

Algebraic and Numerical Algorithms on the Web: a purpose of adaptive teaching and learning environment

Maria Angélica de O. Camargo-Brunetto¹, Douglas Terêncio do Vale², Claiton de Oliveira²,
Cleberson Vidotte Costa²

¹ Computer Science Departmente State University of Londrina - Rodovia Celso Garcia Cid Pr 445 Km 380 Campus Universitário,
Londrina, PR, Brasil. Cx. Postal 6001 CEP 86051-990. Fone: (43) 3371-4000

² Computer Science Students

angelica@uel.br, dtvale@dc.uel.br, coliveira@dc.uel.br, cvcosta@dc.uel.br

www.protem.dc.uel.br/anaweb

Abstract : ANA-WEB (Algebraic and Numerical Algorithms on the Web) is a teaching and learning environment that aims to help on the understanding of some classical numerical methods through the online execution of computer programs with explanations as also by using a Computer Algebra System to solve related problems. The system is composed by two main modules: Numerical Computation and Algebraic Computation that have just been integrated, so they have still presented differences in some aspects. In the context of numerical computation, ANA-WEB provides a set of numerical algorithms that can be performed with different inputs, according to the needs of the user. The system not only deliveries the answer, but also gives explanation about the resolution of each problem. Seeking for more flexibility, the system offers two modes of use, namely **authoring** and **learning**. In the authoring mode, the teacher can adds so theoretical web pages as problem examples associated with the subjects of the numerical algorithms. In the learning mode, the student can navigate among the numerical algorithms, testing different inputs that can be typed by him or can be searched in a set of problem examples that have been built by teachers. Considering the algebraic context, ANA-WEB establishes a connection with the computer Algebra System Maple, providing additional alternatives to solve mathematical problems. Authoring in the algebraic approach allows the author to define a structure of subjects and to each subject to introduce problem examples that can be explored by the student in different ways: ready examples, interactive examples with templates (try yourself) and interactive examples with help (exercises). ANA-WEB was developed using the PHP programming language, MySQL database to manage the users, XML to store problem examples, DOM (Document Object Model) to manipulate the XML files and JavaMath to integrate the Web application with the Computer Algebra System. The purposed system is being used at the State University of Londrina in the discipline of Algebraic and Numerical Computation and the experimental tests have shown good perspectives. The main contribution of this work is to provide an adaptive educational environment that allows interactive and dynamic experiences so in the authoring mode as in the learning mode. It is hoped that with this web application, teachers can improve their educational resources and students can improve their understanding about different mathematical algorithms.

Keywords: Mathematical Computation on the web, e-learning, educational hypermedia

1 Introduction

Internet and World Wide Web has contributed for teaching and learning considering individual differences, giving opportunities to generate adapted content and presentations. Among several new technologies it must be outlined dynamic web pages, hypermedia adaptation techniques, standards of content representation and interchange of data among web applications.

Mathematical Computing is applied in several knowledgment areas and the use of suitable examples can be useful to a well-succeed understanding of how a problem is solved using a specific algorithm.

Research and development activities in producing Mathematics education on the Web have increased in the last years. The first advances reached in this area were the production of static web

pages with a high quality of mathematical output. In this sense, MathML¹, that is a XML² derived language seems to be the most obvious language to help in the task to develop mathematics contents for the web. It is known the existence of limitations in the use of such language, considering the differences among the Web browsers. However this scenario tends to change fast, considering the efforts in this area. It has soon observed the need to produce dynamic Web pages suitable for *Mathematical Computing*. To do this task, several approaches have been studied, including Java applets, CGI, integration with Computer Algebra Systems, and so on.

Advances in the *Presentation of Mathematical Contents* as well *Mathematical Computation on the web* have been strongly observed by results published in the IAMC³ workshops. Several web sites providing courses and tools for mathematic education have been developed in recent years.

Interactive learning environments of Mathematical Computation on the web include ActiveMath (Libbrecht 2001), Livemath⁴, WebMathematica⁵ LiveMath, WebMathematica and so on. More recently it was proposed the WME (Web-based Mathematics Education) (Wang et al 2005) that is a framework to help on creation of interactive material for mathematical education on the Web. MathEdu (Diez 2004) is an authoring tool to solve mathematical exercises using Mathematica. The system provides two ways of use: as a tutor, showing to the student how to solve specific problems corresponding to previously defined patterns or to solve specific problems interactively while it controls the resolution processes. Calculus Machina (Quinney, 2001) defines tests or quizzes whose resolution that is checked by the system using a simple answer. Calculus Wiz (Stroyan) is an interactive infinitesimal calculus course, but the teacher can't extend the subject it covers without a deep knowledge of programming. Wiley Web Test (www.math.unl.edu/webtests/docs/focus2) includes the definition of resolution strategies, but its domain of application cannot be extended.

Interactivity in Mathematical Computation on the web can be improved by including integration of Mathematical web applications with Mathematical Computer Systems as Maple, Macsyma, Mathematica, and others. Using Computer algebra systems may improve the teaching-learning task. In this context, IAMC points a set of efforts that give support to research in this area. Many research has been developed, treating with different needs, including patterns for mathematical data interchange, APIs to establish a communication between web server and mathematical computer systems, interactive learning environment, and so on.

Considering patterns for mathematical data interchange on the web, we should outline MathML (content encoding) and OpenMath (Naylor, 2001).

For APIs, the main results are OMEI - Open Mathematical Engine Interface (Liao et al) and JavaMath (Solomon 2001). OMEI establishes a uniform interface to different computer algebra systems being independent of programming language and can be used under any protocol of mathematical codification. Javamath is an API that allows mathematical programs written in Java to use the facilities of Computer Algebras System as Maple or GAP.

In this context, this article presents an interactive learning and teaching environment (ANA-WEB) that aims to help on the understanding of some classical numerical methods through the online execution of computer programs with explanations and also allows an integration with a Computer Algebra System. At the moment the application addresses the following subjects in the context of Numerical Methods: Floating-point arithmetic, matrix manipulation, resolution of linear systems of equations, interpolation and data fitting methods, integration and derivative numerical

¹ Mathematical Markup Language <<http://www.w3.org/TR/2003/REC-MathML2-20031021/>>

² eXtensible Markup Language <<http://www.w3.org/TR/REC-xml>>.

³ IAMC – Internet Accessible Mathematical Computation (<http://icm.mcs.kent.edu/research/iamc.html>)

⁴ LiveMath – Computer Algebra System & Matheq : <http://www.kivemath.com>

⁵ Webmathematica – <http://www.wolfram.com/products/webmathematica>

methods and resolution of nonlinear equations. In the context of algebraic computation ANA-WEB interacts with Maple Computer Algebra System, using the API JavaMath. The software can be used in two modes: authoring and learning. Adapted Content and Presentation are provided in both modes by means XML as language to represent the examples to be explored.

This article is structured in the following way: section 2 presents the architecture and system overview describing each component. The methodology proposed and the technologies used are explained in section 3. Section 4 presents some experiences of use and an empirical evaluation of the system. Concluding remarks and future works are presented in section 5.

2 ANA-WEB ARCHITECTURE AND SYSTEM OVERVIEW

The architecture of ANA-WEB environment can be viewed in figure 1.

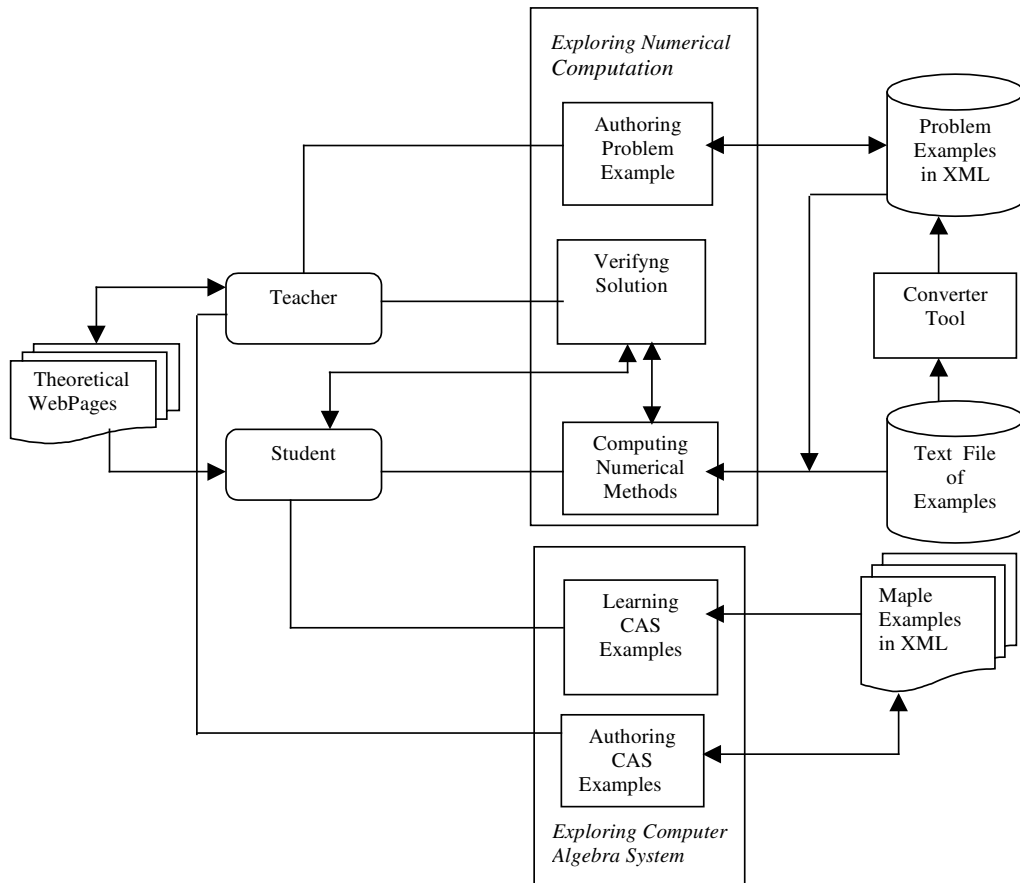


Figure 1 – Architecture of the ANA-WEB application

ANA-WEB application has two main approaches: Exploring Numerical Methods and Exploring Computer Algebra System. Before explaining in details each approach, it will present a brief history of the System: The start idea was to develop a web application to help in the understanding of numerical methods by the online execution with explanations. This application has been named NA-WEB-Numerical algorithms on the Web (Camargo-Brunetto 2004). In parallel it has developed an independent but related module to establish a connection of a web application with a Computer Algebra System. In the first prototype of NA-WEB application, it has been observed the need to extend the variety of problem examples to be explored by student. So, it has been included an authoring problem examples tool. NA-WEB arises to be a complementary material to theoretical Web pages of a discipline of Algebraic and Numerical Computation in a graduate course at State University of Londrina, but it has grown a lot becoming a new application with its own

characteristics. In the sequence we will present the functionalities of both approaches of ANA-WEB application, namely Exploring Numerical Methods and Exploring Algebraic Computation. It must be pointed out that we use the expression “algebraic computation” in a more wide context, that means how to explore facilities of Computer Algebra System to solve mathematical problems, including a hybrid use of numerical and algebraic computation.

2. 1 Exploring Numerical Methods

This module is composed by the modules Authoring Problem Examples, Computing Numerical Methods and Verifying Solution, that will be described in the sequence.

Authoring Problem Examples Module is an authoring tool that allows the teacher to create a library of problem examples that he judges suitable for his students. This module provides an interface where the author fills the fields, according to the subject. The problem examples are stored in XML files. The content representation language XML has show to be suitable to provide adaptability in web applications (Oliveira et al 2003 and Warpechowski 2004). The problem examples follow specifications defined by a DTD (Document Type Definition) previously elaborated that describes the structure of the XML document that store the library of problem examples. One DTD describes the composition of a hierarchic structure, specifies which elements and attributes are presented in this structure, the domain of such elements and the relationship among them. In our application, the XML root’s document was defined by the tag “problems”. Starting from the root, each problem category was inserted as a new child. Then, individual problems were inserted in each category. Each problem has its input data represented in tags. The DTD that describes part of the structure of the library is in the figure 2(a). An example of XML document that follows this DTD is in the figure 2(b).

<pre><?xml encoding="ISO-8859-1"?> <!ELEMENT problems (pontoflutuante linear systems matrices interpolation data fitting integration derivation ode)+> <!ELEMENT derivation (derivative+)> <!ELEMENT derivative (function, initial_value, max_iteration)> <!ELEMENT function (#PCDATA)> <!ELEMENT initial_value (#PCDATA)> <!ELEMENT max_iteration (#PCDATA)></pre>	<pre><?xml version="1.0"?> <!DOCTYPE problems SYSTEM "problemas.dtd"> <problems> <derivation> <derivative> <function>cos(x)</function> <initial_value>1</initial_value> <max_iteration>20</max_iteration> </derivative> </derivation> </problems></pre>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure 2 (a) – Part of DTD file from the numerical integration method

Figure 2(b) – XML document with a derivation problem

Computing Numerical Methods Module: In this module the learner can navigate through several numerical methods organized by subject. The following subjects are covered at the moment: Floating-point arithmetic, matrix manipulation, resolution of linear systems of equations, interpolation and data fitting methods, integration and derivative numerical methods and resolution of nonlinear equations Each numerical method has a computer web program associated. In all the programs, the first step is the obtaining of the data of the problem. The next step is to verify the resolution of the problem, executing the algorithm that corresponds to the selected method. The answer obtained by the program and other information on the partial calculations are exhibited in the screen for the user. The learner can verify the execution of a numerical method through two options: examples or exercises. In the module of examples, he verifies the resolution of a problem step by step. The input of data can be performed through the keyboard, a text file containing the data or a set of problem examples stored in XML, created by the teacher. In the module Verifying Solutions, the learner selects a subject and specifies two data files: the input that he has used and the

output that his computer program delivered. The application reads those data, executes the algorithm and compares the obtained answer by the system with the learner's answer informed in the output file. Figure 3 presents a snapshot of the learner mode of the Computing Numerical Methods. In the Numerical Methods approach the authoring is focused on the problem examples, that are essentially associated with the subjects defined. The adaptation is on the variety of problems, not on the subjects. Additionally, the author can attaches theoretical web pages related to each subject covered by the system.

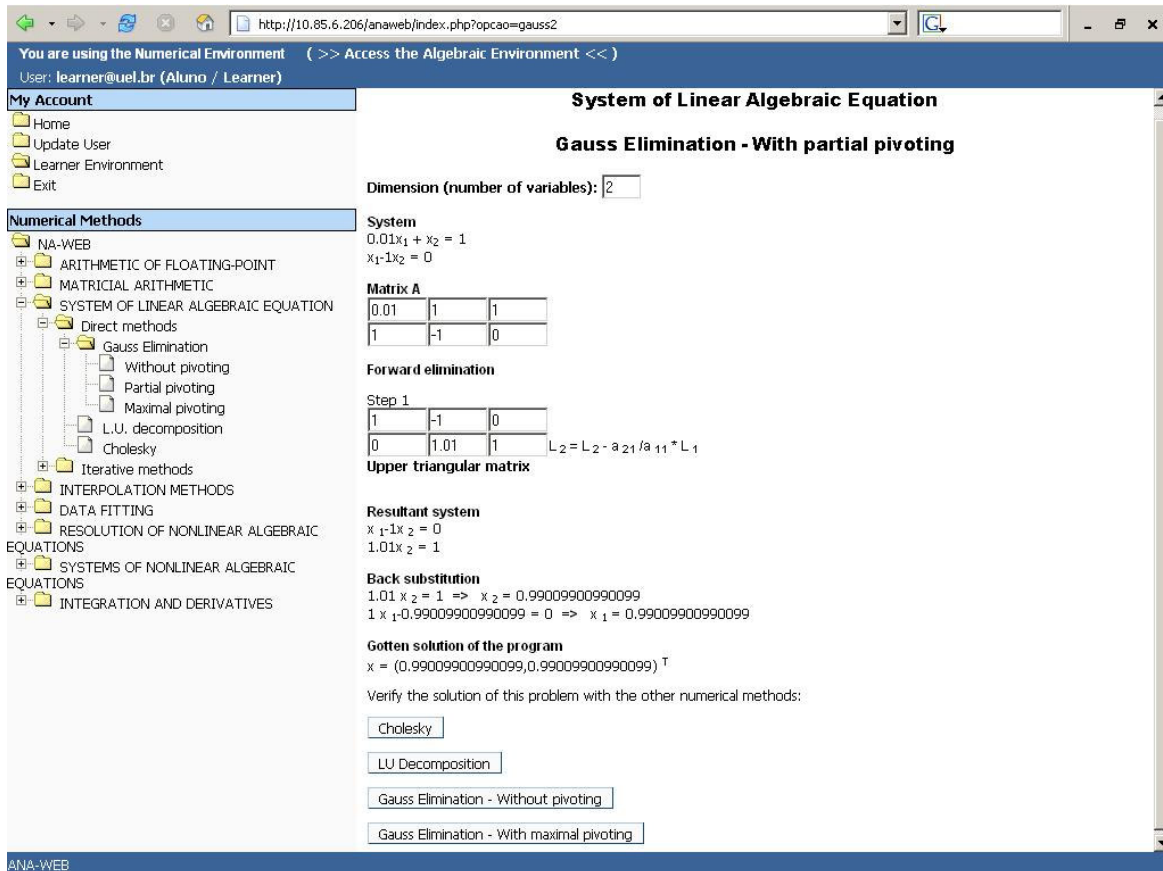


Figure 3 – A snapshot of the interface of ANA-WEB in Numerical mode showing the solution of a Linear System of algebraic equations by the Gauss Elimination method. During the matrix transformations, the system shows at right side what operations have been performed.

2.2 Exploring Computer Algebra System

The module *Exploring Computer Algebra System* aims to offer additional resources to understand how to solve the problems related to the subjects covered by ANA-WEB. This module uses a HTTP Servlet (*WebServer Resin*) and the API JavaMath (Solomon 2001) to establishes interaction with the Maple Computer algebra System and the Client environment (Applet console). This module also provides two modes of use: *authoring* and *learning*. In the authoring mode, the teacher creates a hierarchical content structure that is stored in XML format. Each subject can be presented to the student in three formats: *examples*, *try yourself* and *exercises*. In the format *try yourself*, the teacher prepares a template of sentence to illustrate the resolution of a kind of problem and the system generates a model of command in Maple syntax, where the student fills the parameters needed. In the *example* format, the teacher prepares a sentence or a set of sentences in a complete way to illustrate the resolution of a given problem. The teacher can even load program examples developed in Maple to be executed directly. In the *exercise* format, the teacher formulates

a text presenting the problem and a template of valid sentence to solve that problem. When the system is used by a student, he selects the subject desired and the presentation format (*try yourself*, *example* or *exercise*). A snapshot of the module Exploring Computer Algebra System in **authoring mode** is presented at figure 4 and in **learning mode** at figure 5.

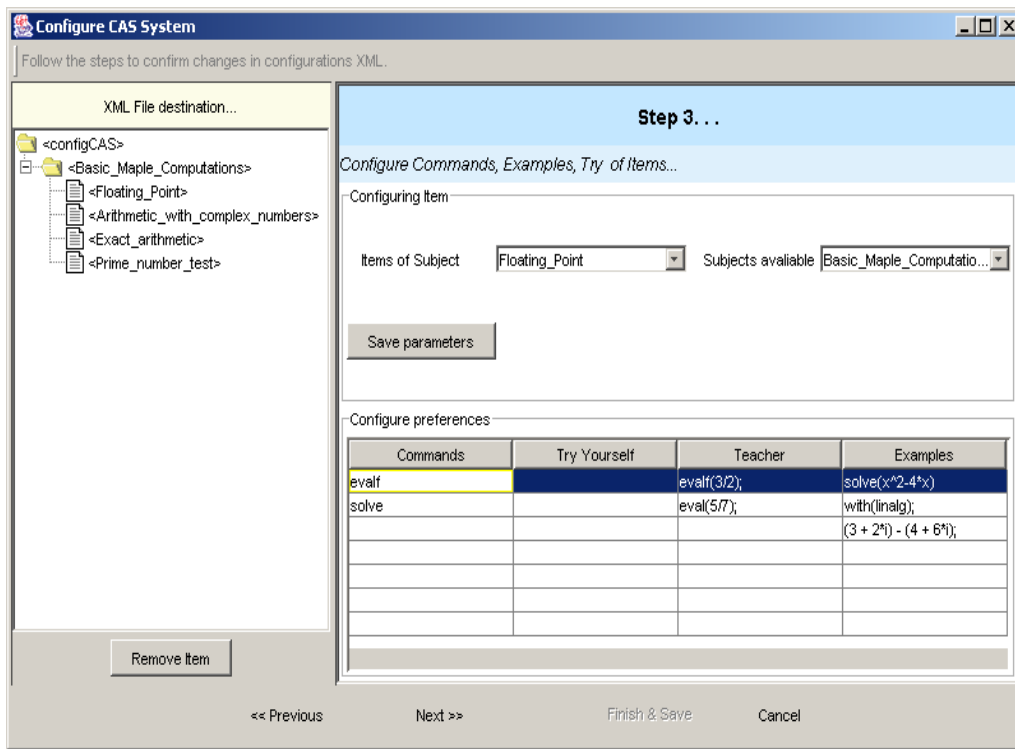


Figure 4 – The interface of the module Exploring Computer Algebra System in the **Authoring Mode**

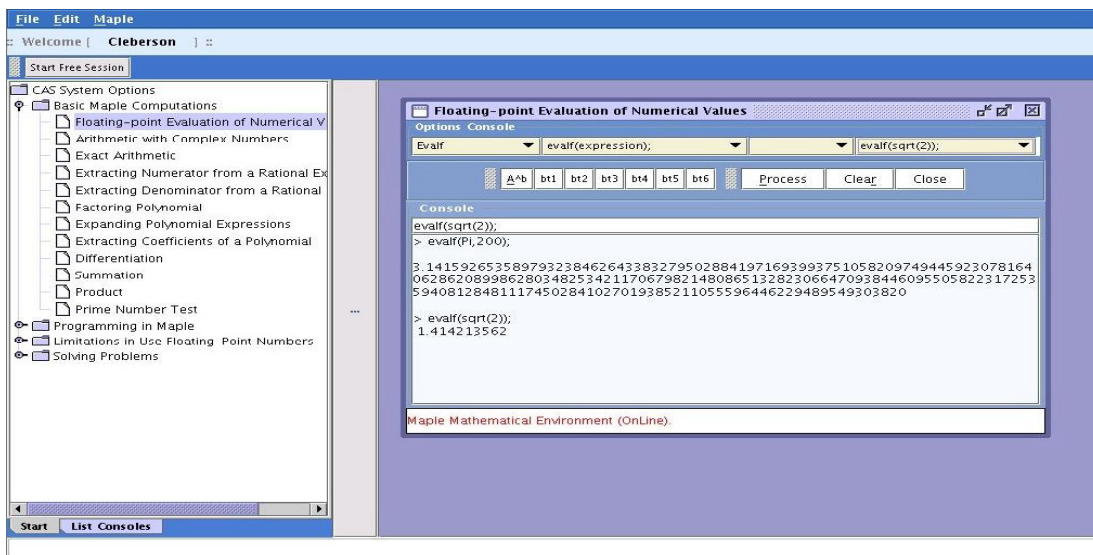


Figure 5 – The interface of the module Exploring Computer Algebra System in the **Learning Mode**

In the authoring mode of the module *Exploring Computer Algebra* the user can adapt the content that is structured by subjects. These subjects are created and manipulated using XML format. A basic structure of subjects stored in a XML file is presented in figure 6.

```

<?xml version="1.0" encoding="UTF-8" ?>
- <configCAS>
- <item title="Basic Maple Computations">
+ <node title="Arithmetic with Complex Numbers">
+ <node title="Exact Arithmetic">
- <node title="Extracting Numerator from a Rational Expression">
- <commands>
  <command>Number</command>
</commands>
- <tryyourself>
  <try>number(expression);</try>
</tryyourself>
- <teacher>
  <program />
</teacher>
- <examples>
  <example>f1 := ((x-1)^2*(x+2)*(x^2+1)*(x^3-2*x+2))/((x+3)*(2*x-4));</example>
  <example>num := number(f1);</example>
</examples>
  <filehelp>http://127.0.0.1/NetApplet/config/quickhelp/Basic_ExtractNum.html</filehelp>
  <comment />
</node>
+ <node title="Extracting Denominator from a Rational Expression">
+ <node title="Factoring Polynomial">
+ <node title="Expanding Polynomial Expressions">
+ <node title="Extracting Coefficients of a Polynomial">
+ <node title="Differentiation">
+ <node title="Summation">
+ <node title="Product">
+ <node title="Prime Number Test">
+ <node title="Floating-point Evaluation of Numerical Values">
</item>
+ <item title="Programming in Maple">
+ <item title="Limitations in Use Floating-Point Numbers">
+ <item title="Solving Problems">
</configCAS>

```

Figure 6 – A XML structure of subjects organized by a teacher in the authoring mode of ANA-WEB in the Explore Computer Algebra System module.

3 Technologies used

The numerical algorithms used in ANA-WEB were developed using the PHP language (*Hipertext Preprocessor*), using Linux. With this language it is possible to create dynamic web pages and considering that these pages are server-side scripts, the user does not need to install any plug-in in his computer.

Concerning to the problem of the ANA-WEB to delivery dynamic examples, and thinking on future levels of adaptation, the problem examples in the Exploring Numerical Methods and the learner sessions in the Exploring Computer Algebra System are stored in XML format. As XML is an extensible markup language to describe data on the Web, it is allowed to create new tags, giving more flexibility, simplicity and easy reading. Further such language allows to include metadata that can be useful to aggregate semantic to the content of the document. To define the structure of XML documents it has been used XML Document Type Definition (DTD). In order to establish an interface between the computer program in PHP e the XML file it has been used the API (Application Programming Interface) DOM (Document Object Model), considering that it is based on a tree structure, independent of plataform and programming language, allowing program and scripts to access and to update dynamically the content, structure and styles of the XML documents.

The integration of the ANA-WEB with the Computer Algebra system requires some technologies. Analyzing different possibilities, including programming language, APIs, communication protocols, the best choice for our experiments was the API JavaMath. Such choice has implied in the use of a HTTP Servlet with support to JAVA. The Computer Algebra System selected was Maple, once such system is available on our laboratory.

HTTP Servlet is composed by the software Apache Server and Resin Server. Apache has the role of http server, once it is a robust and flexible considering the inclusion of new modules. Resin is a Servlet server, developed to provide integration of Java code with HTML code. Servlets are small Java codes that act as intermediate between client and server, allowing dynamic generation of HTML pages. They can be installed to process data transmitted by client under http. When running, these small codes are loaded at the server memory and allow the sharing of resources for several created connections. For each connection it is generated a thread that allows several connections or objects to share a same resource, as for example accessing a database.

Created by Solomon and Struble, **JavaMath** (Solomon 2001) is an API that allows to establish communication between Computer Algebra Systems locally installed and HTTP servlet servers. JavaMath supports natively resources to use the software Maple and GAP. With JavaMath it is possible to create and to handle different sessions of a Mathematical software, to insert other Mathematical systems and has the advantage of providing information processing in a transparent way for the client.

The communication among Java programs that are running is performed using Java RMI - Remote Methods Invocation (Wollrath), becoming possible the sharing of messages and information in an indirect way. Execution of Java RMI is made by means of a service of the Operational System, using a communication gate responsible for the input and output of information.

The sessions manipulated by the client are created by the Computer Algebra System Server and stored in the Java RMI. As it is not possible for the client to establish a direct communication of his session with the CAS server, Java RMI promotes such communication.

In our work, it was added new functionalities on the JavaMath, allowing to create and to update files and directories, linking with the XML files, that contains the configuration of each system user.

The communication client/server implemented was based on applets Java, allowing use of graphics to interact with the client under a web browser. This browser must be installed with the plug-in Java 1.4.x.

4 Some Experiences of Use

Different teaching-learning strategies are being explored in a discipline of Algebraic and Numerical Computation at State University of Londrina.

Considering the module Exploring Numerical Computation, we have been constructed a demo library of problem examples that the students are using along the classes. In the module Exploring Computer algebra System, we structure an hierarchical organized in the following way: an introduction to basic commands in Maple, special Maple commands to facilitate the development of computer programs to solve typical problems of Numerical methods, examples comparing results of numeric and algebraic computation, Maple routines developed by the teacher to solve the typical problems. The integration of ANA-WEB with Maple can be made by command line or by block of commands. In the first case, the language used to data interchange follows the Maple syntax. In the future we intend to use MathML or OpenMath for this task, once Maple can manipulate MathML. In the case of block of commands, as a Maple routine, we has exported Maple files as HTML files, so that ANA-WEB was able to present and runs the Maple program to the learner.

An evaluation of ANA-WEB has been performed with 25 students of the discipline Algebraic and Numerical Computation in the graduate course of Computer Science of State University o Londrina. In the experiment, the students have navigated in the software about two hours and have answered a questionnaire to get some information about the level of system usability. The questionnaire contains 34 objective questions and 6 subjective questions. The main positive points found by the students were: 88% thought that the use of application helps in the understanding of the content of the discipline, 92% appreciated the availability of problem

examples given by the system, 88% found the system easy to use, 88% would recommend the system to their colleagues, 72% appreciated to use the system, 80% judged the system is consistent, 92% had no difficult on navigating in the system. The main asking pointed out by the students were: to improve the initial information about how to use the system in the algebraic mode, to allow that some explanations can be optional, to include more tests of data consistency in some numerical methods. Some bugs reported by the students are being analyzed and must be solved as soon as possible.

5 Concluding Remarks and Future Works

A new teaching-learning environment to provide Algebraic and Numerical Computation on the Web was proposed. The main contributions of this work are: (i) a purpose to extend the content of problem examples to be solved by Numerical methods through the authoring Problem examples storing them in XML files; (ii) provides a web application that allow the student to explore so problem examples of his interest as examples suggested by a teacher; (iii) a purpose of how to explore a Computer Algebra System as an additional resource in the teaching and learning tasks of Mathematical Computation with different levels of help (try yourself, example and exercise).. Also it should be outlined the interactivity of Mathematical Computation on the Web provided by ANA-WEB. As future works, it has been planned to examine other API's to integrate different Computer Algebra Systems with ANA-WEB. We have special interest in establishing a connection of it with Scilab System, once it s a free and opensource mathematical software. Also it should be included the use of MathML as language of data interchange between the ANA-WEB and the Computer Algebra System as well a set of filters in the XML files of problem examples, allowing so teacher as learner to choose problems with specific characteristics.

REFERENCES

Camargo-Brunetto, M. A. O. ; VALE, D. T. ; Oliveira, C. *NA-WEB: Um Sistema Interativo para Aprendizagem de Métodos Numéricos via Web*. [XV Simpósio Brasileiro de Informática na Educação](#). 2004.

Camargo-Brunetto, M. A. O. Numerical and Algebraic Computation in the web: exploring benefits and limitations. ISSAC-2001 Workshop on Internet Accessible Mathematical Computation. 2001. Available on <http://icm.mcs.kent.edu/research/iamc2001.posters/angelica.pdf>

Diéz, F.; Moriyón, R. Solving Mathematical Exercises that involve symbolic Computations. **Computing in Science & Engineering**. January/February 2004.

Freitas, André F. **RMI – Invocação Remota de Métodos**. Escola de Informática da Universidade Católica de Pelotas. 2002. Available in: <http://atlas.ucpel.tche.br/~barbosa/consiso/consiso8/sd/sd10.pdf>

Gang, X.; Galligo, A. *Using WIMS for Mathematical Education*. . ISSAC-2001 Workshop on Internet Accessible Mathematical Computation. 2001. Available on <http://icm.mcs.kent.edu/research/iamc2001.papers/galligo.pdf>

Liao, Weidong; LIN, Dongdai; WANG, Paul S. **OMEI: An Open Mathematical Engine Interface**. Institute for Computational Mathematics. Kent State University. Kent, Ohio, USA.

Libbrecht, Paul. **Integration of Mathematical Systems into the ActiveMath Learning Environment**. ISSAC-2001 Workshop on Internet Accessible Mathematical Computation. 2001.

available in : <http://icm.mcs.kent.edu/research/iamc2001.papers/ActiveMath_IntegrMathSys.pdf>

Naylor, B.; Stephen W. The relationship between OpenMath and MathML. ISSAC-2001 Workshop on Internet Accessible Mathematical Computation. 2001. Available on <http://icm.mcs.kent.edu/research/iamc2001.papers/nay.pdf>

OLIVEIRA, J. P. M., FREITAS, V., CAMARGO-BRUNETTO, M. A. O., LIMA, J. V., RIBEIRO, C. H. F. P., GASPARINI, I., AMARAL, M. A., PIMENTA, M. S., MARÇAL, V. P., PROENÇA JUNIOR, M. L. **AdaptWeb: um Ambiente para Ensino-aprendizagem Adaptativo na Web.** Educação em Revista. Curitiba - PR: , p.175 - 197, 2003.

Quinney, D. Computer-Based Diagnostic Testing and Student Centered Support. Computer-Aided Assessment in Mathematics Series, Cliff Beevers, ed., Learning and Teaching Support Network 2001, <http://ltsn.mathstore.ac.uk/articles/maths-cao-series/nov2001/index.shtml>.

Solomon, A . (2001). Javamath: na API for internet accessible Mathematical Services. Lectures Notes Series on Computing – vol. 9. Computer Mathematics. **Proceedings of the Fifth Asian Symposium on Computer Mathematics (ASCM 2001):** 82-91.

Stroyan, K. D. “Calculus Wiz”; www.wolfram.com/products/student/calcwiz.

Wang, P.; Mikusa, M.; Al-Shomrani, S.; Chiu, D.; Lai, X.;Zou, X. Features and advantages of WME: a Web-based Mathematics Education System. <http://icm.mcs.kent.edu/reports/2005/ICM-200501-0003.pdf>

Warpechowski, M. Oliveira, J. P. M, Souto, M. A.; Camargo-Brunetto, M. A.; Freitas, V. and Proença, M. L. Adaptive Hypermedia in the AdaptWeb® Environment. In Adaptive Hypermedia 2004. Eindhoven-Netherlands. http://www.win.tue.nl/~laroyo/AH2004/Camera-ready/EAW2004/final_submissions/warpechowski_short.pdf

Wollrath, A. and Waldo, J. The Java Tutorial. . <http://java.sun.com/docs/books/tutorial/rmi/>.