



Soliloquy Composer

A tool for building libraries for Soliloquy

Version 1.0

The University of Chicago — Center for the Study of Languages

Introduction



Soliloquy is an innovative iPad application that enables instructors to present audio-based exercises to students, who may record audio of their own, and submit the results to instructors.

Students register their names and user IDs with the Soliloquy application, then open a link distributed by the instructor. The link downloads a library of exercises. When the student selects an exercise, Soliloquy downloads the instructor's audio for the exercise. The student can then play it, and can hold down a "record" button to provide responses when they are called for. Soliloquy provides tools for editing the student's responses, and for returning the completed exercise to the instructor.

Soliloquy supports multiple users, exercises, libraries, and courses.

This article shows how instructors can create downloadable packages of exercises, prepare them for distribution, and receive students' completed work, using Soliloquy Composer, a desktop application for Mac OS X 10.6 and later.

Overview

Products

In order to present exercises, instructors must provide two things:

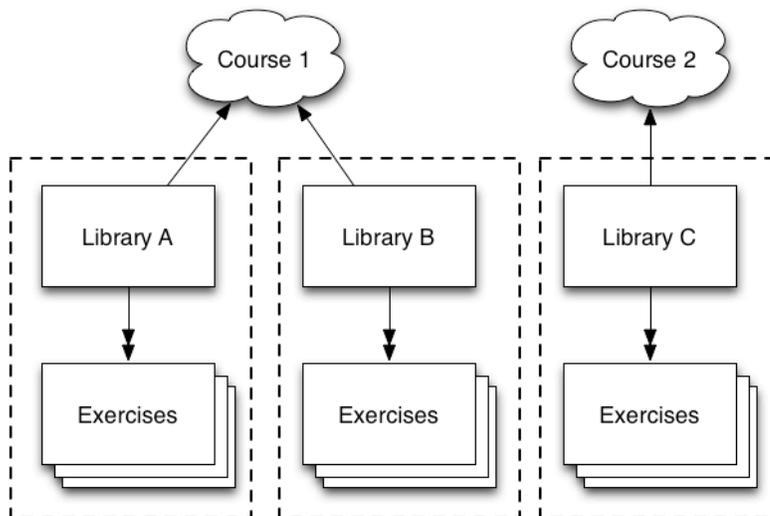
- A *Library file* that encapsulates related exercises. Library files are served to Soliloquy by HTTP (Web) servers.
- A specialized *Cookie URL* that can be distributed by email or through a web page. When the URL is opened from the iPad's Safari web browser or from Mail, it launches Soliloquy and directs it to download the library file.

Soliloquy Composer allows you to create, save, and edit library files, and copy cookie URLs to distribute the libraries.

Concepts

Soliloquy presents exercises in a three-level hierarchy, providing students a natural framework for navigating among them.

- *Exercises* are the student's unit of work. They include a title, a URL for the audio file (the *master audio*) the instructor poses as a task for the exercise, and a rich-text description. It can also include a URL for a WebDAV directory to receive completed exercises over the network. See "WebDAV," under "Completed Exercises," later in this article.



Soliloquy preserves each student's recordings for an exercise, so students can share the iPad on which Soliloquy is installed, and freely switch among exercises.

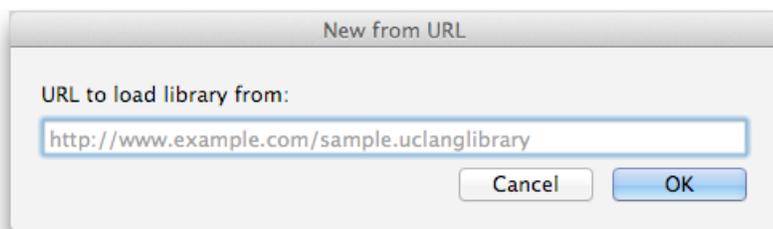
- *Libraries* are the instructor's unit of work, and the unit by which Soliloquy acquires exercises. They group related exercises into logical packages. Students navigate through libraries to find exercises. A Soliloquy Composer document contains one complete library.

- *Courses* group libraries for the purpose of navigation, and to help Soliloquy distinguish among different libraries that might share the same title. No one file keeps a course: It is up to the instructor to make sure that libraries refer to the same course if they are to be grouped. Soliloquy Composer includes a “Course well” that makes this process possible; see the description of the Course field in “The Library Document Window,” below.
- *Students* are users of Soliloquy who have registered themselves with the app. Registration consists of providing a full name, as well as a user ID to identify them with file servers, and in the completed exercise files. Once registered, students can pass Soliloquy around, selecting their own registrations before beginning work. Instructors need not worry about student registrations.

Using Soliloquy Composer

There are two ways to create a new library file.

1. Selecting New from the File menu, or activating Soliloquy Composer when no library file is already open, will create a new, unsaved library document.
2. Selecting New from URL... (command-L) from the File menu lets you enter a URL for an existing library file. Soliloquy Composer will download it and create a *new, unsaved* library document with the same contents. Soliloquy Composer is not able to edit remotely-hosted library files in-place (unless they are loaded from a file server visible in the Finder).



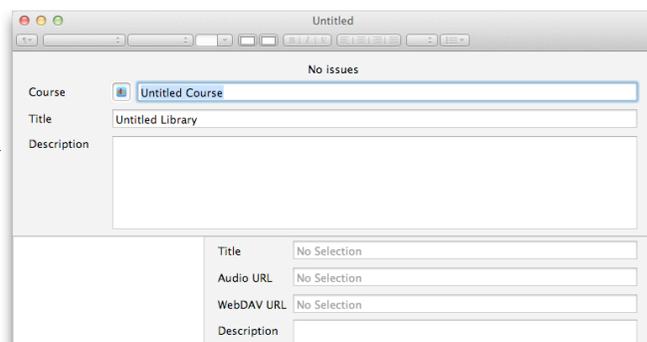
You can edit an existing library file by double-clicking its icon in the Finder, dragging it to Soliloquy Composer's icon in the Dock, or using the Open... or Open Recent commands in the File menu. On Mac OS X Lion, Soliloquy Composer will restore any previously-open libraries when you reopen it.

The Library Document Window

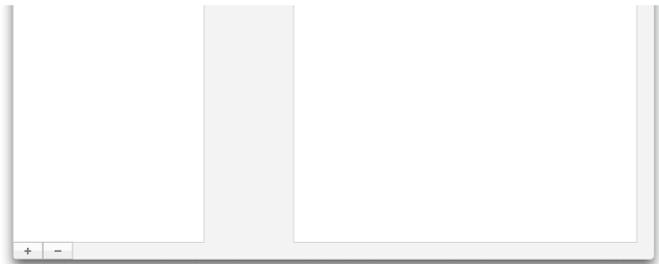
The library editor window consists of three parts:

- At the top is the *library description area*, where you specify the name of the library and course as they will be presented to the student. There is also an area for composing a rich-text description of the library.

Note: Version 1 of Soliloquy does not use the library description.



- At the left of the lower area is the *task list area*, that displays the names of the exercises in the library, and allows you to add to, delete from, or reorder the list.
- At the right of the lower area is the *task editing area*, for providing a title, master-audio URL, an optional WebDAV URL to receive completed exercises, and a rich-text editor for the exercise's description.



Library

The library description area contains three fields.

- The **Course** field sets the name of the course — the organizing container for one or more libraries — that will be displayed in Soliloquy. Courses do not exist in any shared file; each library that shares a course must refer to it separately.

Internally, Soliloquy Composer and Soliloquy maintain a unique identifier so that courses will not be confused even if they have the same name. Simply entering the same course name in two different libraries will result in each having a different course. The icon in the well to the left of the Course field represents the combination of course name and unique identifier. If you drag the icon to the well in another library, the name and identifier will be transferred to that library, and the two libraries will be sorted into the same course.

The Course field must be filled in.

- The **Title** field will identify the library in Soliloquy. It is possible to give more than one library in a course the same name. This will *not* merge the two libraries; they and their contents will be listed separately.

The Title field must be filled in.

- The **Description** text area is a rich-text editor for a description of the library. This is provided for future versions of Soliloquy; Soliloquy 1.0 does not use the contents of this field.

The course Description editor may be left blank.

Exercises

Click the + button in the task list area to create a new exercise. A new item will appear in the task list, and some of the fields in the task editing area will fill with example settings. To add another exercise, click the + button again; clicking the - button will delete the selected exercise; and dragging an exercise up and down the list will reorder the list.

Note: You can duplicate an exercise from another library by dragging it from the source library's list to the destination library's list.

- The **Title** is the name of the exercise as it will be presented to the student. Also, when a

completed exercise is exported to an audio file, the exercise's title will comprise part of the file's name, and the title of the exported track will be the exercise's title.

It is an error to give more than one exercise in a library the same name. Exercises in separate libraries, even if they are part of the same course, may share a name.

- The **Audio URL** is the full URL of the exercise's master file, which the student will play and into which she will record her responses.
- The **WebDAV URL** is the full URL of the WebDAV *directory* that will receive completed exercise files. WebDAV is an optional feature of Soliloquy; if you do not wish to enable it for a particular exercise, leave this field blank.
- The **Description** text area is a rich-text editor for a description of the library. The contents are displayed at the top of Soliloquy's exercise view.

About Description Text

The description editors for libraries and exercises are full rich-text editors. Using the Format menu (and the formatting bar at the top of the window in Mac OS X Lion), you have complete control of text styles and fonts. The Format menu also has options to show a text ruler, in which you can set margins, as well as left, center, and right tab stops.

Further, there are options for inserting and managing special constructs, such as tables, lists, and links. All of these features are standard in rich-text editors in Mac OS X. If you are familiar with the TextEdit application, you are already familiar with the description editors.

Soliloquy responds to most links by referring them to the iPad app most appropriate for the URL: Google Maps URLs will open Maps; YouTube links, the YouTube app; `mailto:` links will open Mail; and most other HTTP links will open in Safari.

The important exception is links to image files. If the link URL ends with `.jpg`, `.jpeg`, `.gif`, or `.png` (whether upper- or lower-case), Soliloquy will download the image and display it in place of the link. The description editors cannot embed images directly, but this mechanism lets you achieve the same effect.

Description text is transmitted to Soliloquy as a single HTML file. While almost all features of the text as written in Soliloquy Composer will be displayed in Soliloquy, the translation process is not perfect, and some formatting will be lost. Further, you should keep in mind that not all fonts available in Mac OS X are available on the iPad. iOS will make the closest substitution, but if a substitution must be made, the presentation will not be the same.

As of early 2012, the following fonts were available in iOS 5. Most are also available in Mac OS X Lion:

Academy Engraved	American Typewriter	Apple Color Emoji
Apple SD Gothic Neo	Arial	Arial Hebrew
Arial Rounded MT Bold	Bangla Sangam	Baskerville

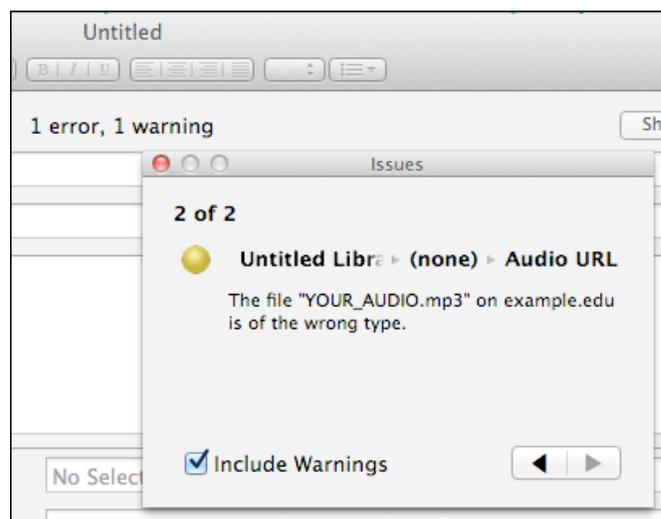
Bodoni 72	Bodoni 72 Oldstyle	Bodoni 72 Smallcaps
Bodoni Ornaments	Bradley Hand	Chalkboard
Chalkduster	Cochin	Copperplate
Courier	Courier New	DB LCD Temp
Devanagari Sangam	Didot	Euphemia UCAS
Futura	Geeza Pro	Georgia
Gill Sans	Gujarati Sangam	Gurmukhi
Heiti SC	Heiti TC	Helvetica
Helvetica Neue	Hiragino Kaku Gothic	Hiragino Mincho
Hoefler Text	Kailasa	Kannada Sangam
Malayalam Sangam	Marion	Marker Felt
Noteworthy	Optima	Oriya Sangam
Palatino	Papyrus	Party LET
Sinhala Sangam	Snell Roundhand	Tamil Sangam
Telugu Sangam	Thonburi	Times New Roman
Trebuchet	Verdana	Zapf Dingbats
Zapfino		

Note It is often not necessary to select non-Latin fonts when writing description text. If the current font does not include a glyph for a letter (such as a Georgian character in Palatino), Mac OS X and iOS will automatically substitute a glyph from a font that includes it.

Validation

It is important that the contents of a library should be complete and consistent. If information is missing, or URLs are misformatted or do not point to data of the expected type, Soliloquy may not function properly, or may reject exercises. Soliloquy Composer informs you of potential problems through its *Issues inspector*, which you can make visible with the Issues Inspector command in the Window menu, or by clicking the Show button in the top-right corner of the library window.

Note: Soliloquy Composer's validation service cannot be turned off.



There are two kinds of issue: *Errors* are conditions that will produce an unusable library. *Warnings* are conditions that *may* cause problems, but Soliloquy Composer cannot say for certain that they will. For example, if you do not provide a URL for an exercise's master audio, Soliloquy Composer will flag the condition as an error; if the URL leads to a password-protected file, Soliloquy Composer will be unable to pursue the link further, and will warn that it cannot fully verify the URL.

A full count of errors and warnings appears at the top of the library window. The Issues inspector

describes each issue, page by page. You can step through the list by clicking the arrow controls in the lower-right corner. The contents of the Issues inspector are:

- A count of the current issue number, and the total number of issues.
- A circle indicating the severity of the issue. Errors get a red dot, warnings a yellow dot.
- A “path” showing the location of the issue, usually by library and exercise title, and the specific field in which the issue was detected. Outer elements may be truncated to fit the available space; hovering the mouse pointer over an element will temporarily expand it to show its full title.
- A description of the issue.
- A checkbox for including or excluding warnings from the inspector display. Unchecking this box is useful if you wish to deal with errors first, but it is recommended that you not release a library before you have considered all warnings.
- Left and right arrowheads to navigate back and forth in the issues list.

Here are the errors and warnings that Soliloquy Composer presents, as of early 2012. The list may change as issues are identified, changed, or removed:

Errors

- Courses must have names.
- Libraries must have titles.
- Exercises must have titles.
- Each exercise must have a different title.
- Audio URLs must be present.
- Audio URLs must be well-formed, and begin with an `http` or `https` scheme.
- Audio URLs must point to servers that exist.
- Audio files, if retrievable, must not exceed 20 MB in size. Soliloquy will reject larger files.
- Servers for Audio URLs must not report an error on access (such as file-not-found).
- WebDAV URLs may be blank, but if present, must be well-formed, and begin with an `http` or `https` scheme. (Use of `http` will result in a warning.)
- Links in description text must be syntactically-correct absolute URLs.
- Descriptions must not directly contain images (and in fact, the editors should prevent this).

Warnings

- Audio URLs should end with a known extension for an audio file.
- The server for an Audio URL may demand a password, but no further checking will be possible.
- If further checking is possible, an Audio URL should point to an audio file.
- WebDAV URLs should use the `https` scheme.

Note: Soliloquy Composer's validation services can detect whether URLs for links in descriptions, or for WebDAV directories, are syntactically valid, but cannot verify that the links are reachable.

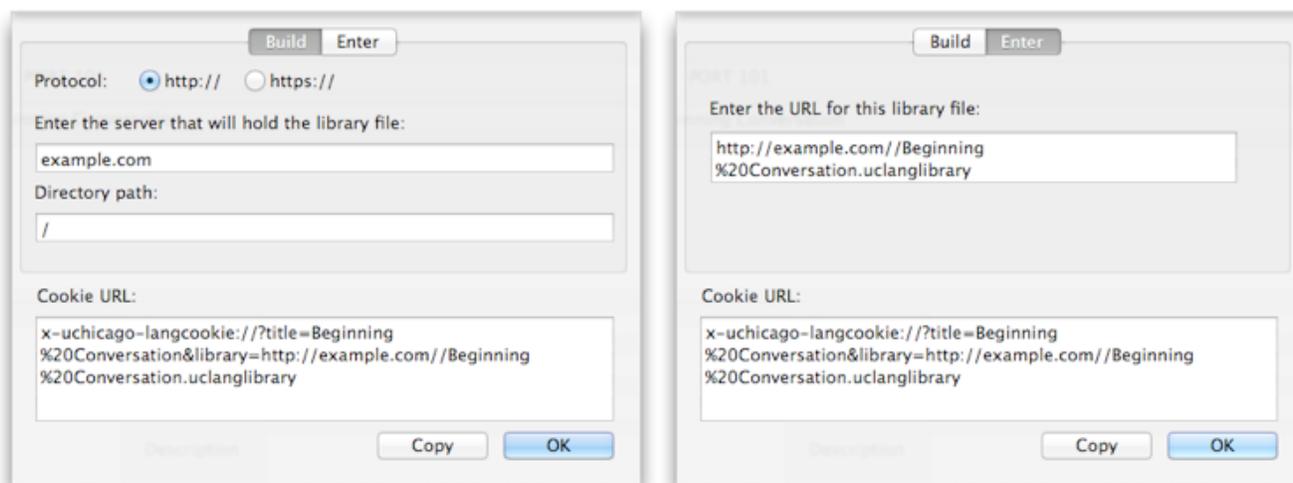
Note: Validation of URLs requires a network connection. If Soliloquy Composer cannot reach the Internet, it will present spurious errors and warnings.

Cookie URLs

Soliloquy finds library files through “cookie” URLs. These are different from Web URLs that can access files through the `http` or `https` protocols. The `x-uchicago-langcookie` scheme used in cookie URLs informs Soliloquy of the title of a library, and where to find it on the Web. When Soliloquy is installed on an iPad, opening a cookie URL will open Soliloquy and cause it to download the library. Cookie URLs have no effect outside of an iPad with Soliloquy installed.

You distribute libraries by distributing cookie URLs. You can simply paste a cookie URL into an email message, or use it as the destination of a link in a Web page (this code is broken into lines for space, and so is not valid):

```
<a
  href="x-uchicago-langcookie://?title=Simple%20Conversation
    &library=http://example.edu/soliloquy/Remote-hosted.uclanlibrary"
>
  Download this conversation.
</a>
```



Note: An `x-uchicago-langcookie` URL may *contain* the URL at which the library file is to be found, but it is *not* that URL. Distributing the `http` or `https` URL for the library will just open the file in Safari, and will never reach Soliloquy.

To build a cookie URL for a library, bring the library's window to the front, and make sure you have saved the file (so Soliloquy Composer will have a file name to put into the URL). Then select URL Helper... (command-U) from the File menu.

There are two ways to build up a cookie URL.

Building from components

The Build tab of the URL helper allows you to assemble the cookie URL from the `http` or `https` URL

for the library file. The Build tab lets you enter each component of the URL:

- Select the protocol Soliloquy is to use for retrieving the library file — `http` or `https`.
- The server name is the unadorned domain name of the machine that will distribute the library file, such as `languages.example.edu`. Do not add a protocol (such as `http://` or `https://`), or a file path.
- The path within the server to the *directory* that will contain the library file.
- URL Helper will supply the file name.

Building from a complete URL

Alternatively, you may select the Enter tab, which exposes a single large text field. Type or paste the full URL for the library — including the file name — into the field. Soliloquy Composer will convert the Web URL into a cookie URL.

Copying the cookie URL

Either way you construct the URL, the completed cookie URL appears in the Cookie URL text area. It is not editable, but you can select its contents and copy them. Or, you can click the Copy button to put the cookie directly into the clipboard. The OK button closes the helper sheet. You can now paste the URL into an email message, or as the `href` attribute of an `<a>` anchor in a web page.

It is not necessary to password-protect the page from which you distribute cookie URLs. They contain no confidential information beyond the mere existence of the library. Password-protecting access to the library files themselves will suffice to protect confidentiality and intellectual property.

Refer to the **Security** section below for additional considerations in password-protecting Soliloquy assets.

Supporting Soliloquy

Caution: Before you take any action to make resources available to Soliloquy over the net, be sure to read and understand the [Security considerations](#) section at the end of this document.

Identifying Students

Students — users of the Soliloquy app — register themselves with the app before beginning work on exercises. Once registered, students can pass Soliloquy around; each student selects her own registration before resuming work.

Registrations consist of a full name, and a user ID. The user ID serves two purposes:

- It is embedded in completed exercise files, and is made a part of the files' names. For this purpose, it doesn't matter what user ID a student enters, so long as no two students with the same

user ID submit the same exercise.

- It is used as the user name when logging in to password-protected servers, whether for access to library and audio files, or for submitting exercises through a WebDAV file server. See the following sections on serving libraries and audio, and “WebDAV” under “Receiving Completed Exercises,” below. For this purpose, the student's user ID *must* match an ID that is permitted access to these services.

Only one user ID can be associated with a student. Soliloquy permits students to change their user IDs freely. Password dialogs in Soliloquy do *not* have a field for user names; the user ID of the current student is used in every case. Once a library or master audio file has been retrieved, it is not necessary to download it again.

Serving Libraries

Library files, the `.uclanlibrary` files that Soliloquy Composer saves, must be served by a Web server accessible through an `http` or `https` URL. They may be protected by a password wall. (Review the **Security considerations** section, above.) The URL may be a “ticket” URL, containing a code instead of a file name, that the server will redirect to the actual file.

Note: All of this means that it's easiest to arrange for storage and retrieval of your library files before using URL Helper to construct a cookie URL for the library.

Once a library file has been retrieved, it will not be necessary to download it again, and it follows that there is no need to re-present a password to access it.

It is possible to revise a library once it has been published, though this is not recommended. Simply save the `.uclanlibrary` file to the same URL, and have users tap on the cookie URL again. Soliloquy will bring its copy of the library up-to-date with the new library file. If you need to change a master audio file, you must save the new file under a different URL, amend the library file, and have users tap on the cookie URL.

Serving Audio

Master audio files must be served by a Web server accessible through an `http` or `https` URL. They may be protected by a password wall. (Review the [Security considerations](#) section, at the end of this document.) The URL may be a “ticket” URL, containing a code instead of a file name, that the server will redirect to the actual file.

If an audio URL is absent, or not syntactically valid, Soliloquy will reject the exercise, and it will not appear in the library's list. If Soliloquy cannot recognize the downloaded file as an audio file, it will be unable to play or record audio for that exercise. In either case, it will present an alert explaining the problem.

Soliloquy Composer's validation facility will catch most of these issues. It does not download complete audio files and confirm that they are apparently playable; and validation stops when it hits a password wall.

If the audio URL was syntactically valid, but could not be downloaded, switching to a different

exercise, then back to the one that had the problem, will retry the download.

Once an audio file has been retrieved, it will not be necessary to download it again, and it follows that there is no need to re-present a password to access it.

Serving Images

Library files cannot contain images; therefore it is not possible simply to drag an image into a description editor, and have the image packaged into the library. You may display an image in Soliloquy by entering text that describes the image, selecting it, and using the Link command in the Format menu to supply a URL linking to the image file on the Web. When Soliloquy detects such a link (which it will identify by the suffix of the file name), it will replace it with an embedded image.

In general, images “embedded” in exercise descriptions must be publicly-accessible, and have URLs that end with a recognized extension for an image file. Without an image suffix, Soliloquy will not be able to replace a text link with an embedded image.

Soliloquy will not solicit passwords for image links; however, if Soliloquy had recently downloaded a library or master audio that used the same server, with the same credentials, the credentials may still be valid for the image. Those credentials, however, will expire, and later users will not be able to see password-protected images. Instructors who want to test exercise descriptions should switch among exercises, and then wait a while, before seeing whether “embedded” images can be loaded independently.

Soliloquy Composer's validation services can detect whether URLs for links in descriptions are syntactically valid, but cannot verify that description links are reachable.

Image files are retrieved each time the descriptions that embed them appear.

Receiving Completed Exercises

When the student taps the Export button in the top-right corner of Soliloquy, the app prepares a completed exercise file by merging her responses into the master audio. The completed file is encoded as an MPEG-4 audio-only track (.m4a). iTunes and most other modern audio players will be able to play it.

The name of the file is:

```
exercise_title-student_ID-task.m4a
```

The file contains *metadata* that makes it easier to identify in iTunes or other programs that manage audio files:

- The track's **Title** is the exercise's title.
- The **Album** is the library's title.
- The **Artist** is the user ID (not the name) of the student.
- The **Release Date** is the moment at which the completed file was built.
- The **Genre** is “Soliloquy exercise.”

Once the completed file is built, the student is offered up to three options for submitting it.

iTunes File Sharing

“iTunes File Sharing” is an iTunes feature that is not well-known. Plug an iPad into the computer with which it has been paired. When it appears in iTunes' source list, select it, and then the Apps tab. At the bottom of the Apps view is a list of installed applications that can transfer documents into and out of the device. If the student selects iTunes as the export option, her completed exercise file will appear in this list. Drag it out of the list to save it to your computer.

Note: If the same registered student exports the same exercise through the iTunes option, any previous export file is overwritten.

iTunes File Sharing is always offered. Note that this option is *not* the same as adding the track to the iPad's Music (iPod) library; this is not possible in Soliloquy.

Email

If the student selects the Email option, Soliloquy will prepare an email message identifying the course, library, exercise, and student name; and attach the completed exercise. There are two important things to understand about identities:

- iOS is a single-user system, and iPads cannot readily change the accounts used by their email client. All exercises will be sent through one email account.
- As with the completed exercise file itself, the submission email will be marked with the name and ID of the student *registered* as the current user of Soliloquy. If the actual user does not select or create a student registration of his own, the email will identify the previous registered user, and not him.
- It is up to the student to fill in the instructor's email address.

Many email systems will not pass attachments above a certain size, typically one MiB. Soliloquy will not offer email as an option if the completed exercise exceeds one MiB in size. If the iPad is not set up with an email account, Soliloquy will not offer the email option.

WebDAV

WebDAV is a simple protocol, based on HTTP, that provides most of the features of a file server. Plug-ins are available for the open-source Apache web server (as a part of the standard distribution), and Microsoft's IIS server. Both Mac OS X and Microsoft Windows (XP or later) can mount WebDAV directories as remote volumes. If you provide an exercise with a URL for a WebDAV directory, you can have direct access to the files students submit; if there is no WebDAV URL, the option to export to WebDAV will not be offered.

The WebDAV directory specified in the exercise must exist before the first student submits a completed exercise. Soliloquy will not create directories.

Note: If the same registered student exports the same exercise through the WebdaV option, any previous export file is overwritten.

You will typically want students' uploads to be authenticated, so only members of a particular class can submit exercises. When Soliloquy's WebDAV receives a server's request for a password, it will present a dialog view to receive the password from the student. The dialog will not solicit a user name; Soliloquy will use the user name the student submitted when she registered with the app.

Setting WebDAV Up

At the University of Chicago, the easiest and most secure way to set up a WebDAV directory is to ask IT Services (via the TECHB@R) to provide a Webshare directory restricted to the members of your class. IT Services is working on a self-help option to make setup easier.

If you are at another institution, consult your IT department to see if a similar option is available.

If you must go it alone, the details of setting up WebDAV directories will vary according to the server and your needs.

Microsoft IIS, Mac OS X Server, and some Linux distributions offer WebDAV as an option in their administrative applications.

[This how-to](#) explains how to configure a non-server desktop Mac (10.5 or later) to serve WebDAV. The same procedure may apply to most UNIX-like systems. [This how-to](#) covers LDAP authentication and SSL link security.

Always back up any configuration files you edit before making any changes.

Some issues to consider are:

- Whether to create dedicated directories for all courses, or a single umbrella directory (Soliloquy will write only to the directory you specify for an exercise).
- Whether to use the passwords your institution provides each student, or set custom passwords for the purposes of WebDAV. We do not recommend a uniform password: Students may forget to switch Soliloquy to their own user IDs, and assigning unique passwords prevents one student's overwriting another's completed work.
- The directory privileges to accord instructors. We recommend reading, writing, and listing.
- The directory privileges to accord students. We recommend writing only. Soliloquy cannot list or read from WebDAV directories, but students may guess directory URLs and access others' work.
- To prevent malicious use of your WebDAV directories, set them up so they will accept only MPEG-4 (.m4a) files, and only of a reasonable size (30-40 MB would be a good starting point).

Security considerations

At the University of Chicago, a student's user ID will typically be the CNetID, authenticated through

the University's LDAP server. The University of Chicago has strict policies on how CNetIDs and passwords are to be used. Visit the [IT Services policies page](#) and click "Policy for Authenticating University of Chicago Users" for details.

The University of Chicago is required by law to keep certain information confidential and secure against tampering. It uses CNetIDs and passwords to guard these assets. Those credentials must *never* pass through the network without being secured by an encryption layer; if they are transmitted without protection, the University must treat them as compromised for all purposes, and is required to reset the offending passwords. In the case of Soliloquy transfers, the `https` protocol must be used for any library, audio, or WebDAV access for which you require a password.

Other institutions almost certainly have similar requirements. Consult the security office in your IT department.

For libraries and audio files, you have two choices:

- Don't put them behind password walls.
- If you must put them behind password walls, specify the `https` protocol in the URL, not `http`. The server must support secure communications through SSL.

You also have the option of not password-protecting WebDAV directories, but this opens the directories to abuse, including disruption of submitted exercises. Further, password protection prevents students from mistakenly overwriting each others' exercises. Instead, *always* use `https` in WebDAV URLs. Soliloquy Composer will warn you if a WebDAV URL does not use the `https` scheme.

Secure password protection requires that your server have a valid SSL certificate. For University of Chicago users only, [this page](#) outlines the procedures for obtaining and installing SSL certificates for servers at the University. The University of Chicago will not provide this service outside the University; consult your institution's IT department for assistance.

At the University of Chicago, the easiest way to set up a WebDAV directory is to create one on Webshare. Contact the TECHB@R for assistance; a self-service option will be available soon.