CS & ECE Systems Course Resources Handbook

Chloe Lopez
Tufts University

January 13, 2009

Revision: 1.0

Formatted for LaTeX
by MultiMarkdown

Contents

Chapter 1

Introduction

This handbook summarizes the array of technical resources made available to instructors in the Departments of Computer Science and Electrical and Computer Engineering by CS & ECE Systems.

Chapter 2

Course Directories

2.1 The /g directory

`/g` is the directory where all course files and directories should be stored. It is also accessible as the `Q:` drive from Windows workstations. Within the `/g` directory, course material is organized in a hierarchy first by course number, then by semester. For example

- Spring 2009 offering of COMP 15: `/g/15/2009s`
- Fall 2008 offering of EE 126: `/g/ee126/2008f`
- Summer 2007 (12-week) offering of ES 3: `/g/es3/2007uc`

The /g naming convention

As you can see, COMP courses are indicated by the bare course number, while all other courses start with the department code. Semester directory names have varied widely in the past (and sometimes weren't even used) but now are standardized as the year followed by a one- or two-letter season code. For example, in 2009, the following codes will be used:

- Fall 2009: `2009f`
- Spring 2009: `2009s`
- Summer (1st session) 2009: `2009ua`
- Summer (2nd session) 2009: `2009ub`
- Summer (12-week session) 2009: `2009uc`

2.2 /comp and /ee: user-friendly pathnames

In practice, however, Unix users don't have to worry about semester codes. This is because the most recently offered semester of every course will always also be available in the `/comp`, `/ee`, `/en`, or `/es` directories. For example, materials for COMP 11 are available at `/comp/11` and materials for EE 102 are available at `/ee/102`. (Note that all department codes appear here, unlike `/g`, where `comp` is omitted.) If your students are using Unix, you should direct your students to course materials using the `/comp` or `/ee` paths, and not `/g`.

`/comp` and `/ee` on Windows?

If your students are using Windows, they will have to find the course materials by course number and semester in the `Q:` drive. CEAS is currently investigating the best way to emulate the `/comp` and `/ee` functionality under Windows.

2.3 Unix groups

Access to course materials is controlled by membership in Unix groups, whether the materials are accessed from Unix or from Windows. For that reason it's a good idea for everyone working with files in `/g` or the `Q:` drive to understand how Unix groups are assigned.

What is a Unix group?

In Unix, every file has an *owner* and belongs to a *group*. If a file belongs to a group, then everyone who is a member of that group can access it according to how the *group permissions* are set. Every user also has a default group to which files they create belong. For students, this is group `student`, and for faculty, it is group `faculty`. But material for courses often needs to be made readable or writable to certain groups of people only. This is where Unix groups come into play.

Types of Unix groups

TA group

TA groups are by far the most often used Unix group. Despite the name, faculty members are also included in this group. (It's hard to imagine a group for TAs that needs to exclude the faculty member.) In fact, a TA group will be used even when there is one faculty member and no TAs! It's best to think of this as the "default" course group.

Top-level course directories in `/g` will be created with the TA group by default.

Grading group

Grading groups are only necessary for courses that are using `provide` for assignment submissions. The membership will be exactly the same as that of the TA group. `provide` manages its own filesystem permissions, so it is important and useful to have a separate group. The `provide` user will also be a member of this group.

Infrequently used groups

The following sorts of groups have been used in the past, but they have been used infrequently or inconsistently.

Instructor group

In the past, Instructor groups, beginning with `instr` or `in` have been used in a number of ways, approximately as a synonym for TA groups. To eliminate this confusion, from now on instructor groups will only be created only when a course *both* has graders who are not full TAs *and* there are files that need to be accessible to faculty members and TAs but *not* to graders. These would presumably be files related to teaching the course and instructing the students, hence the "instructor group" name.

Faculty group

Sometimes faculty need to create files that are not available either to TAs or to students. Assuming that the course has more than one associated faculty member, it's probably best to simply create those files in group `faculty`.

If for whatever reason, access to these files needs to be even narrower, including only faculty for the course, a faculty group with a name beginning with `fac` can be created. Please contact `staff@eecs` if a faculty-only group is needed.

Student group(s)

It's hard to imagine cases where students need to create files that are accessible only to other students, but not to faculty members or TAs. For that reason, what is meant here by a "student group" is a group accessible only to students, faculty members, and TAs, but *not to anyone outside the course.* One example where this might be needed is if there are intellectual property issues with making files world-readable. (Another, better, way to control access in this manner is to use Moodle.)

Once in a while, students in a course are working together in teams, and so they need to share files with each other without allowing access to students in other teams. If you need this functionality, please contact `staff@eecs.tufts.edu`. (Again, this kind of functionality is probably simpler in Moodle, unless source code is being shared.)

2.4 File permissions

When course directories are first created, they are owned by the faculty member and in the TA group. The permissions are set to be owner- and group-writable, but only world-readable. (If a `grading` directory is requested, it will have its own permissions, managed by `prozac`.)

The `setgid` bit

Course directories are also created with the `setgid` bit set. This is indicated by the letter `s` in place of `x` in the permission list when you run `ls -l`. When this bit is set for a directory, any new files created in that directory will be given the same group as the directory. In addition, any directories created in that directory will be given the same group *and* will have the `setgid` bit set.

This is to help ensure that all members of the TA group will have access to each other's work. However, it is still the responsibility of instructors to make sure that the files are *group-readable.*

Inaccessible files

It is very common for TAs to create files in the course directory that no one can read or alter. This can happen one of two ways:

- The files are not group-readable or group-writable
- The files are not in the right group

When this happens, you will need to contact the owner of the files and ask them to change the permissions on the file. To change file permissions:

- `chmod g+rwX <file>` makes the file accessible to the group
- `chgrp <group> <file>` changes the group for `<file>` to `<group>`. (Use `chgrp -R <group> <directory>` to change the group for all the files in a directory.)

If for some reason, the owner of the files cannot be contacted, you may contact `staff@eecs.tufts.edu`, who can change these permissions. However, in general we don't like to do this too much because the owner may have their reasons for not making the files accessible. It is always better to try contacting the owner first.

Why can't anyone access my files?

If you find yourself creating lots of files that others can't access, there are two things you can check. First, the `setgid` bit of the directory you're in may have gotten turned off. Do `ls -ld .` to check, and then do `chmod g+s .` to turn it on.

Second, you may have your `umask` set wrong. Your `umask` determines the permissions used when creating files. When working in course directories, you will want your `umask` to be `002` (or `022` if you don't want people in the group to make changes) when creating files. You can check your `umask` by just typing `umask`. A `umask` of `077` is a very restrictive `umask`, meaning that only you can access the files you create. You can set it to something more useful by typing `umask 002`.

You may also be creating files via Windows. (See below.)

Why can't I access the files of a group I am in?

To see what groups you are in, log into a Unix host and type `groups`. If the group is not listed, please email `staff@eecs.tufts.edu`.

If you are in the right group to access a file, and the file is group-readable and/or group-writable, then it may be that you are in too many groups. Unfortunately there is a very old limit on the number of groups a user may be in. If a user is in more than sixteen groups, additional groups may be ignored at random when accessing files from the file server. For this reason, it is important to request that you be removed from any groups you are not using.

2.5 Windows

The `Q:` drive

The `/g` directory is accessible on Windows as the `Q:` drive.

Windows permissions

Windows permissions for the `Q:` drive are ultimately based on the corresponding Unix file permissions. So even if your course is not using Unix to distribute files, you will still need to understand how Unix file permissions work, as described above. In particular, when you are setting up the files for the first time, it will probably be necessary for a TA or faculty member to log in to Unix and check the permissions from there, to make sure they are what is expected. Windows can sometimes generate unexpected permissions when creating files. Any kind of fine-grained tuning of access will also need to be done from the Unix command line.

Chapter 3

Mailing Lists and Aliases

In all cases, mailing lists will use a student's official Tufts `First.Last@tufts.edu` address, unless you or the student has requested otherwise.

3.1 Choose between a Mailing List and an Alias

Mailing lists are managed via the Mailman mailing list software, whose web interface can be found at `https://www.eecs.tufts.edu/mailman/`. This allows instructors to manage the list subscriptions and moderate discussions themselves in a user-friendly way.

Some instructors, however, wish to avoid the administrative tasks associated with managing a mailing list in Mailman, and would prefer to use a "mail alias." A mail alias is simply an email address that automatically forwards messages unchanged to one or more addresses. Mail aliases do have certain disadvantages, however.

- Mail aliases do not have the additional antispam protections that Mailman has.
- Every request for membership changes must be manually handled by CEAS
- Mailing list messages are clearly marked in subject line as coming from a particular list. Messages sent to mail aliases can be difficult to distinguish from other messages. (If a mail alias is `Bcc:`'d, there's no way to tell at all.)
- There are no message archives for a mail alias. (The keeping of message archives may be important for ABET accreditation.)

In general, CEAS recommends using Mailman for all class lists, and requires it for any list with a more than a few recipients. Occasionally, a mail alias may be appropriate as a way for a faculty member to contact a short list of TAs. For students to contact TAs, however, it is better to use a list, due to the archiving functionality.

3.2 Mailing List Types

There are two basic types of mailing lists, class-wide lists and TA lists. Class-wide lists include all faculty, TAs, and students, while TA lists only include faculty and TAs. TA lists may be used as a means for faculty to contact TAs, or they may instead be a way for students to contact TAs.

3.3 Archiving prior semesters

Be aware that Mailman mailing lists addresses are recycled every semester. This means that if you request a mailing list, you will need to consider whether you want current students to have access to discussions from prior semesters for reference, or whether you want to start with a blank slate to avoid the possibility of revealing solution sets and the like. CEAS can provide either scenario by request. By default, right now, archives from prior semesters are preserved.

Chapter 4

Moodle

Moodle is an open-source course management system comparable to Blackboard.

This document is not intended to explain how to *use* Moodle, but instead to explain what Moodle is, what it can do, and how to get it set up to aid your course. There is a course within Moodle that explains how to use it, and additional documentation is available.

4.1 Accounts

Moodle accounts are based on CEAS Unix accounts, so anyone with a Unix account can log in at `https://www.eecs.tufts.edu/moodle` with their Unix username (usually the UTLN) and password. An account will be created automatically if it didn't already exist.

If a student cannot remember their password, they can reset it at `https://www.eecs.tufts.edu/~accounts/` with their UTLN (Trumpeter) username and password.

If a student does not have a Trumpeter username and password, they should come by the CEAS office in Halligan 231 with their student ID so that we can set it for them.

4.2 Roles

There are four main kinds of "roles" a user may have in a course. First, there is the "student" role. Students may access course materials and participate in activities, but they can't change anything. They also appear in the grade book. "Teaching Assistants" can grade assignments, but they can't change anything else. "Editing Teaching Assistants" and "Faculty" have the ability to change all assignments, activities, and resources. "Faculty" also have the ability to add or remove anyone from any of these roles.

4.3 Enrollments

Moodle student enrollments are handled automatically based on information received from SIS. In the general case, students never have to be added by hand. If a student who you know to be registered in the course is not appearing in your student list, make sure they have tried logging in to Moodle first. Moodle accounts are created automatically when users log in for the first time, and students cannot be added to any course until their account is created.

Faculty and TAs should also automatically appear in the Moodle list under the appropriate categories, but if they do not, please contact `staff@eecs`.

Faculty may add any Moodle user as a student, TA, or editing TA, but keep in mind that this overrides any information in SIS. In particular, it may allows students who have dropped the course to continue accessing course materials. In general it is better, and less work for everyone, to have students make sure they have registered for the course. They will then have access to the course by the morning of the next business day.

4.4 Organize by topic or week

The first choice a faculty member or an editing TA will have to make when setting up a Moodle course is whether the course should be organized by topic or by week. By default, courses are created organized by weeks, with enough weeks to continue through the end of finals. (Weeks in Moodle begin on a Sunday.) This is especially useful if using the course calendar aspects of Moodle. You can increase or decrease the number of weeks as you see fit.

Alternatively, Moodle courses can be organized by topic. Topics are just whatever abstract instructional unit that organize the syllabus. This may be more useful if you need flexibility in the amount of time spent on each topic or unit. As with weeks, topics can be increased or decreased in number.

4.5 Resources

In Moodle, course materials are considered either a "resource" or an "activity." Resources include things like handouts, lecture notes, web pages, and multimedia files, but also interactive items like web forums, wikis, workshops, and surveys.

4.6 Activities

Activities are anything that counts for a grade. Assignments, of course, are one time of activity, but activities also include projects, quizzes, and exams.

Most activities require the student to perform some action, like answer quiz questions or type in assignment responses, while logged into the Moodle session. Alternatively, an assignment can require a student to prepare a response as a file which is then uploaded to Moodle, where graders can retrieve it. It is possible to have students upload more than one file, but if you find yourself needing students to upload many files per assignment, or a zip or tar archive of a directory full of files, you may want to consider making the assignment an "off-line" activity and using `provide` instead.

A special kind of activity is the "off-line" activity. An offline assignment contains the assignment name and a short description, and it creates an entry for the assignment in the Moodle gradebook, but there's nothing for the student to do in Moodle, besides read the assignment information. This is to allow assignments like oral exams, presentations, and handwritten work to be entered into the Moodle gradebook and class schedule without causing confusion.

4.7 Gradebook

As of Moodle 1.9, the gradebook has become quite sophisticated in its means of weighting assignments and calculating final grades. It is possible to set up just about any weighting system for course grades. See Appendix 2 for more details.

4.8 Communicating with students

Students are notified by email about anything that happens to them in the course. They will know automatically when you leave comments on their work, or if you reply to their forum postings. You can also send messages to the students directly, either one at a time, or as a group.

4.9 Reusing course materials

It is in fact possible to use a previous offering of a course as a template for a current offering of a course. This may be easier than recreating a Moodle course from scratch. All resources, activities, and assignments will be copied to the new course, but without including the student data. Do keep in mind that any references to dates (and possibly other things) will have to be changed by hand, so if you intend to use this functionality, it is probably a good idea to design the resources to minimize such references.

Chapter 5

Course Evaluations

In order to comply with ABET accreditation requirements, both departments must collect student feedback for all courses offered at midterm.

5.1 Evaluations for CS

The CS department course evaluations are located at `https://www.eecs.tufts.edu/cscourse/feedback` and collect feedback for all instructors using a standardized set of questions. All responses are completely anonymous. No action needs to be taken by instructors to set up these surveys, but the department recommends that instructors make time in the class schedule for students to use the labs to login and submit their feedback.

Submitting feedback requires that students log in using their CEAS Unix username and password, so please be prepared to help students with that even if your course does not use the CEAS Unix account for anything else.

Mid-term course evaluations open on or around the sixth week of the semester.

5.2 Evaluations for ECE

The ECE course evaluations are located at `http://www.ece.tufts.edu/abet/`. Please contact the ECE department office for more details.

Chapter 6

provide

`provide` is a system of utilities, written by Prof. Alva Couch, for submitting programming assignments on the Unix systems. But it can also be used for submitting any assignment file. `provide` can also function as a drop box, in a pinch.

6.1 Requesting `provide`

To request that `provide` be set up for a class, just email `staff@eecs.tufts.edu` with the following information:

- The name and number of the course
- The list of TAs and graders who will need to be added to the grading group
- Please also note if any of the assignment submissions are expected to be unusually large (i.e, more than about 1MB).

6.2 `prozac`

`provide` uses a sophisticated system of Unix permissions to protect privacy and prevent cheating while also ensuring security. It will check all these permissions, and if any of them are wrong, it will simply refuse to work.

Luckily, `provide` includes a script called `prozac` that will fix these permissions for you. `prozac` is best run by the owner of the `grading` directory, though it may work in certain cases when run by anyone in the grading Unix group. 99% of problems with `provide` can be solved by running `prozac <coursename>`. Anyone in the grading group can run `prozac`.

If `prozac` fails for whatever reason, please contact `staff@eecs.tufts.edu`.

6.3 Configuring `provide`

In most cases, `provide` will do what you need it to do with the default configuration. However, there is one step that a faculty member or TA will need to do before students use it: they will need to specify the assignments in the course.

This is done by editing the `assignments.conf` file in the `grading` directory. Look inside this file for basic advice on how to add and name assignments. Additional configuration,

like due dates, whether to accept resubmissions, or late submissions, is also possible. Type `man provide` for details.

6.4 How students use it

It's very easy for students to use `provide`. They simply type `provide <course> <assignment> <file>`, where `<course>` is the name of the course (as listed in `provide.conf`), and `<assignment>` is the assignment name (as listed in `assignments.conf`). You will have to provide both these names to students before they can submit any assignments. `<file>` is the name of the file or directory they are submitting. More than one file can also be submitted.

6.5 How graders use it

The simplest way to access submitted assignments is simply to go to the grading directory, where submissions will be organized by assignment name and then by the students' usernames. Submissions can then be inspected manually, but there are also tools to make the grading process easier.

`profess`

`profess` is the tool for adding grades and annotating assignments. Log into the Unix systems, change to the assignment submission directory, and run `profess` there. To grade another assignment, change to another assignment directory and rerun `profess`. You can also use `procure` to see what assignments still need to be graded. See the `man` pages for more details.

Automated Grading

In the `provide` system, automated grading can be accomplished by simply writing scripts that do whatever compilation and testing is needed and the call `profess`. No special format or script language needs to be used. This allows for a great deal of flexibility, but some care must be taken. See `man provide` for details and examples.

6.6 Advanced topics

Included in the `provide` package are a number of associated scripts:

- `proboscus`: Converts assignment grades to `.csv` format
- `procure`: Lists assignments that still need to be graded
- `prodigal`: Reports grades for graded assignments
- `profess`: Grades assignments (see above)
- `progress`: Lets students check on grades for their own assignments
- `projoin`: Creates a summary table of grades for a course
- `propensity`: Automaticaly handles lateness penalties

- `prorate: Automates the process of`cd‘ing to assignment directories
- `prozac`: Checks and repairs `provide` grading directory permissions

See the `man` pages for more details. For those scripts that lack `man` pages, you may need to inspect the scripts in `/usr/sup/provide/bin` to see how they work. Email `staff@eecs.tufts.edu` if you have any questions.

Chapter 7

Web Pages

There are two ways to create a web page for a course. The first is to create the web pages from scratch in the `public_html` directory within the course directory on `/g` or the `Q:` drive. In addition to plain HTML and CSS, PHP scripts and Ruby- or Perl-based CGI scripts may also be used. In fact, it is possible to install full content-management, wiki, or web forum applications here, if desired. Most instructors, however, just use this space for course announcements, syllabuses, and course notes.

If more advanced web features are required, or if you need to restrict access to some materials only to those registered for the course, please consider using Moodle before installing other software or writing your own. Most web functionality is provided by Moodle, it ties into the official course enrollment system, and CEAS is able to provide support for it.

7.1 Archives

The web pages for the most recent offering of the course remain available as a reference until the next time the course is offered. This is useful for future instructors and for prospective students. For this reason, when a course is completed, instructors should make sure that any content that they would not like to have available is either moved out of the `public_html` directory or has its permissions set to make it inaccessible from the web. It's best if a syllabus or course description remains available at the very least.

Permissions for prior semesters

All course materials for prior semesters remain in the file system, even after they become inaccessible from the web. There is currently no standard for making these older semesters available over the web, but if there is a need to make them available temporarily, please email `staff@eecs.tufts.edu`

Chapter 8

Computer Labs

There are four main general-use computing labs for students. Two labs of Microsoft Windows workstations are located in Halligan 120 and 122, and two labs of Red Hat Enterprise Linux workstations are located in Halligan 116 and 118. Students rely on these labs to complete coursework, so it is important to ensure that examples, assignments, and course software all work on one of these platforms. Requests for software to be installed in the labs should be made before the beginning of the semester, but needs that come up during the semester can be handled on a best-effort basis.

8.1 EE Labs

In addition, there are three labs designated for EE projects: the circuits lab in Halligan 223 which uses Windows workstations, and the electronics labs in Halligan 225 and 229 which are currently dual-boot Windows and Linux workstations.

8.2 Scheduling

Time in the labs for course instruction is generally in high demand, and yet there needs also to be time available for students to complete their assignments. For this reason, it is very important to schedule instructional time in the labs well in advance. Please contact `staff@eecs.tufts.edu` for more details.

Chapter 9

Instructional Software

9.1 Unix Software

Packaged Free and Open Source Software

Given enough notice, most open source software can be obtained and packaged for general use with the standard Solaris and Red Hat Linux packaging systems. Software packaged in this way will be available under `/usr` in Linux, and `/usr/local` in Solaris. Unless instructors specify otherwise, CEAS will make every effort to support these requests on both Linux and Solaris.

However, there may be times when the desired software is experimental, difficult to package, or conflicts with operating system software, or when instructors expect to make ongoing changes to the software installation. In these cases, instructors should feel free to install software in the course directory in `/g`, with the understanding that it will only be available for the semester, and will not be available for general use. In most cases, when instructors choose to do this, they intend to support the software on only one of the two Unix environments, so be sure to make it clear to students which environment they should use.

If students' Unix environments need to be modified to use such software, instructors should request the creation of `use` scripts (see below).

Commercial software

Commercial software packages often require installation in a directory of their own. CEAS locates such software packages in the `/usr/cots` directory. User environment customization is usually needed to use software in `/usr/cots`, and this is handled by `use` (see below). Examples of software handled in this way include Cadence, TeX Live (non-commercial, but requires its own directory), Matlab, Coverity, Agilent, and Leda.

Unix environments (`use`)

A common problem in managing Unix software is to provide an easy way to modify Unix environment variables like `PATH` and `LD_LIBRARY_PATH` while avoiding conflicts with other software, and without requiring every user to modify their shell initialization files. At CEAS, this problem is solved with the `use` utility. `use` is a shell alias that is able to load any of a number of centrally managed shell initialization files.

For example, let's say that a software product called `foo` requires a number of initalization commands to be run before the software will function. CEAS can place these commands in a file called `foo.csh` (or `foo.sh`) in `/usr/sup/use`. Then users will only have to type `use foo` before they begin using `foo`. If the class COMP 100 has special initialization that needs to be done before using libraries installed in `/comp/100`, then a `use` script could be created that would let students type `use comp100` before they do their compiling.

In general, if the need to modify student shell initialization files comes up, odds are good that the need can be better met by using `use`.

9.2 Windows Software

System Imaging

Software packages on Windows classroom lab systems are reinstalled just before every semester, so if there is some software that will be needed on those systems, it is important to make those requests before the semester begins. It is possible to distribute software to systems once the semester has begun, but it can be disruptive to ongoing use of the labs.

Occasionally, a software package cannot be distributed via our packaging system, which means that it will have to be installed by hand on all systems. This is difficult to do during the semester, so this is another reason to have requests for Windows software made in advance.

Device Drivers

Device drivers, for example, for an EE project peripheral or some custom piece of hardware, also have to be installed manually at every machine with the hardware in question at hand. This is because even if the driver files are distributed, Windows won't install the files until the device is attached, and the installation must be authorized by an administrator. For this reason, if device drivers need to be installed, please provide CEAS with the driver software and an example device before the semester begins.

Chapter 10

How do I . . . ?

10.1 . . . create a drop box?

`provide` is really the best solution for creating a "drop box" where students can leave files for instructors, but students cannot access what they or anyone else has left there. Unfortunately, there is no equivalent to `provide` for Windows, and simple Windows permissions are not going to be sufficient to set up a drop box manually.

Since Moodle does the same thing as `provide`, that is, allow for submissions of assignments, one could also create a web-based drop box as an assignment in Moodle. This is a good solution for those who are already using Moodle and who aren't using the Unix command-line.

10.2 . . . set up web forums and wikis?

Moodle provides the functionality of both web forums and wikis in the form of Moodle activity modules. This is the best solution for those who want a quick set-up of such a system without doing a lot of technical administration.

If your needs are more sophisticated than what Moodle offers, or if you have a particular web forum or wiki software in mind, it is always possible to set it up in a course's `public_html` directory. If the software requires a database, one may be requested by using the form in the Userguide. Keep in mind that web forums and wikis can pose a security risk, so it is important to stay up to date on all security alerts for that software.

10.3 . . . set up an online grade book?

Moodle has the capability for an online grade book. Please note that to do this will mean that every assignment will need to be added to the Moodle course, regardless of whether the assignment is actually submitted *via* Moodle. Such assignments are called "off-line activity" in Moodle. They will appear like any other assignment, but there will be no way for students to submit their work. Instructors will just enter the grade when it is ready.

In fact, it is completely possible to use Moodle *only* as an online gradebook, by making all the assignments "off-line." Students would only log into Moodle to check their grades (and possibly contact instructors).

10.4 . . . allow students to submit programming assignments?

The best solution for submitting a directory tree of programming assignments is `provide`. It is possible, however, to have students upload `tar` archives or `zip` files of their source tree into Moodle. If there are many students in the class, however, downloading, extracting, and running the code may prove time-consuming.

If the students are submitting single files of pseudocode or the like, `provide` or Moodle will likely work equally well.

10.5 . . . perform automatic grading?

Automatic grading will always require instructors to write custom grading scripts. If the assignments were submitted with `provide`, then the custom grading scripts can call the `profess` tool from the `provide` package to record grades in the `provide` system.

Moodle does not provide an automatic grading facility, but the Moodle gradebook does allow for the quick entry and recording of grades.

10.6 . . . distribute files to students?

There are three ways to distribute files to students.

The first and simplest way is to place the files in the `public_html` directory of the course directory in `/g` or `Q:`. The students can then download the files via the web. This method does not provide any way of controlling who has access to the files, however. This method is often used for course notes, syllabuses, and free software.

The second method is to place the files in the course directory, but outside the `public_html` directory. This way the files are available only to the CEAS network and systems.

If you need to control access to the files, making them available only to those enrolled in the course, consider using Moodle. Files of any type or format can be uploaded as "resources" to Moodle, and in many cases Moodle can embed them directly in Moodle web pages. Students will have to be enrolled in the Moodle course to get access to these web pages, and you also have the ability to grant access to these files only on certain dates. For example, you may want solution sets to be available only after the assignment is due.

10.7 . . . make multimedia content available?

Multimedia files can be placed directly in the `public_html` directory of the course directory, but if you need access control, consider using Moodle. Moodle also allows you to embed these files within a web page, allowing for better integration with other course materials.

10.8 . . . contact students in the course?

The easiest way for faculty to send a quick email to all students in a course is to log in to SIS and use the class email functionality there. This is a web form, and isn't an address you can send to from your usual email client.

For ongoing emails to students, one should have a mailing list set up. Contact `staff@eecs` for details.

Moodle also provides a way to send messages within the Moodle system that students will see the next time they use it. Moodle will also send users email when they have messages waiting in their Moodle inbox.

10.9 . . . support student groups?

Moodle provides full support for dividing students into groups or teams to work on projects or assignments. These groups will be reflected when Moodle computes grades. It is also possible to control access to various materials on the basis of these groups.

If you need to control access to `/g` and `Q:` files on the basis of student groups, please contact `staff@eecs.tufts.edu` with the names of each of the members of the groups, and whether TAs or faculty members should also be a member.

Chapter 11

Appendix 1: Naming Conventions

11.1 Course Directory Naming Conventions

The main directory in `/g` for a course is named for the department code, if it is something other than `comp`, followed by the course number. So, the example course of COMP 100 will be located at `/g/100`, and EE 100 would be located at `/g/ee100`.

Within these directories, the directory for the semester will be named for the calendar year followed by `f` for fall semesters, `s` for spring, `ua` for summer, first session, `ub` for summer, second session, and `uc` for summer, 12-week session.

The most recent offering of a course is also mounted under the `/comp` or `/ee` directories, e.g., `/comp/100` or `/ee/100`.

11.2 Web Naming Conventions

Web pages can be expected to appear at either `www.ece.tufts.edu` or `www.cs.tufts.edu`, followed by the mount point for the course in lower case.

Examples:

- `/comp/11/public_html` appears on the web as `http://www.cs.tufts.edu/comp/11`.
- `/ee/126/public_html` appears on the web as `http://www.ece.tufts.edu/ee/126`.

As of today, there is no standard for accessing the web pages for semesters other than the most recent one. If you need prior web pages to be accessible on the web, please contact `staff@eecs.tufts.edu`. You will also need to clear it with the faculty member who taught the course that semester.

11.3 Groups Naming Conventions

Group names are made up of three parts:

- The group type prefix (`ta`, `grade`, `instr`, etc.)
- The department code, but only for departments other than COMP (e.g., `ee`)
- The course number and section code (if necessary)

So, taking the example courses COMP 100 and EE 100

- TA group: `ta100` and `taee100`
- Grading group: `grade100` and `gradeee100`
- Instructor group: `instr100` and `instree100`

11.4 Mailing Lists Naming Conventions

Student mailing lists just use the department code and course number as the name of the list, but TA mailing lists use the same name as TA groups (see above).

So, taking the example courses COMP 100 and EE 100:

- Student mailing list: `comp100` or `ee100`
- TA mailing list: `ta100` or `taee100`

11.5 Crosslistings

When courses are crosslisted within the department, one listing should be chosen to be the one used for all names. This is usually the highest-numbered course.

When courses are crosslisted with a course offered in another department, the `COMP` or `EE` course will be the one used for all names. (In the past, EN 47 had been an exception to this rule, but from now on this course will instead be taught as COMP 9.)

11.6 Section Numbers

For most courses, only one section of the course is offered, and the course is listed with a section number of "01." In these cases, the section number is completely ignored for all names used for course-related resources.

When more than one section is being offered for a course, one of two things may happen.

If all the sections of the course are taught by the same faculty member, or if the various faculty members are pooling their resources, sharing TAs, and teaching the same curriculum, then it may make sense to treat all the sections of the course as if they were one big section. In these cases, section numbers will be ignored in all resource names, and mailing list memberships and Moodle enrollments will place enrolled students from all sections will be placed in a single mailing list and Moodle course.

At other times, when the various faculty members are teaching from different syllabuses, or if a single faculty member just wants to keep the sections separate, then the course resources will have to be divided by section, and the names of the resources will have to reflect that. In general, the section numbers will be appended to the course resources name with a hyphen, e.g., `-01`. This will be the case for mailing lists, Moodle courses, web pages, and `/comp` and `/ee` directory paths. For directory paths in `/g`, however, the section number will be appended to the name of the semester-specific directory, e.g., `/g/11/2009s-01`. This keeps all course material within the `/g/11` directory. As you might expect, this directory would then appear at the `/comp/11-01` mount point.

Section numbers will not be used for unix groups. This does mean that even if two sections of a single course are both using provide, they will have separate `grading` directories, and

the students will to specify which section they are submitting assignments for, but the resulting files will have the same unix group.

"Special Topics" Courses

Certain course numbers are used to refer to "special topics" courses, courses that have no officially set curriculum and may be only offered once. These courses numbers are 10 (for CS only), 50, 150, 250, 193, 194, 293, and 294. As a result, courses with these numbers must be distinguished from other courses that share those numbers, whether they are offered in the same semester or different semesters. Similarly, if such a course does happen to be offered more than once, there needs to be a way to connect these courses for continuity's sake.

In the past, the registrar would assign these courses three-letter section codes that corresponded to the course's title, and thus to the content of the course. That is no longer the case, at least for COMP and EE. Now, all courses have a two-digit section codes. For special topics courses, this poses a problem. A 150 course that has a section number of 01 this semester will very likely be completely different in content from a 150 course with the same 01 section number the next semester.

The way we get around this problem is to assign three-letter codes ourselves, and use them within the department when referring to course resources and also the course more generally. If you are not sure what the three-letter code is for your course, please ask the appropriate faculty member. It may be that a code has not been assigned yet.

Numeric section numbers should still be used in course listings and in any place where the expected audience is prospective students. Otherwise they will have no idea which section to register for!

Chapter 12

Appendix 2: The Moodle Gradebook

The Moodle gradebook supports most grading systems, with a few subtleties that need to be taken into account.

12.1 Grade Categories

The key concept for the Moodle gradebook is the grade "category". A grade category is not based on assignment or activity types (though you could set it up that way). Rather, a category is a group of grades that are averaged together (or otherwise aggregated), and the result is used to compute a final grade.

For example, if a course has a grading policy that says that quizzes will count for 20% of the final grade, then that course will need a grade category called "Quizzes," and all the quiz-related activities (whatever activity type they may actually be) will need to go in that category. Notice that if you were to calculate these grades by hand, you would average all the quizzes together, and then use the result in a later calculation.

Moodle calls this "average them all together" step "aggregation." Moodle provides many aggregation strategies:

- mean
- weighted mean (where you specify the weights explicitly)
- simple weighted mean (assignments are weighted by the number of points they are worth)
- median
- mode
- sum
- highest grade
- lowest grade

You can also exclude a given number of lowest grades from this aggregation.

It's best to consider what your categories are and to create them before creating assignments, so that you add assignments to categories as you go, rather than having to put them in the right categories at the end.

12.2 Computing the Final Grade

So how do you compute the final grade from these category grades? In Moodle, the entire course grade is just a big grade category itself, so you can just pick one of the aggregation strategies above for the whole course. So, going back to our example, to make the quiz average worth 20% of the final grade, we would choose "weighted mean" for the course aggregation, and set the weight for the Quizzes category to 20 (assuming that the total of the weights for all the categories is 100.)

12.3 Weighting Assignments with Point Values

Some instructors prefer a simple point value system, where, for example, quizzes are worth 10 points, assignments 20 points, and exams 100 points, and the final grade is just the sum of all these points. If you use this system, then you don't actually need to use categories at all, and you would use either the simple weighted mean or the sum aggregations to get the final grade. But categories might still be convenient pedagogically, however, because they allow students and faculty to break out their averages for each of the assignment types.

12.4 Making Your Own Grade Computations

If none of the provide grade aggregations do what you need them to do, you can create a custom calculation for any category, or even the whole course. If you click on "Edit calculation for category total," there will be a place to enter a spreadsheet-style formula (something like `=AVG(A1,A2,A3)+AVG(Q1,Q2)`). This will override the aggregation for the category (or for the course, if desired). So if the system seems too limiting, remember that this option is always there.

12.5 Extra credit

Extra credit is something of a blind spot for Moodle in 1.9. Right now there is no way to give, for example, 105 points out of 100 to an assignment. (The developers claim that this is a peculiarly American practice that they hadn't considered.) This will be fixed in Moodle 2.0.

In the meantime, if you intend to award extra credit, you will need to need to think carefully about how you would like to do it.

If you want to add extra points to a category that has "mean" as its aggregation, then you are in luck. You can change the aggregation to "mean (with extra credit)" and then set the "extra credit" value for each extra credit assignment. The "extra credit" value also appears for the "sum" aggregation. Be aware, however, that doing this will never raise the category result to be greater than 100%. For this to work, all the assignments in the category need to count equally. Split them inte separate categories if they do not.

Then you can create a grade item like "Extra Credit for Assignment 2" and set the grade to 5.

If what you need to do instead is just have an extra assignment that's averaged in with the rest, just create it like any other assignment, and only enter a grade for the students who complete it. Be sure to choose "Aggregate only non-empty grades" for the category, and don't forget to give zeros to any student who fails to complete required assignments in the category.

Finally, you can create your own grade computations, as described above, for any grade category or for the whole course.