Concept Mapping: Renewing Learning and Thinking
Proc. of the Eighth Int. Conference on Concept Mapping
Medellín, Colombia 2018

ECMAP: AN EMBEDDABLE WEB-BASED CONCEPT MAP EDITOR

Alberto J. Cañas, Roger Carff, & James Lott
Institute for Human & Machine Cognition (IHMC), USA
Email: {acanas, rcarff, jlott}@ihmc.us
www.ihmc.us

Abstract. The embeddable concept map editor eCmap is part of the CmapTools software suite that enables the embedding of a Cmap editor into a host Web page. By providing a set of flexible APIs to the host page, different sets of functionalities can be made available to the user depending on the needs of the application where eCmap is embedded. eCmap is designed to be part of the CmapTools network and can fully interact with CmapServers and the Cmap Cloud. In this paper we describe eCmap, and we present several sample cases of its use, including CmapTools in the Cloud (the Web-based version of CmapTools), the CmapViewer, an integration to implement Cmap-based test items in online tests, and a Web page to evaluate Cmaps automatically.

Keywords*:* CmapTools, concept map, Cmap, editor, Cmap Cloud, embeddable, eCmap

1 Introduction

Since their creation in the 1970s, the use of concept maps (Novak & Gowin, 1984) has grown extensively throughout the world in all domains of knowledge, by users of all ages and for all kinds of applications (Novak & Cañas, 2010). The construction and manipulation of concept maps (Cmaps) is of course simplified when supported by the use of technology, just as a word processor facilitates the writing of text. More so, in our research efforts we have found that the combination of concept mapping with technology such as the Internet and WWW, as exemplified in the software suite we developed, CmapTools (Cañas et al., 2004) , has significantly extended the applicability and use of concept mapping, and we are always surprised by the innovative uses that CmapTools users have found for the software.

Novak & Cañas (2010) asked themselves and the Cmappers community: Why aren't concept maps more ubiquitous? Even with the emergence of concept mapping tools – and there are many more available nowadays, particularly Web-based – it is not easy to embed a concept map editor in any Web page as one can embed a text editing box. Or to embed an interactive concept map with links to resources into a site. In this paper we describe eCmap, an embeddable concept map editor that is part of the CmapTools software suite, and present several cases that show how eCmap has been incorporated into different types of applications.

2 The CmapTools Network

The CmapTools Network consists of a set of client and server programs that work together to facilitate the construction, publishing and sharing of concept maps (Cmaps) and resources (Cañas, Hill, Granados, Pérez, & Pérez, 2003). CmapTools client programs access CmapServers to store, retrieve and share Cmaps, videos, images, documents, links to Web pages, etc. and integrate them into Knowledge Models (collection of Cmaps and related linked resources on a particular domain). The embeddable Cmap Editor eCmap is another client program within the CmapTools network.

3 eCmap: The Embeddable Cmap Editor

eCmap is the component of the CmapTools software suite which enables the embedding of a concept map editor into a web page. With this client-side, JavaScript / HTML5 library[1], a Cmap editor can easily be embedded, by calling a single function to initialize the editor and attach it to an element on the page. Multiple eCmaps may be embedded within a single page, attached to unique elements.

With eCmap, users can create and edit a Cmap in the same fashion (with the same gestures) as they do in the CmapTools desktop client (see Figure 1). Double-clicking on the background creates a concept. Selecting the concept displays the link creation icon which can be dragged to create a linking phrase to connect concepts. Double-clicking

[1] Code is developed in Java and compiled to JavaScript with Google Web Toolkit (GWT).

a concept or linking phrase allows for editing text. Selected objects can be moved with the mouse. There are also cut, copy, paste, undo and redo capabilities accessible from a right-click context menu. Since concepts may contain resource links, which users are able to click on and open, the library is also capable of displaying these in new browser windows or tabs or, if a callback handler has been registered, pass the link info to the handler to display.

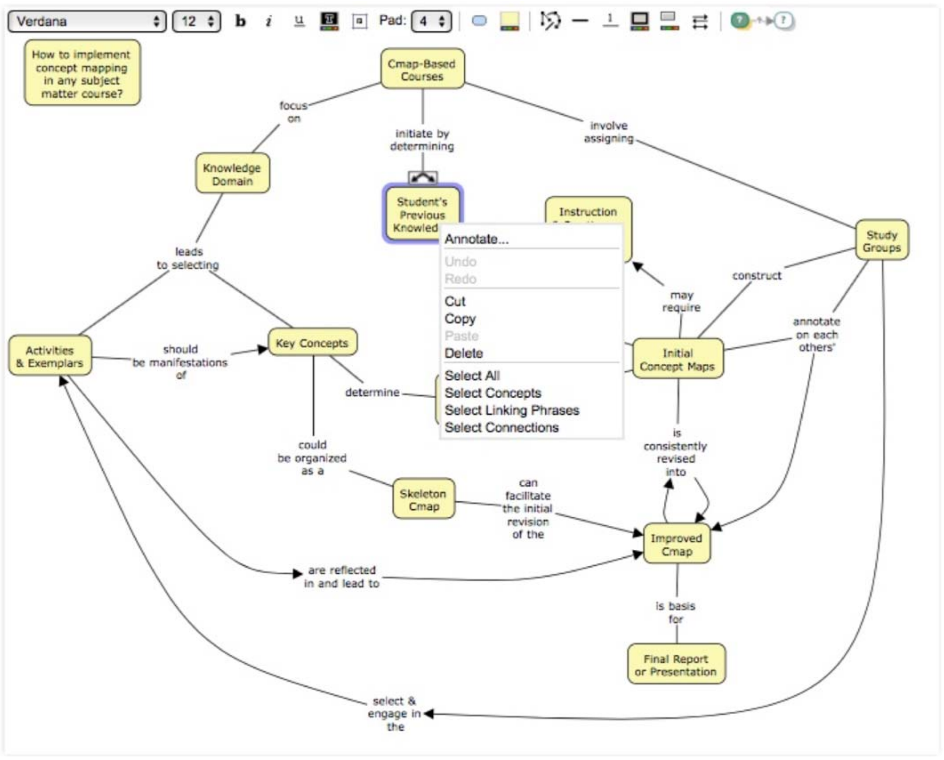


Figure 1. The eCmap embedded concept map editor, showing the toolbar at the top, and an example of right-click menu.

A toolbar is available for manipulation of the Cmap's style (Figure 1), in the same way as the Style Palette in the desktop version of CmapTools. Selecting objects in the Cmap and then selecting different icons in the toolbar will affect the selected objects. The toolbar may also be hidden (e.g. in view-only mode).

The eCmap library supports several different modes of editing: whether the Cmap should be viewable only, annotatable only (users can add Annotations to the Cmap) or fully editable. The Cmap can also be scaled to fit the size of its container object or displayed in a scrollable view at full size.

When initializing eCmap, a URL may be specified to load an existing Cmap into the editor. This URL may refer to a Cmap in a CmapServer, as eCmap is frequently used in conjunction with a CmapServer for Cmap and Resource storage. The eCmap library includes support for communication with a CmapServer, either directly or through a proxy (see Cmap Cloud section for a proxy example). In addition, Cmaps may be loaded from local storage (e.g. a file:// URL) or imported/exported as CXL through the eCmap API, allowing for integration with any storage system supported by the host environment.

eCmap is also integrated with Google Analytics. If the web page containing the embedded editor has Google Analytics initialized, then the library will register success/failure events when resources are opened, closed, and saved. It will also record Cmap level change events such as concept added/deleted, linking phrase added/deleted, and resource link opened.

Currently eCmap supports the majority of the functionality of the desktop version of CmapTools, including drag and drop of resources from the host page to concepts in the Cmap, Annotations, and style-related features such as fonts, colors and background images. However, some advanced features are only supported by the desktop version of CmapTools, including Presentation Mode, Recorder, Nested Nodes, and Discussion Lists.

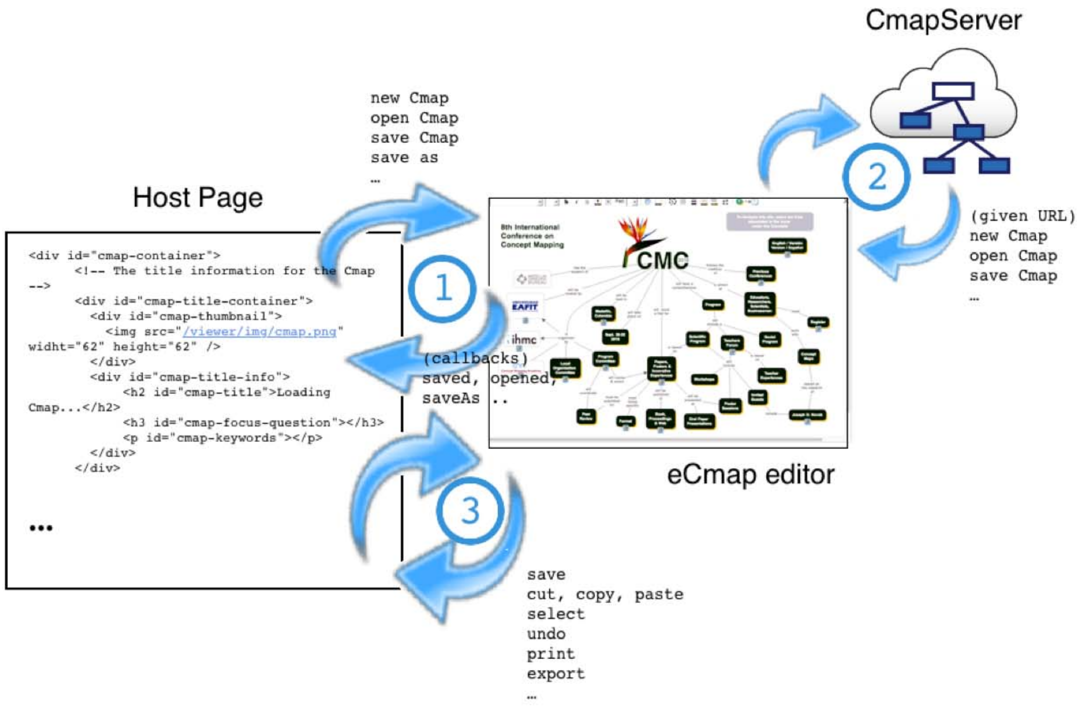
3.1 API Details

To simplify the description, we depict eCmap as having three APIs, as shown in Figure 2.

3.1.1 eCmapEditor API

Through the eCmapEditor API (#1 in Figure 2), eCmap exposes methods which allow the host page to initialize and interact with the embedded concept map editor. This API includes functions for creating, loading, and saving Cmaps, as well as getting and setting the Cmap's metadata and CXL. This API also includes a callback-style architecture, so eCmap can interact with its host environment. eCmap uses these callbacks as feedback functions to let the host page know that a Cmap or resource has been opened, or saved, or is being saved, or an error has occurred; the host page can choose to display this feedback as appropriate.

The eCmapEditor API also allows some overriding of functionalities that eCmap provides internally. This enables the host page to further customize the behavior and look-and-feel, by registering callback handlers for certain events such as: requests to show resource links, requests to edit resource link information, and dialogs for saving Cmaps. For example, if the host page wants to customize the behavior of the Save Cmap dialog, it registers as a ViewsOperations callback handler. When a save is to take place, the Cmap's info is passed to the showSaveDialog callback function. The host page then does whatever it wants with the information. It could simply show a custom Save dialog for the environment's look-and-feel, in place of eCmap's built-in Save dialog. When the user clicks save, the metadata can be passed back to eCmap which then deals with saving the Cmap and its parts back to the CmapServer. Or, instead of saving to a CmapServer, the host page could just ask eCmap for the CXL of the Cmap and upload it, for example, to Google Docs if desired.



```
<div id="cmap-container">
      <!-- The title information for the Cmap
-->
      <div id="cmap-title-container">
        <div id="cmap-thumbnail">
          <img src="/viewer/img/cmap.png"
widht="62" height="62" />
        </div>
        <div id="cmap-title-info">
              <h2 id="cmap-title">Loading
Cmap...</h2>
              <h3 id="cmap-focus-question"></h3>
              <p id="cmap-keywords"></p>
        </div>
      </div>

•••
```

Figure 2. Through APIs eCmap interacts with the Host Page and CmapServer.

3.1.2 eCmapServer API

When Cmaps are stored on a CmapServer, eCmap uses the eCmapServer API (#2 in Figure 2) to retrieve Cmaps in CXL format, save Cmaps, and export Cmaps to CXL, JPEG and SVG formats. In general, this API includes functions for managing resources and folders on a CmapServer, including uploading, creating, copying, deleting, renaming, moving Cmaps and resources, retrieving thumbnails and folder lists, and dealing with permissions. This interaction can be directly with the CmapServer or through a Proxy, as we'll see in the examples later in this paper. Once the operation is completed, eCmap would use the callback mechanism to let the host page know if it was successful or not. An additional feature of the eCmapServer API is that it is also available to the host page. This is demonstrated by the Cmap Cloud (and other examples), described in section 4, which can show a user's current list of resources and folders and allows the user to rename, delete, move, and create resources.

3.1.3 eCmapActions API

The eCmapActions API (#3 in Figure 2) is the mechanism by which eCmap provides the host page with a set of Actions relevant to the Cmap being edited, along with their enabled state. These actions include operations like Select All, Select Concepts, Select Linking Phrases, Select Connectors, Cut, Copy, Paste, Undo, Redo, Change Properties, Print, Export (in several formats). It's optional whether to display all of the Actions or not, as we'll see in the examples below. The host page can show all of the actions to the user (e.g. in a menu integrated with the look and feel of the page), or it may decide to customize the actions available to the user, and only display certain actions (perhaps as buttons in a toolbar). When the user selects an Action (dependent on how the actions are displayed), the host page can notify the action's callback function, and the action will be handled by eCmap.

4 Examples of Embedding eCmap

The flexibility in the design of eCmap together with the integration with the CmapTools architecture has facilitated its embedding in many unique environments. In this section we describe several of these use-cases.

4.1 Cmap Cloud

As Internet connections get faster and Web browsers more capable, counterparts for desktop applications have become more common and capable, and for many users, have replaced the desktop applications, as exemplified by Google Docs, Office 365, simplifying sharing and collaboration and the construction of simple documents. Developing a CmapTools editor that would run on a Web browser made sense.

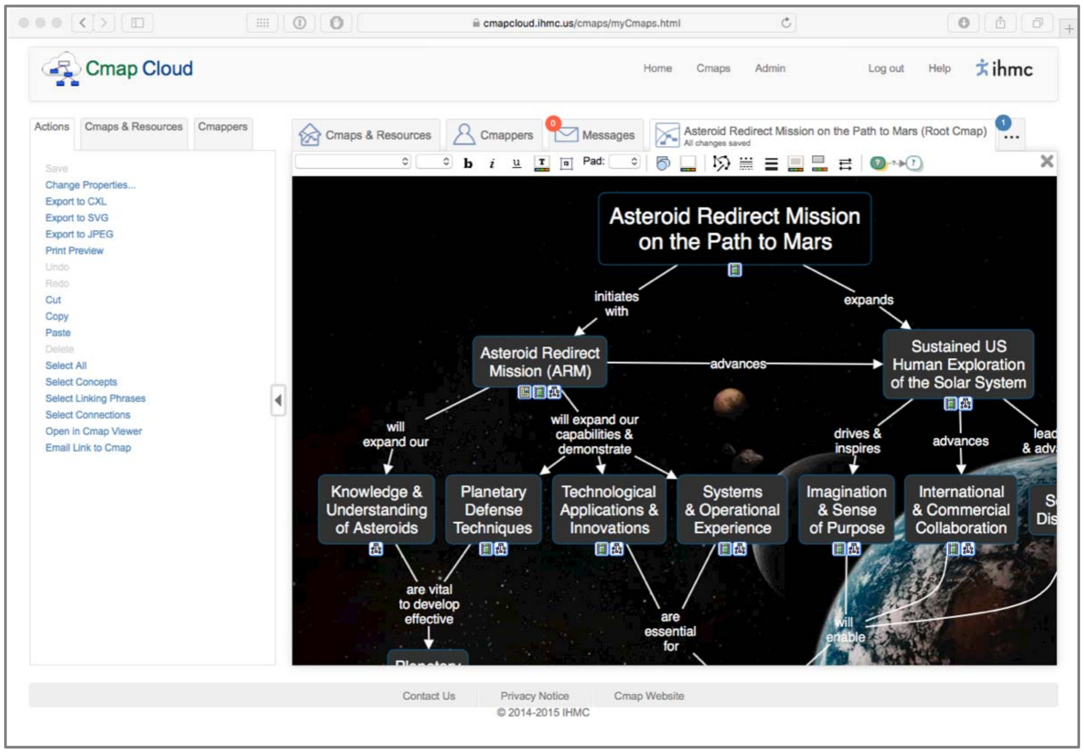

cmapcloud.ihmc.us/cmaps/myCmaps.html
Cmap Cloud
Home
Cmaps
Admin
Log out
Help
ihmc
Actions
Cmaps & Resources
Cmappers
Save
Change Properties...
Export to CXL
Export to SVG
Export to JPEG
Print Preview
Undo
Redo
Cut
Copy
Paste
Delete
Select All
Select Concepts
Select Linking Phrases
Select Connections
Open in Cmap Viewer
Email Link to Cmap
Cmaps & Resources
Cmappers
Messages
Asteroid Redirect Mission on the Path to Mars (Root Cmap)
All changes saved
Pad:
Asteroid Redirect Mission on the Path to Mars
initiates with
expands
Asteroid Redirect Mission (ARM)
advances
Sustained US Human Exploration of the Solar System
will expand our
will expand our capabilities & demonstrate
drives & inspires
advances
Knowledge & Understanding of Asteroids
Planetary Defense Techniques
Technological Applications & Innovations
Systems & Operational Experience
Imagination & Sense of Purpose
International & Commercial Collaboration
are vital to develop effective
are essential for
will enable
Contact Us
Privacy Notice
Cmap Website
© 2014-2015 IHMC

Figure 3. CmapTools in the Cloud incorporates eCmap in a tabbed environment, with Actions list on the left to operate on the opened Cmap.

Building on the existing CmapServer and the CmapTools Network Architecture, we implemented the Cmap Cloud, a centralized sharing site for users to store and share their Cmaps, resources and Knowledge Models and access them from the three versions of CmapTools: (a) CmapTools (for Desktop), (b) CmapTools for iPad, and (c) CmapTools in the Cloud (the Web version of CmapTools). Through their personal accounts, users can synchronize CmapTools for the iPad with the root of their Home folder in the Cmap Cloud. Using the desktop version of CmapTools, users can access the Cmap Cloud resources in their Home folder, from the Views window. And through CmapTools in the Cloud, which we implemented using eCmap, the user can access and share resources and edit them using a Web browser.

For CmapTools in the Cloud, eCmap is embedded in a tab-based view which supports opening multiple Cmaps and quickly switching between them. The site uses the eCmapActions API, described above, to display context-sensitive menu items (Actions) in a panel on the left (see Figure 3), as well as the eCmapEditor API to register ViewsOperations callbacks for handling opening linked resources (either in an internal tab or through the browser/OS), showing feedback to the user (saving, error), and managing drag and drop of resources to the editor. In addition, the Cmaps and Resources tab (Views) uses the eCmapServer API to communicate with the CmapServer, to show the contents of folders and allow operations on resources (create, delete, copy, move, edit properties, etc.), similar to the desktop version of CmapTools.

To support single-sign-on (SSO), the Cmap Cloud site enables eCmap's proxy option, which re-directs all requests to the CmapServer to route through a proxy running on the Cmap Cloud webserver. This allows the Cmap Cloud site to attach the user's credentials to CmapServer requests, and then forward the requests to the CmapServer; when it receives a response, it is forwarded back to eCmap. Recently we completed the implementation of OAuth2 on the CmapServer, and intend to modify eCmap to take advantage of it instead of using the proxy server.

CmapTools in the Cloud has exercised eCmap's functionality, as it has grown to over 518,000 registered users, and has become very popular, in particular with academic students, faculty, and researchers.

4.2 Cmap Viewer

The CmapViewer is a software utility to view and embed Cmaps that are stored in CmapServers. It is the default Web viewer for Cmaps stored in the Cmap Cloud, the IHMC Sample Knowledge Models Place, and several other IHMC CmapServers. Implemented as an add-on for the CmapServer, it replaces the internal viewer; the CmapViewer JavaScript code is automatically loaded by the user's browser whenever they open a URL to a Cmap. Internally, the CmapViewer uses the eCmap code to render the Cmap, in a view-only mode without any toolbars. It is not part of the standard downloadable CmapServer and its therefore not available on all CmapServers.

Figure 4 shows how the Cmap in Figure 1 is displayed by the CmapViewer with its default settings, including the Cmaps' name as the Title, the Cmap's Focus Question, and icons for Annotate, Save and Embed. As can be seen, the CmapViewer code is itself the host page for eCmap, and has implemented the Annotate and Save button for the user to interact with the Cmap. The Annotate button allows the user to make Annotations on the Cmap (permission to Annotate is needed), and the Save button saves the annotations.

The Embed button shows how to embed the CmapViewer displaying this Cmap in another web page (and thus embed eCmap). This button is also shown when a Cmap being edited in the desktop version of CmapTools is stored in a CmapServer that has the CmapViewer installed, as shown in the lower right of Figure 5. Clicking on the icon displays the dialogue shown in Figure 5. Clicking on the icon in the CmapViewer displays a similar dialogue. The options selected[2] determine the HTML code, needed to embed the Viewer, which is displayed for the user to copy and include in their Web page. Notice that allowing Annotations is an option. We have used the CmapViewer in a number of sites, including the Cmaps displayed in the Concept Mapping Conference (Cmap) website[3] to display each CMC's Cmaps.

[2] More details on how to embed the CmapViewer are available at: https://cmap.ihmc.us/docs/cmapviewer-client.php
[3] CMC's website: http://cmc.ihmc.us.

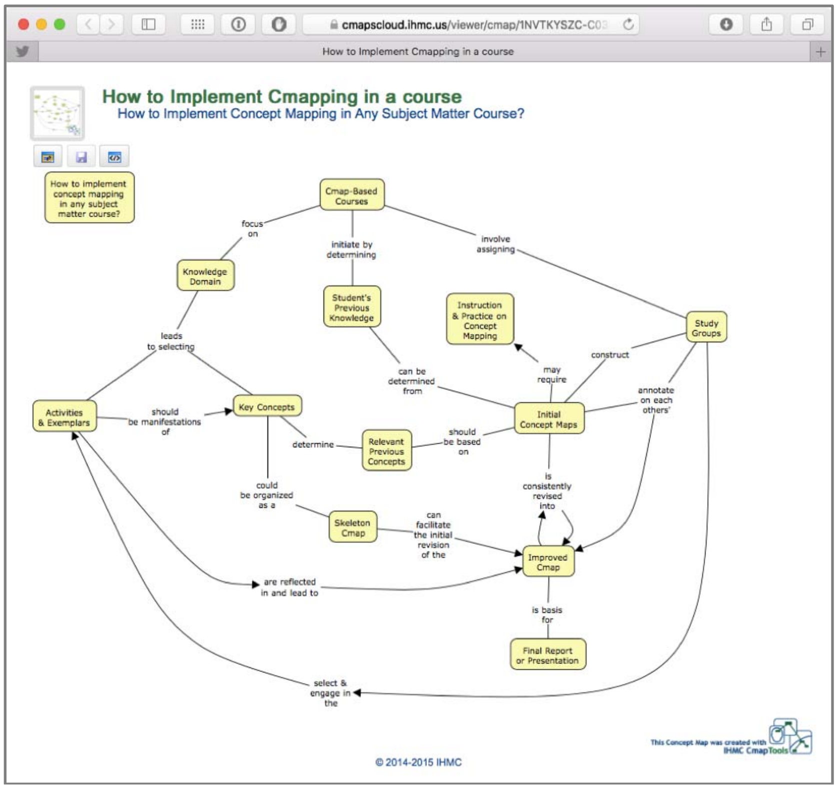
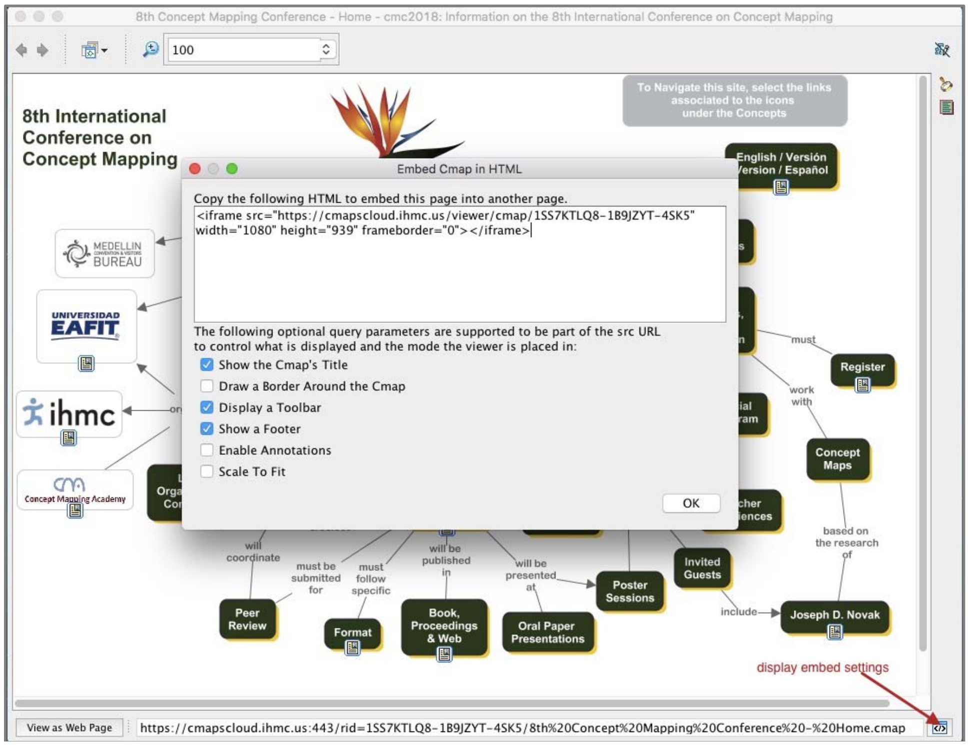

cmapscloud.ihmc.us/viewer/cmap/1NVTKYSZC-C0
How to Implement Cmapping in a course
How to Implement Cmapping in a course
How to Implement Concept Mapping in Any Subject Matter Course?
How to implement concept mapping in any subject matter course?
Cmap-Based Courses
focus on
initiate by determining
involve assigning
Knowledge Domain
Student's Previous Knowledge
Instruction & Practice on Concept Mapping
Study Groups
leads to selecting
construct
can be determined from
may require
annotate on each others'
Activities & Exemplars
should be manifestations of
Key Concepts
Initial Concept Maps
determine
Relevant Previous Concepts
should be based on
could be organized as a
is consistently revised into
Skeleton Cmap
can facilitate the initial revision of the
Improved Cmap
are reflected in and lead to
is basis for
Final Report or Presentation
select & engage in the
This Concept Map was created with IHMC CmapTools
© 2014-2015 IHMC

Figure 4. The Cmap in Figure 1, as displayed by the CmapViewer with its default settings



Copy the following HTML to embed this page into another page.

```
<iframe src="https://cmapscloud.ihmc.us/viewer/cmap/1SS7KTLQ8-1B9JZYT-4SK5"
width="1080" height="939" frameborder="0"></iframe>
```

The following optional query parameters are supported to be part of the src URL to control what is displayed and the mode the viewer is placed in:

- ✓ Show the Cmap's Title
- ☐ Draw a Border Around the Cmap
- ✓ Display a Toolbar
- ✓ Show a Footer
- ☐ Enable Annotations
- ☐ Scale To Fit

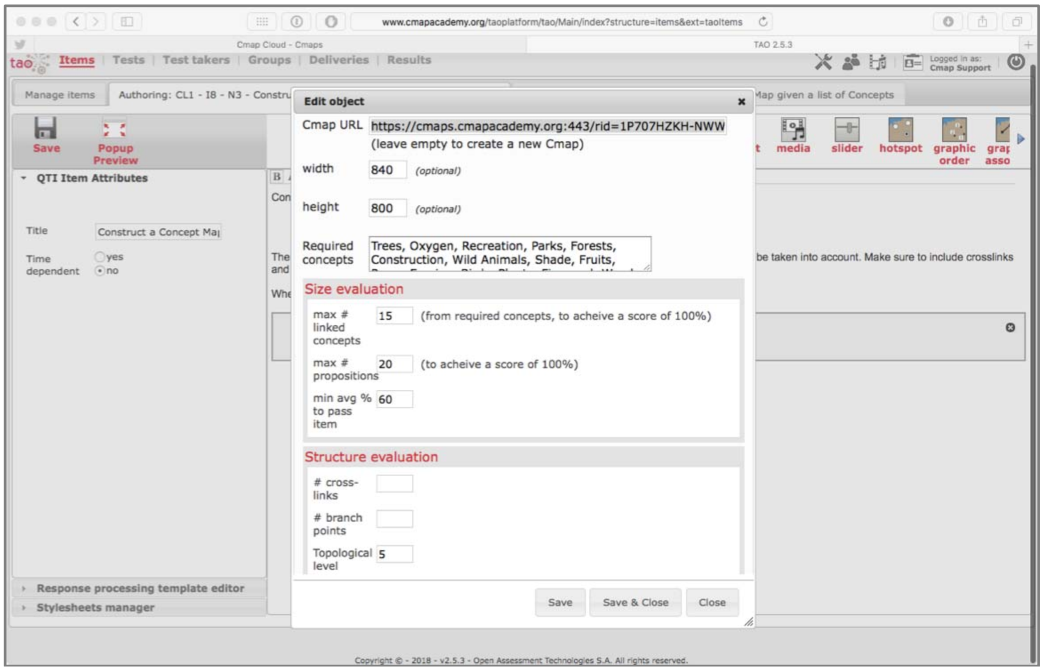
Figure 5. Dialogue to construct the HTML code necessary to embed a Cmap using the CmapViewer.

4.3 Cmap-based Test Items

As part of the Concept Mapping Academy website, which is described below, we needed test items that involved having the test-taker construct or modify a Cmap, and for the system to evaluate the correctness of the response. For this purpose, we decided to use TAO[4], a Web-based open-source architecture for creating, delivering, and evaluating tests. We embedded eCmap into the TAO architecture, which allowed embedding Cmaps into TAO test items. In addition, we integrated CmapAnalysis (Cañas, Bunch, Novak, & Reiska, 2013), a software library for assessing Cmaps against evaluation criteria, into TAO's test evaluation process, to provide an end-to-end solution for using Cmaps as a method of testing and evaluation.

In the TAO test item editor, we added a button to the toolbar which embeds eCmap into a test item. Clicking on the placeholder for the editor brings up a settings dialog which allows the user to specify the evaluation criteria, as shown in Figure 6. The test author must provide the URL of the Cmap (which may be stored in a CmapServer), the size at which the Cmap should display, and fill in a set of criteria. In the example in Figure 6, a set of concepts are required to be present in the Cmap, which should also have a maximum of 15 concepts (from the list), 20 propositions, and a topological (Cañas et al., 2006) level of 5. The item now goes into TAO's items bank, and can be incorporated into any TAO test. After the user completes a Cmap test item, it is evaluated by the CmapAnalysis library, and the results are stored in the TAO test results database.

We have used the Cmap TAO items in the Concept Mapping Academy website described in the next section.


www.cmapacademy.org/taoplatform/tao/Main/index?structure=items&ext=taoItems
Cmap Cloud - Cmaps
TAO 2.5.3
tao
Items | Tests | Test takers | Groups | Deliveries | Results
Logged in as: Cmap Support
Manage items
Authoring: CL1 - I8 - N3 - Constru
Map given a list of Concepts
Save
Popup Preview
media
slider
hotspot
graphic order
grap asso
QTI Item Attributes
Response processing template editor
Stylesheets manager
Save
Save & Close
Close
Copyright © - 2018 - v2.5.3 - Open Assessment Technologies S.A. All rights reserved.

Title: Construct a Concept Ma
Time dependent: yes [✓ no]

be taken into account. Make sure to include crosslinks

Edit object

Cmap URL: https://cmaps.cmapacademy.org:443/rid=1P707HZKH-NWW
(leave empty to create a new Cmap)

width: 840 *(optional)*

height: 800 *(optional)*

Required concepts: Trees, Oxygen, Recreation, Parks, Forests, Construction, Wild Animals, Shade, Fruits,

Size evaluation

max # linked concepts: 15 (from required concepts, to acheive a score of 100%)

max # propositions: 20 (to acheive a score of 100%)

min avg % to pass item: 60

Structure evaluation

cross-links:

branch points:

Topological level: 5

Figure 6. Configuring a test item in TAO that will present the test-taker a Cmap using eCmap.

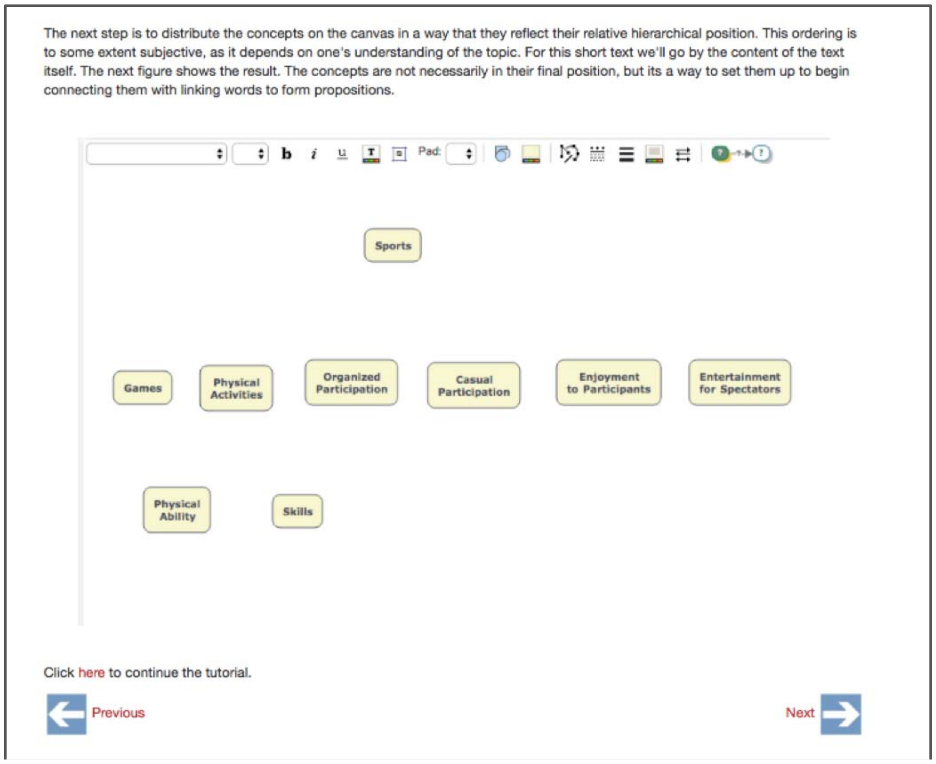
4.4 Concept Mapping Academy Tutorials and Certification Tests

The Concept Mapping Academy's website is a work-in-progress effort to offer concept mapping tutorials and a Professional Certification Program on concept mapping. As part of the tutorials on concept mapping, several Cmap-based test items are offered in between the tutorial text to test the learners' understanding of the topics being covered.

Figure 7 shows the result of embedding a TAO test, that consists of a single Cmap test item, into the middle of a tutorial page. In this test item the learner is expected to construct a Cmap with the concepts provided. As can be seen in the Figure, this results in eCmap being used to display the Cmap retrieved from the CmapServer. Once the user

[4]TAO is available at https://www.taotesting.com.

completes the Cmap, it's assessed by the CmapAnalysis code according to the criteria selected when the test item was constructed, and if the response is correct the user can continue, otherwise he or she is asked to try again with another, similar item.


The next step is to distribute the concepts on the canvas in a way that they reflect their relative hierarchical position. This ordering is to some extent subjective, as it depends on one's understanding of the topic. For this short text we'll go by the content of the text itself. The next figure shows the result. The concepts are not necessarily in their final position, but its a way to set them up to begin connecting them with linking words to form propositions.

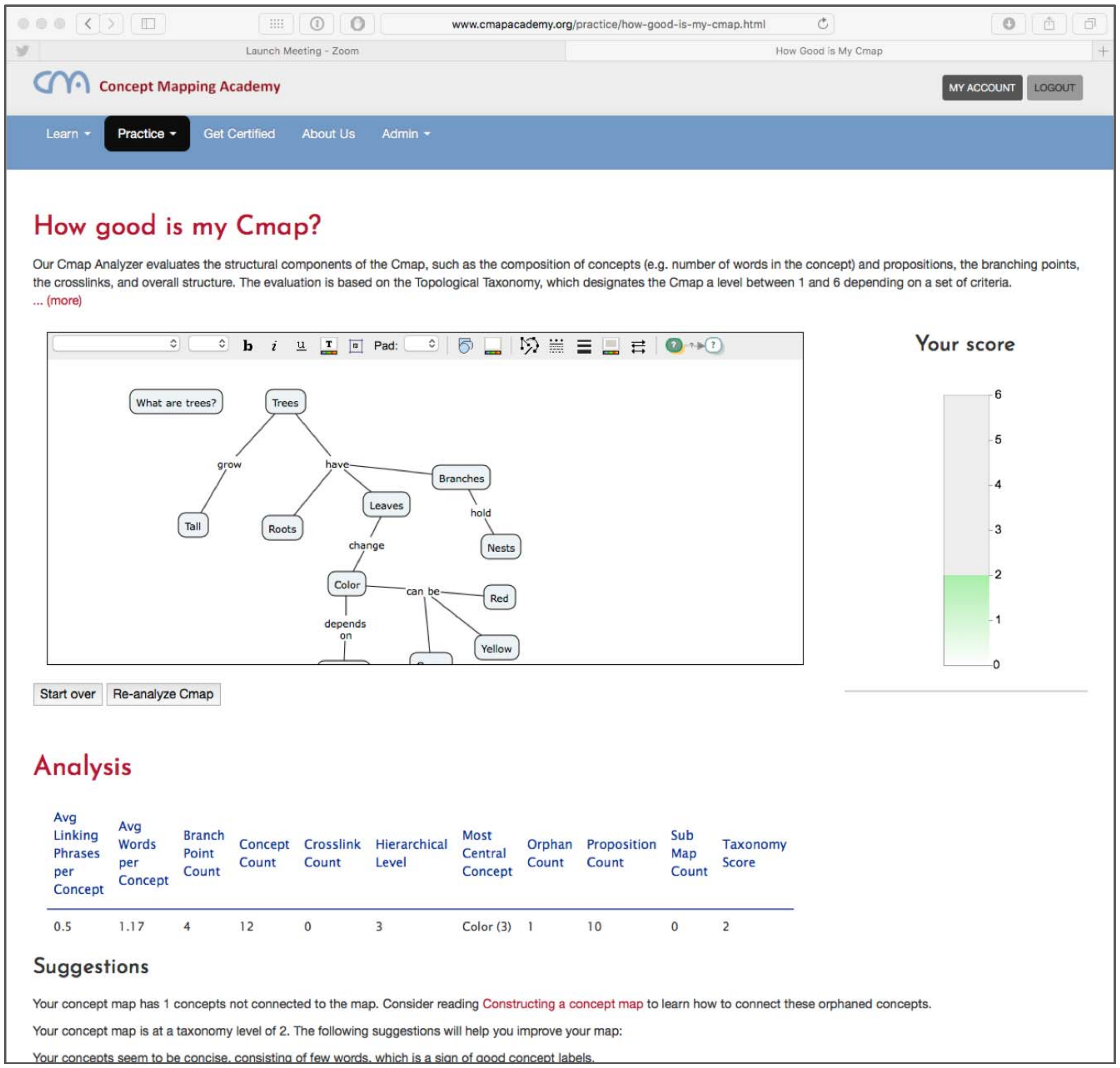
Click here to continue the tutorial.

Figure 7. A TAO test consisting of a single Cmap test item is displayed as part of a concept mapping Tutorial page at the Concept Mapping Academy website.

4.5 *How Good is My Cmap?*

The last example we will present is the "How Good is My Cmap?"[5] webpage of the Concept Mapping Academy website. This page allows you to analyze a concept map you have built with CmapTools and evaluate its ranking according to the Topological Taxonomy, using the CmapAnalysis library previously mentioned. In addition to the taxonomy score, the analysis results include several additional metrics along with feedback on how to improve your Cmap, and provide links to sections of the tutorials related to the suggestions. You can upload an existing Cmap in CXL or Java-binary format, provide the URL for a Cmap stored in a CmapServer, or construct your own Cmap. To display the supplied Cmaps, or to allow the users to construct their own Cmaps, eCmap is embedded into the webpage, as shown in Figure 8. This example demonstrates how the host page can pass the uploaded or retrieved Cmap to eCmap for display, or retrieve the constructed Cmap from eCmap for evaluation. The APIs provided by eCmap allows for significant flexibility on how the host page interacts with the eCmap.

[5]Although the rest of the website is not yet completed, the "How Good is My Cmap?" page is functional https://www.cmapacademy.org/practice/how-good-is-my-cmap.html



How good is my Cmap?

Our Cmap Analyzer evaluates the structural components of the Cmap, such as the composition of concepts (e.g. number of words in the concept) and propositions, the branching points, the crosslinks, and overall structure. The evaluation is based on the Topological Taxonomy, which designates the Cmap a level between 1 and 6 depending on a set of criteria.
... (more)

Analysis

Avg Linking Phrases per Concept	Avg Words per Concept	Branch Point Count	Concept Count	Crosslink Count	Hierarchical Level	Most Central Concept	Orphan Count	Proposition Count	Sub Map Count	Taxonomy Score
0.5	1.17	4	12	0	3	Color (3)	1	10	0	2

Suggestions

Your concept map has 1 concepts not connected to the map. Consider reading Constructing a concept map to learn how to connect these orphaned concepts.

Your concept map is at a taxonomy level of 2. The following suggestions will help you improve your map:

Your concepts seem to be concise, consisting of few words, which is a sign of good concept labels.

Figure 8. In the "How Good is My Cmap" webpage, eCmap displays the uploaded or retrieved Cmap being evaluated, or allows you to construct your own Cmap.

5 Conclusions and Future Work

With the flexible and modular design of the embeddable CmapTools editor eCmap, we have moved towards making concept maps ubiquitous and universal (Novak & Cañas, 2010). eCmap has been implemented in a large variety of scenarios, from being the editor component of CmapTools in the Cloud, the Web-based CmapTools with thousands of users, to being part of Cmap-based test items in an online test preparation and delivery system. We believe that its flexibility will facilitate integration with a wide range of future applications.

We plan to continue enhancing eCmap by including some of the unimplemented advanced features mentioned above that are available in the desktop version of CmapTools. In addition, we are looking to extend the eCmap API to allow the host page to modify a Cmap in real-time by adding, moving, changing, and deleting its contents (and to listen for such events), which we expect will open a whole new category of embedding applications.

References

Cañas, A. J., Bunch, L., Novak, J. D., & Reiska, P. (2013). CmapAnalysis: an Extensible Concept Map Analysis Tool. *JETT*, *4*(1), 36-46.

Cañas, A. J., Hill, G., Carff, R., Suri, N., Lott, J., Eskridge, T., . . . Carvajal, R. (2004). CmapTools: A Knowledge Modeling and Sharing Environment. In A. J. Cañas, J. D. Novak, & F. M. González (Eds.), *Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping* (Vol. I, pp. 125-133). Pamplona, Spain: Universidad Pública de Navarra.

Cañas, A. J., Hill, G., Granados, A., Pérez, C., & Pérez, J. D. (2003, 2003). *The Network Architecture of CmapTools*. Technical Report. Institute for Human and Machine Cognition, Pensacola, FL.

Cañas, A. J., Novak, J. D., Miller, N. L., Collado, C. M., Rodríguez, M., Concepción, M., . . . Peña, L. (2006). Confiabilidad de una Taxonomía Topológica para Mapas Conceptuales. In A. J. Cañas & J. D. Novak (Eds.), *Concept Maps: Theory, Methodology, Technology. Proceedings of the Second International Conference on Concept Mapping* (Vol. 1, pp. 153-161). San Jose, Costa Rica: Universidad de Costa Rica.

Novak, J. D., & Cañas, A. J. (2010). The Universality and Ubiquitousness of Concept Maps. In J. Sánchez, A. J. Cañas, & J. D. Novak (Eds.), *Concept Maps: Making Learning Meaningful. Proceedings of the Fourth International Conference on Concept Mapping* (Vol. 1, pp. 1-13). Viña del Mar, Chile: Universidad de Chile.

Novak, J. D., & Gowin, D. B. (1984). *Learning How to Learn*. New York, NY: Cambridge University Press.