a specific resource customization [34]. The number of contemporary
random client requests was set to vary from 0 up to 500. Figure 19 reports the average
time required to receive, analyze the request, adapt and deliver contents back to the users,
depending on the number of contemporary requests.
As shown in Figure 19, the system results (essentially) scalable, as response times are
only partially affected by the increase of the number of contemporary requests.

![Figure 19. Average Adaptation Time](image)

5. Conclusions
We have presented LAUGH, a system for the cooperative creation and sharing of SMIL
multimedia resources. LAUGH is employed in typical collaborative e-learning contexts
where users may enrich didactical materials for use of an e-learning community. In
particular, LAUGH can be used to add captions and annotations to an original multimedia
resource, using an user friendly Wiki interface.
LAUGH is inspired by the Web 2.0 concept, according to which knowledge sharing and
collaborative production processes may lead to the emergence of a distributed
intelligence. In the case of LAUGH, this improves the quality of the didactical material
distributed over the network. We have shown that LAUGH extends and improves the
SMIL standard. This has motivated the title of the article.

Acknowledgments
This work is supported by the Italian M.I.U.R. under the MOMA and DAMASCO
initiatives.

References
Object Aggregation Model (SCORM) Version 1.3, retrieved October, 2006. Available from


4th IEEE Communications and Networking Conference (CCNC 2007), Las Vegas (USA), IEEE Communications Society.