

Using Assembla in PracT_EX Production

Mark Eli Kalderon

Email eli@markelikalderon.com
Website <http://markelikalderon.com>
Address Department of Philosophy
University College London
Gower Street
London, WC1E 6BT
United Kingdom

Abstract A short introduction to the use of Assembla's tools and services in the collaborative production of your PracT_EX article.

1 Overview

So PracT_EX has accepted your proposed article or has invited you to write an article. Congratulations! From here on out, you will be working with the editor, a production editor, and any coauthors you may have to produce your masterpiece. A production editor will be working closely with you, offering advice about revisions and answering any technical questions you may have. This is a coordination problem of potentially international proportions—all of these individuals may very well be scattered quite literally across the globe. How are they to effectively coordinate to help you write your completed article?

Fortunately, PracT_EX appreciates the challenges of working with a globally distributed team and has addressed these challenges by using Assembla's tools and services¹. Having accepted your proposal, or issued you an invitation, the editor will send you an e-mail inviting you to join an Assembla team. This e-mail will contain a url leading to your start page where you can accept or decline your invitation, [Figure 1](#). To accept the invitation simply click on **Accept**.

1. <http://www.assembla.com>

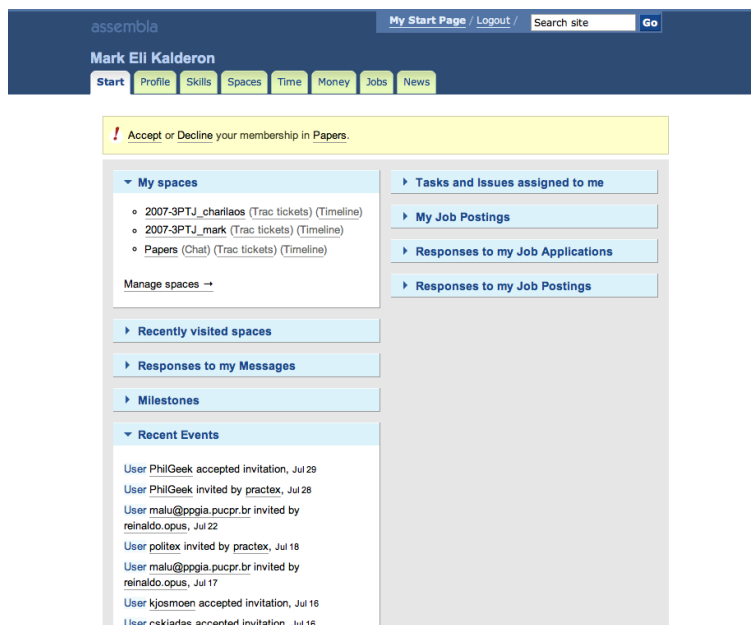


Figure 1: Assembla Start Page

On your start page, there is a portion called “My spaces”. Think of a space as where your project lives. There will be a list of one or more project spaces, depending on how many LaTeX projects you are involved with. Each space is associated with three links. The first link leads to your project space’s page, the second to a ticket list, and the third to a timeline. Sometimes, a chat tool is added. The ticket list and timeline are part of a trac tool to help you and your team track the progress of your article. Let’s review these in turn.

If you click on your project space’s page you can find a link to the trac tool, the ticket list, the timeline, and the Subversion url for the space. If activated by the admin, the chat tool can be accessed in My spaces or by a tab in the project space. As you may know, Subversion² is a free, open source version control system that keeps the history of the revisions of your document in an encrypted database that only you, your production editor, and any coauthors have access to. When a revision is committed to the Subversion repository, you, your production editor, and any co-author’s will receive an e-mail notifying them of the committed

2. <http://subversion.tigris.org>

revision along with the commit message. This e-mail will contain a url of the changeset, or diff, representing the changes that have been made. If you click on this link, it will open this page in your browser, [Figure 3](#). Changesets will be explained further later in this section.



Figure 2: Project Space

The ticket list is a list of prioritized messages about what needs to be done or comments, more generally, about revisions of your paper. Tickets offers several actions, such as active (leave the ticket open—the problem or issue it raises still persists) or closed (the problem or issue is now fixed or otherwise resolved). The link from your space’s page leads to the list of your active tickets. These are comments or queries that need to be addressed. To view the ticket simply click on an item on the list. The ticket system is an efficient way for the production editor to communicate with you or for you to communicate with your production editor or coauthors. To issue a ticket simply click on “New Ticket” on the trac tool and fill in the relevant fields. When a new ticket is issued, you will receive an e-mail notification.

The timeline is a list of project-related events ordered by date. These include tickets issued and changesets (a representation of the changes you or your coauthors have made to your paper since the previous committed revision). A change-set or diff only really works on plain text documents, so if you have committed

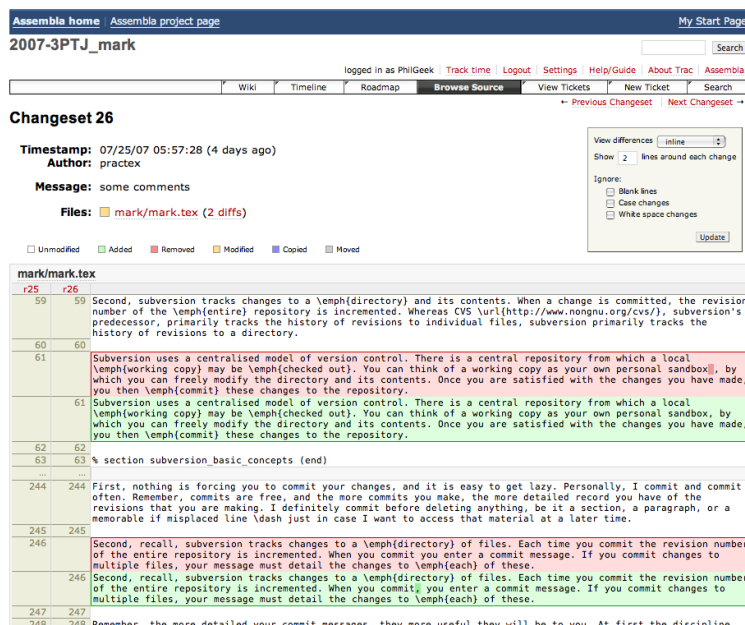


Figure 3: Changeset

the PDF generated by your \LaTeX source, or any associated graphics, the results can look scary. But don't be put off, because the changesets for your \LaTeX source are as informative as the changesets of binary files are ugly. (It is unnecessary and undesirable, anyway, to commit the PDF and auxiliary files such as .aux or .log files.)

The changesets can be accessed another way. In the trac tool, click on "Browse Source". You will see a folder representing your project materials. Clicking on this folder will reveal its contents. If you click on your \LaTeX source, it will be displayed in your web browser. At the bottom of this page is a button called "View changes". This is a link to a page that asks you for two revisions. Say you want to compare the 29th and 26th revision. By clicking on "View changes" again, the changeset, or diff, will be displayed.

The chat tab on the project space's page will open a page where you can chat with the production editor or any coauthors, if they are online. See Figure 4. The right column lists who on your team is available to chat and a link to view the history of the chat sessions. To chat, simply type your message in the text

field and press return. The ability to chat online with your production team and coauthors is an incredibly useful tool that can speed up the production process.

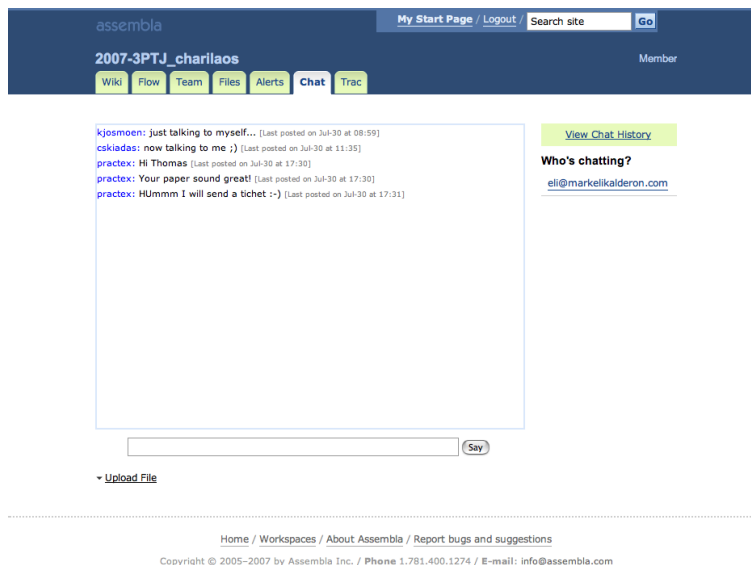


Figure 4: The Chat Page

The trac tool associated with your space is an efficient means for you and your global team to bring your masterpiece to fruition. There's no need for the UN to solve this international coordination problem. PracT_EX and Assembla has solved it for us all.

2 Getting Started with Subversion

At the heart of Assembla's suite of tools is the Subversion repository that tracks the history of the revisions of your paper. Here are some instructions, organized by platform, about how to get started using Subversion in writing your PracT_EX article.

As I mentioned to you before, the PracT_EX production team will create your project's Subversion repository for you on the Assembla server. From this point what you need is to do is described in one of the three following subsections

depending on your operating system (Windows, Mac OS X, or *nix). Pick the appropriate subsection and perform the steps presented below to be synchronized with your team.

2.1 Windows

A Windows binary for Subversion is available online³. Get the best (that is, the latest) available version of Subversion. Please download and install this file. You should also download and install the latest version of TortoiseSVN⁴. TortoiseSVN allows you to access Subversion commands and view the status of your files all within Windows explorer. For more information see the TortoiseSVN website⁵.

1. You need to create a folder for your working copy. Call it **PTJ-workplace**. This is where your project files must be.
2. Right click on **PTJ-workplace** folder and choose **Checkout** from menu. This action only needs to be done *once*.
3. In the dialog window paste the url of your project's repository. The url will be of the form:

`http://tools.assembla.com/svn/2007-3PTJ_surname/surname-theme`
and will be provided by the production editor. A link to it can also be found on your project's space (see [Figure 2](#)).
4. Click on the **Checkout** button.
5. A dialog box will ask you if the login and password can be saved for further access. Click on the **OK** button.
6. Enter your username and password previously created when you accepted the Assembla invitation.
7. Again, click on the **OK** button. Notice that the files stored in the Subversion repository are now being downloaded to your local **PTJ-workplace**.
8. Open your **PTJ-workplace** folder.

3. http://subversion.tigris.org/project_packages.html

4. <http://sourceforge.net/projects/tortoisesvn>

5. <http://tortoisesvn.tigris.org>

9. Open your \LaTeX source in your \LaTeX -aware editor, edit it, and save the changes. If you do not already have a \LaTeX -aware editor, you might try LEd⁶ or PCTEX⁷.
10. Notice, after you have saved your changes, the file icon will be changed to (!). This means that your source has changes not yet committed to the Subversion repository.
11. Right click over the file with (!) and choose **Commit**.
12. This will open a dialog window. Briefly describe the changes you have made to your text. This will be useful to you as a record of your revisions as well as to the production editor and your coauthors, if any, so that they know what you have done.
13. Click **OK**.

Congratulations! You have just committed your first revision to your space in Assembla's Subversion repository.

Before your next editing session be sure to update your working copy by right-clicking on **PTJ-workplace** and choosing **Update**. This will propagate any changes committed to the Subversion repository by the production editor or your coauthors to your working copy. After that, follow steps 8–13 to make further changes and commit these.

2.2 Mac OS X

A Mac OS X binary for Subversion is available online⁸. Get the best (that is, the latest) version. Unzip it, double click on the package installer, and follow the instructions to install. You should also download and install the latest version of SCPlugin⁹. The goal of the SCPlugin project is to integrate Subversion into the Mac OS X Finder. For more information about installation see the SCPlugin website¹⁰. If you are comfortable with the command line you can also follow the *nix instructions in [subsection 2.3](#).

6. <http://www.latexeditor.org>

7. <http://www.pctex.com>

8. <http://homepage.mac.com/martinott>

9. <http://scplugin.tigris.org/servlets/ProjectDocumentList>

10. <http://scplugin.tigris.org/installation.html>

1. You need to create a folder for your working copy. Call it **PTJ-workplace**. This is where your project files must be.
2. Right click on **PTJ-workplace** folder and choose **Checkout** from the Subversion submenu. This action only needs to be done *once*.
3. In the dialog window paste the url of your project's repository. The url will be of the form:

`http://tools.assembla.com/svn/2007-3PTJ_surname/surname-theme`
and will be provided by the production editor. A link to it can also be found on your project's space (see [Figure 2](#)).
4. Click on the **Checkout** button.
5. A dialog box will ask you if the login and password can be saved for further access. Click on the **Always allow** button.
6. Enter your username and password previously created when you accepted the Assembla invitation.
7. Again, click on the **OK** button. Notice that the files stored in the Subversion repository are now being downloaded to your local **PTJ-workplace**.
8. Open your **PTJ-workplace** folder.
9. Open your L^AT_EX source in your L^AT_EX-aware editor, edit it, and save the changes. If you don't already have a L^AT_EX-aware editor, you might try TeXShop¹¹ or TextMate¹².
10. Notice, after you have saved your changes, the file icon will be changed to (!). This means that your source has changes not yet committed to the Subversion repository.
11. Right click over the file with (!) and choose **Subversion Commit**.
12. This will open a dialog window. Briefly describe the changes you have made to your text. This will be useful to you as a record of your revisions as well as to the production editor and your coauthors, if any, so that they know what you have done.
13. Click on the **OK** button.

11. <http://www.uoregon.edu/~koch/texshop>

12. <http://macromates.com>

Congratulations! You have just committed your first revision to your space in Assembla's Subversion repository.

Before your next editing session be sure to update your working copy by right-clicking on **PTJ-workplace** and choosing **Update**. This will propagate any changes committed to the Subversion repository by the production editor or your co-author to your working copy. After that, follow steps 8–13 to make further changes and commit these.

2.3 Unices

To build Subversion from source, download the most recent distribution tarball¹³. Unpack it, and use the standard *nix procedure to compile:

```
./configure
make
sudo make install
```

Be sure to read the document “Install” first, though, to check for dependencies.

1. Create the directory PTJ-workplace:

```
mkdir PTJ-workplace
```

2. Navigate to PTJ-workplace:

```
cd PTJ-workplace
```

3. Check out your working copy:

```
svn co --username username --password password http://
tools.assembla.com/svn/2007-3PTJ_surname/surname-theme
```

username and password were previously created when you accepted the Assembla invitation. The url will be of the form:

```
http://tools.assembla.com/svn/2007-3PTJ_surname/surname-theme
```

and will be provided by the production editor and a link to it can be found on your project's space (see [Figure 2](#)). The files in your Subversion repository are now being downloaded to your local PTJ-workplace. This only needs to be done *once*.

13. <http://subversion.tigris.org/servlets/ProjectDocumentList>

4. Open your \LaTeX source in your \LaTeX -aware editor, edit it, and save the changes. If you don't already have a \LaTeX -aware editor, you might try Kile¹⁴; emacs and vim also both have excellent \LaTeX environments.

14. <http://kile.sourceforge.net>

5. Commit your changes:

```
svn commit -m "your commit message" foo.tex
```

"your commit mesasge" should briefly describe the changes you have made to the text. This will be useful to you as a record of your revisions as well as to the production editor and your coauthors, if any, so that they know what you have done.

Congratulations! You have just committed your first revisions to Assembla's Subversion repository.

Before your next editing session be sure to update your working copy by navigating to PTJ-workplace and entering `svn up`. This will propogate any changes committed to the Subversion repository by the production editor or your co-author to your working copy. After that, follow steps 4–5 to make further changes and commit these.