Technical Infrastructure: The WebReports site

Gordon Simpson, Yishay Mor, Jesper Holmberg, Richard Noss, Jakob Tholander, Ken Kahn, Celia Hoyles

D3.2.2 Technical Infrastructure: The WebReports site

Overview

The WebReports website allows children to create text-based and interactive content that is accessible on line and to comment on each other's work, thereby being provided with opportunities to collaboratively build knowledge around a range of different learning domains. In this document we first summarise our key findings. We then provide an account of the evolution of the system during the three years of the project, giving the rationale for our major changes. Finally, we describe the functionality and architecture of the final version of the WebReports system.

Key findings

1. The evolution of site structure through iterative design

The final version of WebReports was the result of a long process of iterative design, in which the WebLabs community and the site structure co-evolved.

One example of this process is the navigational structure of the WebReports site. The first mockup ignored this issue almost completely, focusing on the individual aspect of publication and reacting to comments. As soon as we started publishing reports, even using our crude *notes* mechanism (described below), it was obvious that these need to be indexed in some way. We began to organize them manually by group, topic and author. At this point we made no distinction between different types of reports and different types of groups. This organization was carried over to our next system, which was based on a Wiki. The advantage of a Wiki is in its flexibility, and it gave us the opportunity to experiment easily with different ways of organizing our content. We started with the same categories we had before, but soon added several others: tools, games, papers and conferences, local info, tasks and links. However, the ease of making changes meant that the site structure was messy and unstable. It mixed categories of very different functional levels and audience, and most participants found it too hard to navigate. This led us to several decisions:

- Negotiate a simple structure for the site, with a minimal set of top level categories, and fix it.
- The site should be structured from a student's perspective; content that is directed towards teachers or researchers should be moved to the background.

Observing the emerging usage patterns, we also acknowledged:

- Content should be accessible by topic and by site. These two dimensions needed to be separated.
- There are several distinct patterns of using webreports, and these should be supported by the system. This issue is discussed in detail below. At this point, we will only note that this led to the emergence of the functional dimension of report classification.
- User actions should be restricted to the area in which she is browsing. Users can only edit reports in their folder, the report indices are generated automatically by the report meta-data and only researchers can edit the topic and site front pages.

The evolution of the site structure did not stop at this point. Once the new (plone based) site was set up, the community's patterns of activity adapted to it and gave rise to new requirements. In fact, minor changes were made occasionally up until the last months of the project.

A second example regards the evolution of our methodology itself. Early on in our experiments we noticed two critical transitions in our educational cycle: the first from group discussion to individual construction and the second from construction to reflective reporting. In our preliminary experiments, we would discuss the mathematical question at hand, derive a task, describe that task verbally and then typically ask: "*does everyone know what to do now*"? Of course, this question was received with a round of nods across the class. However, when we then asked the students to turn to the computers and start working, many would demonstrate that they were unsure what to do.

At first, we addressed this issue by using traditional means: printed worksheets. These worksheets would start with a summary of the big question being investigated, followed by a description of the programming task in text and accompanying images of ToonTalk components. These worksheets helped, but at the price of students having to switch between three different media: ToonTalk, WebReports and paper. Eventually we began to feel uneasy with the need to scaffold computer-based activity with paper artefacts.

At the same time, the WebReports system had evolved to address issues which emerged from our first prototypes. Apart from general issues of ease of use (e.g. WYSIWYG editing), we introduced features derived from our educational methodology. One such feature was **streamlined embedding of ToonTalk models** in webreports. Another was report **templates** which scaffold students' writing. Both were designed with the difficulties of reflective reporting in mind. Once a construction task is completed, the student can upload the model she created into a new report by "holding" it in her hand and hitting the pause button. When the model is uploaded into the report, an image of it is automatically embedded in the text. Templates are reports that include ready-made headers and prompts, designed to scaffold students' reflection and articulation. Users can click a button at the top of a template to start a new report from it, which will automatically include all the elements in the template. Templates were tailored to the specific activities, with specific tools to use and questions to explore.

Only after these features were available did we realize that they enabled us to create a new tool, and a new related practice, which we call *active worksheets*. These are report templates which include task instructions and questions. The novelty of this tool is that all the tools required for the task are embedded in the template. Students click on the tools they need, work their way through the modelling task, and eventually replace the question text in the template with their own observations. After experimenting with this approach for a while, we began to embed not just the tools, but the *task itself* as a ToonTalk object. This practice was known as *task-in-a-box*: (Figure 1): a ToonTalk box with an untrained robot, an input box, and output nest, and instructions. The task-in-a-box serves several purposes at once. First, it helps students overcome the shift in medium from the (mainly textual) web page to the animated programming environment. More important, it scaffolds their work by providing the input box to be used in training. Last, it implicitly sets a standard for packaging and sharing ToonTalk models.

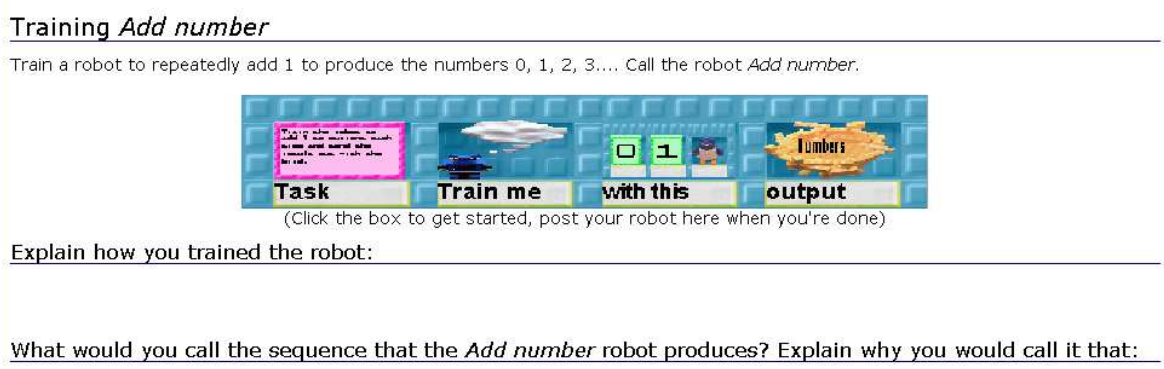


Figure 1: `Add a number` task-in-a-box

2. Developing a solution for constructing multiple representations

By embedding ToonTalk objects in webreports we facilitated the inclusion of these objects as elements in students' emerging shared repertoire. To support this capacity, we needed to ensure that objects could be easily carried from one context to another. This required simultaneous enhancements to ToonTalk and the WebReports system: ToonTalk had to support a smooth export functionality, which was tied in to the WebReports' import functionality. Both features were developed in tandem but independently. Thus, ToonTalk can now export to other environments and WebReports offers an open import API, allowing learners to link representations.

The transition between ToonTalk and WebReports is not just a technical one. It marks a shift between the different phases of the learning process. ToonTalk programming is the main tool for individual construction, while webreports are geared towards collaborative knowledge building. Moving between the two requires students to move between different states of mind. While we see the synergy of the individual and the collaborative as one of the strongest innovations of the project, it does pose some obvious challenges. Having a streamlined method of carrying objects from one environment to the other scaffolds students' transition between the two phases of their learning. This mechanism works in both directions: students can select an object in ToonTalk and carry it over to a webreport, or click the object's image in a webreport to open it in ToonTalk.

3. The difficulties of designing for interaction

Viewed as an activity system, there are six components that we had to take into account in designing the WebReports system. These were:

- The structure of the community (or communities) of researchers, teachers and students.
- The division of labour between these three groups and within them.
- The social rules which govern interactions between students and between students and teachers / researchers.
- The web of connections which tie local groups and global communities.
- Other instruments in the environment, such as the programming environment and spreadsheets, traditional tools, such as whiteboards and paper, as well as specifically designed objects for collaborative group activities.
- The mathematical and scientific objects which are explored and the educational outcomes of these explorations.

Using an activity-oriented perspective helps us understand both our successes and our disappointments.

For example, the *Guess my Robot* game discussed below made very intensive use of the facilities provided by the system: the game is played by exchanging ToonTalk objects between proposers and responders using the report and commenting mechanisms. In fact the design of this activity shaped the requirements of the system. One such feature was the capacity to include ToonTalk objects in comments, which required substantial development effort, and was driven in large by the needs of the game. However, the game would not have succeeded had we not paid careful attention to the other constituents of the activity system, such as the social rules and division of labour within the community. These range from norms of courtesy when responding to a challenge to the role of a facilitator that directs the dynamics of the game.

In other cases lack of attention to conflicts between our design and existing practices resulted in disappointments and eventually called for redesign. One such example is the issue of notifications. From the offset it was agreed that we would need some form of notification, to alert users when new content in their domains of interest was available. When we implemented the plone based system, it included a mechanism for e-mail notifications. This mechanism would send users an email message whenever a report was published in one of the topics they registered for. As the number of reports increased (and indeed greatly exceeded our expectations) users found this feature disruptive to their daily work, and eventually we disabled it. On the other hand, we acknowledged that notification of comments was essential for discussion dynamics in such an asynchronous environment. We observed that sometimes it may take days, or even weeks, from the time a report is published to the time a comment is posted. When that happens, the author might not notice the comment and the opportunity for a fruitful interchange is lost. To overcome this, we introduced an internal messaging system which alerted users to comments on their reports (as well as responses to their comments).

Finally, some features we included with high hopes were seldom used. Drawing on the success of blogs in web culture, we invested effort in a feature we called “blabs”. This feature allowed users to maintain a personal on-line journal of their activities. We hoped that this tool would promote and assist students in their personal reflection on their experiences. However we found that this feature was seldom used and eventually we removed it. We see two main reasons for its lack of success. First, we did not enable commenting on blab entries. This meant that communication through the blab mechanism was one-way. When students go to the effort of publishing content on the web, they do so to provoke feedback and engage in social interaction. This observation is supported by our evidence from other activities, were students expressed disappointment at not receiving comments on their reports. Furthermore, in several cases students used the standard webreport mechanism to publish journal-style content. While it may be the case that this is derived from their familiarity with the tool, we suspect that they found reports more attractive precisely because they offered an option of interaction. The second issue relates directly to activity design. As a rule, the features that were used successfully are those that played a role in an activity. Most activities involved using templates, publishing reports and including ToonTalk objects in them. Students became very proficient in using these tools and developed surprising new uses for them. On the other hand, none of the activities required students to edit their personal pages or post blab items, and these tools were indeed seldom used.

4. Classifying collaboration

We identified four distinct ways in which webreports were used in practice by students for the purpose of collaboration:

a. Group reports

b. The WebLabsPaedia

c. “Guess my X” style reports

d. Reports used for group presentations.

Group reports were authored by a group of students at a local site, with the writing being orchestrated by a teacher/ researcher. This typically occurred after some set of activities in which students first published individual reports. Figure 2 shows a group report with associated cross-site comments. It is interesting to note that many comments were posted in a thread of discussion, and that some of the comments included multimedia (such as the photo visible in the middle screenshot). This type of collaboration was in fact the closest we came to our original vision of students sharing their evolving understandings and working models in particular knowledge domains. We had also hoped that students might collaboratively co-author group reports *across* sites, but this did not eventuate. Whether this was due to lack of system support or to inherent cross-site collaboration difficulties is not entirely clear. It is certainly true that aspects of the system did not promote this sort of collaboration - for example, the mechanisms to form groups and to have multiple report authors were cumbersome. But even if support for multiple authors had been improved, there are still significant procedural difficulties to overcome e.g. how could we ensure that both groups "sign up" to the content and changes made to a report, and how could a sense of ownership be maintained by the students? Note that pragmatic difficulties were also significant – groups at different sites were often engaged in activities at different times and had different native languages for example.

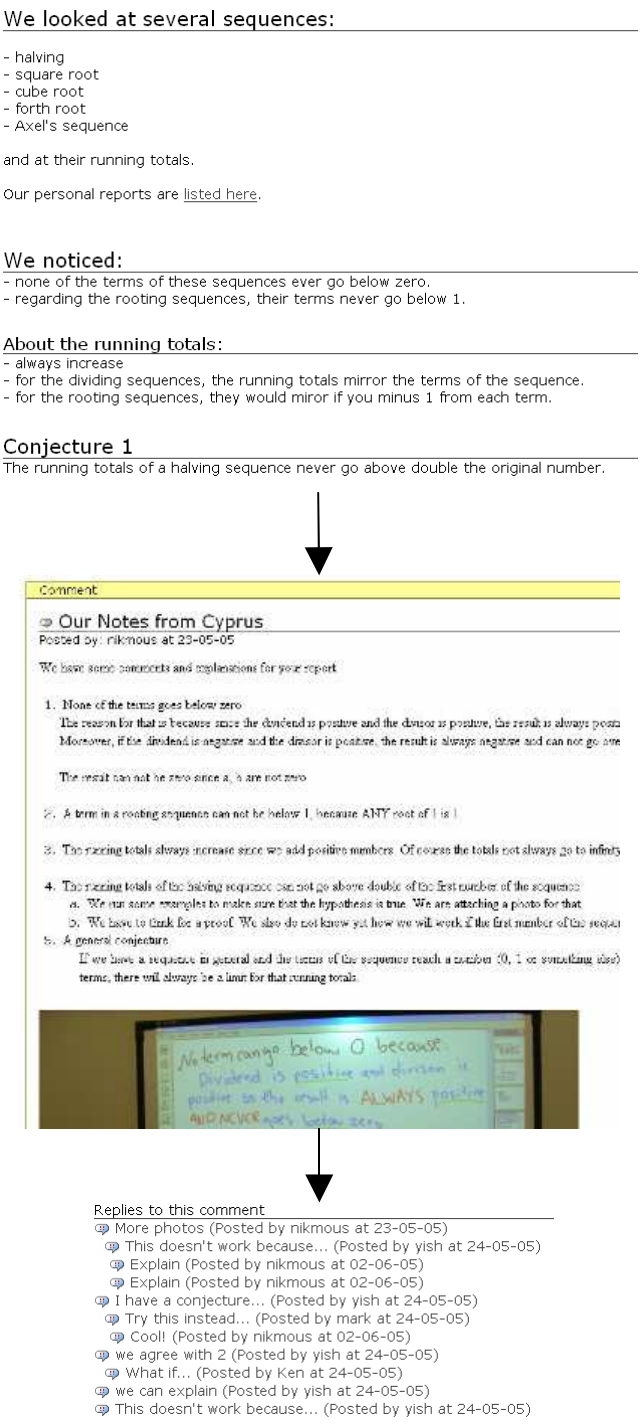


Figure 2. Illustration of a group report and cross-site comments. The top screenshot is of the original report, the middle shows a comment that was posted from another site, and the bottom shows the thread of comments posted in the ensuing online discussion.

The second type of collaborative report was the **"WebLabsPaedia"**: an encyclopaedia in the Model systems and randomness knowledge domain. This was tightly linked to activities that required successive refinement in the definition of terms (such as "predictable" and "random"). Students at one site worked together to define the terms and cross-site comments were posted

between groups. The WebLabsPaedia can be considered as a set of group reports with a focus on definition and refinement of particular terms.

The third way in which webreports were used for collaboration was in **"Guess my X" style individual reports**. These reports took a challenge-response type structure in which a student from one site posted a challenge to be reproduced by students from another site (see Figure 3). The Guess my Robot activity was very successful in the Number Sequences domain during the second year of the project and we subsequently generalised the format to the Lunar Lander (Guess my Graph) and Model systems and randomness (Guess my garden) domains during the third year.

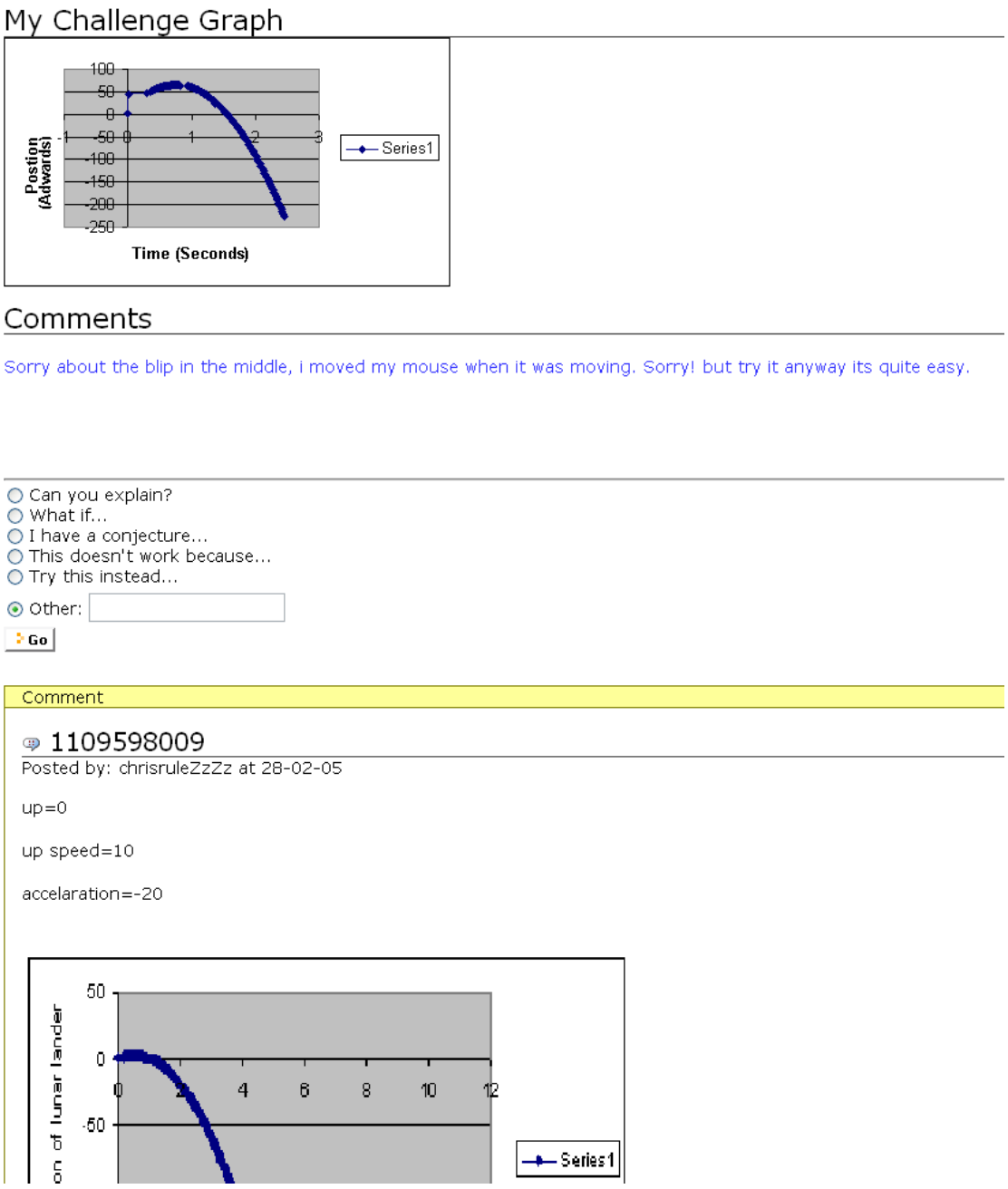


Figure 3. An example of a Guess my Graph webreport showing the original challenge and a response posted as a comment from another student.

The fourth and final way in which we identified webreports being used for collaboration was when they were used for the purposes of **group presentation**. In this case, webreports became objects of discussion that students used to support their views and arguments. Although we did not design webreports for this purpose we found it to be a very successful way of scaffolding students' presentations.

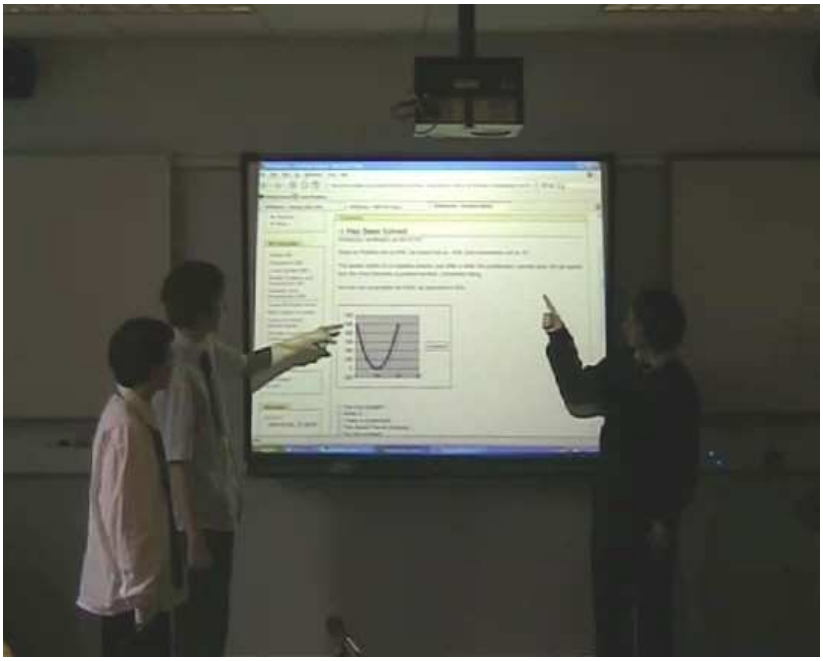


Figure 4. A group of students use a projection of their webreport to present their views to the rest of the group.

5. Outstanding Challenges

We have identified two major challenges that remain outstanding: support for dynamic group forming and language translation issues. By dynamic group forming, we mean the ability for users to form new groups or join existing ones that are then supported as objects in the system e.g. automatic generation of homepages with associated reports and members. Native support for this is not built in to the Plone architecture and would require significant development and testing which we did not have time to implement. As it is, the system classifies users by *site* and *topic group*. Although this is useful, we later realised that we really wanted a way for students *across different sites* who were working on the *same activities at the same time* to be connected. The forming of such groups was essentially manually managed by the relevant researchers and teachers, who pointed their students towards the work of students at other sites who they knew were working on the same topic. We believe that dynamic groups could be a valuable addition to the system, allowing students working on the same activities to share a common space and collaborate more readily. It could also ease the administrative burden of manually managing the "groups".

Language was another major outstanding issue, and is discussed in detail in the "Translation and the language barrier" section below. In summary, although there is support within the system for automatic translation of the UI and for multi-language versions of reports, the actual translation of user content still needs to be done manually. Our solution was essentially to use English as the common language for communication and collaboration, although we probably underestimated the administrative overhead required for translation. We also had success with using ToonTalk code

and mathematical objects such as graphs as the language of communication in activities with a well-defined structure (thus lessening the need for natural language). The language issue cannot at this point in time be solved simply with a technical solution, and will remain an obstacle to effective collaboration across languages and cultures, especially where children are involved.

Historical account of iterative design

Year 1

Over the course of the first year, we experimented with four prototypes for the web reports system. The first prototype (mock-up) was not an operational system, but rather a conceptual demo. The second prototype (JSP templates and "notes") was used by project members to share developing ideas and reports on activities. The third prototype (Blogs) was used by the IOE team to log their activities mainly as a test of the technology. The fourth prototype (Wiki) was used to author and share real web reports between partners and participating students. We also reviewed several other systems and collaborative websites: as possible infrastructure solutions for web reports, or sources of inspiration.

Prototype 1: Mock-up

An html/javascript mock-up was constructed to provide a forum to discuss the desired mechanism for web-reports. The mock-up consisted of several static pages, which were linked so as to demonstrate a possible student workflow. The main objective of this exercise was to provide an object for experimentation and discussion of the features and workflows we envisaged in the web-reports system.

Prototype 2: Notes

We created several templates for authoring webreports. Additional templates were designed for worksheets and slideshows. We developed a software mechanism called "notes", to be used for commenting, and thus to assist collaborative interchange. Our notes are used as special tags which could be inserted into web pages. Where they are present, the web server generates a special link which allows the reader of the page to add comments to it. Typically, a summary of the comment is inserted into the page, and the rest can be seen in a pop-up window. The following figures illustrate the developed templates and the notes mechanism.

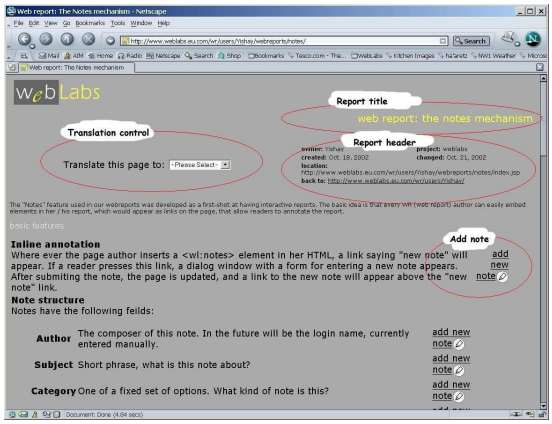


Figure 5: WebReport page structure

A sample webreport page. The template includes a report title, report header, translation control and associated style sheets (CSS). A number of note links can be seen toward the bottom right.

We identified a number of drawbacks with this system:

- Implementation needs refinement: The templates are based on JSP technology. It is possible to edit reports in a WYSWIG editor, but this requires some "massaging" of the edited document before publishing. Uploading of reports is done using FTP to the web server. This method requires some technical know-how.
- Unimplemented features: Some necessary features, such as e-mail notification, note editing and deletion, authorization and additional templates were not implemented.
- Missing features: Some features are hard to address in this framework, such as versioning, personal portals, repository and file attachments. In particular, it was hard to identify a note with a particular piece of text, to trace 'conversations' through notes and to distinguish notes of different characters (meta-points, specific disagreements).

We concluded that the notes mechanism would need significant development and would also need to be used in conjunction with another mechanism, which provides features it cannot address.

Prototype 3: Blogs

During the project meeting in Cyprus, the idea of using a "blog"-like tool arose. Blog is an internet jargon, short for web-log. It refers to a personal diary that is conducted on the web. Typically blogs are created using specialized web servers, which provide users with an easy to use interface. This interface allows the user to add consecutive, dated entries to their blog by using a simple web-form. The idea that was discussed in the project meeting was to employ a similar mechanism for supporting research diaries, both personal and group managed.

The IOE team experimented with using a Blog, provided for free by a commercial supplier (http://www.blogger.com). We managed to integrate the Blog seamlessly into the existing web site, and for some time recorded our activities on it.

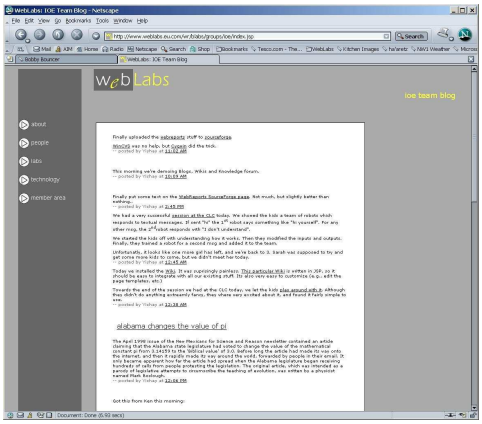


Figure 6: IOE team's blob page

Blogs were easy to use and could be edited on line, in plain text. Each entry includes the date it was created. Although content can be entered as plain text, the blog software recognizes html, so we could embed images and links. We could easily edit the page wrapping the blog, to include links, the WebLabs logo and so on. However blogs offer restricted functionality and limited ability to modify or add to existing functionality. For example there is no real sense of structure or user management, personal portals, repository, email notification etc. We concluded that blogs were not really suitable for our full webreports system, although they could potentially form a part of it.

Prototype 4: WebLabs Wiki

After experimenting with the notes mechanism, we realized that although it was quite rich in features, these were not accessible to less technical participants. At this stage, we wanted to start pilot tests of collaborative activities, and needed a system we could use, even if it did not support all the features we needed. Therefore, we decided to search for existing technology which would support most of our needs, and would be easy to use, install adapt and maintain. After a brief research phase, we opted for using Wiki technology, and specifically the JSPWiki implementation.

We constructed a "webreport" site using JSPWiki technology. Wiki's themselves typically do not impose limitations on the way content is added, structure is imposed by the conventions of the user community. The following figures serve to illustrate the various conventions and Wiki features of our site:

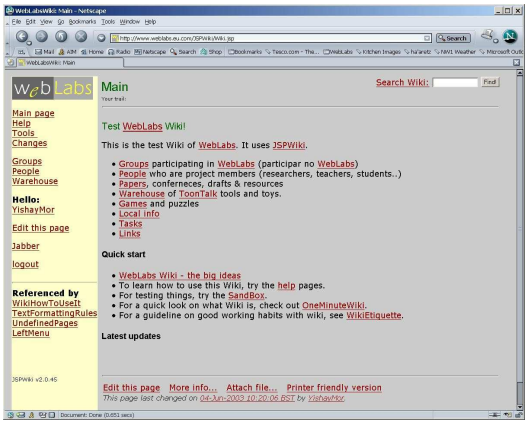


Figure 7: WebLabs Wiki front page

Our front page showing the menu down the left and various links to major subsections. Important structures in Wiki are:

- Groups: for the main foci of the work of WebLabs
- Warehouse: where tools are displayed, with a directory-like map at the top giving the user a sense of location
- People: where researchers, teachers or children can display their pages.

The Wiki system provided us with a number of features:

- Web form editor (using Wiki markup).
- Open source, so we can change at our pleasure.
- Very powerful editable template mechanism. We easily modified the page templates to match our "look and feel", removed some elements to reduce clutter, and added others (e.g., our "notes").
- Seamlessly integrates with any other JSP / servlet based tools (e.g., our "notes", calendar servlets, etc.)
- File attachments.
- Easily extendable by plug-in mechanism.
- Versioning of pages and attachments.
- Recent changes list.

- Search and Index tools.
- RSS feed.
- Tag library for easy template editing.
- Bread crumbs (a trail of previous pages visited).

The Wiki was successfully used for trial collaboration between kids in different sites. A warehouse of ToonTalk code was developed as a Wiki subsection. We thought that the Wiki might be 'just good enough' for our purposes. However, after testing with researchers and students we identified a number of drawbacks:

- Wikis are by nature "flat", i.e., they have no hierarchical structure. Some Wiki clones support a partial hierarchy, either by a mechanism of "sub-Wiki" of by using "categories". In our Wiki pages we imposed some structure by adding context links to the pages themselves. However, this mechanism relies on the good will of the page editors.
- No user management.
- No draft-publish cycle.
- Limited support (bug fixes are supported).
- Wikis are all-purpose and potentially powerful to the initiated user. But the markup language is far from natural, and there is some undesirable "noise" associated with creating and editing pages.

Years 2 and 3

At the beginning of year two of the project, a critical evaluation of the previously implemented Wiki site was performed. Some major limitations with that solution were identified. Firstly, navigation was found to be cumbersome both for children and adults, given the fact that the Wiki was a hyperlink structure with no other means of navigation. Secondly, the basic editing of content by users required learning of the Wiki-specific mark-up language, which was found difficult for the targeted user group. Thirdly, the Wiki contained no architectural support for ownership and control of permissions, which would require considerable implementation effort from the site developers.

This evaluation led to a reconsideration of other potential site-building technologies. After some investigation, the project decided to use the Plone architecture, which proved to have solutions to the abovementioned limitations already implemented. Among other things, Plone provides the users with a personal homepage, WYSIWYG editing of content and the possibility to easily implement functionality for commenting on the work of other users.

The following sections of the document describe the final version of the WebReports system, highlighting the changes made during the third year. The system is based on the Plone architecture adopted during the second year, with various refinements being undertaken to make the system better suit our needs. Note that a paper detailing the iterative development of the system was presented at the 10th anniversary Computer Supported Collaborative Learning (CSCL) conference.

The WebReports system

Technology architecture

The WebReports collaboration site is built using the Plone (http://www.plone.org) content management framework. Plone is an open source architecture designed to be used as a starting point for developers to create collaborative web sites. Plone itself runs on the Zope web application server (http://www.zope.org), which is built to support large web sites with dynamic content. Zope is written in the Python programming language (http://www.python.org).

Among other things, Plone provides the users with a personal homepage, user-friendly editing of content and the possibility make pages that can be commented by other users of the system. The physical server itself runs a version of Debian GNU/Linux (http://www.debian.org), with the Apache web server (http://www.apache.org). The site currently has around 400 registered users. More than 764 reports have been published since the new architecture was deployed in October 2003.

Navigation

Figure 1 displays a screen shot of the interface of the WebReports website, as it appears when first entering the system. To aid navigation, a site-wide search field is presented at the top of the interface. Through this field, the user can search for any type of content. An issue that was raised after the second year of the project was that searching for content on the site is quite cumbersome, as it was found that the search functionality usually returns too much unclassified data. We had been planning to improve the search interface by offering more structured search results, helping the user to find information more quickly. However, since the current search function was part of the internal Plone infrastructure we decided to keep the current status of the search function. To facilitate navigating the website, we instead redesigned the front page of the system, as well as the structure and naming of interface tabs.

The tabs on the upper left (“Welcome”, “Sites”, “Tools” and “Teacher Guide”) are site-general options, i.e. clicking these links gets the user to content which belongs to the site as a whole. Previously, we also had a tab named “Topics”, but now the information on that page is instead present of the first welcome page of the system. On the upper right are two links, "Undo" and "Log out", which make it possible for a user to undo any number of actions made to his content, or to log out, respectively.

Down the left-hand side are small yellow boxes of content which are specific to the individual users. First of these is the user box, which contains links to the user's home page, reports page and blab. The next box is the “Favourites” box, where links to the topics groups and site which the user belongs to are displayed. A report counter is displayed next to each topic group, giving the user an indication of activity in the domains of her interest. The user can also add any page to this box, which makes it possible to create quick links to pages she visits often. The last box is the messages box, where incoming site messages are displayed to the user. Clicking on any of these links shows the message, and the user can either choose to reply to the message or discard it.

At the beginning of the third year of the project, the “active users”-slot was removed from the students’ view of the website. This was because it was found that the students often put too much focus on who was currently online, and also since we found that this feature appeared buggy and inconsistent. Instead, we added a link to the students own user-group in the favourites-slot, which shows all users who are working within the same topic. This directs students to contact users at other sites who may not currently be online, rather than contacting only with users within the same classroom.

The body of the front page of the site (Figure 8) displays links to the most common actions that users may perform in the system, links to the five topic groups, and a table with the ten most recently published reports.

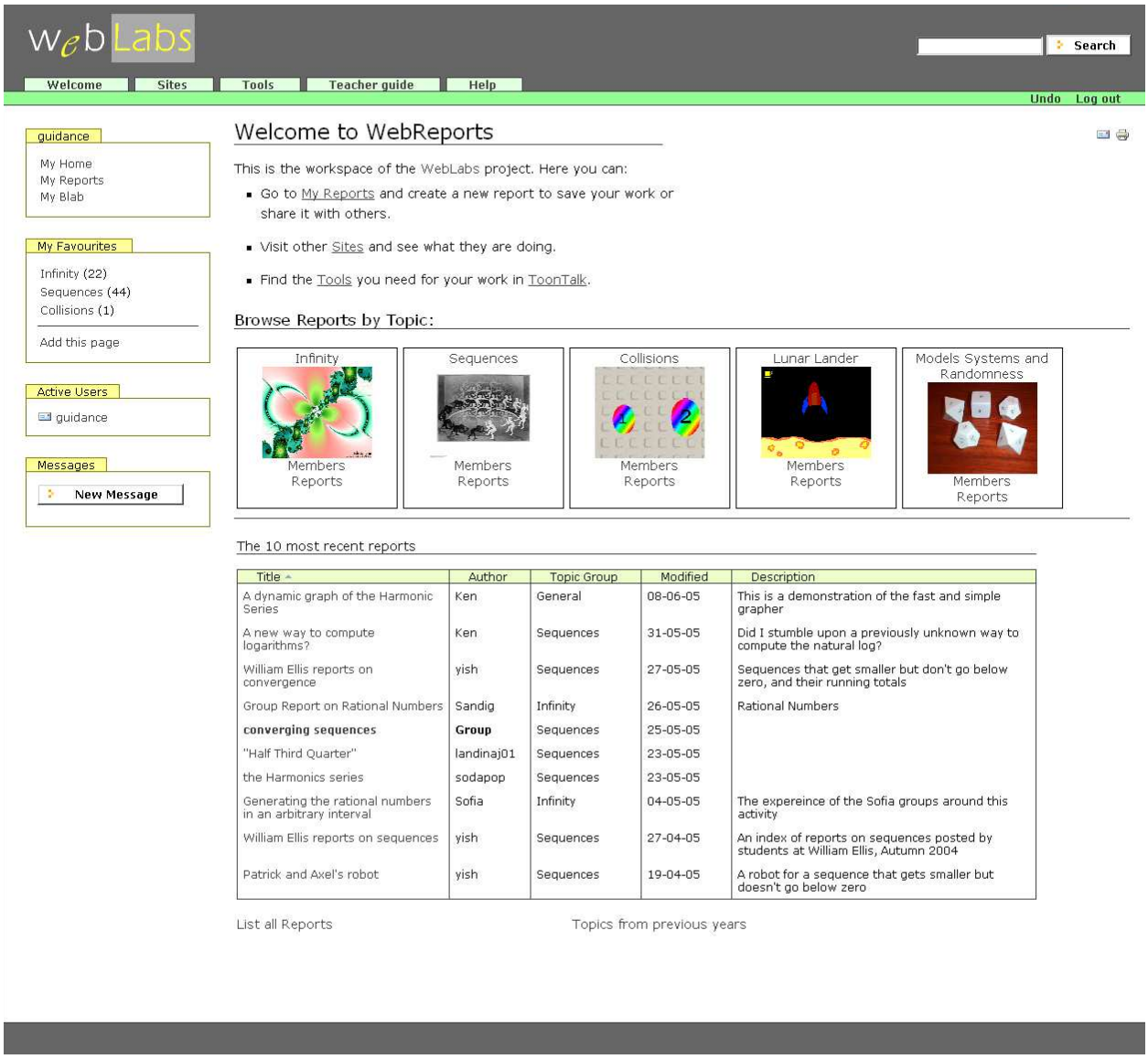


Figure 8. General User Interface. The welcome page.

Grouping of content

The content on the site, as well as its users, are grouped in two dimensions: the topic groups and the participating sites. Each topic group and each site is presented with its own index page, and users and reports are presented on these pages. Members and reports thus show up on several different index pages, depending on through which dimension the user chooses to browse the content. Each participating site is also represented on the web site. Figure 2 shows the presentation of all the participating sites, which are currently 16 different groups of students.

The topic groups represent the various fields of interest in which the users participate, and currently include “Infinity”, “Sequences”, “Lunar Lander”, “Collisions”, and “Models Systems and Randomness”. Topic groups from previous years have also included “Randomness”, “Fibonacci”,

"Primes and Factors", "Force and Acceleration" and "Ecoliteracy". You can access these topic groups by clicking on the link "topics from previous years" on the welcome page.

Each topic and site has a presentation page, where the general presentation of the topic or site has been created by the researchers. These presentations can easily be created in several languages (see below). Through these pages, links to the members of a specific topic group or site are provided along with the reports they have created.

On her Preferences page, the user can select to which groups she wants to be associated. Likewise, the user can choose to which topic group a certain report should belong (see the section "Reports"). The choice of topic group of a report is limited to the topic groups the user has chosen to be part of. Analogously, a user must choose to be associated with a site. All teachers and researchers in the projects are associated with the special group "Teachers and Researchers" instead of a site. This is partly because many researchers collaborate with several groups of students simultaneously, and also to separate the teachers from the students within the community.

Among the topic groups listed on the welcome page, one can see how many reports are published under each topic group and site (within parentheses). It is also noted in the same way in the favourites-box.

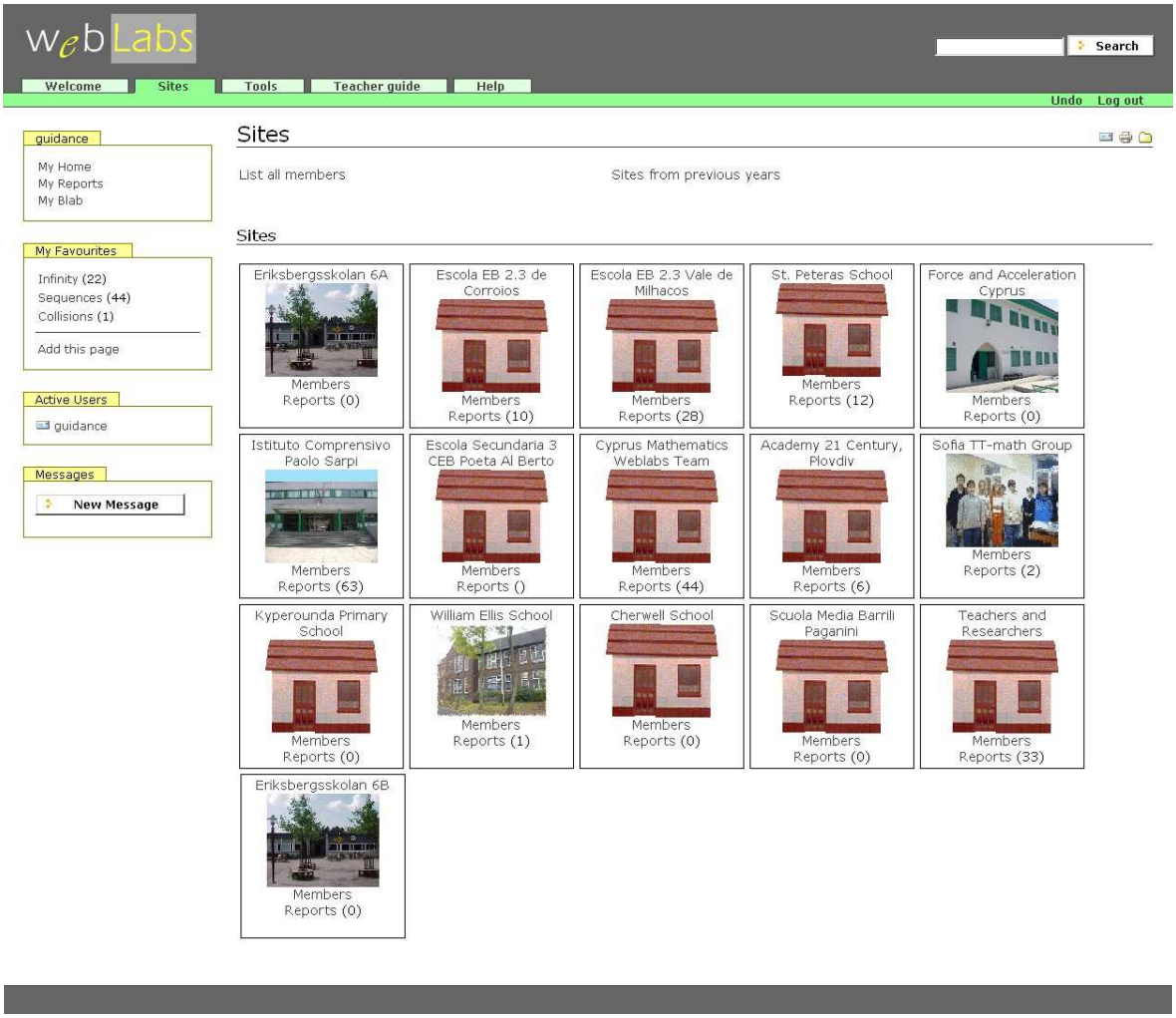


Figure 9. Participating Sites. Each site is represented by a picture.

Adding Content

Personal home pages

Each user has her own home web page, where she can present herself to other users. Many of the children in the project spent much time working on presenting themselves on their personal home pages. Generally, the children had no difficulties learning to edit personal pages or reports. Figure 3 is an example of a home page made by one of the students. The header of the home page of a user (the beige box underneath the name of the user) includes information about her groups, and site. Her email is displayed to other members if she has given permission to show it under her "Preferences". The default is not to show it. There is also a button with which one can send a message to her.

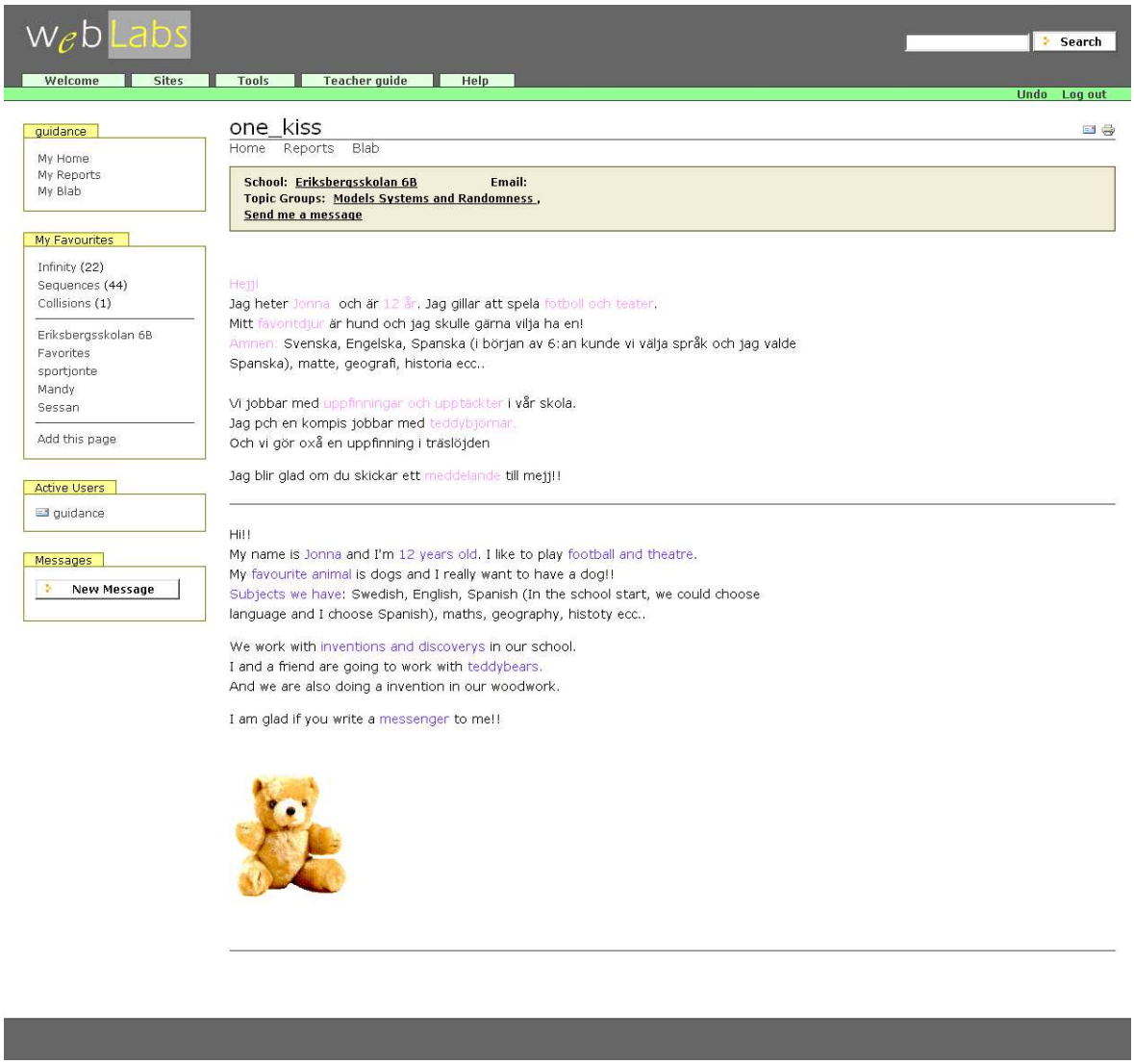


Figure 10. The home page of a user.

Reports

The main piece of functionality on the WebReports site is the authoring of reports. Users create reports about their work, and then publish them on the site, meaning that they are publicly available to all other visitors. Several users may have the right to edit a report, if the original creator of the report chooses to give other users those permissions. Each user has her own reports page, where all reports she has created are presented – including reports not yet published (Figure 4).

To start a report, the user is first taken to a screen where information about the report, such as the title and the general topic of the report can be specified. Researchers, whom are treated differently by the system (see below) can additionally indicate that they want this report to be of a special type, such as a report template, a tutorial, or a ToonTalk tool. These properties can later on be edited through the properties tab of the report. These properties will affect the classification of the report and its listing on the site. They will also appear on the report header. Users can also choose a template to use for their report. Templates are further discussed below.

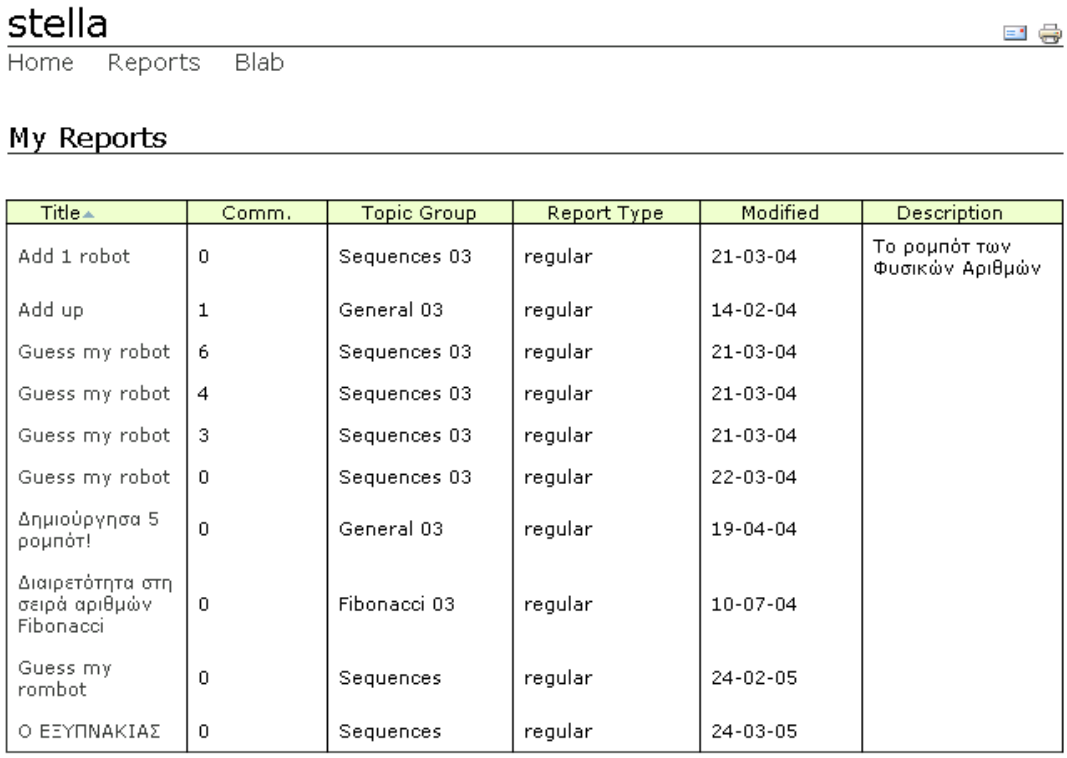


Figure 11. "My Reports". A list of all reports created by a user.

After specifying the details about the report, the user can then start editing the report proper. This can be done through a WYSIWYG editor interface, similar to the basic functionality of Microsoft Word (Figure 5). Through selecting text with the mouse and clicking the appropriate buttons, the user can easily create a text containing several levels of titles, tables and in-line images.

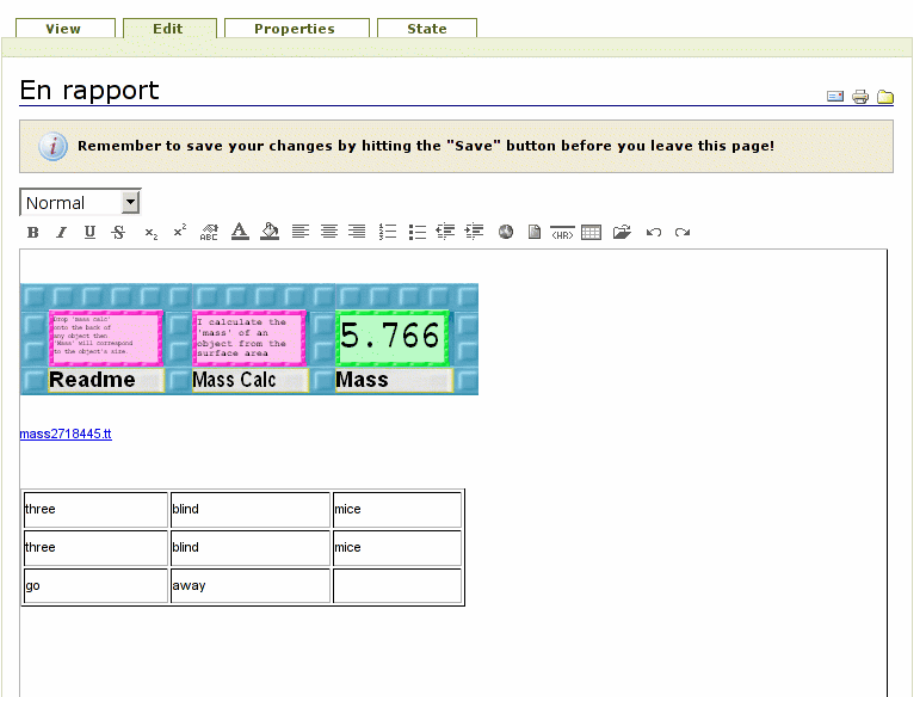


Figure 12. Editing a report through WYSIWYG. The editing buttons resemble those of common word processors.

An important part of writing reports is the possibility of including objects such as ToonTalk files, applets or images in the text. This can very easily be achieved through a standard file selector: the user selects the file she wants to include from her own local computer, and the file is automatically uploaded and included in the document. The Webreports server treats the uploaded file differently depending on the type of file:

- Images. If an image file is selected, the image will be inserted into the report.
- ToonTalk objects. If a ToonTalk object is chosen, the object will be added to the site, and an image representing the object will be inserted into the report. This image is a link, and clicking it will open the object in the user's ToonTalk environment. During the third year, the report publishing functionality has been further integrated with ToonTalk so that objects created in ToonTalk can be more directly inserted into the WebReports site from within ToonTalk.
- ToonTalk applets. A zip-file containing an applet created with ToonTalk, will be inserted into the report, so that it can be run in the web browser of other users.
- Other content. Any other file (Word files, Excel files etc) selected will be added to the site, and a text link to it will be inserted into the report.

The user can at any time save her work, which will automatically take her to the regular view of the report, showing her what the report will look like to other users. Switching between edit mode and view mode can then be done any number of times.

The third part of the report interface is the "State" tab. When the report is deemed good enough to display to others, the user can choose to publish it. Once the report is published, it will show up under the public parts of the site, under the appropriate topic and site. A published report can no longer be edited, but if the user wants to edit it again, it can be retracted through the "State" tab. Now it can be edited again, and once the editing is completed, republished.

In addition to the content added by the author, a report is always presented with a set header, which displays the topic group, owner and creation date of the report.

Tools, templates, tutorials

Researchers can, in addition to regular reports, choose to publish a report either as a template, a ToonTalk tool, guidance or a tutorial. The most interesting of these are the templates, which makes it possible for other users to take the template as a starting point for their own report. In this way, researchers and teachers can make sure that all reports published in a certain topic have a similar structure. This may serve as scaffolding for children's learning, by providing an initial structure to their reports. Templates have also been used to support "active worksheets": reports which include ToonTalk objects, task descriptions and questions. The students start their work by instantiating such a task template, and edit it to replace the questions with answers as they go along.

Guidance reports provide a way for researchers and teachers to produce guidance which can be treated separately from the reports created by the children. The tools and the tutorials are not different from regular reports, but the marking of a report as either tutorial or tool makes it show up on different places in the site, providing a way to group reports intended for a special purpose in the same location.

Blabs

Several other types of content can be added to the Webreports server in a similar way as the reports. In addition to the personal home page each user has a "Blab" (Figure 6), which is a page where short comments about the day's work can be noted, similar to the ubiquitous Blogs.

Figure 13. Example blabs

During the user evaluations during the second year of the project, we found that the Blabs functionality was only rarely used by the students, and we therefore considered removing it or modifying it substantially. However, after further project discussions we decided to keep this functionality without revisions.

Comments

Comments can be made by any user to all content published on the website. Each page - such as a report, a home page etc - has a comment button in the footer. Pressing this button takes the user to a

comment editing dialog, where comments can be edited through the same interface as other content. Comments show up at the bottom of the page. Furthermore, comments can also be made to other comments, and in this way nested threads of discussion between the users can easily be created (Figure 7). It is also possible for a user to remove own comments, or comments referring to own content.

The comment-editing dialog provides the same means as the report editing interface to include ToonTalk objects and other files in the content. This provides a convenient way for users to, for example, provide a suggestion for an improved solution for a particular problem.

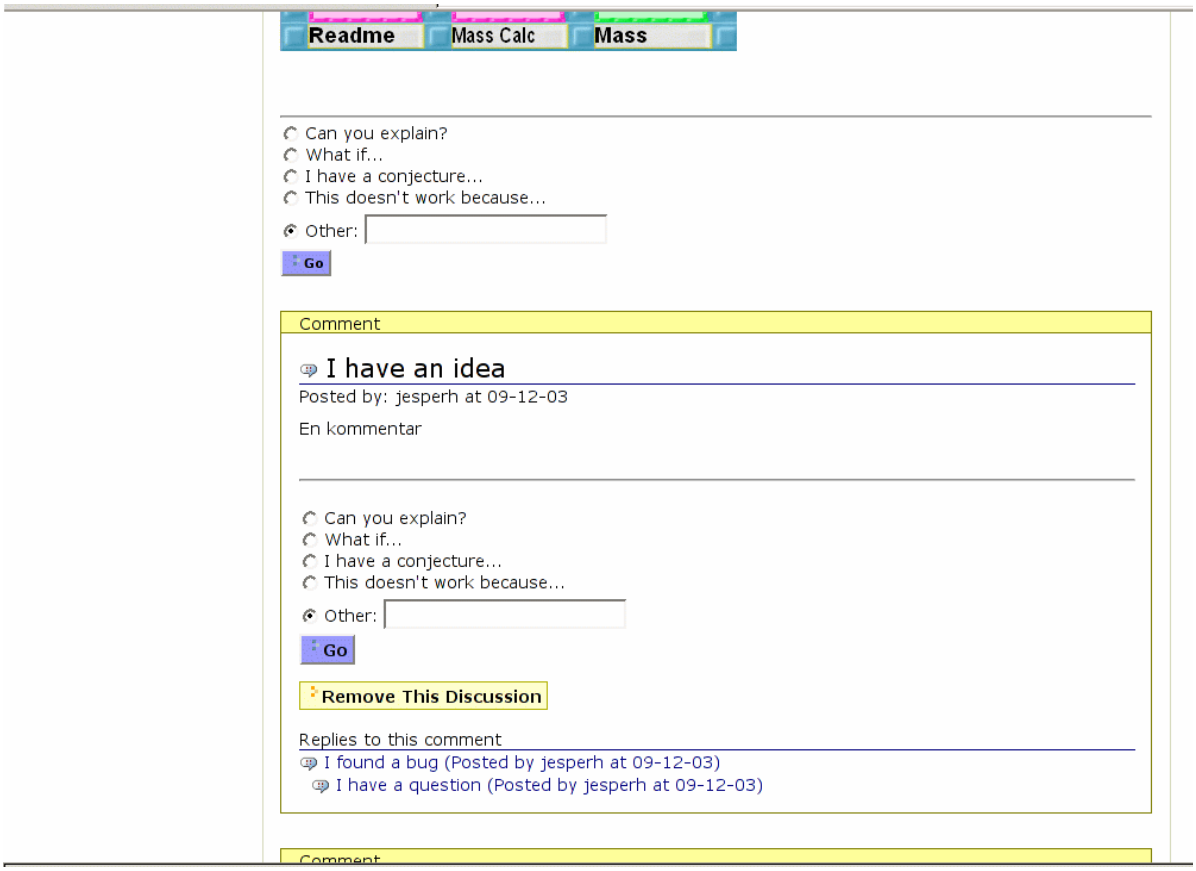


Figure 14. Adding comments to a report. The comment is attached to the report, and nested discussions can be created.

The comment system includes a set of predefined comment types, which gives the general title of the comment. These are available as radio buttons next to the comment button. The comment types provide a sort of scaffolding for the type of comments that can be made to a particular document. The chosen comment type becomes the title for the comment made. There is also the possibility to define a free title to a comment if the user does not wish to use any of the predefined types.

The owner of an object is notified through a site message (see below) whenever a user makes a comment to something she has created. This also works for nested comments, which makes it easy to keep track of the discussion around a particular topic. In all listings of reports, the number of comments made to a particular report provides a way for everyone to quickly find where on the site new discussions are taking place.

Email notification

For a couple of months, we used a system of email notifications, where email was sent out to all member of a particular topic group when a new report within that topic group was published. The intention was to make it easy to keep track of interesting additions to the site. In the end, however, the users found it too distracting to receive a large number of emails about updates, and this functionality was turned off. In its place, users are now notified by site messages when new content which might interest them is published. Currently, users are notified with personal messages only when somebody has responded to own content. All the most recently published material is instead present on the welcome page.

Group belonging

An important goal of the site was that the children should get a sense of group belonging with the other children working in the same domain, regardless of their geographic location. After the second year of the project, we realised that this did not occur spontaneously and that we needed to find ways to more actively encourage work-related communication between the children, and to stimulate the children's interest in each others' work.

The way the idea of topic groups was initially implemented proved unsatisfactory for organizing the collaboration of children working simultaneously within the same activity sequence. The domain topic group generally contained many more users than were active within a particular activity sequence at the same period of time. The domain topic groups contained users who were no longer engaged in the project, children working with other activity sequences in the same domain, as well as researchers and teachers. This led us to consider the idea of implementing support for "work groups" (see Figure 8).

To ease the collaboration between children working within the same topic at the same time, we considered various ways to specify dynamic work groups. Members of the same work group need to see the presence of other group members, thereby further facilitating communication around topics worked with during a particular period in time. The final solution for this was through manual management of the topic home pages by teachers and researchers in the project. On the topic pages, the researchers help grouping links to the children's pages, so it becomes clear to them who were to be working together. Between some sites, the teachers arranged for the children to become personal "pen pals", which was found useful for achieving a sense of community.

The Challenges

Challenges from the Istituto Comprensivo Paolo Sarpi	Author(s)	Responded by
Challenge by Lollo, Molly, Teo	Lollo - Molly - Teo	Shadow - Kaneko
Challenge by Andri, Ciufciuf, Dende	Andri - Ciufciuf - Dende	Tobias902 - Mandy
Challenge by Vale, Gio, Sogo	Vale - Gio - Sogo	Mettalicaman - Trizz - Proffesorn
Challenge by Nando, Todi, Bungia	Todi - Bungia	
Challenge by Jeka, Pepe, Je	Jeka - Pepe - Je	Amelie - Sessan - Joys

Figure 15. Small project groups listed on the "models, systems and randomness" topic page, managed by the Italian partner.

In our initial implementation of the website, there was no way for the children to distinguish researchers from children working on the site. This led the children to often try to interact with researchers - who more often were online - rather than with other children. Therefore, an important issue during the third year has been to change the group belongings of all teachers and researchers

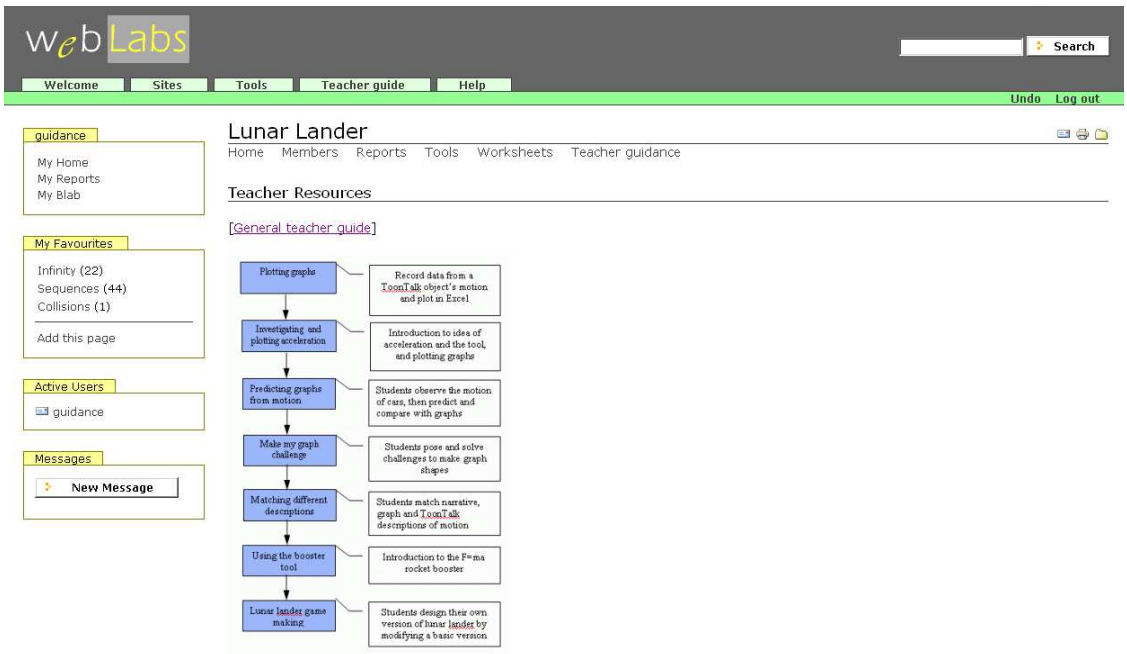
into the special group "teachers and researchers", so that only students are listed together on the school pages.

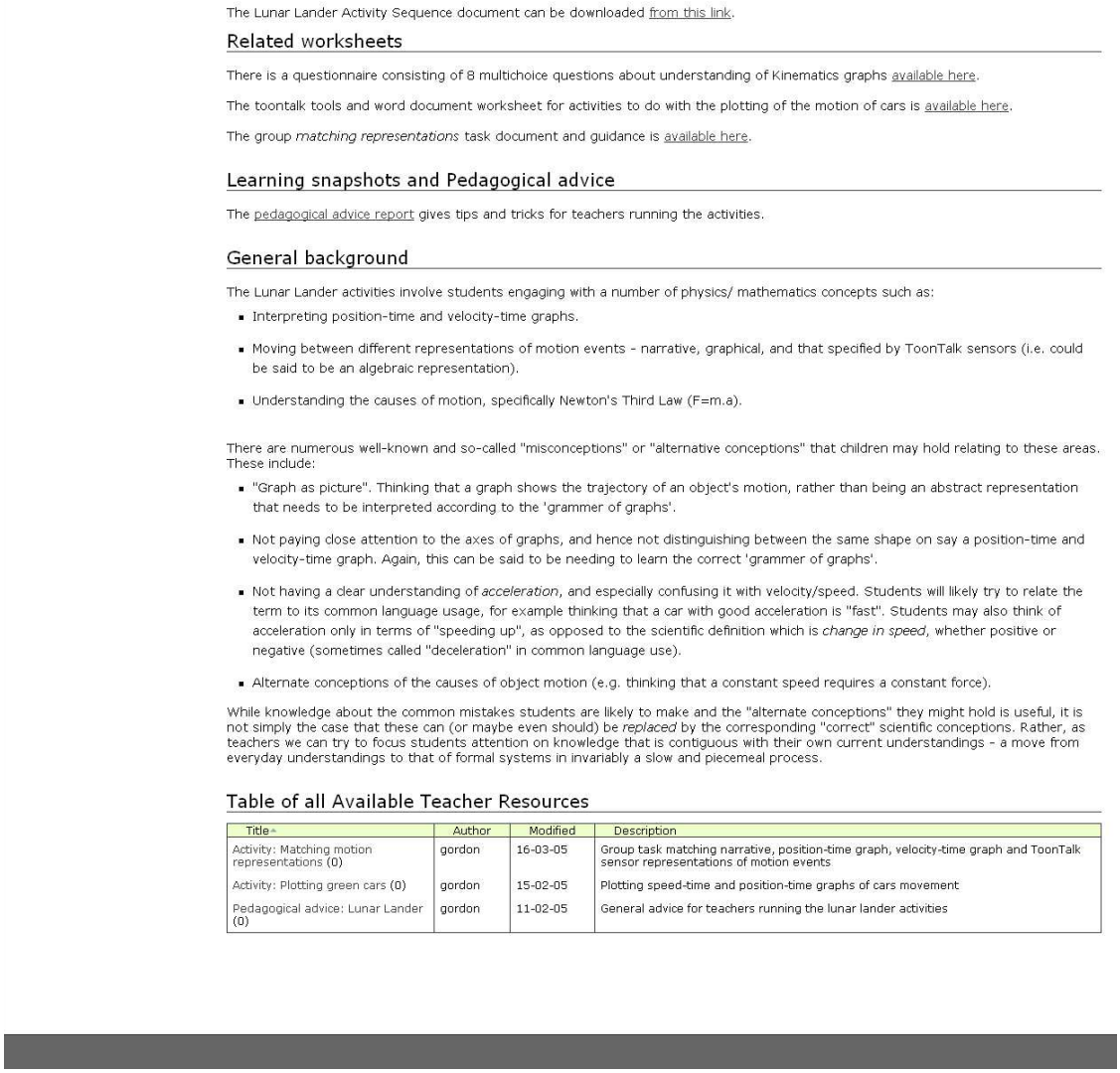
Another issue which has been discussed but not yet implemented was to make it easier to spot "group reports", i.e. reports created by a group of students. These reports will be used as a more deliberate summary of work done on a particular subject, and would therefore need to be easily distinguishable.

On-line Help

We have redesigned the topic-based 'teacher guidance' page, to lead the teacher through the various layers of guidance (see Figure 9). The Activity Sequence diagram (a concise map of the activities in that topic area) is at the top of the page (and this can be enlarged, by clicking directly over the diagram), with a link to the activity sequence (as detailed above), and below are the worksheets which relate directly to particular activities. The following section lists the learning snapshots, pedagogical advice and teaching tips relevant to the topic area. And below this is a general background to the area (as detailed above). All the guidance reports are listed in a table at the bottom of the page, for quick reference.

There are five core topic areas – Infinity, Sequences, Collisions, Lunar Lander and Models, Systems & Randomness. Sequences has two sub-levels (which follow the same structure as the top level) – Fibonacci and Convergence & Divergence. These sub-levels can be accessed by clicking on the diagram ('Fibonacci Activity Sequence' and 'Convergence and Divergence Activity Sequence') or by using the links directly below the diagram.





The Lunar Lander Activity Sequence document can be downloaded from this link.

Related worksheets

There is a questionnaire consisting of 8 multichoice questions about understanding of Kinematics graphs available here.

The toontalk tools and word document worksheet for activities to do with the plotting of the motion of cars is available here.

The group *matching representations* task document and guidance is available here.

Learning snapshots and Pedagogical advice

The pedagogical advice report gives tips and tricks for teachers running the activities.

General background

The Lunar Lander activities involve students engaging with a number of physics/ mathematics concepts such as:

- Interpreting position-time and velocity-time graphs.
- Moving between different representations of motion events - narrative, graphical, and that specified by ToonTalk sensors (i.e. could be said to be an algebraic representation).
- Understanding the causes of motion, specifically Newton's Third Law (F=m.a).

There are numerous well-known and so-called "misconceptions" or "alternative conceptions" that children may hold relating to these areas. These include:

- "Graph as picture". Thinking that a graph shows the trajectory of an object's motion, rather than being an abstract representation that needs to be interpreted according to the 'grammer of graphs'.
- Not paying close attention to the axes of graphs, and hence not distinguishing between the same shape on say a position-time and velocity-time graph. Again, this can be said to be needing to learn the correct 'grammer of graphs'.
- Not having a clear understanding of *acceleration*, and especially confusing it with velocity/speed. Students will likely try to relate the term to its common language usage, for example thinking that a car with good acceleration is "fast". Students may also think of acceleration only in terms of "speeding up", as opposed to the scientific definition which is *change in speed*, whether positive or negative (sometimes called "deceleration" in common language use).
- Alternate conceptions of the causes of object motion (e.g. thinking that a constant speed requires a constant force).

While knowledge about the common mistakes students are likely to make and the "alternate conceptions" they might hold is useful, it is not simply the case that these can (or maybe even should) be *replaced* by the corresponding "correct" scientific conceptions. Rather, as teachers we can try to focus students attention on knowledge that is contiguous with their own current understandings - a move from everyday understandings to that of formal systems in invariably a slow and piecemeal process.

Table of all Available Teacher Resources

Title	Author	Modified	Description
Activity: Matching motion representations (0)	gordon	16-03-05	Group task matching narrative, position-time graph, velocity-time graph and ToonTalk sensor representations of motion events
Activity: Plotting green cars (0)	gordon	15-02-05	Plotting speed-time and position-time graphs of cars movement
Pedagogical advice: Lunar Lander (0)	gordon	11-02-05	General advice for teachers running the lunar lander activities

Figure 9. Example of teacher guidance page.

On the top level ‘teacher guide’, there are additional general guidance materials about our pedagogic approach, as well as links to the topic specific ‘teacher guidance’ pages.

The ‘help’ page brings together all general non-teaching guidance materials. There is information and technical help on the WebReports system and ToonTalk, also some general details about the WebLabs project, and some technical tutorials about creating particular types of reports. We have removed the "Manual FAQ Tutorials About" links that previously appeared on the help pages. Now that we have a simple list of materials, we no longer need this (rather confusing) menu system.

Translation and the language barrier

The Plone architecture provides powerful support for translation of both site content and the user interface (UI). The Webreports site uses both these mechanisms to present the site in the various languages used within the project.

All text within the user interface, such as tabs, error messages and titles to the various navigation elements, are separated in text files. To translate the UI, one has only to go through the text files and provide a translation of each string. This text file in the new language can then be fed back to

the Webreports site. The user can through her browser indicate which language she prefers, and the UI will be presented to her in that language. The Webreports site, excluding content created by its users, is currently completely available in English, Portuguese and Swedish, and work is being done on the Italian and Bulgarian translations. During the third year, most translations of the site were finalised, so that the interface can be presented to the students in their own language.

Translating the content added by our users is a more difficult problem. So far, we have implemented a support for multi-language content of all group presentations and the help pages. Each document is wrapped in a language sensitive mechanism (completely transparent to the user), and this wrapper then presents the user with the appropriate version of the selected document. In this way, users from Portugal can read the presentation of their group in Portuguese, whereas the users in Sweden are presented with the same page in Swedish. As long as the researchers and teachers have provided these different versions, the change of language is completely automatic.

However, the issue of translation has still been a major issue in children's communication through the website. The children naturally create reports most easily in their own native language, and in order to facilitate cooperation between children from different sites, some translations of reports have been provided by the researchers. This has however not worked out as well as one could hope in all cases. The language barrier is a major obstacle in creating a sense of community between the children working with the same activity sequence. Various ways of providing translations have been tried out, but we have not yet found a method where the children actually perceive the translated text from another child to communicate with. The main obstacle, however, is not the effort that teachers and researchers have to put into the actual translation process. Instead, from our point of view problems seems to be that when a report or a comment is translated by two different adults (e.g. from Swedish to English and from English to Italian) the children seem to lose their sense of ownership in the report. The translation process makes the reports something that is more important for the adult than the child. The children prefer to use the means the limited knowledge of any foreign language that they know rather than being provided with a report perfectly translated by an adult. One solution has been to design activity sequences so that they do not necessarily require communication through written language, but instead provide possibilities of communicating through the models that children build. Indeed, in the "Guess my Robot" and "Guess my Graph" activities, the object of discussion was really centred around ToonTalk robots and Excel graphs respectively – objects that are virtually language independent. It is no coincidence that these have been among the most successful activities with regard to promoting cross-site interactions.

Access control

The WebReports website has a sophisticated support access control to different parts of the site. First of all, as part of the registration process, a new user must register with an individual password. In addition to this, the new user must also know our project password, which has to be provided by a teacher or researcher, a security measure we've taken to ensure that only people who in one way or another are associated with our project can be accepted as users.

Much care has been taken to ensure that users cannot accidentally or intentionally harm the work of others. Each user can by default only edit the content she has created herself. An owner can however choose to grant other users status as co-owners of selected parts of her own content.

Researchers can do several things that kids cannot do on the site: add ToonTalk tools, tutorials, guidance materials and templates, and edit the presentations of the different topic groups and sites, as well as the help section.

In order to protect users' privacy, some of the information present on the site is restricted to registered users in the following way:

- The personal information such as email address etc is not visible to non-registered users.
- The user portraits are only visible to other registered users.

During the third year, we added the topic pages to be visible for all users, so that curious visitors could check the published material and overall content of the different topic groups, without gaining access to the pages of individual children.

Data Mining

Each time a report is published, a full copy of it is stored away and backed up, providing a way for the researchers to study the way content on the site has been developed and refined.

Prior to the third year, we had plans for implementing better support for statistics generation and for tagging content with “meta-data”. The statistics generation would allow the researchers to mine the data for interesting information. This would include usage statistics, as well analysis of content and communication on the site. In order to facilitate the researchers’ navigation of the content on the site, we explored the possibility to tag content with meta-data, allowing more refined classification of children’s contributions. However, after discussions with the researchers involved in the project, it was found that research needs for such kinds of analysis were weak, hence no actual support for such “data mining” the contents have been implemented.

Other minor changes made during year 3

- New colour scheme of the website, to better match the project content and logo.
- Default portrait pictures for users were changed from the standard Plone icon showing a silhouette of an adult man, to instead show the ToonTalk picture of Marty. The default school portrait was changed into a ToonTalk house. This was partly to visually connect the website to the activities in ToonTalk, and also to emphasise the young age of the target users.
- Rearrangement of the subsection on the topic home pages, e.g. change of 'Guidance' to 'Teacher Guidance', and cosmetic changes to how the subsections for each topic group were presented.
- Restructuring of the content on the tools page
- Major changes to the presentation of content on the help/guidance section, as outlined earlier
- Picture of school automatically added at the top of the welcome page for each school.