

## CMAPANALYSIS: AN EXTENSIBLE CONCEPT MAP ANALYSIS TOOL

*Alberto J. Cañas & Larry Bunch*  
*Institute for Human & Machine Cognition (IHMC), USA*  
*www.ihmc.us*

*Priit Reiska*  
*Tallinn University, Estonia*  
*www.tlu.ee*

Abstract. CmapAnalysis is a software tool that facilitates performing various analysis measures on a collection of concept maps. A set of measures that consider size, quality and structure properties of the maps are included. The program is designed to be extensible, allowing users to add their own measures. The output of CmapAnalysis is an Excel spreadsheet that can be further analyzed.

### 1 Introduction

Concept maps have been used for many years as a tool for people of all ages and all domains of knowledge to express their understanding about a topic. More specifically, in education concept maps have been shown to be useful as a tool for teachers to assess students' understanding, whether at the beginning, process, or end of the study of a topic (Novak & Cañas, 2004). However, concept maps themselves have to be "assessed" – that is, the concept maps constructed by students need to be assessed by the instructor to get an appreciation of the student's understanding (and possibly assign a grade). Many assessment rubrics have been proposed, beginning with that proposed by Novak & Gowin (1984).

Although Ruiz-Primo and Shavelson (1996) have reported problems in using concept mapping as an assessment tool, there are many studies showing that concept mapping is an appropriate tool for testing students achievement (Fischler, et al., 2002; McGaghie, McCrimmon, Thompson, Ravitch, & Mitchell, 2000; Reiska, 2005; West, Pomeroy, Park, Gerstenberger, & Sandoval, 2000). Some of the studies also show that there is a high correlation between the concept mapping and other knowledge tests (Mikelskis, 1999) but some studies did not prove the correlation between concept map scores and e.g. multiple choice exam performance (McGaghie, et al., 2000).

One of the difficulties of applying assessment rubrics or algorithms is the time it takes to apply them to a large number of concept maps. If the class is large, or if the assessment is being done for research purposes on a large number of Cmaps, the time to manually analyze the maps, or transcribe their content for analysis, can be considerable.

Automatic assessment of concept maps has been included as part of several concept mapping tools, e.g. C-Tools (Harrison, Wallace, Ebert-May, & Luckie, 2004), COMPASS (Gouli, Gogoulou, Papanikolaou, & Grigoriadou, 2003) and CRESST (Herl, O'Neil, Chung, Dennis, & Lee). These systems are concept map editors with an assessment tool incorporated. They don't lend themselves to evaluating a large collection of concept maps for research purposes, or applying different types of assessment criteria to a set of maps.

CmapAnalysis is a software tool that facilitates the analysis of sets of concept maps utilizing various algorithms, rubrics and techniques. The tool provides a set of assessment options by default, and can be extended by the user to apply other assessment techniques that he/she defines. We don't propose or recommend that CmapAnalysis be used as an "automated" concept map assessment tool, and that instructors take its results as grades to be assigned to the maps. On the contrary, we propose that by automatically performing the routine operations of the analysis, instructors, and researchers, can dedicate more time to evaluate the results of the analysis and what they mean and represent. CmapAnalysis does not include a concept map editor, it presumes that the concept maps have already been constructed and stored in the CXL format (Cañas, Hill, et al., 2006).

## 2 Assessment Algorithms & Techniques

The content of a concept map can be divided into three general categories (Reiska, 2005):

1. Size
2. Quality
3. Structure

**Size** describes how many concepts, linking words and propositions are in a concept map. Typical measures in this category are Number of Concepts (or Concept Count), Number of Linking Words and Number of Propositions. Measures from this category also describe students' knowledge. Students with more knowledge about certain topic and focus question usually include more concepts and linking words in their concept maps. However, measures from this category alone can also mislead, because large number of proposition does not mean always that the student has good knowledge – the propositions can also be incorrect. All measures in category Size are quantitative and can be calculated automatically.

**Quality** describes what kind of concepts, linking words and propositions are in a concept map. Typical measures in this category are *Number of Correct Propositions* (or *Correct Proposition Count*), *Average Rating of Propositions*, and *Relevance of Concepts*. The evaluation of propositions is mostly carried out with expert rating. That means that propositions will be rated by experts or compared with the propositions from expert concept map or an expert reference matrix. *Quality* measures can be qualitative and quantitative. To calculate the measures in this category additional information is needed. This information is commonly provided via expert ratings in the form of concept maps or ratings matrices.

**Structure** describes how the concepts are connected to each other. Typical measures in this category are *Centrality of Concepts*, *Number of Cross Links*, *Density*, and *Inter-Cluster Proposition Count*. The measures from this category provide information on how well the concepts are connected, such as whether there are any central concepts, are there any separate sub maps, is the map a “chain”, a “tree” or a “star” (one central concept). The structural measures provide useful information, however the use of these measures without measures from other categories gives us limited information about the knowledge of students. All measures in category *Structure* are quantitative and can be calculated automatically.

While the measures from first two categories (*Size* and *Quality*) provide information that can be gathered with other testing methods (e.g. essay or multiple choice test), the *Structure* measures are unique for concept maps. Combining the measures for *Structure* with the measures for *Quality* offers unique information. Analyzing concept maps from this point of view has not well developed yet. One of our aims with *CmapAnalysis* is to develop a flexible tool that allows the researcher (or teacher) to create their own measures, in particular measures that combine *Quality* and *Structure*. Soika, Reiska & Mikser (2010) used *CmapAnalysis* to find out how the knowledge and the structure of mental models are influenced by the animation and the paper instruction by upper secondary level chemistry students.

## 3 CmapAnalysis

CmapAnalysis is a cross-platform application that enables users to define and execute analyses over *sets of concept maps*. The result of the analysis is a Microsoft Excel spreadsheet containing one row for each concept map in the analyzed set with columns for each of the desired measures. *Measures* range from simple counts of concepts and propositions to more complex calculations such as identifying the top three most central concepts in each map.

CmapAnalysis was designed with several objectives in mind:

- a. CmapAnalysis should support *Size*, *Quality* and *Structure* measures in the analysis of concept maps, as described above.
- b. CmapAnalysis should take as input Cmaps in the open CXL file format in addition to the .cmap format, allowing the analysis of concept maps developed by concept mapping programs that utilize CXL.
- c. CmapAnalysis can handle large sets of concept maps, calculating the measures in (a) to all the maps.

- d. In addition to the analysis measures incorporated into the program described below, CmapAnalysis should be extensible, meaning that users (with technical inclination) should be able to add other measures to the program.
- e. The results provided by the program should be in a format that lends to further analysis (e.g. Excel spreadsheet).

## 4 Using CmapAnalysis

Selecting the set of maps to analyze involves browsing for a folder in the computer's file system. The tool recursively searches the selected folder for Cmaps in either the CmapTools (Cañas, et al., 2004) binary format (.cmap) or the concept map open XML format (.cxl). The CXL format is preferred since the analysis tool leverages standard XML technologies to perform its task (see Section X).

Configuring the content of the analysis involves selecting the set of desired measures from a menu where each measurement defines a column in the resulting table, as shown in Figure 1. The tool has several groups of predefined measures including basic Cmap information, topological characteristics and topological taxonomy score (Cañas, Novak, et al., 2006), and centrality measures concerning the level of connectedness among concepts in the map. These groups of measures can be applied with very little if any configuration. The groups of measures for concept clusters and for scoring propositions require the user to provide additional input such as a list of concepts or propositions. The results of running CmapAnalysis on a set of expert Cmaps as specified in Figure 1 are shown in Figure 2.

### 4.1 Measures included in CmapAnalysis

*Basic Cmap Info Measures:* include all of the metadata properties of the concept map, many of which are editable through the CmapTools map Properties window. The measures include:

- *Author Email:* the email address of the creator of the concept map that can be blank.
- *Author Name:* the name of the creator of the concept map that can be blank.
- *Author Organization:* the name of the author's organization that can be blank.
- *CmapTools Version:* the version of the CmapTools client last used to save the map.
- *Date Created:* the date the map was created in the format YYYY-MM-DDTHH:MM:SS-GMT, for example 2006-01-05T16:02:46-06:00.
- *Date Last Modified:* the date of the last save made to the concept map in the same format as Date Created.
- *Language:* a two character ISO-639 code (<http://www.loc.gov/standards/iso639-2>) representing the language of the map content, for example 'en' for English and 'es' for Spanish.
- *Size:* the size of the concept map in bytes as of the last time the map was saved.
- *Title:* The name of the concept map that is the same as the file name without the .cmap or .cxl extension.

*Topological Taxonomy Measures* include the topological taxonomy score (Cañas, Novak, et al., 2006) between 0 and 5 where higher scores typically indicate higher quality concept maps. This group of measures also includes the following individual aspects of the concept map that are considered in calculating the taxonomy score:

- *Avg Words per Concept:* the total count of words, as separated by whitespace, in all concepts divided by the number of concept in the map. Concise concepts are important to the taxonomy score.
- *Branch Point Count:* the total number of concepts and linking phrases that have at least one incoming connection and more than one outgoing connection.
- *Concept Count:* the number of concept in the map.
- *Linking Phrase Count:* the number of linking phrases in the map.
- *Orphan Count:* the number of concepts in the map that have no connections.
- *Proposition Count:* the number of propositions (i.e. concept-linking phrase-concept) in the map.
- *Root Child Count:* the number of concepts in the map that have an incoming connection from a root concept. A root concept is defined as one that has outgoing connections but no incoming connections.
- *Sub Map Count:* the number of root concepts found in the map.
- *Taxonomy Score:* the topological taxonomy score computed for the map.

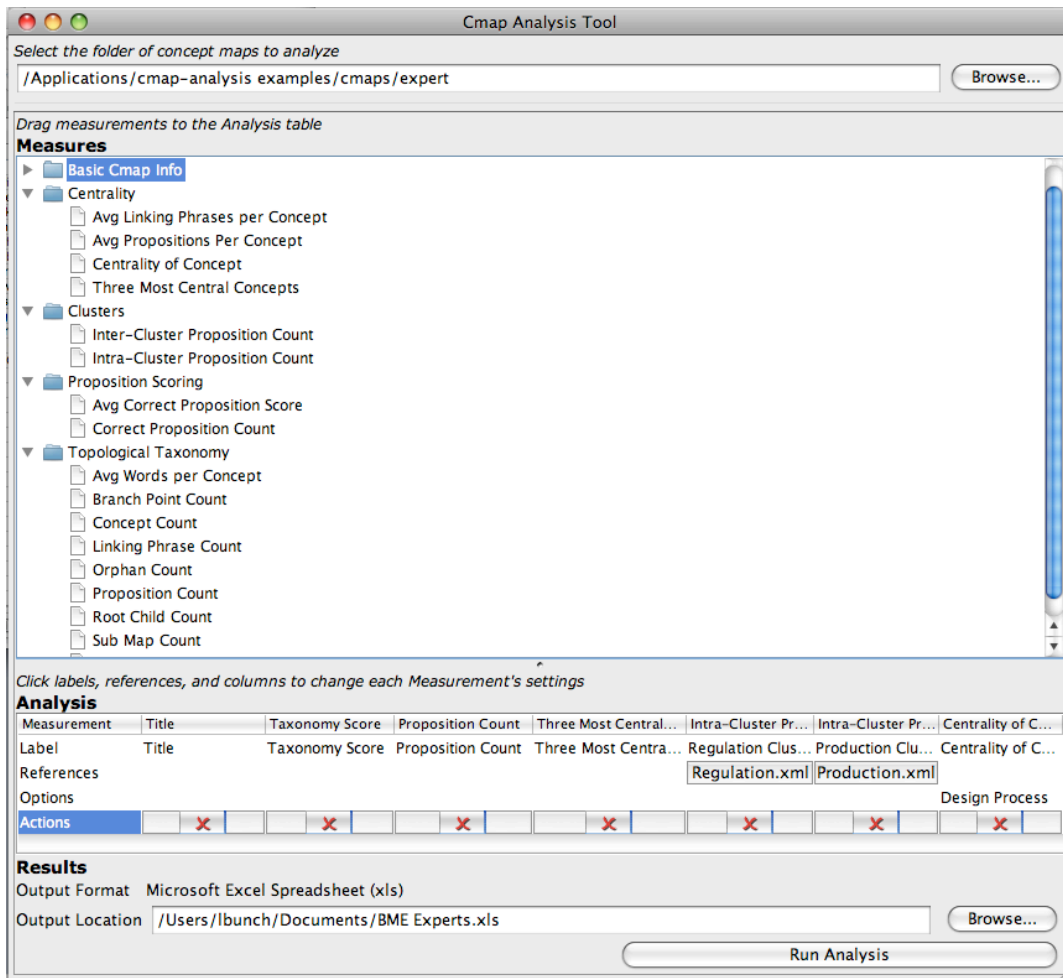


Figure 1. A Screenshot of the CmapAnalysis tool's dialogue box to define the analysis measurements.

*Centrality Measures* concern the connectedness of the concepts in the map, and include:

- *Avg Linking Phrases per Concept*: the ratio of linking phrases to concepts.
- *Avg Propositions per Concept*: the ratio of propositions to concepts.
- *Centrality of Concept*: the number of connections into and out of a given concept. The user provides a concept label in the Option field for this measure. The tool searches for the first occurrence of a concept with a matching label, ignoring whitespace and the case of the letters in the concept, and computes the Centrality of this concept.
- *Three Most Central Concepts*: a comma-separated list of the text labels from the three concepts with the highest centrality measure in the map.

*Proposition Scoring Measures* involve comparing the propositions in the map to a list of propositions provided by the user as an XML file that is selected using the References field of the measurement definition. We are aware that editing XML files is not a user-friendly interface and hope to correct this in future versions of CmapAnalysis. The following example illustrates the XML format for the list of propositions:

```
<proposition-score-list xmlns:cm="http://cmap.ihmc.us/xml/cmap/">
<cm:proposition score=".75">
  <cm:concept label="BME Design Process"/>
  <cm:linking-phrase label="involves"/>
  <cm:concept label="brainstorming"/>
</cm:proposition>
<cm:proposition score=".5">
```

```

    <cm:concept label="FDA approval"/>
    <cm:linking-phrase label="leads to"/>
    <cm:concept label="product on the market"/>
  </cm:proposition>
</proposition-score-list>

```

- **Correct Proposition Count:** the number of propositions in the concept map that match one of the propositions in a given list. Whitespace and letter case are ignored for the comparison. Proposition scores are not considered by this measurement.
- **Avg Correct Proposition Score:** sums the score value for all propositions in the map that match one in the given scored proposition list, then divides this score sum by the number of propositions in the map that were matched.

*Cluster Measures* provide a way to analyze relationships among groups of concepts with a map, for example the number of propositions in the map that relate key physics concepts with key chemistry concepts. These measures are configured with a single XML file that defines a list of one or more clusters where each cluster is a named list of concepts. We expect that a future version of CmapAnalysis will provide a more user-friendly interface for defining the clusters.

```

<cluster-list xmlns:cm="http://cmap.ihmc.us/xml/cmap/">
  <cluster id="Design">
    <cm:concept label="Design Process"/>
    <cm:concept label="design"/>
    <cm:concept label="innovation"/>
  </cluster>
  <cluster id="Production">
    <cm:concept label="Marketing"/>
    <cm:concept label="product"/>
    <cm:concept label="manufacturing"/>
  </cluster>
</cluster-list>

```

- **Intra-Cluster Proposition Count:** the number of propositions in the map that occur *within* a given cluster. Intra-cluster propositions have both concepts defined in a single given concept cluster. For this measure, the provided XML cluster definition file should include only one cluster.
- **Inter-Cluster Proposition Count:** the number of propositions in the map that occur *between* concepts in the given clusters. Inter-cluster propositions have a concept in one of the given clusters and a concept in another cluster. For this measure, the XML cluster definition file should include at least two concept clusters.

## 5 CmapAnalysis Implementation

The analysis tool takes advantage of the CXL format of the concept maps to apply powerful standard XML processing and query tools to implement the analysis. The analysis execution step is defined as a transformation from the collection of maps as XML documents to a single resulting XML document in the Microsoft Excel format. We use a combination of the Extensible Stylesheet Language Transformation (XSLT) and XML Query Language

	A	B	C	D	E	F	G
	Title	Taxonomy Score	Proposition Count	Three Most Central Concepts	Regulation Cluster Count	Production Cluster Count	Centrality of Concept
1	Concept Map 1		42	Process=6,Genesis of design process=5,Market analysis=5	1	0	Design Process=0
2	Concept Map 2	6	34	Regulatory requirements=5,Process=4,Management skills=4	0	5	Design Process=0
3	Concept Map 3	2	59	scientific needs=7,Market Opportunities and Constraints=6,Genesis of design process=6	0	7	Design Process=0
4	Concept Map 4	5	30	design review=6,Market analysis=5,return on investment=4	0	0	Design Process=0
5	Concept Map 5	6	32	BME Design=8,genesis of design process=6,Technical knowledge=4	0	3	Design Process=0
6	Concept Map 6	5	29	Industry needs=4,Technical background=2,Regulatory requirements=2	0	0	Design Process=0
7	Concept Map 7	5	25	Market Analysis=6,BME Design=4,Customer Needs=4	0	1	Design Process=0
8	Concept Map 8	4	59	Customer Needs=6,Management-selected cross-functional team=6,Prototypes=5	0	3	Design Process=0
9	Concept Map 9	5	37	Market Opportunities and Constraints=6,Design Team=6,Project Engineer=5	0	2	Design Process=0
10	Concept Map 10	5	35	BME Design=7,Product=5,Literature review=4	0	3	Design Process=0
11	Concept Map 11	4	29	BME Design=3,Process=3,Market analysis=2	0	2	Design Process=0
12	Concept Map 12	4	70	BME engineering activity and professional practice=12,BME Design=6,societal concerns=6	0	6	Design Process=0
13	Concept Map 13	4	101	Product=15,Prototype=13,BME Design=13	1	7	Design Process=0
14							
15							
16							

Figure 2. Excel spreadsheet with the results of running CmapAnalysis on a set of Cmaps from experts as specified in Figure 1.

(XQuery) to walk through the set of concept maps and compute each of the selected measurements for that map. To illustrate the power of XSLT, the code to obtain the count of concepts in a map is simply:

```
count( $map/cm:cmap/cm:map/cm:concept-list/cm:concept )
```

where count is a built-in XSLT function, \$map is the collection of CXL documents, and the remaining text is just the path to the <concept> elements within the XML document structure.

XQuery similarly provides a very concise and expressive language for analyzing the content of XML documents. For example, the code to find the list of propositions involves ‘joining’ each linking phrase to the list of connections and concepts:

```
for $lp in $map//cm:linking-phrase
let $conn1 :=
$map/cm:cmap/cm:map/cm:connection-list/cm:connection[@to-id = $lp/@id],
$conn2 :=
$map/cm:cmap/cm:map/cm:connection-list/cm:connection[@from-id = $lp/@id],
$cpt1 :=
$map/cm:cmap/cm:map/cm:concept-list/cm:concept[@id = $conn1/@from-id],
$cpt2 :=
$map/cm:cmap/cm:map/cm:concept-list/cm:concept[@id = $conn2/@to-id]
for $i in (1 to count($cpt1)),
  $j in (1 to count($cpt2))
return
  <cm:proposition> {$cpt1[$i], $lp, $cpt2[$j]} </cm:proposition>
```

This XQuery function returns a list of propositions represented in XML:

```
<proposition><concept/><linking-phrase/><concept/></proposition>
```

Since the results are also in XML format, they can be used as input to other XQuery calculations. This is exactly how the ‘Correct Proposition Count’ measurement is implemented. It first obtains the proposition list from the map, joins this intermediate XML result with the user-provided list of correct propositions to identify matches.

Another useful feature of XQuery is the ability to call-out to a procedural language such as Java. The topological taxonomy score is computed by using such a call to a Java method with the current concept map’s XML content as a parameter. This allowed us to reuse the Java implementation of the taxonomy scoring that was implemented for use within CmapTools.

The analysis tool defines a ‘shell’ XQuery algorithm that handles setting up the collection of CXL documents, iterating through this set of maps, invoking each measurement’s XQuery code on each CXL map, and organizing the results into a table of rows and columns that Microsoft Excel can understand. Each measurement defines just the XSLT or XQuery code necessary to calculate the value for a single concept map and produces a text or numeric result. Our hope is that this level of modularity will enable others to easily create and share new measurements without needing to understand the code of the analysis tool itself, but rather just the CXL format and the standard XML technologies.

## 6 Extending CmapAnalysis

The analysis tool contains a directory named ‘measurements’. Each folder in this directory defines one of the groups of measurements that appear in the user interface. Within each of these measurement group folders is an XML file for each measurement. The folder names and file names are used directly by the analysis tool to populated the user interface, so creating a new group of measurements or reorganizing them involves simply creating new folders and rearranging the XML files within.

Measures are defined as XML documents with the following format.

```

<cmmap-measure>
  <Label>Avg Words per Concept</Label>
  <Description></Description>
  <DataType>Number</DataType>
  <XQueryExpression>
    if (count( local:concept-words($map)/cm:concept) > 0) then
      (
        round-half-to-even(
          count( local:concept-words($map)/cm:concept/cma:word)
          div
          count( local:concept-words($map)/cm:concept)
          , 5 )
        )
      else 0.0
    </XQueryExpression>
  <XQueryFunctions>
    declare function local:concept-words($map as document-node()) as element(concept-word-list)
    {
      <concept-word-list>
      {
        for $concept in $map/cm:cmap/cm:map/cm:concept-list/cm:concept
        return
          <cm:concept>
          {
            for $i in tokenize($concept/@label, "\s+" )
            return <cma:word>{$i}</cma:word>
          }
        </cm:concept>
      }
    </concept-word-list>
  };
</XQueryFunctions>
</cmmap-measure>

```

- *Label*: the default text to use as the column heading for the measure.
- *Description*: the text description of the measurement.
- *DataType*: identifies whether the results are a Number or a String (i.e. text). This is used to display the column of results correctly in Excel.
- *XQueryExpression*: the XQuery code that will be executed to compute the measurement. This code is inserted into the ‘shell’ XQuery that the analysis tool defines so that the variable \$map will always be defined and contain the XML for the current concept map. The code defined by the measurement computes the value for a single map and the analysis tool takes care of applying this measure to all of the selected maps.
- *XQueryFunctions*: XQuery supports defining functions that can be called from within the XQueryExpression as shown in the example above where the ‘Avg Words per Concept’ expression calls the ‘concept-words’ function to obtain a list of all the words in a given concept. These functions typically return an XML result that is then used by the calling XQuery expression.
- The XQuery code for the analysis shell (*analysis-shell.xml*) is also provided with the tool for developers to potentially modify or extend to suit their needs. This shell document defines several functions that can be reused and called from within the XQueryExpression or XQueryFunctions sections of any measurement definition. These functions include obtaining an XML list of propositions and branch points as well as other useful features like finding the list of concepts connected to a given one.

## 7 Future Work

CmapAnalysis is fully operational and in use by researchers and instructors. However, being in its first version the user interface needs further work. The dialogue box used to define the measurements for the analysis needs adjustments, and using XML to specify clusters and list of propositions needs to be replaced by a proper graphical dialogue box.

## 8 Conclusions

CmapAnalysis is a tool that enables the automated assessment of a large number of concept maps utilizing a set of predefined measurements. It is written in Java and thus runs in a large number of platforms, including Windows, OS X and Linux. CmapAnalysis is extensible, allowing the user to define new measurements in XML. The results of CmapAnalysis' processing a set of Cmaps are stored in an Excel spreadsheet that can be further analyzed. CmapAnalysis provides both instructors and researchers with a mechanism for in-depth analysis of concept maps.

## 9 References

- Cañas, A. J., Hill, G., Bunch, L., Carff, R., Eskridge, T., & Pérez, C. (2006). KEA: A Knowledge Exchange Architecture Based on Web Services, Concept Maps and CmapTools. 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. 304-310). San Jose, Costa Rica: Universidad de Costa Rica.
- Cañas, A. J., Hill, G., Carff, R., Suri, N., Lott, J., Eskridge, T., et al. (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., Novak, J. D., Miller, N. L., Collado, C. M., Rodríguez, M., Concepción, M., et al. (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.
- Fischler, H., Peuckert, J., Dahncke, H., Behrendt, H., Reiska, P., Pushkin, D., et al. (2002). Concept Mapping as a Tool for Research in Science Education. In H. Behrendt, H. Dahncke, R. Duit, W. Gräber, M. Komorek, A. Kross & P. Reiska (Eds.), *Research in Science Education - Past, Present, and Future* (pp. 217-224): Springer Netherlands.
- Gouli, E., Gogoulou, A., Papanikolaou, K., & Grigoriadou, M. (2003). COMPASS: An Adaptive Web-based Concept Map Assessment Tool. 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. 1). Pamplona, Spain: Universidad Pública de Navarra.
- Harrison, S. H., Wallace, J. L., Ebert-May, D., & Luckie, D. B. (2004). C-TOOLS Automated Grading for Online Concept Maps Works Well with a Little Help from "WordNet". In A. J. Cañas, J. D. Novak & F. M. González (Eds.), *Concept Maps: Theory, Methodology, Technology. Proc. of the First Int. Conference on Concept Mapping* (Vol. 2, pp. 211-214). Pamplona: Universidad Pública de Navarra.
- Herl, H. E., O'Neil, H. F., Jr., Chung, G. K. W. K., Dennis, R. A., & Lee, J. J. (1997). Feasibility of an On-line Concept Mapping Construction and Scoring System.
- McGaghie, W. C., McCrimmon, D. R., Thompson, J. A., Ravitch, M. M., & Mitchell, G. (2000). Medical and Veterinary Students' Structural Knowledge of Pulmonary Physiology Concepts. *Academic Medicine*, 75(4), 362-368.
- Novak, J. D., & Cañas, A. J. (2004). Building on Constructivist Ideas and CmapTools to Create a New Model for Education. In A. J. Cañas, J. D. Novak & F. M. González (Eds.), *Concept Maps: Theory, Methodology, Technology. Proceedings of the 1st International Conference on Concept Mapping*. Pamplona, Spain: Universidad Pública de Navarra.
- Novak, J. D., & Gowin, D. B. (1984). *Learning How to Learn*. New York, NY: Cambridge University Press.
- Reiska, P. (2005). *Experimente and Computersimulationen. Empirische Untersuchung zum Handeln im Experiment und am Computer unter dem Einfluss von physikalischem Wissen*. Frankfurt a. M.: Peter Lang.
- Ruiz-Primo, M. A., & Shavelson, R. J. (1996). Problems and issues in the use of concept maps in science assessment. *Journal of Research in Science Teaching*, 33(6), 569-600.
- Soika, K., Reiska, P., & Mikser, R. (2010). *The Use of Animation as a Visual Method in Learning Chemistry*. Paper presented at the 4th International Conference on Concept Mapping, Viña del Mar, Chile.
- West, D., Pomeroy, J. R., Park, J., Gerstenberger, E., & Sandoval, J. (2000). Critical thinking in graduate medical education. *Journal of the American Medical Association*, 284(9), 1105-1110.