VITELS

Scheduling Howto

Thomas Jampen
University of Bern
Neubrückstrasse 10
3012 Bern
+41 31 631 33 12
jampen@iam.unibe.ch

1 Introduction

This Howto is divided into three sections. The first section contains this introduction. The second section explains the front-end to the VITELS scheduling system that is used by the students in order to reserve timeslots and by module administrators in order to configure their module. The third section shows other module developers how to access the important LDAP directory entries with the help of a provided PHP file.

2 Front-end to the VITELS Scheduling System

This Section describes the front-end to all the LDAP queries. The GUI is programmed as a set of websites that can be viewed in every browser. The websites do not contain special code that requires certain plug-ins, they are just shown to the user as normal HTML pages containing basic JavaScript code.

2.1 Login

At the beginning, the user (student or module administrator) is prompted to enter his username and password. He also has to choose from a pull-down list to which educational institute he belongs (Figure 2.1). Module administrators choose "VITELS Staff" in order to get the appropriate privileges!

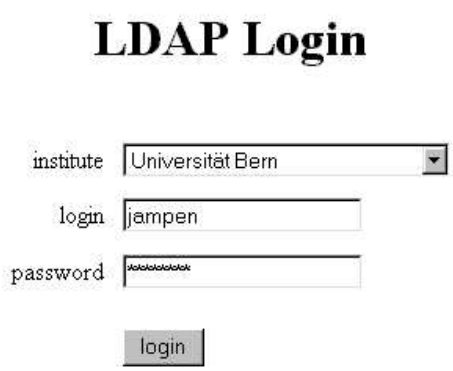

Figure 2.1: Login Page

If the login is successful the user is directly brought to the online timetable. If not, the same login site is displayed again.

2.2 Timeslot Reservation

After a successful login the currrent week's timetable of one module is displayed (Figure 2.2). There are three symbols used in order to visualize the state of each timeslot within the timetable. A red cross indicates that the slot has already been reserved by another student. A green circle shows that this slot is still available and can be reserved simply by clicking on the circle. The blue check

mark is used to mark slots that have been reserved by the user himself. These slots can be freed by clicking on the blue symbol.

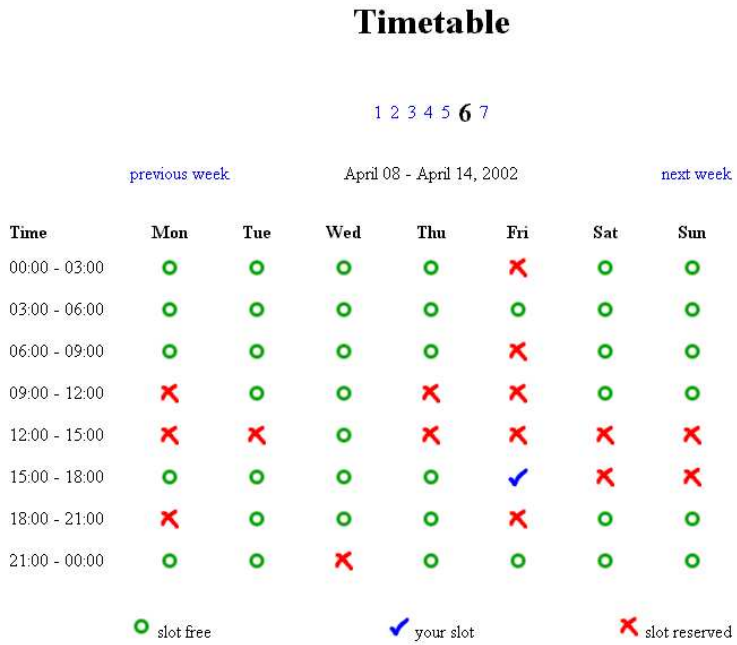

Figure 2.2: Timetable

Above the timetable the displayed week is indicated. On the left and on the right side links have been placed in order to change to the previous or to the next week respectively.

On top, there is a list of all available modules. The module that is currently selected is marked with a bigger font face. If you click on another module id, the timetable of that module is displayed, if you click on the same module again, the timetable is reread from the database. The selected week stays the same as before changing the module.

In the case that the directory does not contain entries for the currently selected week and module, the message "no timeslots available" is displayed instead of the timeslot information symbols.

2.3 LDAP Directory Administration

If a VITELS administrator (staff member or module administrator) is logged in, below the timetable an administration menu is displayed (Figure 2.3).

2.3.1 Display Usernames of Reserved Slots

An administrator is allowed to see who has reserved which slot. This allows him to track students that reserve lots of slots and block other students. A module administrator is even allowed to free slots. He is only allowed to do this for

Figure 2.3: Administration Menu

modules he administrates. In order to free a slot he has to click on *view names* first and then, select the slot he likes to free. When the administrator moves the mouse over the a displayed username the user's real name appears within a little window or in the status line of the browser (the behavior is browser dependent). The administrator can only free slots, when clicking on a username whose slot he intends to free. It is not possible to free slots without seeing the corresponding username (e.g. when seeing only the symbols). This is done not only for error prevention but also because a module administrator should be allowed to reserve slots aswell and thus being able to solve the module or parts of it himself in order to make sure everything works as planned.

2.3.2 Adding and Removing Timeslots

When an administrator clicks on *add/remove slots*, he is presented a page where he can add and remove timeslots (Figure 2.4).

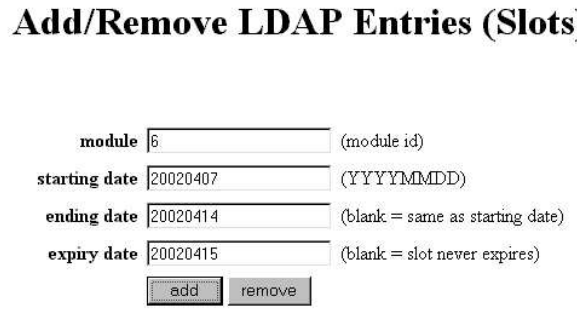

Figure 2.4: Adding and Removing Timeslots

The first field has to be filled with the module identification. Unfortunately, this cannot be filled automatically because the access restrictions for administrators are not saved within the LDAP directory tree but in the LDAP server configuration file. Thus, it is not possible for the scheduling script to determine whether an administrator is allowed to do changes to just a single module, to several of them or to none. This is not a security problem because if an administrator tries to add or remove slots to or from a module that he does not administrate, the LDAP server daemon prohibits the changes. There is a status report field in the administration menu that reports whether and how many entries have been added or removed.

The second and third field must be filled with the date for the first and last

slot to be added or removed respectively. The format for the date is of the form "YYYYMMDD". Make sure to always start with a Monday and to end with a Sunday. This is not mandatory (as it is not verified by the script), but it is strongly recommended because the timetable of a week is only displayed if every slot of this week is present in the directory. This means that if you add slots for Monday till Friday, the week won't be displayed because the slots for Saturday and Sunday are missing!

The last field allows the administrator to specify an expiration date for the slots he intends to add. This field is ignored when removing slots. A daily cron job will check the LDAP tree for expired slots and remove them automatically. Slots without expiration date will have to be removed by the module administrator! A reasonable value could be the Monday of the succeeding week (when adding only one week), or a date in the semester holidays if you are adding the slots for the whole semester at once.

Before any slots are deleted the administrator is displayed a pop-up window, explaining the steps that will be executed. Now, he has the possibility to commit the changes or to abort without changing anything.

2.3.3 Change Module Settings

After clicking on *change module settings* the administrator is directed to a page where the timeslot duration, the number of timeslots per day and the starting time of the first slot can be changed (Figure 2.5).

Caution: this should only be done at the beginning of a semester because all information stored within the module's timetable will be deleted when changing these values (because the slots might not be consistent anymore)!

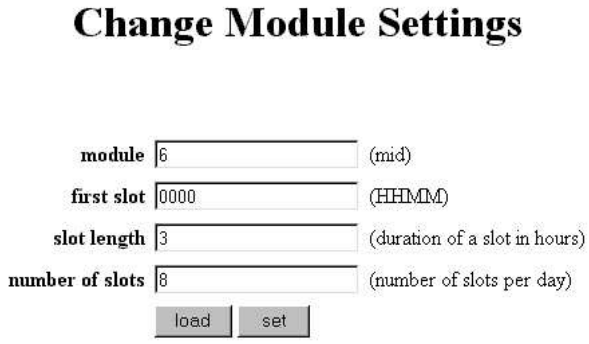

Figure 2.5: Change Module Settings

The first field takes the identity of the module to change. Administrators are stronly recommend to press the *load* button after entering the module identity in order to check the current values before performing unnecessary changes.

The second field takes a four digit number specifying the starting time of a day's first slot. The format is "HHMM". Starting with this time all the other slots will be calculated without any break inbetween.

The duration of a single slot can be entered in the third field. Make sure to specify a number representing the duration time in hours!

The last field allows you to specify how many slots you would like to make available per day. Make sure you enter a reasonable value (based on the above described values), otherwise the execution will fail or lead to unwished results!

Before any changes to the LDAP directory tree are applied the administrator has to validate the changes and press the OK button within a pop-up window.

3 PHP Interface for Accessing a Module's Current User

In order to send cookies with slot information to the browser when students visit a module's website, the module's current user and the its slot settings have to be accessible (at least for module 6). The file `vitelsldap.inc.php` offers an easy to use interface in order to check whether a user is the current user of the specified module, whether the supplied password is correct and what the starting time and the ending time of the current slot is.

The needed functionality is provided by the class `VitelsLDAP`. There is only one function that should be called from outside:

`get_slot_if_current_user($dn, $pwd, $mid)`

This function returns an array containing the starting and ending time of the slot only if the user specified by `$dn` is the current user of the module `$mid` and if his password is correct. Otherwise, `false` is returned.

```
<?php

require('vitelsldap.inc.php');

// when composing the dn make sure you know from which
// institute the user comes!
// choose one of:
// $unibe_base, $unifr_base, $unige_base, $unine_base, $ingfr_base
$dn = "uid=a-username,$unibe_base";

$pwd = "some-password";

// for example module 6
$mid = "6"

// access to the LDAP directory
$ldap = new VitelsLDAP();
$result = $ldap->get_slot_if_current_user($dn, $pwd, $mid);

if ($result) {
    // the date is specified as seconds since 1.1.1970
    $starting_time = $result["starttime"];
    $ending_time   = $result["endtime"];

    // get printable date, e.g. 20020426 1840
    $start = date("Ymd Hi", $starting_time);
```

```
    $end   = date("Ymd Hi", $ending_time);

    //...
}
else {
    print "wrong password or not current user!";
}
```

The above example shows that the `vitelsldap.inc.php` has to be included. Then you have to obtain the the user identification and compose the distinguished name (dn). Please make sure you know from which educational institute the user comes! Then assign the appropriate base dn for this institute. After that, a new instance of the class `VitelsLDAP` has to be created. With the use of this object, it is possible to call the function `get_slot_if_current_user($dn, $pwd, $mid)`.

The array returned can be accessed as follows. `$result["starttime"]` contains the starting time of the slot and `$result["endtime"]` contains the ending time of the slot in seconds past since January, 1^{st} 1970. These values can be easily converted into the preferred format using the PHP function `date($format, $time)`. `$format` can be specified as in the above example or as specified on the PHP website (`http://www.php.net/`).

The second file provided with this little API is `ldaperror.php`. This file contains the HTML code for an error message that is displayed automatically if the LDAP server is not reachable at the moment.