

Learning *Online* Network with CAPA

Author's Tutorial And Manual

27th March 2003

John Williamson, Felicia Berryman, Jeremy Bowers

Laboratory for Instructional Technology in Education
and the
Office of Medical Education Research and Development
Michigan State University

Contents

1	Introduction to LON-CAPA	4
1.1	About This Manual	4
1.2	Login as Course Author	4
1.3	Author Remote Control	4
2	Creating Content Using LON-CAPA	7
2.1	Description of the Construction Space	7
2.2	How to Create New Content Pages	7
2.3	How to Edit Existing Content Pages	8
2.4	Creating Online Problems Using LON-CAPA	9
2.5	Problem Types	9
2.6	Foils	9
2.6.1	Radio Response Problems	10
2.6.2	Option Response Problems	10
2.6.3	String Response Problems	10
2.6.4	Numerical Response Problems	10
2.6.5	Formula Response Problems	10
2.7	Creating Radio Response Problems	10
2.7.1	Randomization	14
2.8	Option Response Problems	14
2.8.1	Option Response Problems with Concept Groups	14
2.8.2	Example: Concept Group	15
2.8.3	Example: Matching Problem	16
2.8.4	Creating Option Problems	16
2.8.5	Simple Option Response: No Concept Groups	18
2.9	Creating a String Response Problem	19
2.10	Creating Numerical Response and Formula Response Problems	20
3	Publishing Your Resources	21
3.1	What is Metadata?	21
3.2	Publishing A Resource	21
4	Creating A Course: Maps and Sequences	24
4.1	Creating Sequences	24
4.2	Creating a Simple .sequence With The Simple Editor	24
4.3	Creating a Simple .sequence With The Advanced Editor	26
4.4	Page Maps	30

4.5	Creating a Course: Top-level Sequence	30
5	Numerical Response And Formula Response Questions	30
5.1	The Parts of a Numerical Response Problem	31
5.2	Simple Numerical Response Answer	33
5.3	Simple Script Usage	34
5.3.1	Variables in Scripts	35
5.3.2	Variables in the Text Block	35
5.3.3	Variables in the Answer Block	36
5.4	Calling Functions	36
5.4.1	Numerical Response Randomization	37
5.5	Dynamic, Randomized Problems: Putting It All Together	37
5.6	Units, Format	38
5.7	For More Information	38
5.8	Formula Response	38
5.8.1	Sample Specifications	40
5.8.2	Formula Notes	41
5.8.3	Exaple Formula Response	41

1 Introduction to LON-CAPA

LON-CAPA is a web-based interface that helps to organize and present your course website, deliver and manage problems, and manage student enrollment. All author functions are done through a web browser (Netscape 4.x or higher, a recent Mozilla, or IE 5+ required).

At this time, you should have:

- developed your objectives for your course.
- developed your problems for input into LON-CAPA and determined the appropriate question formats.

1.1 About This Manual

Throughout this manual, keywords and phrases literally present on the computer screen will be referred to in **bold type**. Function names and scripts will be shown in a **typewriter font**.

Much of this document can be used as a tutorial that will introduce you to the authoring system. In particular, chapters ?? through 4 comprise a basic tutorial that can get you started using LON-CAPA, even with no previous LON-CAPA experience.

For additional help, visit our FAQ at <http://help.loncapa.org/>.

1.2 Login as Course Author

To begin using LON-CAPA, you first need to log in to your account on LON-CAPA. Open your web browser and navigate to your local LON-CAPA URL. You will be presented with a log in screen.

Fill in the Username and Password boxes with your information. Then press the Login button. This will take you to your LON-CAPA User Roles menu.

Note: Your Username and Password will be given to you by your system administrator.

Both are case sensitive, so make sure you type them with the correct case.

1.3 Author Remote Control

The Author Remote Control will automatically load whenever you log in to LON-CAPA as the course instructor. The Author Remote Control is a separate window in your browser, and is automatically sized and placed in the upper left of the screen. The Remote Control is a tool that allows you to switch between functions and roles within LON-CAPA.

When you move your mouse over the buttons in the remote, the sixteen gray boxes will show a reminder of what that button does.

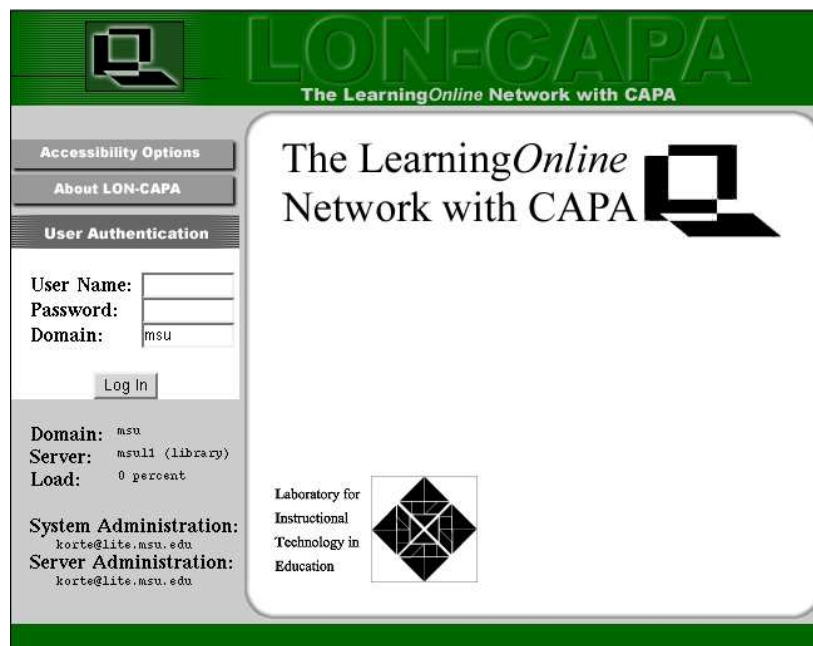


Figure 1: LON-CAPA Log in screen

- **ROLES (CHOOSE ROLE)** allows you to select which user role to assume for this session.
- **COM (COMMUNICATION)** allows you to access the communication functions in the system.
- **CUSR (USER ROLES)** brings up a page that allows you to create new users and change user privileges.
- **CSTR (CONSTRUCT)** displays the construction space for your account.
- **RES (RESOURCE SPACE)** allows you to browse the LON-CAPA network directory.
- **SRC (SEARCH LIBRARY)** brings up a screen that lets you search the LON-CAPA resources using multiple criteria.
- **PREF (PREFERENCES)** brings up a screen that allows you to change some preferences.
- **EXIT (LOGOUT)** will log you out of the LON-CAPA system.



Figure 2: Author Remote Control

2 Creating Content Using LON-CAPA

LON-CAPA provides three types of resources for organizing your course website. LON-CAPA refers to these resources as Content Pages, Problems, and Maps. Maps may be either of two types: Sequences or Pages. You will use these LON-CAPA resources to build the outline, or structure, for the presentation of your course to your students.

- A **Content Page** displays course content. It is essentially a conventional HTML page. These resources use the extension “.html”.
- A **Problem** resource represents problems for the students to solve, with answers stored in the system. These resources are stored in files that must use the extension “.problem”.
- A **Sequence** is a type of **Map** which is used to link other resources together. The users of this resource can use directional buttons on their remote or the NAV button to follow the sequence. Sequences are stored in files that must use the extension “.sequence”. Sequences can contain other sequences and pages.
- A **Page** is a type of **Map** which is used to join other resources together into one HTML page. For example, a page of problems will appear as a problem set. These resources are stored in files that must use the extension “.page”.

2.1 Description of the Construction Space

The Construction Space is the section of LON-CAPA where you create and manage your course resources. The figure explains what each button does.

2.2 How to Create New Content Pages

Content Pages are HTML documents that display the course information you are presenting.

Many users use tools such as Dreamweaver to create web pages. To upload HTML files generated with such tools, you can use the **Browse** button in the Construction Space, locate your HTML file, and use the **Upload File** button to create a content page in LON-CAPA. Remember to upload any graphics your generated web pages may have included.

To create new Content Pages, do the following:

1. Click the **CSTR** button on the LON-CAPA remote. Your web page will change to your Construction Space.



Figure 3: Construction Space

Contents of the Construction Space:

Button Name	Description
Publish this Resource	Opens the Resource Publishing window.
List Directory	Lists the contents of the current working directory
Copy	Type a new name in the entry box to make a copy the current resource
Browse	Helps you select a file to upload
Upload File	Uploads the selected file to your Construction Space
Retrieve Old Version	Load an older version of a resource if you have multiple versions
Delete	Deletes the current resource
Rename	Type a new name in the associated entry box to rename a resource
New Subdirectory	Type a name in the entry box to create a new directory

2. In the Location bar of your browser, type in the full URL of the new Content Page. Make sure the last part of the URL ends with “.html”, for example, *http://www.s10.lite.msu.edu/priv/username/new_resource.html* . Press the Return or Enter key.
3. Type the content into the editor, *OR* copy and paste HTML source code obtained through the use of some other HTML authoring program into the editor.
4. Optionally, click the **View** button to preview your Content Page.
5. Finally, click the **Save this** button *OR* click the **Save and then attempt to clean HTML** button.

Repeat this process as many times as necessary to create your Content Pages.

If you’re following this as a tutorial, create at least one content page, which we’ll use later as raw material. Visit the FAQ at <http://help.lon-capa.org/> if you get “unmatched tag” warnings.

2.3 How to Edit Existing Content Pages

You may edit any Content Pages that have been created.

To edit Content Pages:

1. Click the **CSTR** button on the LON-CAPA Remote. Your web page will change to your Construction Space.
2. Click on the link for the name of the Content Page to edit. The Content Page editor will load and display the current edition of the Content Page.
3. Press the **Edit** button. Edit the HTML code, or copy and paste HTML source code into the editor.
4. Finally, click the **Save this** button *OR* click the **Save and then attempt to clean HTML** button. If you do not do this, your work will not be saved.

Once you've saved your page, you can click the **View** button to preview your Content Page.

2.4 Creating Online Problems Using LON-CAPA

If you're following this as a tutorial, go ahead and make one of each of these problem types now. We'll be using them later as raw material to assemble maps and sequences.

While several problem types are listed here, in LON-CAPA, all problems are actually the same. All problems are written in XML, which can be obtained and edited with the **EditXML** button you'll see. The problem types listed in this manual are actually just templates. As your knowledge advances, you may wish to play with the XML representation directly to see what you can do.

2.5 Problem Types

In this manual we will cover five basic types of problems: Radio Response, Option Response, String Response, Numerical Response, and Formula Response. You will need to identify which types of problem you want to use and create appropriate questions for your course.

The problem editor gives you a testing area where you can try your problems out, with several different randomizations by varying the **Random Seed**. If you answer a problem correctly and can no longer enter new answers, you can get the answer field back by hitting the **Reset Submissions** button.

2.6 Foils

In the LON-CAPA system, a **Foil** is the statement after the drop-down box or radio button in a Radio Response or Option Response problem. Foils do not need to be text; they can be images or other resources.

2.6.1 Radio Response Problems

Radio Response problems present a list of foils, with buttons in front. The student can select *one* of these statements by clicking the appropriate radio button.

2.6.2 Option Response Problems

Option Response problems present foils to the student with drop-down boxes. The student can select the matching choice for the foils from a list of choices. Optionally, the foils may be bundled into Concept Groups, and the system will select one foil from each displayed group to display to the student.

2.6.3 String Response Problems

String Response problems are problems in which the student submits a string of characters for the answer. Examples of string response questions are vocabulary tests, short answer and entering chemical formulas.

Note that it is easy to abuse String Response problems. For instance, consider the question “Who wrote ‘Huckleberry Finn’?” If you tell the system the answer is “Mark Twain”, and a student answers “Twain”, the system will mark it wrong. If they answer “Samuel Clements”, then the student will definitely get it wrong. There is some room for flexibility in the string processing, but it can be difficult to get it all right. Before you use a String Response problem, be sure you can easily characterize correct answers.

2.6.4 Numerical Response Problems

Numerical Response problems are answered by entering a number and (optionally) a unit, such as 2.5 m/s². Tolerance and required significant digits can be specified as well.

2.6.5 Formula Response Problems

Formula response problems asks the student to type in a formula as an answer. If the answer is $2x^2 + 4$, the student is allowed to type “2*x*x+4”, “x*x + x*x + 4”, “2*x^2 + 4 - 10”, or any other equivalent expression. Formula Response problems have many of the same characteristics of Numerical Response problems, including the ability to run scripts, dynamically generate answers, etc.

2.7 Creating Radio Response Problems

To create a Radio Response problem, create a new resource as described in section 2.2. This is a “problem” resource so the URL must end in “.problem”. You should see a screen as in

The screenshot displays the LON-CAPA interface for creating a 'Formula Response Problem'. The interface is organized into several horizontal sections, each with a title bar and a content area. The sections are as follows:

- Script**: A light beige section with a 'Delete' button and a dropdown arrow.
- Text Block**: A yellow section with a 'Delete' button and a dropdown arrow.
- Response: Formula**: A green section with a 'Delete' button and a dropdown arrow, and an 'Insert' dropdown.
- Answer:**: A green section with a text input field.
- Sample Points:**: A green section with a text input field.
- Parameters for a response**: A pink section with a 'Delete' button and a dropdown arrow. It contains fields for 'Name' (tol), 'Type' (tolerance), 'Description' (Numerical Tolerance), and 'Default'.
- Single Line Text Entry Area**: A blue section with a 'Delete' button and a dropdown arrow. It contains a 'Size' field with the value 50.
- Hint**: A light blue section with a 'Delete' button and a dropdown arrow, and an 'Insert' dropdown.

Figure 4: Formula Response Problem

The requested file /~jerf/new.problem doesn't exist. You can create a new problem

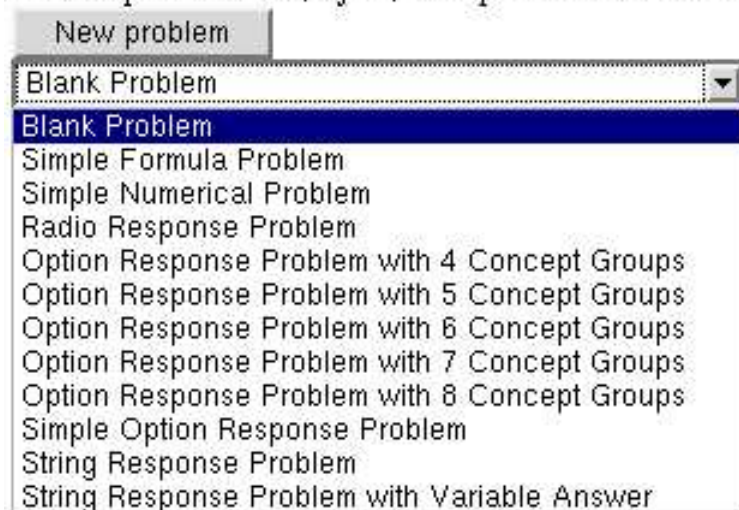


Figure 5: Creating A New Problem Resource

figure 5. You will need to specify the question text and foil statements.

1. In the drop-down option box, select **Radio Response Problem**, and click the **New Problem** button.
2. Click the **Edit** button above the sample problem to enter edit mode. You should see an editing screen.
3. In the **Text Block** at the top of the problem, remove the sample text and type the question for your problem. For example, "What is two plus two?"
4. Locate the **Response: One of N statements** element. In the **Max Number of Shown Foils** text box, place the number of foils you wish to display to the student.
5. Locate **Foil 1**. Remove the text that is in the text box and put the *correct answer* for the problem in the **Text Block**. For example, "Four." Make sure this is set to **true** in the **Correct Option** field.
6. Below it, you will see **Foil 2**. Remove the text in the text box and put an *incorrect answer* for the problem. For instance, "Purple." Make sure this is set to **false** in the **Correct Option** field.

View EditXML undo

Submit Changes

Insert:

Text Block

Enter the text of the question here.

Response: One of N statements

Max Number Of Shown Foils:

Collection of Foils

Foil

Name: Correct Option:

Text Block

This foil1 and it it is true. All other answers must be "false" or "unused".

</foil>

Foil

Name: Correct Option:

Text Block

Figure 6: Radio Response Creation Form

9.

The screenshot shows a web-based interface for creating content. At the top, there's a yellow header bar with a 'Group' dropdown and an 'Insert:' dropdown. Below this, there's a 'Text Block' section. It has a 'Delete:' dropdown and a text area with the placeholder text 'Add hint text here.' The interface is framed with a green border.

Figure 7: Hint Element

7. Repeat the previous step until you've filled in all of the other incorrect answers you wish to offer the students.
8. Once you've filled in all the incorrect answers, change the **Correct Options** on the other foils to **Unused**.

Scroll down to the Hint element. Type some text that will help students when they answer incorrectly. You may delete the hint by selecting **Yes** from the **Delete** drop-down box.

10. Click the **Submit Changes** button located at the top of the frame. If you do not do this, none of your changes will be saved.

The **Correct Option** drop down box controls whether or not a given answer will be accepted as a correct answer. If it is set to **true**, that answer will be considered a correct answer. Any number of foils can be marked **true**, but only one will be shown to any given student. If it is set to **false**, it will be considered an incorrect answer. If it is set to **Unused**, the system will not use that foil.

2.7.1 Randomization

LON-CAPA will randomize what choices are presented to each student, and randomize the order they are presented in. If you wish to present each student the same choices, make sure the **Maximum Number of Shown Foils** box contains the number of incorrect answers, which will force them to all be displayed.

2.8 Option Response Problems

2.8.1 Option Response Problems with Concept Groups

Each Option Response problem can have three parts:

Edit EditXML Random Seed: 1023307703 Change Reset Submissions ☐ Show All Foils

Enter question text here.

▼ This is statement ThreeA of concept Three. True

▼ This is statement OneC of concept One. True

▼ This is statement FourB of concept Four. False

▼ This is statement TwoA of concept Two. True

Submit Answer

Tries 0/2

Figure 8: Option Response Problem

1. The Concept Groups
2. The options for the students to select, by default “True” and “False”
3. The hint for the student

Each **Concept Group** has some number of foils representing questions which are conceptually related. Option Response Problem Templates are available for 4 and 8 Concept Groups. When the Option Response problem is presented to a student, the LON-CAPA system will randomly select one foil from each Concept Group and present it to the student. In order to receive credit for the problem, the student must select the corresponding option from the drop-down box for each given foil.

2.8.2 Example: Concept Group

Example: A Concept Group may contain the following True/False questions:

- “Mark Twain” is the pen name of Samuel Clemens.
- Mark Twain wrote “The Call of the Wild”.
- Mark Twain wrote “Huckleberry Finn”.
- Mark Twain spent most of his life in the Congo.

For each foil, the author marks it True or False. When the student logs on and attempts to answer this question, the student will see only one of the four choices for that concept group. They then go on to do the remaining three to seven Concept Groups in this question before submitting their answer.

2.8.3 Example: Matching Problem

Option Response problems can be used as matching problems.

For example, you might want to ask the student to match musical compositions with their composers. You could create an Option Response problem with 4 Concept Groups, and place the following four foil groups each in its own concept group:

- Claire de Lune, Ballade (Debussy)
- The Pastoral Symphony, The Ninth Symphony (Beethoven)
- Sleeping Beauty Suite, The Dance of the Sugar Plum Fairies (Tchaikovsky)
- Slavonic Dances, New World Symphony (Dvorak)

You could then add the following options to the option list:

- Debussy
- Beethoven
- Schubert
- Tchaikovsky
- Dvorak

The same answers can be used more than once, or not at all, as you see fit. It is conventional to place such a warning in the **Text Block** describing the problem to the students.

2.8.4 Creating Option Problems

To create an Option Response problem, create a new resource as described in section 2.2. This is a “problem” resource so the URL must end in “.problem”. You should see a screen as in figure “Option Response Editor”.

1. In the drop-down option box as seen in figure 5, select **Option Response Problem with N Concept Groups**, where N is the number of Concept Groups you wish the problem to have, and click the **New Problem** button.
2. Click the **Edit** button above the sample problem to enter edit mode. You should see the Option Response page open up.

Text Block		Delete:	<input type="button" value="v"/>
<div>In class on Wednesday, we discussed the phases of cellular division. Recall our discussion and answer the following True or False questions.</div>			

Multiple Option Response Question		Delete:	<input type="button" value="v"/>
Max Number Of Shown Foils: <input type="text" value="4"/>			
Select Options	Add new Option:	<input type="text"/>	Delete an Option: <input type="button" value="v"/>
Insert: <input type="button" value="v"/>			
Concept Grouped Foils		Delete:	<input type="button" value="v"/>
Insert:		<input type="button" value="v"/>	
Concept: <input type="text" value="Division Phases"/>			
Foil	Delete:	<input type="button" value="v"/>	Insert: <input type="button" value="v"/>
Name: <input type="text" value="Anterophase"/>		Correct Option: <input type="text" value="False"/>	
Text Block		Delete:	<input type="button" value="v"/>
<div>Anterophase is a phase of mitosis.</div>			

Figure 9: Option Response Editor

3. Replace the text in the **Text Block** with text that explains the conditions for your problem.
4. Locate the **Max Number of Shown Foils** element and type a number from 1 to 8 to display that number of questions. You cannot display more than one foil from each concept group, so this option will only reduce the number of foils displayed, if it is less than the number of concept groups in your Option Response problem.
5. Now you must define the options the students can select. For each option you wish to add to the Option Response question, type the option into the **Add new Option** box in the **Select Options** section, then hit the **Save Changes** button. If you do not hit the **Save Changes** button, your option will not be selectable below. (You will delete unwanted options in the last step.)
6. Now, you need to define the question foils. Look for the foil with the name “One”. Type the question into the text box, and select the correct option for that question from the **Correct Option** drop-down menu. Click **Submit Changes** to save this question foil. Repeat this step for all remaining foils.
7. Locate the foils that are not being used. In their **Delete** menus, set the value to **Yes**. Once you’ve set the Delete menu value correctly for all the foils, click the **Save Changes** button.
8. In the Hint area, provide a helpful hint for users who get the problem incorrect, and click the **Save Changes** button.
9. Make sure all the options you want to delete are not used for any of your foils. If a deleted option is used in a foil, it will appear in a text box in the **Correct Option** area for that foil. To make the drop-down box reappear, type an option already defined in the **Select Options** field, and hit **Submit Changes**. A drop-down box will reappear. To delete the irrelevant options from the Option Response question, select that option from the **Delete an Option** drop down, and hit the **Save Changes** button. Do that for each option you wish to remove.

2.8.5 Simple Option Response: No Concept Groups

If you select **Simple Option Response** from the drop-down box, you get a template that will allow you to enter up to eight foils with no grouping. The system will randomly mix these foils when presenting them to the student. You can have more foils than the **Max Num of Shown Foils** so that each student will not have the identical foils.

View EditXML undo

Submit Changes

Insert: ▼

Text Block Delete: ▼

```
The 3 types of string answers are: <br />
cs: Case Sensitive <br />
ci: Case Insensitive <br />
mc: Multiple Choice, Order of characters unchecked.
<br /> <br />
The answer is NaCl and it is case sensitive.
```

Response: String Delete ▼ Insert: ▼

Answer: NaCl Type: cs ▼

Single Line Text Entry Area Delete ▼

Size:

Hint Delete ▼ Insert: ▼

Text Block Delete: ▼

Add hint text here..

Submit Changes

Figure 10: String Response Editor

2.9 Creating a String Response Problem

To create a String Response problem, create a new .problem resource (described in 2.2). This is a “problem” resource so the URL must end in “.problem”.

1. In the drop-down option box as seen in 5, select **String Response Problem**, and click the **New Problem** button.
2. Click the **Edit** button above the sample problem to enter edit mode. You should see the String Response editor page open up, which should look something like what you see in the “String Response Editor” figure.
3. Clear the text from the **Text Block** at the top of the problem, and type in your problem.

4. In the **Answer Box**, type the correct answer.
5. Select the answer condition from the drop-down. There are three cases to choose from:
 - (a) **cs**: This means “Case Sensitive”. For example, this is useful in chemistry, where HO and Ho are completely different answers. The student must match the case of the answer.
 - (b) **ci**: This means “Case Insensitive”. The system does not use the case of the letters to determine the correctness of the answer. If the correct answer is “car”, the system will accept “car”, “CAR”, “Car”, “caR”, etc.
 - (c) **mc**: This means “Multiple Choice”. The student’s answers must contain the same letters as the question author’s, but order is unimportant. This is usually used to give a multiple choice question in the question’s **Text Block**, which may have several correct parts. If the author sets the correct answer as “bcg”, the system will accept “bcg”, “cbg”, “gcb”, etc., but not “bc” or “abcg”.

It is conventional to inform the students if the problem is case sensitive, or that the order of the answers doesn’t matter.

6. Optionally, locate the **Single Line Text Entry Area** block and set a length in the Size box. This will only affect the size of the box on the screen; if you set the box size to 2, the student can still enter 3 or more letters in their answer.
7. Scroll down to the **Hint** element, and type some text that will help students when they answer incorrectly, or delete the hint by setting the **Delete** field to **Yes**.
8. Click the **Submit Changes** button.

2.10 Creating Numerical Response and Formula Response Problems

Numerical Response problems are answered by entering a number and an optional unit. For instance, a numerical response problem might have an answer of $2m/s^2$. Formula Response problems are answered by entering a mathematical formula. For instance, a numerical response problem might have an answer of $x^2 + 11$. The answer may be in any equivalent format. For instance, for $x^2 + 11$, the system will also accept $x * x + 11$ or $x^2 + 21 - 10$.

Creating Numerical Response and Formula Response problems starts the same as the other problem types, but because of the power of Numerical Response and Formula Response problems, they are covered in their own section after the end of the tutorial. For more information about these problem types, please see section 5 for Numerical Response problems and section 5.8 for Formula Response problems.

3 Publishing Your Resources

In order to make the content you've created available for use in courses, you must publish your content. LON-CAPA provides an easy interface for publishing your content pages, problem resources, and sequences. You can specify title, author information, keywords, and other metadata. LON-CAPA uses this metadata for many things, and it's important to fill the metadata out as accurately as possible.

3.1 What is Metadata?

Metadata is *data about data*. Metadata can often be thought of as a label on some bit of information that can be useful to people or computer programs trying to use the data. Without metadata, the person or computer trying to use the original information would have to just guess what the original data is about. For instance, if you create a problem and neglect to say in the title or subject of the problem what it is about, then a human who wants to use that problem would have to read the problem itself to see what it was about, which is much more difficult than just reading a title. A computer trying to do the same thing would just be out of luck; it is too stupid to understand the problem statement at all.

One example of metadata is the <title> of a web page, which usually shows up in the title bar of the browser. That's information about the web page itself, not actually part of the web page. People use the title information when they bookmark a page, so they know what the page is. Search engines use it as a clue about the content of the web page.

3.2 Publishing A Resource

To publish a resource, log in and choose your role to be an Author. Then click **CSTR** to go to your construction space. You should see something like the "Construction Space for Publishing". Click on the **Publish** button for the resource you wish to publish. You'll get a metadata screen that should look something like the "Publishing Metadata Screen" figure. Fill out the form. If you are creating resources that may be used in several courses, you should talk with the other authors and establish some sort of standard title and subject scheme in advance.

The **Language** is the language the problem is written in. The **Publisher/Owner** is the LON-CAPA user who owns the problem.

The **Keywords** and the **Abstract** are more information about the problem.

The **Keywords** are words that are strongly connected to your problem; for instance a physics problem about a pulley might include "pulley" as a key word. LON-CAPA pulls out words used in the text of the resource for you so you can just click on their check boxes to make them keywords. **Additional keywords** allows you to add any keyword to your

Construction Space Directory /

Actions	Name	Title	Status	Last Modified
Publish	numericalResponse.problem		Unpublished	Thu May 30 13:44:50 2002
Publish	optionResponse.problem		Unpublished	Thu May 30 12:45:38 2002
Publish	radio.problem		Unpublished	Thu May 30 09:43:14 2002
Publish	stringResponse.problem		Unpublished	Thu May 30 13:17:04 2002
Publish	temp.problem		Unpublished	Wed May 29 15:14:48 2002

Figure 11: Construction Space for Publishing

problem that are not actually in the problem. For instance, on that same problem a physicist might add the keyword “statics”, even though it doesn’t appear in the original problem, because Physics uses that as a classification of problem type. **Additional Keywords** are also useful when publishing graphics.

You need to set the copyright and distribution permissions in the **COPYRIGHT/DISTRIBUTION** drop-down. This setting controls who is allowed to use your resource as follows:

- **Limited to courses in the domain published** means that only courses running in the same domain as you can use your content. Talk to your LON-CAPA administrator if you want more information about your domain.
- **Free** means that anyone can find and use the resource.
- **Private - visible to author only** means that it can’t be used for any course.
- **Public - no authentication required** means anyone can find and use the resource.

Not all of these choices may be visible, depending on the nature of the resource.

Now when you click **Finalize Publication**, your resource will be published and usable (unless you set the distribution to “private”).

If you’re following this as a tutorial, publish your resources so we can use them in the next section.

Title:

Author(s):

Subject:

Keywords:

<input type="checkbox"/> anterophase	<input type="checkbox"/> cellular	<input type="checkbox"/> class	<input type="checkbox"/> concept	<input type="checkbox"/> discussion	<input type="checkbox"/> division	<input type="checkbox"/> fourb	<input type="checkbox"/> hint	<input type="checkbox"/> mitosis	<input type="checkbox"/> oneb	<input type="checkbox"/> onec	<input checked="" type="checkbox"/> phase
<input type="checkbox"/> phases	<input type="checkbox"/> questions	<input type="checkbox"/> recall	<input type="checkbox"/> statement	<input type="checkbox"/> text	<input type="checkbox"/> threea	<input type="checkbox"/> twoa	<input type="checkbox"/> twob	<input type="checkbox"/> wednesday			

Additional Keywords:

Notes:

Abstract:

LANGUAGE:

Publisher/Owner:

COPYRIGHT/DISTRIBUTION:

Figure 12: Publishing Metadata Screen

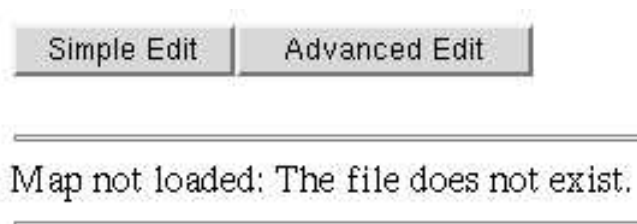


Figure 13: Map Editor Selection

4 Creating A Course: Maps and Sequences

In order to create a useful course, we need to arrange our raw materials so that students can use them.

4.1 Creating Sequences

A **Sequence** is a series of resources that can be navigated using the **NAV** remote control button, or by using the arrow keys on the remote control.

To create a Sequence resource, create a new resource as described in section 2.2. This is a “sequence” resource so the URL must end in “.sequence”. After you enter in the URL ending in “.sequence”, you should see a screen as in figure 13. You can use either the advanced editor or the simplified editor.

4.2 Creating a Simple .sequence With The Simple Editor

After creating a new .sequence resource and getting the editor selection prompt (as in the “Simple Map Editor” figure), click the **Simple Edit** button to get to the simple map editor, which appears in the figure.

The Simple Editor can create .sequences and .pages which are linear, which means they have no branches or conditions.

On the right side of the simple editor is the **Target**, which represents the map you are currently building. On the left side is the **Import** area, which represents a work area you can use for your convenience to load and manipulate resource you may wish to include in your map. Using the three buttons in the middle of the screen, from top to bottom you can cut things out of the Target, copy from the Target to the Import, and copy from the Import to the Target, respectively.

On both sides of the screen, you can do a Group Search and a Group Import. A Group Search allows you to run a search, then import selected results from that search into either

Map not loaded: The file does not exist.

/~jerf/totallyNew.sequence

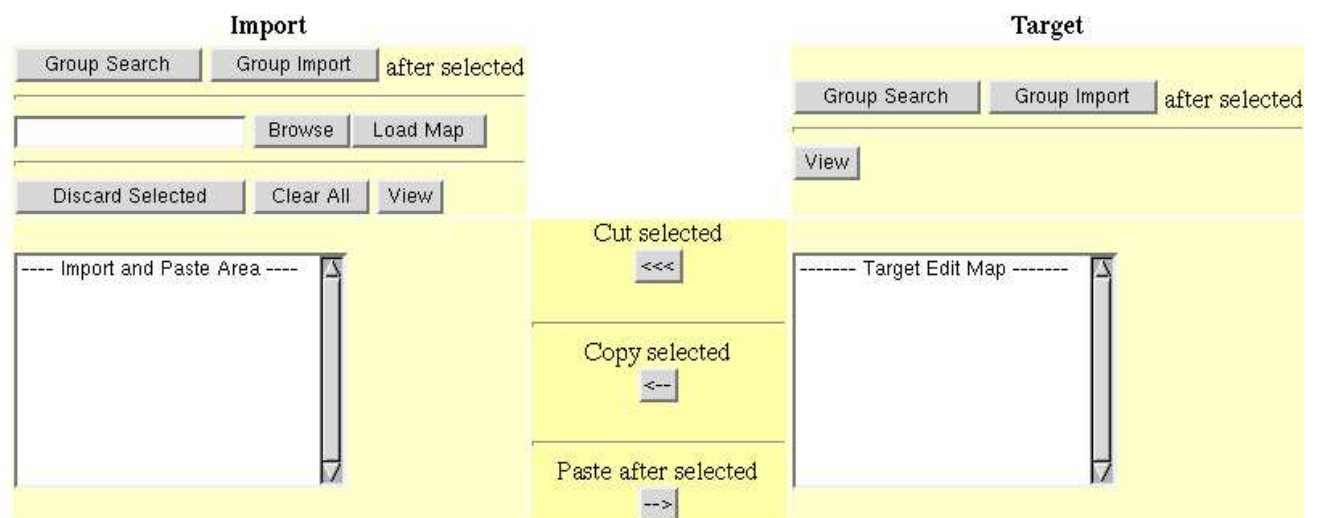


Figure 14: Simple Map Editor



Figure 15: Initial Map Editor

directly into your Map, or into your Import space. Checkboxes will appear next to the results in the Group Search, and you can click the resource you wish to add to your map in the order that you want them added. After you select the resource, you will be presented with a screen that allows you to change the order of the selected resources, then you will be able to import the selected resources and work with them.

A Group Import works in a similar fashion, but allows you to use the LON-CAPA network browser to select your resources.

On the Import side, you can also browse for another Map, and load the resources used in that map into your Import workspace. You can also discard the selected resources, clear all the resources, and view the selected resource from the buttons on the Import side of the screen.

Both list boxes support standard multi-select mechanisms as used in your OS.

4.3 Creating a Simple .sequence With The Advanced Editor

After creating a new .sequence resource and getting the editor selection prompt (13), click the **Advanced Edit** button to get to the advanced map editor. You should see the initial map editor as shown in the “Initial Map Editor” figure. Note there are two windows: One is the workspace, and one is a secondary window which will contain information as you add resources.



Figure 16: After clicking **Start** in the Map Constructor

Click the **Start** box. You'll see what you see in the "After clicking **Start** in the Map Constructor" figure. Click **Link Resource** in the secondary window, then click on the **Finish** box. After that, click **Straighten**. You should see something looking like the "Straightened Map" figure. This creates a simple map that flows from beginning to end.

To insert a resource into the flow, click the black line with two arrows, seen between the **Start** and **Finish** boxes in the "Straightened Map" figure. In the secondary window, you'll see something like the "Inserting a Resource" figure. Click **Insert Resource Into Link**. A new resource box will appear in the link. Click the resource, which will have the label **Res**.

3. Click **Browse**, and the **Network Directory Browser** will appear, looking something like the "Network Directory Browser" figure. Press the **SELECT** button that is next to the resource you want to place in the chosen resource box. Once you've done that, if you look back at the window that popped up when you clicked on **New Resource**, you'll see something like the "Resource Chosen" figure. You can type the **URL** and **Title** into the secondary window, if you prefer, following the format you see when you've successfully browsed to a resource. After you click **Save Changes**, your changes will be set, and the icons for the resource will appear in the **Res** box, as shown in figure



Figure 17: Straightened Map

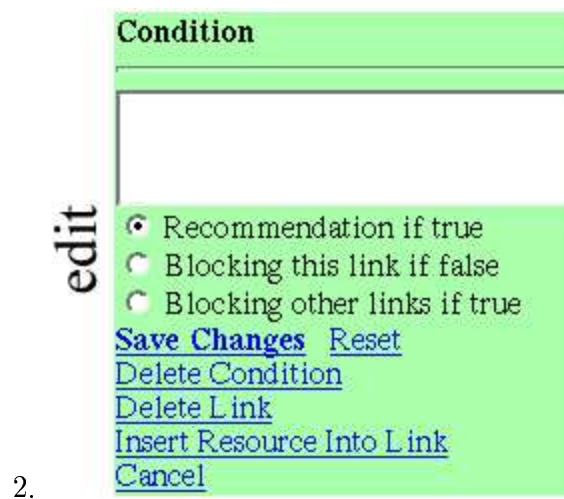


Figure 18: Inserting a Resource

The LearningOnline With CAPA Network Directory Browser

Display file attributes

☐ Size ☐ Last access ☐ Last modified ☐ All versions
☐ Author ☐ Keywords ☐ Language

Name	
	Up
	user1
	you
<input type="button" value="SELECT"/>	aPage.html (metadata)

Figure 19: Network Directory Browser

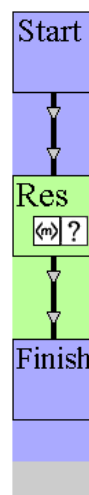


Figure 20: Resource Chosen

20. Click **Save Map** in the bar above your map to save the map.

Clicking on the left icon for a resource will open a new browser window with an informational page about that resource. Clicking on the right icon for a resource will open a new browser window taking you to the rendering of that resource.

4. Repeat steps two and three for as many resources as you'd like to bind together into one page. You can insert the new resources anywhere you'd like.
5. When you are done adding resources, click the **Save Map** link to save the map.

In addition to manually adding in resources, the Advanced Editor also has the ability to import resource in the same way that the Simple Editor can: From a LON-CAPA network browser window, from a Group Search, or from another Map.

The advanced editor has many more capabilities which you can explore.

4.4 Page Maps

Creating a .page map is the same as creating a sequence map, except that when choosing the name of the resource, the URL will end with “.page”. Then all resources you add in the map editor will appear on one page together.

Pages are often used to connect problems in a homework set.

4.5 Creating a Course: Top-level Sequence

In order to view sequences, they need to be part of a **course**.

Courses have a Top-level Map which defines the whole course. This top-level map will often itself contain maps corresponding to homework assignments, chapters, or units. To view your maps, you will need to make them part of a course. Only Domain Coordinators can make courses and set their Top-level maps, so work with your Domain Coordinator if you need to view your maps.

5 Numerical Response And Formula Response Questions

Numerical Response problems are very powerful. In fact, they are so powerful it would be impossible to fully explain what is possible in a simple document. This chapter will focus on getting you started with Numerical Response problems, and showing you some of the possibilities, with no prerequisite knowledge necessary. The more you learn, the more you will find you can do.

Create a new Course

Course Title

Top-level Map

 [Browse](#)

Course ID/Number (optional)

Course Coordinator

Username:

Domain: ▼

Open Course

Figure 21: Creating a New Course

If you like, you can follow this chapter along as its own tutorial. Create a Numerical Response problem using the instructions in section 2.2, ending your resource name with “.problem”, and create a new **Simple Numerical Response** problem.

5.1 The Parts of a Numerical Response Problem

A Numerical Response problem has seven major parts by default:

1. The **Script**. The script is the heart of advanced Numerical Response problems. It can be used to decide some of the parameters of the problem, compute the answer to the problem, and do just about anything else you can imagine. The Script language is **Perl**. You do not need to know Perl to use the **Script** block, as we will be stepping through some advanced examples in this chapter, but knowing Perl can help.
2. Like other problem types, the **Text Block** is used to display the problem the student will see. In addition, you can place variables in the **Text Block** based on computations done in the **Script**.
3. The **Answer** is the answer the system is looking for. This can also use parameters from the **Script** block, allowing the answer to be computed dynamically.

The image shows a software interface for editing numerical response questions. It consists of several stacked, color-coded sections:

- Script** (tan background): Contains a text area with the placeholder "#Enter the computations here" and a "Delete" button with a dropdown arrow.
- Text Block** (yellow background): Contains a text area with the text "What is 2 + 2?" and a "Delete" button with a dropdown arrow.
- Response: Numerical** (green background): Includes an "Insert:" dropdown, an "Answer:" field with the value "4", and "Unit:" and "Format:" fields.
- Parameters for a response** (pink background): Two identical sections. Each has a "Delete" button with a dropdown arrow and fields for "Name", "Type", "Description", and "Default".
 - First parameter: Name: "tol", Type: "tolerance", Description: "Numerical Tolerance", Default: "5%".
 - Second parameter: Name: "sig", Type: "int_range,0-16", Description: "Significant Figures", Default: "0,15".
- Single Line Text Entry Area** (blue background): Includes a "Delete" button with a dropdown arrow and a "Size:" field.
- Hint** (light blue background): Includes an "Insert:" dropdown and a text area containing "Text Block" and "Delete:" with a dropdown arrow.
- Text Block** (yellow background): Contains a text area with the text "This should be easy!" and a "Delete" button with a dropdown arrow.

Figure 22: Numerical Response editor

4. A **tolerance** parameter determines how closely the system will require the student's answer to be in order to count it correct.

For technical reasons, it is almost never a good idea to set this parameter to zero. Computers can only approximate computations involving real numbers. For instance, a computer's [decimal] answer to the simple problem $\frac{1}{3}$ is "0.3333333333333331". It *should* be an infinite series of 3's, and there certainly shouldn't be a "1" in the answer, but no computer can represent an infinitely long, infinitely detailed real number. Therefore, for any problem where the answer is not a small integer, you *need* to allow a tolerance factor, or the students will find it nearly impossible to exactly match the computers idea of the answer. You may find the default too large for some problems.

There are two kinds of tolerance. For some answer a and a tolerance t ,

- (a) an **Absolute** tolerance will take anything in the range $a \pm t$. So if $a = 10$ and $t = 2$, then anything between 8 and 12 is acceptable. Any number in the tolerance field *without* a % symbol is an absolute tolerance.
 - (b) a **Relative** tolerance will take anything in the range $a \pm at$, where t is interpreted as a percentage. Any number in the tolerance field *followed by* a % symbol is a relative tolerance. For example, $a = 10$ and $t = 10\%$ will accept anything between 9 and 11.
5. A **significant figures** specification tells the system how many significant figures there are in the problem, as either a single number or a range of acceptable values, expressed as **min,max**. The system will check to make sure that the student's answer contains this many significant digits, useful in many scientific calculations. For example, if the problem has three significant digits, the significant digit specification is "3", and the answer is "1.3", the system will require the students to type "1.30", even though numerically, "1.3" and "1.30" are the same. A significant figure specification of "3,4" means both "1.30" and "1.300" are acceptable.
 6. The **Single Line Text Entry** area, as in other problem types, allow you to manipulate the text entry area the student will see.
 7. Finally, the **Hint** should contain text which will help the students when they answer incorrectly.

5.2 Simple Numerical Response Answer

Along with showing the Numerical Response editor, figure 22 also shows the parameters for one of the simplest possible types of numerical response. The **Text Block** has the problem's

question, which is the static text “What is $2 + 2$?” The **Answer** is “4”. The **Hint** has been set to something appropriate for this problem. Everything else has the default values from when the problem was created.

If you create a problem like this, hit **Submit Changes**, then hit **View** after the changes have been submitted, you can try the problem out for yourself. Note the last box in the HTML page has the answer LON-CAPA is looking for conveniently displayed for you, along with the range the computer will accept and the number of significant digits the computer requires when viewed by an **Author**.

As you’re playing with the problem, if you use up all your tries or get the answer correct but wish to continue playing with the problem, use the **Reset Submissions** button to clear your answer attempts.

5.3 Simple Script Usage

Totally static problems only scratch the surface of the Numerical Response capabilities. To really explore the power of LON-CAPA, we need to start creating dynamic problems. But before we can get to truly dynamic problems, we need to learn how to work with the **Script** window.

A script consists of several **statements**, separated by **semi-colons**. A **statement** is the smallest kind of instruction to the computer. Most problems will be built from several statements.

A script can contain **comments**, which are not interpreted as statements by the computer. Comments start with **#**, and go to the end of that line. Thus, if a line starts with **#**, the whole line is ignored. Comments can also begin in the middle of a line. It is a good idea to comment more complicated scripts, as it can be very difficult to read a large script and figure out what it does. It is a *very* good idea to adopt some sort of commenting standard, especially if you are working in a group or you believe other people may use your problems in the future.

- One of the simplest statements in LON-CAPA is a **variable assignment**. A **variable** can hold any value in it. The variable name must start with a **\$**. In the **Script**, you need to assign to variables before you use them. Put this program into the **Script** field of the Numerical Response:

```
$variable = 3;
```

This creates a variable named **variable** and assigns it the value of “3”. That’s one statement.

Variable names are *case sensitive*, must start with a letter, and can only consist of letters, numbers, and underscores. Variable names can be as long as you want.

There are many variable naming conventions, covering both how to name and how to capitalize variables¹. It is a good idea to adopt a standard. If you are working with a group, you may wish to discuss it in your group and agree on a convention.

If you **Submit Changes** and **View** the problem, you'll see nothing has changed. That's because in order for a variable to be useful, it must be used. The variable can be used in several places.

5.3.1 Variables in Scripts

Variables can be used later in the same script. For instance, we can add another line below the `$variable` line as such:

```
$variable2 = $variable + 2;
```

Now there is a variable called `$variable2` with the the number "5" as its value.

Variables can also be used in *strings*, which are a sequence of letters. The underlying language of the script, Perl, has a very large number of ways of using variables in strings, but the easiest and most common way is to use normal double-quotes and just spell out the name of the variable you want to use in the string, like this:

```
$stringVar = "I have a variable with the value $variable.";
```

This will put the string "I have a variable with the value 3." into the variable named "stringVar".

If you are following this chapter as a tutorial, add the previous two lines to your **Script** and submit the changes for the problem. There's no need to view it; there's still no visible change.

5.3.2 Variables in the Text Block

Once you've defined variables in the **Script**, you can use them in the **Text Block**. For example, using the previous three-line script we've created so far, you can place the following in the **Text Block**:

```
See the 3: $variable<br />  
See the string: <b>$stringVar</b><br />
```

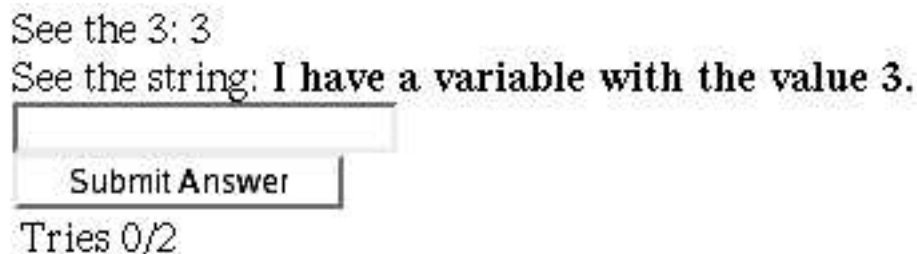


Figure 23: Result of Variables in the Text Block

If you save that and hit **View**, you should get what you see in figure 23. Note how the “\$variable” was turned into a 3, and the “\$stringVar” was turned into “I have a variable with the value 3.”

5.3.3 Variables in the Answer Block

You can use variables in the **Answer** part of the question. This means you can compute an answer to a question. If you set the answer of the question to be **\$variable**, **Save Changes** and **View** it, you’ll see that LON-CAPA is now expecting “3.0” as the answer, plus or minus 5%.

5.4 Calling Functions

With variables, you can store strings or numbers. **Functions** allow you to manipulate these strings or numbers. Functions work like mathematical functions: They take some number of arguments in, and return one argument, usually a number or a string for our purposes. There are a lot of functions available in LON-CAPA. You can see a complete list at <http://mileva.lite.msu.edu/loncapadocs/homework/homework5.html>. For now, let’s just look at some simple examples.

In the **Script** block, function names start with **&**. Some example function calls are shown in figure 24. You can see that functions can take either variables, numbers, or the results of other function calls as parameters. The **&sin** function returns the sine of an angle expressed in radians. **&pow** raises the first parameter to the power of the second parameter. **&abs** returns the absolute value of the argument.

¹The author favors `capsOnNewWords`. Some people use `underscore_to_separate_words`. Many use uppercase letters to specify constants like `PI` or `GOLDEN_MEAN`. Some people always `StartWithCapatalization`. What’s really important is to be consistent, so you don’t have to guess whether the variable you’re thinking of is `coefFriction`, `CoefFriction`, `COEF_FRICTION`, or something else.

```

$a = -3.0;
$b = &sin($a);
$c = &pow(3.0, &abs($a));

```

Figure 24: Some Function Calls

The screenshot shows the LON-CAPA problem editor interface with the following sections:

- Script** (light blue background): Contains the following code:


```

      $slope1 = &random(1, 4, .1);
      $slope2 = &random(-4, -1, .1);
      $yint1 = &random(-10, 10, .1);
      $yint2 = &random(-10, 10, .1);
      $answer = ($yint2 - $yint1)/($slope1 - $slope2);
      
```
- Text Block** (yellow background): Contains the following text:


```

      For the lines defined by the following equations:<br /><br />
      <tt>y = $slope1 x + $yint1<br />
      y = $slope2 x + $yint2<br /><br /></tt>
      At what value of <i>x</i> do these lines intersect?
      
```
- Response: Numerical** (green background): Includes a "Delete" button and an "Insert:" dropdown menu.
- Answer:** (green background): Includes a text input field containing "\$answer", a "Unit:" label, and a "Format:" dropdown menu.
- Parameters for a response** (pink background): Includes a "Delete" button and a table of parameters:

Name	Type	Description	Default
tol	tolerance	Numerical Tolerance	.05

Figure 25: Slope Problem Parameters

5.4.1 Numerical Response Randomization

If you're doing this as a tutorial, try a few random seeds to see what happens.

5.5 Dynamic, Randomized Problems: Putting It All Together

Now you have all the tools to create those wonderful dynamic, randomized problems that you've seen in LON-CAPA.

Try filling out your problem with the parameters shown in the "Slope Problem Parameters" figure.

When creating randomized problems, you want to make sure that the problems always have an answer. Consider what might happen if I had chosen the two slopes *both* with the expression `&random(-1.0,1.0,.2)`. One out of ten students would get a problem where both slopes were equal, which has either no solution (for unequal y-intercepts) or an infinite number of solutions (for equal slopes and y-intercepts). Both of these cause a division-by-

zero error on the division that computes the answer. There are many ways to avoid this, one of the easiest of which is picking one slope negative and one positive. This same problem can show up in many other places, too, so be careful.

5.6 Units, Format

Numerical Response problems can require units. In the problem editing form, place the desired unit in the **Unit** field. For information about what units the system accepts, see <http://capa4.lite.msu.edu/demolibrary/Links/UnitsSymbolsT2.html>. The computer will accept the answer in any of its accepted unit formats. For example, if the answer to a problem is “1ft”, the computer will accept “12in” as correct.

You can format the number displayed by the computer as the answer. For instance, if the answer is one-third, the computer will display that it computed “.33333333” as the answer. If you’d like to shorten that, you can use the **Format** field. Format strings like “2E” (without the quotes) will display three significant digits in scientific notation. Format strings like “2f” will display two digits after the decimal point.

5.7 For More Information

The full power of Perl is well outside the scope of this document. Looking in the function list at

<http://mileva.lite.msu.edu/loncapadocs/homework/homework5.html>

can give you some ideas. O’Reilly has some good Perl books. The Perl 5 Pocket Reference will contain more than what you need to know to use LON-CAPA, available at

<http://www.oreilly.com/catalog/perlpr3/>.

If you have any problems, consult <http://help.loncapa.org/fom/cache/5.html>. If you don’t find the answer to your problem, please help us expand the FAQ by submitting a new pending question.

Our advanced users often come to prefer the XML interface for the problems, available through the **EditXML** buttons. Covering the XML format is beyond the scope of this manual, but you can learn a lot by using the editor to make changes and seeing what happens to the XML.

5.8 Formula Response

Formula response problems asks the student to type in a formula as an answer. If the answer is $2x^2 + 4$, the student is allowed to type “2*x*x+4”, “x*x + x*x + 4”, “2*x^2 + 14 - 10”, or any other equivalent expression. Formula Response problems have many of the

Script

Text Block

Response: Formula

Answer: Sample Points:

Parameters for a response

Name: Type: Description: Default:

Single Line Text Entry Area

Size:

Hint

Figure 26: Formula Response Problem

same characteristics of Numerical Response problems, including the ability to run scripts, dynamically generate answers, etc.

5.8.1 Sample Specifications

As you may know, it is extremely difficult to determine whether a given expression is exactly equal to another expression in general. For example, is $\sin 2x = 2 \sin x \cos x$? Symbolically proving it one way or another is impossible in general. Therefore, LON-CAPA uses a sampling system. If your answer and the student's answer agree at the sampling points within your given tolerance factor, the student's answer will be accepted, otherwise it will be rejected.

To specify where to sample the formulas for determining whether the student's answer is correct, you need to put a sampling specification in the **Sample Points** field. The sampling specifications takes the following format:

1. A comma separated list of the variables you wish to interpret,
2. followed by "@" (not in quotes),
3. followed by any number of the following two things, separated by semi-colons:
 - (a) a comma separated list of as many numbers as there are variables, which specifies one sampling point, OR
 - (b) a comma separated list of as many numbers as there are variables, followed by a colon, followed by another list of as many numbers as there are variables, followed by a #, followed by an integer.

The first form specifies one point to sample. The second form specifies a range for each variable, and the system will take as many random samples from that range as the number after the #.

For $2x^2 + 4$, with one variable "x", one could specify

- "x@2", which will sample the answers only at 2. (This is generally a bad idea, as the student could get lucky and match at that point)
- "x@1:5#4" will takes 4 samples from somewhere between 1 and 5.
- "x@1:5#4;10" will takes 4 samples from somewhere between 1 and 5, and also sample at 10.

For $2x^2 + 3y^3$, which has two variables, one could specify

- "x,y@4,5:10,12#4;0,0", which take four samples from the box determined by the points (4, 5) and (10, 12), and also sample the point (0, 0).

5.8.2 Formula Notes

- The formula evaluator can not handle things of the form “ $x + - y$ ”. If you have a random variable that may be positive or negative (as in the example following this section), you can try wrapping the references to that variable in parenthesis. As always, it is a good idea to try out several randomized versions of your problems to make sure everything works correctly.
- **Never use relative tolerance in Formula Response problems.** Relative tolerance is poorly defined in Formula Response problems. Always use absolute tolerance.

5.8.3 Exaple Formula Response

A very simple formula response problem:

In the **Script**, place the following:

```
$slope = &random(-5.0,5.0,.5);  
$yint  = &random(-5.0,5.0,.5);  
$answer = “$slope*x + ($yint)”;
```

- In the **Text Block**, place the following: “For a line with slope \$slope and y-intercept \$yint, what is y equal to?”
- In the **Answer**, place the following: \$answer
- Set the Tolerance to .000001.
- Set the **Sample Points** to x@0;1;2;3 .