

---

**Jenzabar CX**

# **Jenzabar-CX Interface Setup**



**JENZABAR**

**Installation Guide**

Copyright © 2001 Jenzabar, Inc.

You may print any part or the whole of this documentation to support installations of Jenzabar software. Where the documentation is available in an electronic format such as PDF or online help, you may store copies with your Jenzabar software. You may also modify the documentation to reflect your institution's usage and standards. Permission to print, store, or modify copies in no way affects ownership of the documentation; however, Jenzabar, Inc. assumes no responsibility for any changes you make.

Filename: injenza

Distribution date: 10/29/2001

Contact us at [www.jenzabar.com](http://www.jenzabar.com)

Jenzabar CX, and QuickMate are trademarks of Jenzabar, Inc.

INFORMIX, PERFORM, and ACE are registered trademarks of the IBM Corporation

Impromptu, PowerPlay, Scenario, and Cognos are registered trademarks of the Cognos Corporation

UNIX is a registered trademark in the USA and other countries, licensed exclusively through X/Open Company Limited

Windows is a registered trademark of the Microsoft Corporation

All other brand and product names are trademarks of their respective companies

JENZABAR, INC.  
JENZABAR-CX INTERFACE SETUP INSTALLATION GUIDE

TABLE OF CONTENTS

<b>SECTION 1 – INTRODUCTION TO THE INTERFACE .....</b>	<b>1</b>
Overview.....	1
Purpose of the Interface .....	1
Purpose of the Upload Process.....	1
Purpose of this Guide .....	1
Intended Audience.....	1
Implementation Process.....	2
Introduction.....	2
Setup .....	2
Data Transfer.....	2
Host Links.....	2
Optional Steps .....	2
Adoption.....	3
Interface Diagram.....	4
Introduction.....	4
Upload Process Description.....	4
Upload Process Description.....	5
Introduction.....	5
Tracking Uploaded Files.....	5
Instructions for Uploading Files.....	5
<b>SECTION 2 – IMPLEMENTATION CONSIDERATIONS.....</b>	<b>7</b>
Overview.....	7
Introduction.....	7
System Requirements .....	7
Implementation Steps.....	8
Establishing the Jenzabar-CX Administrator’s Account.....	8
Defining Your Website’s URL.....	8
Defining Your Website’s URL Without Single Login.....	8
Defining Your Website’s URL With Single Login.....	10
Obtaining Your Institution’s ID.....	11
Populating the \$TXTPATH/jenzabar Directory.....	11
Accessing the Jenzabar API Menu .....	12
Accessing the Jenzabar Maintenance Submenu.....	13
Creating the Jenzabar Initialization Record (jenzini_rec).....	14
Setting Up Department Codes .....	18
Setting up Alternate Address Functionality .....	18
Reviewing Error Messages .....	19
Posting Files .....	19
Identifying In-Process Files .....	20
Identifying the Jenzabar-CX Records.....	20
Managing Sequence Numbers.....	21
Uploading Information .....	21
Running the Jenzabar Batch Upload Data Process.....	22
Using Registration Code Coupon Functionality .....	23
Enabling/Disabling the "Process All New Coupons" Option .....	23
Running the Create/Upload Data File Process (jenzupload) .....	24
Resetting the Session and Year Parameters .....	25
Engaging and Disengaging the API Process .....	26

Using <i>cron</i> to Execute Jenzabar Processes .....	27
Introduction .....	27
Determining Environment Variables for <i>cron</i> .....	27
Setting Up the Jenzabar-CX API Update and Registration Code Coupon <i>cron</i> Processes .....	28
Setting Up <i>cron</i> Manually .....	29
Executing the <i>cron</i> Processes for Jenzabar-CX API Data Processing .....	30
Setting Up Authentication Files .....	32
Introduction .....	32
Setting Up the DBM File for Web Authentication .....	32
Using Troubleshooting Tools.....	33
Processing Help Messages.....	33
Running in Debug Mode.....	33
<b>SECTION 3 – POST-IMPLEMENTATION CONSIDERATIONS.....</b>	<b>35</b>
Overview.....	35
Introduction.....	35
In This Section.....	35
Subsequent Uploads of Data .....	36
Introduction.....	36
Jenzabar Tables in the CX Database .....	37
Jenzabar Tables.....	37
Transaction Types.....	37
Table Relationships.....	38
Advantages to Maintaining a Copy of the Jenzabar Tables.....	39
Maintaining a Copy of Jenzabar Tables.....	39
<b>APPENDIX A: PROGRAM FLOW .....</b>	<b>41</b>
<b>APPENDIX B: MAPPING OF DATA ELEMENTS .....</b>	<b>43</b>
Purpose .....	43
<b>INDEX .....</b>	<b>49</b>

# SECTION 1 – INTRODUCTION TO THE INTERFACE

## Overview

### Purpose of the Interface

The Jenzabar-CX API (Application Program Interface) enables faculty and students who access Jenzabar.com to see course information from their institution. Students can click a link to the Jenzabar CX Web Server where they can view, add, and drop courses. In turn, the CX Web Server provides a link enabling the student to return to Jenzabar.com with a single click.

### Purpose of the Upload Process

The CX-to-Jenzabar Data Upload process determines what pertinent data has changed within Jenzabar CX, then uploads this data to Jenzabar.com, allowing students access to data that is refreshed according to a time period specified by the CX institution. Data that is transferred to Jenzabar.com is stored in CX in a comma-delimited ASCII format in accordance with the specifications of the Jenzabar API.

### Purpose of this Guide

Use this guide as a source for information about the interface, and for setting up and maintaining files and data the interface uses.

### Intended Audience

This guide is for use by systems users at your institution's computer center. Systems users include the Jenzabar system coordinator, Jenzabar system administrator, Jenzabar software administrator, and programmer/analyst.

# Implementation Process

## Introduction

The implementation of the Jenzabar-CX interface consists of:

- Setup
- Data transfer
- Host links
- Optional components
- Adoption

This installation guide addresses these implementation considerations.

## Setup

The setup step for the Jenzabar-CX interface requires the institution to:

- Determine the Jenzabar administrator
- Obtain the unique Jenzabar college ID from Jenzabar.com
- Enter the Jenzabar college ID number into Jenzabar CX
- Ensure that Informix 7.3 and Perl 5.6 are installed on your institution's system

## Data Transfer

Data transfer includes both the retrieval (Get method) and storage (Post method) of data.

### **Get Method (Jenzabar polls host system)**

- Configure Jenzabar for polling in Test mode
- Run data transfer in Test mode
- Review Jenzabar and host logs and verify they are correct
- Configure Jenzabar for regular polling in Live mode

### **Post Method (Host system sends data to Jenzabar)**

- Configure the host system for posting in Test mode
- Run data transfer in Test mode
- Review Jenzabar and host logs and verify they are correct
- Configure the host system for regular posting in Live mode

## Host Links

The host links step for the Jenzabar-CX interface requires the system administrator to:

- Determine host links to be implemented
- Determine which host links use single-login
- Configure host links in Jenzabar in Off mode, if desired
- Test host links
- If tested in Off mode, turn on the links to make them live

## Optional Steps

If desired, the system administrator may:

- Review the Jenzabar look-and-feel settings, uploading school graphics if available
- Review the Jenzabar access settings, adjusting as necessary

## **Adoption**

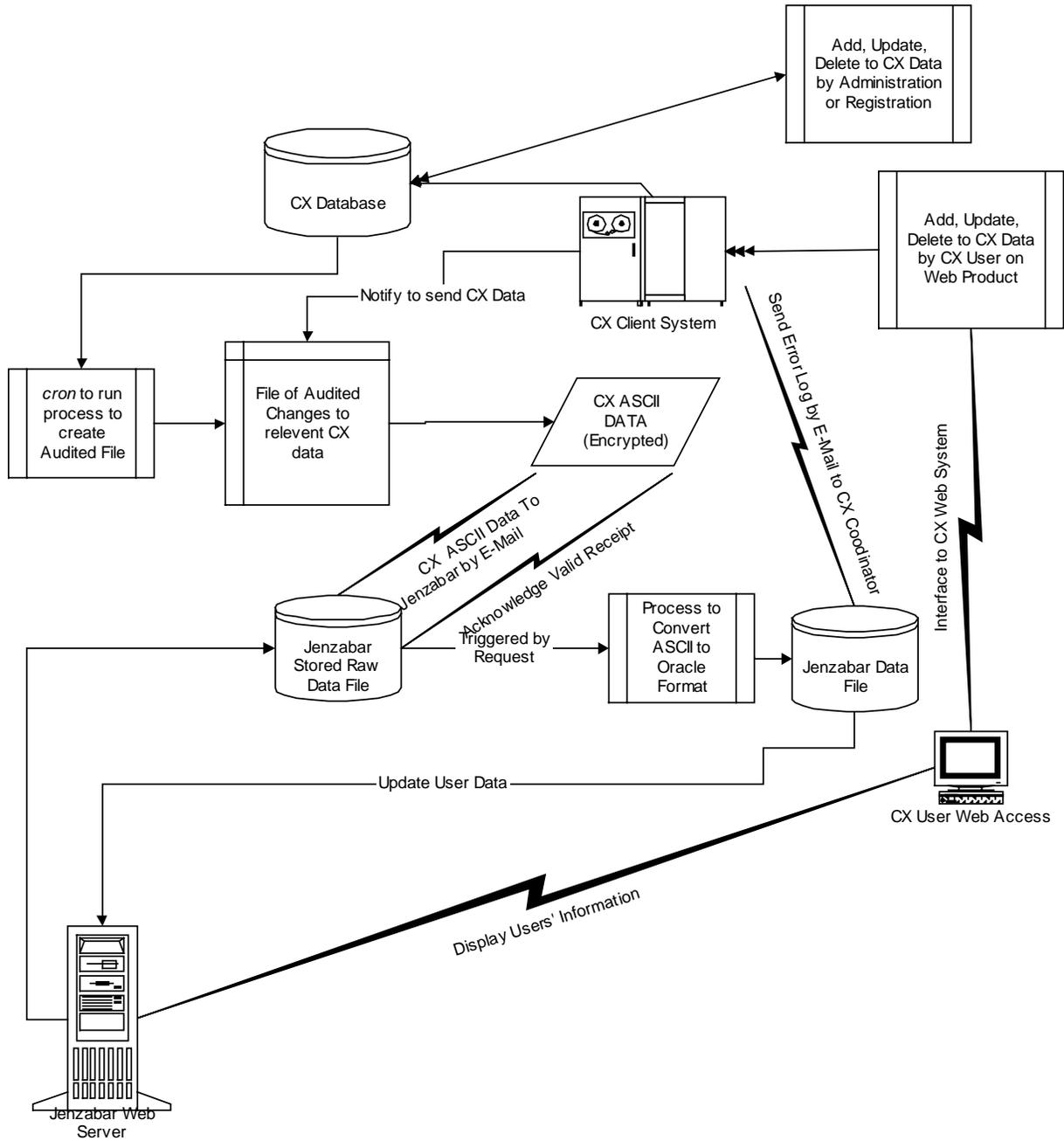
Also considered the Go Live step, the Adoption step requires the system administrator to:

- Run the host system's Jenzabar registration e-mail distribution utility, if available
- Run mail merge or another method for users without e-mail
- Monitor the Jenzabar Administrator Status screen for usage rates
- Run the Jenzabar Inactive Code extractor to re-notify users as needed

# Interface Diagram

## Introduction

The following diagram shows the flow of data between Jenzabar and the CX database via the CX Web products.



# Upload Process Description

## Introduction

Each institution using the Jenzabar-CX interface maintains tracking information about the information that flows between the two entities.

## Tracking Uploaded Files

The Jenzabar-CX Data Upload process maintains a serial number in the Jenzabar initialization file, located in `$CARSPATH/schema/student`. The parameters are stored in a table called `jenzini_rec`. After every successful upload, the serial number is incremented by 1. The script generates the ASCII data dump in the file `xxx.log` where `xxx` is the serial number for this batch upload file. The script also generates a log with the details of the upload. In case of error, the log file will have the name `xxx_date_time.log`. For a successful upload, the log file will have the name `xxx.log`. The system stores the log files in the directory path `TXTPATH/jenzabar`.

## Instructions for Uploading Files

Institutions run the CX-to-Jenzabar Data Upload process according to parameters in the Jenzabar initialization file located in `$CARSPATH/schema/student/jenzini`. The system administrator defines these parameters in accordance with instructions in this section.



## SECTION 2 – IMPLEMENTATION CONSIDERATIONS

### Overview

#### Introduction

This section contains information about the setup and use of the CX software that supports data maintenance and data uploads to Jenzabar.com. This information addresses:

- Establishing the Jenzabar-CX Administrator's account
- Defining your Website's URL
- Obtaining your institution's ID
- Populating the \$TXTPATH/jenzabar directory
- Accessing the Jenzabar API menu
- Creating the Jenzabar Initialization record
- Setting up department codes
- Setting up alternate address functionality
- Reviewing error messages
- Posting files
- Identifying Jenzabar-CX records
- Uploading information
- Running the Jenzabar Batch Upload Data script
- Using the Registration Code Coupon functionality
- Resetting the Session and Year parameters
- Setting up *cron* jobs for the *jenzupload* and *jenzcoupon* scripts
- Running *cron* jobs for the *jenzupload* and *jenzcoupon* scripts
- Setting up Authentication files

**Note:** For specific instructions regarding the setup and use of your institution's accounts on Jenzabar.com itself, including setup and login information for the Jenzabar-CX Administrator, faculty, and students, contact Jenzabar directly.

#### System Requirements

To use the Jenzabar-CX interface, you must have installed the following software on your system:

- Informix 7.3
- Perl 5.6

**Note:** For information about installing Perl 5.6, see the README accompanying CX SMO 12614.

# Implementation Steps

## Establishing the Jenzabar-CX Administrator's Account

Every institution using the Jenzabar-CX interface must designate an administrator who will have an account on jenzabar.com. To establish the account, contact Jenzabar at 617-492-9099 to obtain the name of the contact person to whom to send the following e-mail message:

I am with <official college/university name>, a client of Jenzabar-CX. Please set up our administrator's account using the following information:

Name: <administrator's name>

E-mail: <administrator's e-mail address>

URL: <link to your college/university; the main link, not the registration application>

School Colors: <if desired, specify school colors; otherwise, the default Jenzabar colors will be used>

## Defining Your Website's URL

Jenzabar must be able to locate the URL (the address of your web server) to call when a student or faculty member accesses your CX web server from Jenzabar. You have the option of defining your Website's URL with or without a single login.

- Single log-in allows a faculty member or student to log directly into the indicated CX Web page from Jenzabar.com without having to enter a username and password. The system performs the login automatically in the background.
- If you do not use the single log-in feature, a faculty member or student must enter a username and password to access the CX Web server from Jenzabar.

## Defining Your Website's URL Without Single Login

To set up your Web site's URLs without single login, you must first determine what the URL should be. You may have an URL for students and an URL for faculty, both of which must be in this format:

`http://<web server>:<port #>/<path>/<script>`

Examples:

`http://myweb:9041/cgi-bin/student/frame.cgi`

`http://myweb:9041/cgi-bin/faculty/frame.cgi`

To define your web site's URL without single login, follow these steps:

1. Log on to Jenzabar.com as the Jenzabar-CX administrator.
2. Click on **Administrator** at the top of the page.  
**Note:** If you are maintaining several sites, a drop-down list may display from which you will select the appropriate site.
3. Click on **Campus Links** on the left side of the page. The Campus Link Configuration page appears.
4. Click on **Add**, then complete the various sections as follows:
  - In the Link section, click the **On** radio button next to Status.
  - In the Link Text field, enter the text you want to display for the link to your registration application (e.g., College Registration, or Faculty Access).

- In the URL field, enter the link to your registration application. For example:

http://myweb:9041/cgi-bin/student/frame.cgi

or

http://myweb:9041/cgi-bin/faculty/frame.cgi

- In the Location field, select the location where you want this link to display. Typically, this link appears in the Course Manager area, the place where students and faculty view a list of their courses. An (L) indicates the link will appear on the left side of the screen, while an (R) indicates the link will appear on the right side of the screen.
  - In the Variables Passed section, click the radio button for On URL.
  - In the Permissions section, click a check box for each entity that should have permissions to this link.
5. On the Pop-up Screen, click the **Off** radio button.
  6. In the Redirect Variable section, click the **On** radio button.
  7. In the Variable Name field, enter **page**.
  8. Enter one of the following in the Redirect Value field to specify the default page to which a student or faculty member will be directed when they click this link:
    - For a student link, enter **regist.cgi** or any other valid cgi script name
    - For a faculty link, enter **classl.cgi** or any other valid cgi script name

**Note:** The regist.cgi and classl.cgi scripts represent default pages to which students and faculty will be directed via the special links. You can decide which pages will be best for your institution's needs. For a complete list of default pages, refer to the menu.cgi file in the appropriate student or faculty directory as follows:

\$CARSPATH/web/cgi/student  
\$CARSPATH/web/cgi/faculty
  9. In the Single Log-In Variable section, click the Status radio button to **Off**.
  10. Click the Open in New Browser radio button to **Yes** or **No**.
    - If Yes, an entirely new and separate session will open for the CX Web link. When this session is closed, the parent Jenzabar.com session will remain open.
    - If No, the same browser session will be used to link to the CX Web. When this session is closed, you cannot return to Jenzabar.com unless you use the **Back** button on the Browser.
  11. In the From Variable section, enter a variable named either **from** or **link**.
    - Enter **from** if the Open in New Browser radio button is set to Yes.
    - Enter **link** if the Open in New Browser radio button is set to No. With this option, you cannot use the Back to Jenzabar link. You must use the **Back** button to return to Jenzabar.com.

**Note:** When the link to the CX Web is opened in a new browser, a special "Back to Jenzabar" link appears in the CX Web application. Clicking this link closes the browser, but the parent browser linked to Jenzabar.com still exists. Setting the Variable Name to From informs the application to open a new browser, while Link indicates the same browser will be used to open the CX Web.
  12. If you are ready to commit your changes, click **Save** and the link is ready to use.
- If you need assistance, contact the following:
- For help determining which URLs to use for your web site, contact Support Services.
  - For help setting up the URLs on Jenzabar.com's web site, contact Jenzabar.com's Customer Support.

## Defining Your Website's URL With Single Login

The Single Login feature allows a faculty member or student to log directly into the CX Web page from Jenzabar.com without having to enter a username and password. With single login, logging in is performed automatically.

To set up your Web site's URLs with single login, you must first determine what the URL should be. The same URL is used for both student and faculty access and must be in this format:

`http://<web server>:<port #>/<path>/<script>`

Example:

`http://myweb:9041/cgi-bin/public/entry.cgi`

In the above example, the entry.cgi script interprets the single login information received on the incoming URL and uses it to automatically log in authorized users of the URL.

**Note:** Unless you have specifically modified your server to use the Single Login logic contained in entry.cgi, you must use the script entry.cgi as shown above for single login to work.

To define your website's URL with single login, follow these steps:

1. Log on to Jenzabar.com as the Jenzabar-CX Administrator.
2. Click on **Administrator** at the top of the page.
3. Click on **Campus Links** on the left side of the page. The Campus Link Configuration page will appear.
4. Click on **Add**, then complete the various sections as follows:
  - In the Link section, click the **On** radio button next to Status.
  - In the Link Text field, enter the text you want to display for the link to your registration application (e.g., College Registration or Faculty Access).
  - In the URL field, enter the link to Single Login script. For example:  
<http://myweb:9041/cgi-bin/public/entry.cgi>
  - In the Location field, select the location where you want this link to display. Typically, this link appears in the Course Manager area, the place where students view a list of their courses. An (L) indicates the link will appear on the left side of the screen, while an (R) indicates the link will appear on the right side of the screen.
  - In the Variables Passed section, click the radio button for **On URL**.
  - In the Permissions section, click a check box for each entity that should have permissions to this link.
5. On the Pop-up Screen, click the **Off** radio button.
6. In the Redirect Variable section, click the Status radio button to **On**.
7. In the Variable Name field, enter **Redirect**.
8. In the Redirect Value field, enter the redirect path as follows:
  - The format is: `<web server>:<port #>/<path>/<script>&<default page>`
  - Example of student URL:  
`myweb:9041/cgi-bin/student/frame.cgi&page=regist.cgi`
  - Example of faculty member URL:  
`myweb:9041/cgi-bin/faculty/frame.cgi&page=classl.cgi`

**Note:** The regist.cgi and classl.cgi scripts represent default pages to which students and faculty will be linked via Single Login. You can determine which pages are best suited for your institution. For a complete list of default pages, refer to the menu.cgi file in the appropriate student or faculty directory as follows:

\$CARSPATH/web/cgi/student  
\$CARSPATH/web/cgi/faculty

9. In the Single Log-In Variable section, set the Status radio button to **On**.
10. In the User ID field, enter **p1**.
11. In the Password field, enter **p2**.
12. Click the Open in new browser radio button to **Yes**.
13. In the From Variable section, enter **student** or **faculty** in the Variable Name field, as applicable.
14. If you are ready to commit your changes, click **Save** and the link is ready to use.

If you need assistance, contact the following:

- For help determining which URLs to use for your web site, contact Jenzabar-CX Support Services.
- For help setting up the URLs on Jenzabar.com's web site, contact Jenzabar.com's Customer Support.

### Obtaining Your Institution's ID

To upload files to Jenzabar correctly, you must set up the Jenzabar-CX Administrator account on Jenzabar.com and obtain a unique institution ID.

To obtain your unique institution ID:

1. Access the Jenzabar.com address (e.g., [www.jenzabar.com](http://www.jenzabar.com)).
2. Log on as the Jenzabar-CX Administrator.
3. Click on **Administrator** from the options at the top of the page. The Status page will appear.
4. Scroll to the bottom of the page to locate the line that starts with the words, "Your college code is ...". Make note of the code for later use.

### Populating the \$TXTPATH/jenzabar Directory

The directory \$TXTPATH/jenzabar must exist, must have read/write/execute permissions set to 770, must be owned by carsu and must have the group set to carsprog. Following is an example of the permissions and ownership for this directory:

```
drwxrwx--- 2 carsu carsprog 99 May 16 13:19 jenzabar
```

## Accessing the Jenzabar API Menu

Jenzabar provides menu access to tools to help you maintain the Jenzabar API. This menu access can be enabled or disabled with the macro `ENABLE_FEAT_JENZABAR_API` defined in `$CARSPATH/macros/custom/student`. The macro value defaults to "Y", so the Jenzabar API menu is enabled in the standard product. This menu is password-protected according to the menu password set up for the Registrar's module and should be located on your system in a path similar to the following:

- [b] Student Management
- [a] Registrar
- [o] Jenzabar API

**Note:** With the exception of the Jenzabar Initialization option, errors that occur during the running of the options on the Jenzabar menu and maintenance submenu are e-mailed to the Jenzabar-CX Administrator and/or the individual executing each option.

The menu options are:

### Jenzabar Initialization

Allows access to the Jenzabar Initialization record (`jenzini_rec`) where variables for the Jenzabar API process are defined.

### Create/Upload Data File

Allows the user to run the Jenzabar-CX Data Upload process manually. If the `JENZABAR_MANUAL_UPDATE_DATA` macro in `macros/custom/student` is set to "Y", then data files created by this process will be automatically uploaded to `Jenzabar.com`. If the macro is set to "N," then the files will be created but will only be stored in `$TXTPATH/jenzabar` and will not be automatically uploaded to `Jenzabar.com`.

### Upload Single Data File

Allows the user to upload a data file with updated faculty and/or student information to `Jenzabar.com`. Unlike the Create/Upload Data File option, this option does not create datafiles; it only sends files that have already been created and stored in `$TXTPATH/jenzabar`.

### Process All New Coupons

Allows the user to generate Registration Code Coupon e-mails en-masse for faculty and students. The Registration Code Coupon e-mail includes the faculty or student's Registration Code, a special commercial offer (if available), and instructions for setting up an account on `Jenzabar.com`. Usually, an individual will only be sent a Registration Code Coupon once. When someone has been sent a coupon, the process flags that person as having received a coupon, and that person will not be sent a coupon again automatically. Since this option should only be run when faculty and students are to be sent Registration Code Coupons, a warning message displays asking you to confirm your intentions to this option.

Four date fields are defined in the Jenzabar Initialization record controlling the production of Registration Code Coupons. The Coupon Begin Date and Coupon End Date fields specify the beginning and ending dates for the inclusion of a special offer (i.e., coupon). The Regcode Begin Date and Regcode End Date fields specify the beginning and ending dates for all of the Registration Code Coupon output. If the current date falls outside this begin and end date range, Registration Code Coupons will not be sent to any faculty or students via this menu option.

### Process Single Coupon

Allows the user to view, print, or send a single Registration Code Coupon to a student or faculty member indicated by an ID number. This option is useful in cases where a faculty or student has misplaced or did not receive an original Registration Code Coupon. The Jenzabar-CX API process will not automatically send an individual more than one

Registration Code Coupon. However, by running the Process Single Coupon option, an operator may view or re-send a coupon to the indicated individual.

Registration Code Coupons may be produced by this process regardless of whether or not the faculty member or student's jenzprofile\_rec.reg\_code\_sent value is Y or N. However, if a coupon is actually e-mailed to an individual, the value is set to Y; thus preventing another coupon e-mail from being sent the next time the Process All New Coupons option is run.

If the current date falls outside the Coupon Begin Date and Coupon End Date range, the special offer will not be included in the Registration Code Coupon that is produced. However, a Registration Code Coupon is always produced via the Process Single Coupon option regardless of the values in the Regcode Begin Date and Regcode End Date fields in the Jenzabar Initialization record.

The Process All New Coupons and Process Single Coupon options in the Jenzabar API allow you to automatically send Registration Code Coupon e-mails to faculty and students. These e-mails contain any special offers associated with signing up on Jenzabar.com as well as instructions for the sign up. You can also generate hardcopy Registration Code Coupons using the Process Single Coupon option.

You use the Coupon Text field in the Jenzabar Initialization screen to create the text for Registration Code Coupons. See the section, *Creating the Jenzabar Initialization Record*.

## Accessing the Jenzabar Maintenance Submenu

You access this submenu from the Jenzabar API Menu. This submenu provides access to special Jenzabar API maintenance features and is password-protected according to the password set up for the Registrar's module.

**Note:** With the exception of the Jenzabar Initialization option, errors that occur during the running of the options on the Jenzabar menu and maintenance submenu are e-mailed to the Jenzabar-CX Administrator and/or the individual executing each option.

The options available from this submenu are:

### Engage API Process

Allows you to engage the API process and run the Create/Upload Data File option. Since the Create/Upload process is disengaged if an error occurs, you may need to run the Engage API Process option to re-engage the Create/Upload Data File process after resolving errors. This option also enables the jenzupload script to run from a *cron* file.

**Note:** This option removes the jenzpost.ip file if it exists in \$CARSPATH/events. This file prevents the API processes from running.

### Disengage API Process

Allows you to disengage the API process which disallows running the Create/Upload Data File option. You may want to run this option to research and resolve data or posting errors. If you select this option, you will need to select the Engage API Process option to re-engage the Create/Upload Data File option. This option also prevents the jenzupload script from running from a *cron* file.

**Note:** This option creates the jenzpost.ip file in \$CARSPATH/events. This file prevents the API processes from running.

### Reset Session and Year

Allows you to reset the Session and Year parameters in the Jenzabar Initialization Record. These parameters should be changed occasionally (such as each semester) to prevent the volume of data from the API create/upload process from becoming too large. When you

change these parameters, the process purges the temporary API data for past CX sessions. This purge does not affect any CX live data nor does it remove any data from Jenzabar.com.

While this process is running, it creates the jenzpurge.ip file in \$CARSPATH/events. To preserve data integrity, this file prevents other API processes from running while the Session and Year are being reset, and while data is being purged. This file is removed when the Reset Session and Year process is complete.

### **Purge All Jenzabar Data**

This process removes *all* data in the Jenzabar-CX API records. Run this option *only* in cases where you know for certain you want to delete all the data in the Jenzabar-CX records. For example, when you are setting up the Jenzabar API interface for the first time and you need to create a new initial upload file. You may also want to run this option if the CX data no longer corresponds correctly to the Jenzabar data; you may want to remove all data, then rerun the process to create a new data file similar to the initial upload. The jenzpurgeall script in \$CARSPATH/modules/regist/scripts controls this process.

**CAUTION:** Run this option only when absolutely necessary.

### **Creating the Jenzabar Initialization Record (jenzini\_rec)**

After you install the Jenzabar API SMO, the Jenzabar Initialization record must be set up according to the requirements for the API.

If the macro ENABLE\_FEAT\_JENZABAR\_API is set to "Y" in \$CARSPATH/macros/custom/student, you can access the Jenzabar Initialization record from the following menu path:

- [b] Student Management
  - [a] Registrar
    - [o] Jenzabar API
      - [a] Jenzabar Initialization

The fields on this screen are as follows:

#### **School Id**

The unique school ID assigned to your institution by Jenzabar. To locate your institution's code, follow the procedure *Defining Your Institution's ID* in this document.

#### **Session**

The beginning session which is the first session to use in the batch data process. All session-based data greater than or equal to this starting session will be included in the data files created. For example, if the session value is set to "FA," then data will be uploaded for the session FA and all sessions following it within the current session year. Therefore, if the academic year is 2000, data for "FA 2000" and "SP 2001" would be selected by the process, but data for "SP 2000" would not be selected. This field allows students to see information for current and future sessions.

**Note:** The Session field can only be edited the first time the Jenzabar Initialization Record is added. After that, you can only edit the Session and Year values by using the Reset Session and Year option on the Jenzabar Maintenance submenu.

#### **Year**

The beginning year which is the first year to use in the batch data process. All year-based data greater than or equal to this starting year will be included in the Jenzabar uploads. For example, if the year value is set to "2000," then data will be uploaded for the year 2000 and all years beyond, including years 2001, 2002, etc. This field allows students to see information for current and future sessions.

**Note:** The Session field can only be edited the first time the Jenzabar Initialization Record is added. After that, you can only edit the Session and Year values by using the Reset Session and Year option on the Jenzabar Maintenance submenu.

### **Data Export Mode**

The data export mode, either TEST or LIVE. The default value is TEST, and it should remain as TEST until Jenzabar advises you to change it. When you are instructed to change this value to LIVE, you must reset the "Serial Number" parameter to 1 to restart the sequential numbering process for the LIVE data. The first data file uploaded to Jenzabar.com must always be 1.dat unless you are instructed otherwise.

### **Admin E-mail Address**

The institution's e-mail address for the Jenzabar-CX Administrator. Error and status messages encountered during the batch data process will be sent to this e-mail address. Note that error and status messages are also sent to any user running the Jenzabar options, even if he/she is not the administrator.

### **Serial Number**

The starting sequential serial number. This is the number that the system will assign to the next batch data file that is created. When you first edit this field, you should set the value to 1. If you change the Data Export Mode from TEST to LIVE (or from LIVE to TEST), you must reset the serial number to 1. The system increments this number after every successful upload file creation; therefore, the serial number then becomes the name of the batch data file in \$TXTPATH/jenzabar with a .dat extension.

If this number becomes corrupted, you can determine what it should be by determining which is the most recent .dat file created and stored in \$TXTPATH/jenzabar. The Serial Number value is one greater than the number of the last batch data file created. For example, if the last data file created was 92.dat, then the value for Serial Number should be updated to 93.

### **Signup URL**

The URL to which students will be directed to sign up on Jenzabar.com. For example:

[www.jenzabar.com/info/login/signup.asp](http://www.jenzabar.com/info/login/signup.asp)

This URL is automatically included on Registration Code Coupon forms generated by the Process All New Coupons and Process Single Coupon options.

### **Jenzabar Post URL**

The URL for posting data update files to Jenzabar.com. This is the URL to which the sequentially numbered data update files are sent to Jenzabar.com via an HTTP Post operation (e.g., <http://post.jenzabar.com/post.htm>).

Data files sent to this URL must be sent in the order in which they were produced. The API process manages this automatically unless an error occurs. If an error occurs, the API disengages and the errors must be corrected in the data file in which they occurred. Then you must manually send the corrected .dat file to Jenzabar.com to complete the sequence. Any errors set back to the CX side from the indicated Jenzabar.com URL will contain the last sequence number that was processed on Jenzabar.com. If you send a file out of sequence, an error is displayed informing you of the file you need to resend to correct the file order sequence.

### **Proxy Post URL**

The proxy URL for your CX Web server. This URL will be different for each institution; not all institutions have a proxy URL. If a proxy URL is used, its value will resemble "http://www-gw.yourserver.edu:8080". For help in determining your proxy URL, contact the Systems Desk in Jenzabar-CX Support Services.

**Coupon Text**

The text for any special offers associated with account signup on Jenzabar.com. If you are not currently aware of any special offers, you may leave this blob field blank. If it is not expired, this coupon will be included in the e-mail sent to faculty and students. The following is an example text for a coupon. It is only an example and does not reflect any actual or implied offer from the Nike corporation.

```
*****!!GREAT OFFER FROM NIKE!!*****
*
*   Wouldn't you look good in a pair of           *
*   NBA Basketball star Sammy Brown's           *
*   Boom Baby! (tm) High Performance Action Shoes? *
*
*   Sign up on Jenzabar.com and become eligible to *
*   receive a $10 gift certificate toward the purchase of *
*   Sammy Brown Boom Baby! (tm) Action Shoes or any other *
*   Nike Product. Sign up now! This great offer expires *
*   on November 3, 2000.                             *
*
*****
```

**Signup Text**

The text containing the instructions for setting up an account on Jenzabar.com. These instructions are included in the Registration Code Coupons that are e-mailed to students. The students can refer to the instructions to help them through the Jenzabar registration process.

**Note:** It is recommended that you use this text for the Registration Code Coupon, including the variables SIGNUP\_URL, EMAIL, USERTYPE, REGCODE, and JENZABAR\_URL. The variables will be interpreted and executed.

## Sample instructions are:

On Jenzabar.com, you can access course materials, research registration and financial aid information, contact your fellow classmates, and experience a wide range of member-only opportunities.  
And when you sign up on Jenzabar.com, you get a free e-mail account and can use Jenzabar.com as your home-base for surfing the Web. Best of all, these services are offered to you absolutely FREE!!! So sign up on Jenzabar.com today!

HERE FOR THE FIRST TIME? READY TO ACTIVATE YOUR JENZABAR ACCOUNT?  
Click on the link below and follow the on-screen instructions!

`$$SIGNUP_URL?email_address=$EMAIL&user_type=$USERTYPE&registration_code=$REGCODE`

NOT READY TO SIGN UP YET? Click here for more information

`$$JENZABAR_URL`

ALREADY HAVE A JENZABAR ACCOUNT, but need to enter a NEW Registration Code from another college or university? Activate your new Registration Code now! Here's how:

1. go to `$$JENZABAR_URL`
2. Log in with your Username and Password.
3. Click the "I have a Registration Code" link under the calendar to access the Registration Code Entry Screen.
4. Enter your NEW Registration Code. Your Registration Code for Jenzabar University is `$$REGCODE`.

IF YOU HAVE DIFFICULTY SIGNING UP, or if this is a printed copy, follow these simple instructions:

1. Go to `$$JENZABAR_URL`.
2. If you DO NOT already have a Jenzabar account, then click on "Sign Me Up" and follow the instructions to get your new Username and to choose your Password.
3. Click the "Calendar" link to start using your account.
4. Click the "I have a Registration Code" link under the calendar to access the Registration Code Entry Screen. Your Registration Code is: `$$REGCODE`.

Once you have an account, here are just a few of the many things you can do!

- \* Click the "Calendar" link to view your current schedule. Hint: If you don't see your classes on the schedule, be sure you are in the correct month and day.
- \* Click the "Courses" link to view your current and future courses.
- \* Set up your own free Jenzabar e-mail account.
- \* See who is in your classes and get a chat session going.
- \* ... Or just spend some time exploring your new portal to the Internet!

So, WHAT IS JENZABAR anyway?

Jenzabar.com is the single most useful Web site for college students. From any computer with Internet access, you can reach your own customized Jenzabar site filled with the information you need every day. Whether you want to buy the books for your classes, join a discussion group open only to your classmates, research your final paper, or much, much, more, you'll find it at Jenzabar.com.

### Coupon Begin Date

The first date on which the "coupon" (i.e., the special offer) is first available to be sent in the Registration Code Coupon e-mail.

### Coupon End Date

The date on which the "coupon" (i.e., the special offer) expires. If a coupon is expired, it will not be sent as part of the Registration Code Coupon e-mail. Note that even if a "coupon" is not included in the Registration Code Coupon e-mail, the instructions for signing up on Jenzabar.com will still be included, according to the Regcode Begin and End dates (see below).

### Regcode Begin Date

The first date for which the overall Registration Code Coupon e-mail is first available to be sent to faculty and students.

### Regcode End Date

The date after which the overall Registration Code Coupon e-mail will no longer be sent. If the "Process All New Coupons" option is run on a date after the Regcode End Date, the administrator will receive an e-mail similar to the following:

```
The current date falls outside the date range for e-mailing Registration Code Coupons.  
No one will be e-mailed Registration Code coupons at this time.
```

**Note:** Regardless of the values of the Regcode Begin and End Dates, a Registration Code Coupon can always be generated using the Process Single Coupon option.

### Setting Up Department Codes

Department codes require special setup for the Jenzabar API. To set up department codes, define the department codes in the Course Record (crs\_rec) and in the Department Table (dept\_table). Make sure all departments defined in the crs\_rec.dept field in the Course Record (crs\_rec) are also defined in the Department Table (dept\_table).

**Note:** To determine if you have department codes defined in the crs\_rec that are not defined in the dept\_table, run the following SQL script:

```
select unique crs_rec.dept from crs_rec  
where crs_rec.dept not in (select dept from dept_table);
```

This script should list any department codes that are not defined in the dept\_table. Update/Add rows to the dept\_table as necessary, then re-run the dept.sql script as shown above until all department codes have been properly defined in the dept\_table.

In the standard CX menu, you can access the Department table (dept\_table) from Student Management / Registrar / Table Maintenance / Registrar (D-E) /Division/ Department. Your menu path may be different if your Jenzabar Coordinator has modified menus.

### Setting up Alternate Address Functionality

If you are planning to use the automated e-mailing process for Registration Code Coupons, the Alternate Address Record (aa\_rec) must contain valid e-mail addresses for all faculty members and students to whom they will be sent.

For proper e-mail setup, the aa\_rec.aa field must contain the e-mail code prefix specified in the macro EMAIL\_AA\_CODE\_PREFIX in \$CARSPATH/macros/custom/common. The prefix defined in this macro is usually EML, so EML will be used as an example here. The aa\_rec.aa field must contain a value using the macro value (e.g., EML or EML1) and the aa\_rec.line1, line2, and line3 fields must contain the e-mail address. If a faculty or student has more than one e-mail address in the aa\_rec according to this scheme, then the priorities for the aa codes must be set up correctly in the aa\_table.

The Alternate Address table (aa\_table) must contain valid aa codes for the e-mail code prefix specified in the macro EMAIL\_AA\_CODE\_PREFIX in \$CARSPATH/macros/custom/common. The prefix defined in this macro is usually EML, so EML will be used as an example here. If more than one EML aa code exists, then these codes MUST be prioritized (i.e., the aa\_table.priority field must be set up to put the EML codes in priority order). For example, the EML1 aa code might have a priority of 1, and the EML2 aa code might have a priority of 2, and so on. Do not assign the same priority for more than one EML aa code; the priority value must be unique for each e-mail aa code.

**Note:** For every e-mail entry in the aa\_table, the email flag must be set to Y.

## Reviewing Error Messages

No special setup for error messages is required. Note, however, that if the process encounters errors, it sends messages to the e-mail address indicated in the Admin E-mail Address field in the Jenzabar Initialization Record (jenzini\_rec), and also to the user who runs each process.

For Create/Upload Data File (jenzupload) and Upload Single Data File (jenzpost) processes, errors that occur during the data file creation and/or posting process will be sent to the screen and e-mailed to the Jenzabar-CX API Administrator and/or the person running these processes. The error messages will contain an explanation of the error(s) that occurred and will include the sequence number of the data file in which the error(s) occurred.

In addition to this error notification, each .dat batch file created on the CX side in \$TXTPATH/jenzabar will have a corresponding .log file. If no error has occurred, this file contains the start and finish times for the creation of the upload file, the tables that were populated, and the number of records deleted, added, and updated in each table. The name of this log file is {serial#}.log. If an error has occurred, the .log file contains process errors and/or database error messages that you can use to investigate the problem. The name of a log file containing an error is in the format of {serial#}\_{date}\_{time}.log. The number of records deleted, added, and updated are displayed for each set of data. Some data types do not allow deletes or updates. In this case, the fields corresponding to the delete and update actions for these data types are marked with N/A.

When the jenzpost process detects an error, the Create/Upload Data File (jenzupload) process is disengaged, and a file is placed in \$CARSPATH/events called jenzpost.ip. The presence of this file indicates that a posting error occurred and the jenzupload process will not be allowed to run until the errors are resolved. To re-engage the process, you must select the Engage API Process option on the Jenzabar Maintenance menu.

If you encounter any of the following error messages when running the Create/Upload Data File or Upload Single File menu options, the reason could be that the Jenzabar Post URL in the Jenzabar Initialization Record is inaccurate. Verify the definition of this URL. If the URL is correct and the errors persist, try to resolve the problem or contact the Jenzabar Support Desk.

- 400 Invalid Request
- 405 Method not allowed
- 504 DNS Timeout

## Posting Files

When a data upload file is uploaded to Jenzabar.com using either the Create/Upload Data File or Upload Single Data File option, the process creates a file in the format of {serial#}.pst in \$TXTPATH/jenzabar. This file contains the time when the data file post was initiated, when it was completed, where it was sent, and any errors that occurred during the post operation. The creation of this file is automatic and does not require any special setup.

## Identifying In-Process Files

The Jenzabar-CX API process uses a series of in-process files indicating that certain processes are currently running. These files are created in the \$CARSPATH/events directory and are removed when a process finishes successfully. The files and their corresponding uses are:

jenzupload.ip	Created when the Create/Upload Data File process (jenzupload) is run to process changes to the API data. It prevents other API processes from running before the current process is completed. When the jenzupload process is finished, the jenzupload.ip file is removed.
jenzpost.ip	Created when a posting error has occurred as a result of an HTTP posting operation to Jenzabar.com. This file remains in \$CARSPATH/events until the errors are resolved. As long as the file exists, other API processes are prevented from running. After the posting errors are resolved, you may remove this file from \$CARSPATH/events using the Engage API Process option on the Jenzabar Maintenance menu.  This file is also created by the Disengage API Process option which can be invoked at any time to prevent the API processes from running. This may be necessary in order to research errors, or to make sure that data files are not continually being created by <i>cron</i> and uploaded to Jenzabar.com.
jenzpurge.ip	Created when the Reset Session and Year option is run. Since resetting the session and year parameters involve the purging of temporary CX API data from past sessions, the presence of this file will prevent other API processes from running CX while the purge is taking place. The jenzpurge.ip file is automatically removed when the process is finished.

## Identifying the Jenzabar-CX Records

To facilitate updates to Jenzabar.com for changes made in CX since the last updates on Jenzabar.com, a replica of the data on Jenzabar.com is stored on the CX database in several Jenzabar-CX API records. These records are:

<b>jenztrm_rec</b>	Contains Jenzabar-CX API TRM data
<b>jenzdpt_rec</b>	Contains Jenzabar-CX API DPT data
<b>jenzprs_rec</b>	Contains Jenzabar-CX API PRS data
<b>jenzcst_rec</b>	Contains Jenzabar-CX API CST data
<b>jenzccd_rec</b>	Contains Jenzabar-CX API CCD data
<b>jenzcrs_rec</b>	Contains Jenzabar-CX API CRS data
<b>jenzsch_rec</b>	Contains Jenzabar-CX API SCH data
<b>jenzcrp_rec</b>	

Contains Jenzabar-CX API CRP data

### **jenzprofile\_rec**

Contains profile information unique to each faculty member or student in the Jenzabar-CX API data.

These records are only accessible interactively by using the senter2 command; however, Jenzabar-CX strongly encourages you to *not* directly modify the data in these records. Any changes you make could cause the data on Jenzabar.com to not match up appropriately with CX data. Contact Jenzabar before changing any values in these database records.

## **Managing Sequence Numbers**

A sequence number is sent to Jenzabar.com with each data file that is uploaded. When a data file is created and stored in \$TXTPATH/jenzabar, its name is derived from the serial sequence number in the form of {serial\_no}.dat. The serial sequence number is stored in the Jenzabar Initialization Record (jenzini\_rec.serial\_no).

When a file is uploaded to Jenzabar.com, the sequence number is obtained from the filename and sent as a variable to the HTTP posting process. The sequence number helps Jenzabar.com determine if the data files being posted to the website are in the proper sequence. This becomes very important if errors occur. Errors must be corrected in data files and resent in sequence. If errors occur on Jenzabar.com during a posting operation, the sequence number is not incremented on Jenzabar.com, so the erroneous data may be corrected and resent in sequential order.

The API process manages the sequential order of data files automatically unless an error occurs. If an error occurs on a file posted to the URL, the API process is automatically disengaged and the errors must be corrected in the data file in which they occurred. You must manually resend the corrected .dat file to Jenzabar.com to complete the sequence.

Any errors returned to the CX side from the indicated Jenzabar URL will contain the last sequence number that was processed on Jenzabar.com. If you send a file out of sequence, an error is displayed informing you of which file you need to send to resequence the file order.

This serial sequence numbering system helps ensure the integrity of the data that is uploaded to Jenzabar.com. It also helps the API Administrator and his/her associates resolve errors that occur, then repost corrected data files in the proper sequence.

## **Uploading Information**

When the Jenzabar-CX data upload script (jenzupload) is run via the Create/Upload Data File option, it generates a message containing various types of information, including process parameters, file numbers, etc. When you run the process, review the output to make sure the process completed successfully.

The process generates a message similar to the following:

```
School id:          109152
Beginning Session:  FA
Beginning Year:    2000
X-fer mode:       TEST
Admin email:      yourname@yourcollege.edu
Batch#:           48
Data file:        /usr/carsi/text/jenzabar/48.dat
Jenzabar POST URL: http://post.jenzabar.com/post.htm
Proxy Server URL: http://www-gw.yourserver.com:8080

Processing Data:   TRM DPT PRS CCD CST CRS SCH CRP

Batch# incremented to: 49
Log file: /usr/carsdevi7/text/jenzabar/48.log
```

**Note:** The Proxy Server URL is blank on your server if you do not have one.

## Running the Jenzabar Batch Upload Data Process

After you complete the setup of the interface, you can run the batch data upload process. To create a successful batch upload data file and to send this file to Jenzabar.com, perform the following steps:

1. Verify that the JENZABAR\_MANUAL\_UPDATE\_DATA macro is defined (Y or N) according to your institutional requirements.

**Note:** The default setting for the JENZABAR\_MANUAL\_UPDATE\_DATA macro in macros/custom/student is "N", which means that when the Jenzabar File Upload process is run from the menu, data files containing updates to Jenzabar data will *not* be automatically uploaded to Jenzabar.com. This is to prevent an institution from attempting to send a data file to Jenzabar.com prematurely. When the Jenzabar File Upload process creates data files, it stores them in the directory \$TXTPATH/jenzabar and names them with a sequential number with a .dat extension. If the JENZABAR\_MANUAL\_UPDATE\_DATA is "N", then these files are not uploaded to Jenzabar.com automatically. This enables an institution to test the process before actually uploading data to Jenzabar.com.

Jenzabar-CX recommends that you carefully consider when it is appropriate to set this macro to Y, thus enabling the transfer of data files to Jenzabar.com.

2. To enable the file upload feature now, enter the following at the UNIX prompt:

```
% cd $CARSPATH/macros/custom
% make co F=student
% vi student
```

3. Change the macro value:

```
% make cii F=student L="Modify JENZABAR_MANUAL_UPDATE_DATA macro."
```

4. Install the menu option to see the change to the macro:

```
% cd menuopt/regist/scripts
% make reinstall F=jenzupload
```

5. Run the Jenzabar File Upload Process, using the following CX menu path:

```
[b] Student Management
    [a] Registrar
        [o] Jenzabar API
            [a] Jenzabar File Upload
```

If the JENZABAR\_MANUAL\_UPDATE\_DATA macro is set to "Y", the process automatically uploads any existing .dat files to Jenzabar.com. If however, the macro is set to "N", the process will write the .dat file in \$TXTPATH/jenzabar; it will *not* automatically upload the .dat file to Jenzabar.com. Regardless of the value of the macro, the Jenzabar Single Upload menu option enables you to upload to Jenzabar.com any .dat file that you specify.

When you run the Jenzabar File Upload Process option, it generates a status message indicating the sequential number for the .dat file being created. It stores the .dat file in \$TXTPATH/jenzabar using a file name in the format {serial\_no}.dat.

6. Check e-mail for any status and error messages.

If any errors occur during the process, these error messages are stored in a .log file in \$TXTPATH/jenzabar. The process also sends an error message e-mail to the Jenzabar-CX Administrator using the e-mail address specified in the Admin E-mail Address field in the Jenzabar Initialization Record (jenzini\_rec). If someone other than the administrator has run the process that generated the error, he/she will also receive the error message e-mail.

7. Repeat the above steps periodically to make sure the changes that occur on your system are uploaded to Jenzabar.

### Using Registration Code Coupon Functionality

With the "Process All New Coupons" option, Registration Code Coupons can be automatically emailed to all faculty and students who were included in the Create/Upload Data File process. The coupons can also be viewed, printed, or sent manually via the Process Single Coupon option.

The Registration Code Coupon includes the faculty or student's Registration Code, a special commercial offer (if available), and instructions for setting up an account on Jenzabar.com. The Registration Code text is defined in the Jenzabar Initialization Record (jenzini\_rec) in the Signup Text field (jenzini\_rec.signup\_blob). The script that controls the Registration Code Coupon process is called "jenzcoupon" and is located in \$CARSPATH/modules/regist/scripts.

When Registration Code Coupons are sent through email, faculty and students may click a link that automatically inputs their Registration Code, thereby simplifying the signup process. There are some variables used to accomplish this feature, and they are as follows:

\$SIGNUP\_URL = URL for signing up on Jenzabar.com, as indicated in jenzini\_rec.signup\_url.

\$EMAIL = E-mail address of the faculty or student to which the Registration Code Coupon has been sent. The email address is obtained from the aa\_rec.

\$USERTYPE = The indicator for the person to whom this Registration Code Coupon has been sent. STU for a student, FAC for a faculty member.

\$REGCODE = The faculty member or student's Registration Code.

\$JENZABAR\_URL = The URL for the main Jenzabar.com website. This value is obtained by trimming the \$SIGNUP\_URL to the .com portion of the address.

The jenzcoupon script determines what the above values should be, then substitutes those values in the text of the Registration Code Coupon before producing it.

### Enabling/Disabling the "Process All New Coupons" Option

When you run the Process All New Coupons option after running the Create/Upload Data File option, you will end up sending Registration Code Coupon emails to all of the faculty and students selected by the process.

Jenzabar recommends that you use this option only when you are sure you want to send Registration Code Coupons to all students. The macro called JENZABAR\_ALLOW\_ALL\_COUPONS allows you to alternately enable or disable the Process All New Coupons option.

The JENZABAR\_ALLOW\_ALL\_COUPONS macro is set to "Y" in the standard product, but if you set it to "N" and perform the appropriate reinstalls, the "Process All New Coupons" option will be removed as an option from the menu.

If you do decide to disable this option, set the JENZABAR\_ALLOW\_ALL\_COUPONS macro to "N" in the \$CARSPATH/macros/custom/student file, then perform reinstalls on the following files:

```
$CARSPATH/menusrc/student/regist/jenzabar/menudesc
```

```
$CARSPATH/menusrc/student/regist/menudesc
```

This should effectively remove this option in a local customization, and would prevent the accidental or premature sending of Registration Code Coupons to faculty and students.

### Running the Create/Upload Data File Process (jenzupload)

After all the necessary setup has been accomplished, you should be ready to run the batch data upload process.

To create a successful batch upload data file and to send this file to Jenzabar.com, perform the following steps:

1. Verify that the JENZABAR\_MANUAL\_UPDATE\_DATA macro is defined according to your institutional requirements.

**\*\*\*IMPORTANT\*\*\***

The default setting for the JENZABAR\_MANUAL\_UPDATE\_DATA macro in macros/custom/student is "N", which means that when the Create/Upload Data File process is run from the menu, data files containing updates to the API data will NOT be automatically uploaded to Jenzabar.com. This is to prevent an institution from attempting to send a data file to Jenzabar.com prematurely.

When the Create/Upload Data File process creates data files, it stores them in the directory \$TXTPATH/jenzabar. The files are named with a sequential number with a .dat extension. If the JENZABAR\_MANUAL\_UPDATE\_DATA is "N", then these files are not uploaded to Jenzabar.com automatically. This gives an institution the chance to test the process before actually uploading data to Jenzabar.com.

We recommend that you carefully consider when it is appropriate to set this macro to Y, thus enabling the transfer of data files to Jenzabar.com via the Create/Upload Data File option.

If you are prepared to enable these features at this time, then you will need to set the JENZABAR\_MANUAL\_UPDATE\_DATA macro to "Y" and perform the commands that follow. Otherwise, we suggest that you leave the macro set to "N" until you are familiar with the processes and procedures involved in the CX-Jenzabar API.

2. To enable the file upload feature now, perform the following steps:

```
% cd $CARSPATH/macros/custom
% make co F=student
% vi student
```

3. Make the change necessary for your institutional requirements.

```
% make cii F=student L="Modify JENZABAR_MANUAL_UPDATE_DATA macro."
```

4. Install the menu option to see the change to the macro:

```
% cd menuopt/regist/scripts
% make reinstall F=jenzupload
```

5. Run the Create/Upload Data File.

You should find the Create/Upload Data File process in a menu path similar to the following:

```
[b] Student Management
[a] Registrar
```

[o] Jenzabar API

[a] Create/Upload Data File

If the JENZABAR\_MANUAL\_UPDATE\_DATA macro is set to "Y", then any .dat file created will automatically be uploaded to Jenzabar.com. If, however, the macro is set to "N", then the .dat file will be created in \$TXTPATH/jenzabar, but it will not be automatically uploaded to Jenzabar.com.

**Note:** Regardless of the value of the macro, the Upload Single Data File menu option will allow you to upload any .dat file that you specify to Jenzabar.com.

When the Create/Upload Data File option is run, it will output a status message indicating the sequential number for the .dat file being created. The .dat file will be stored in \$TXTPATH/jenzabar in the form of {serial\_no}.dat. If any errors occur during the process, an error message email will be sent to the Jenzabar API Administrator's email account as specified in the Admin E-mail Address field in the Jenzabar Initialization Record (jenzini\_rec), and to the user running the process, if different from the administrator.

#### 6. Check e-mail for any status and error messages.

If any errors occur during the process, these error messages are stored in a .log file in \$TXTPATH/jenzabar and are also sent to the Jenzabar API Administrator and the user running the process.

Repeat the above steps occasionally to make sure that the changes that occur on your system will be uploaded to Jenzabar.com.

Once a data file has been created successfully in \$TXTPATH/jenzabar, it will be automatically sent to Jenzabar.com if the JENZABAR\_MANUAL\_UPDATE\_DATA macro is set to "Y". If this macro is set to "N", however, or if the need arises to re-send a particular batch data file, then you may run the Upload Single Data File option which allows you to specify the name of the data file to send, in the form of {serial\_no}.dat.

Note that when the Create/Upload Data File process (jenzupload) is running, it creates a file called "jenzupload.ip" in \$CARSPATH/events. The presence of this file indicates that the process of creating and/or uploading a file to Jenzabar.com is in progress, and another instance of the process cannot start until the current one has finished. The Reset Session and Year option is also prevented from running based on the presence of the jenzupload.ip file. When the Create/Upload Data File process finishes, the jenzupload.ip file is removed.

### Resetting the Session and Year Parameters

Jenzabar recommends that you run the "Reset Session and Year" option from time to time (once each year or so) to reset the population of data being selected by the API process, and to free up space in the database.

Since it is not necessary to retain Jenzabar-CX API data from past CX sessions, resetting the Session and Year values has the benefit of removing unneeded data from the Jenzabar-CX API records on CX and freeing up space in the database. The absence of previous session data will also enhance the performance of the data comparison process.

The "Reset Session and Year" process, accessed from the Jenzabar Maintenance submenu, will update the Session and Year values in the jenzini\_rec according to the parameters specified on the "Reset Session and Year" option.

This process will also maintain the API data on CX by performing a series of purges to delete unneeded data from the API records on the CX database. No live data is deleted from CX during this purging process, and no data is deleted from Jenzabar.com's database. This maintenance action is necessary, however, to prevent multiple deletion actions from being sent to Jenzabar.com to delete past session data from the website's database when the session and year values are changed. This would be undesirable in cases where such data should be

retained on Jenzabar.com for historical purposes. Removing this data from temporary Jenzabar API records on the CX database removes such data from comparisons, and eliminates unneeded data; an action that frees up space in the database.

The "jenzupload" script in modules/regist/scripts performs the task of resetting the Session and Year values and performing the necessary data purges. Two parameters, -s for Session and -y for Year, indicate the new session and year values.

While the "Reset Session and Year" process is running, it creates a file called jenzpurge.ip in \$CARSPATH/events to indicate that a purge may be in process as a result of the resetting of the session and year. The Create/Upload Data File process (jenzupload) should not be allowed to run when the Session and Year parameters are being reset. The presence of the jenzpurge.ip file prevents the jenzupload script from running. This file also prevents another "Reset Session and Year" process from running before the first one completes. The jenzpurge.ip file is removed when the process is finished.

### **Engaging and Disengaging the API Process**

The Jenzabar API Administrator and his/her associates have the ability to engage or disengage the API process in order to continue the process after an error has occurred, or to halt the process in order to investigate errors or various data issues. The two options that control this process are Engage API Process and Disengage API Process. These options are located on the Jenzabar Maintenance submenu.

The script "jenzproc," located in \$CARSPATH/modules/regist/scripts, controls whether the API processes is engaged or disengaged. Two parameters are used:

The "-o" parameter will engage (i.e., allow) the jenzupload script to create and/or transmit data files to Jenzabar.com.

The "-x" parameter will disengage (i.e. disallow) the jenzupload script from creating and/or transmitting data files to Jenzabar.com.

Since the Create/Upload Data File (jenzupload) process is automatically disengaged if an error occurs, you may need to run the Engage API Process option to re-engage the Create/Upload Data File process after resolving errors. This process effectively removes the jenzpost.ip file from \$CARSPATH/events if it exists. It is the existence of the jenzpost.ip file which indicates that the jenzupload process should be disengaged.

You may choose to run the Disengage API Process option to disengage the Create/Upload process so you can research and resolve data or posting errors. This process effectively disengages the jenzupload process by creating the file jenzpost.ip in the \$CARSPATH/events directory.

# Using *cron* to Execute Jenzabar Processes

## Introduction

You can set up both the Jenzabar-CX API Update process (*jenzupload*) and Registration Code Coupon processing (*jenzcoupon*) to execute automatically as batch processes in *cron* by:

- Determining the environment variables for *cron*
- Logging in as root
- Accessing the crontabs directory and establishing the carsu crontab file
- Defining permissions for the carsu crontab file

You execute the processes with *cron* by:

- Accessing the crontabs directory as user carsu
- Preparing the required *cron* commands
- Starting the carsu crontab file

This section contains information about completing these steps.

## Determining Environment Variables for *cron*

The *cron* processes that run *jenzupload* and *jenzcoupon* require special variables. These variables can be determined by creating and running a simple script in your home directory or in any other directory of your choice. To do this, perform the following steps:

1. At the UNIX prompt, enter:

```
% cd ~  
% vi environs.scp
```

2. Read or type in the following commands into this file:

**Note:** Do not edit or run the environs.scp file as root or carsu; edit and run the script under your own login. The first line of the file indicated below executes the UNIX c-shell and should not be confused with a root command. Type in this script exactly as follows, including the use of quotation marks (") with the echo statement.

```
#!/bin/csh -f  
echo " "  
echo CARSDB=$CARSDB  
echo CARSV=$CARSV  
echo CARSPATH=$CARSPATH  
echo INFORMIXDIR=$INFORMIXDIR  
echo ONCONFIG=$ONCONFIG  
echo INFORMIXSERVER=$INFORMIXSERVER  
echo TXTPATH=$TXTPATH  
echo " "
```

```
<Esc> :wq
```

```
% chmod +x environs.scp  
% ./environs.scp
```

The output from this script will resemble the following example:

```
CARSDB=cars  
CARSV=carsi  
CARSPATH=/usr/carsi  
INFORMIXDIR=/opt/informix
```

```
ONCONFIG=onconf.hpdev
INFORMIXSERVER=carshpdev
TXTPATH=/usr/carsi/text
```

Write down or print out these values and keep them available for reference, as you must use them in subsequent setup steps.

### Setting Up the Jenzabar-CX API Update and Registration Code Coupon *cron* Processes

To complete the setup for *cron* processing of Jenzabar-CX API Update (*jenzupload*) and Registration Code Coupon (*jenzcoupon*):

1. Login as root by entering the following commands at the UNIX prompt:

```
% SU csh
Password for user: <password>
```

2. Verify the user "carsu" is in the group "carsprog".

Since group permissions for the \$TXTPATH/jenzabar directory are for "carsprog", the user "carsu" must also be in this group. To check the groups for this user, perform the following:

```
# groups carsu
```

If the user "carsu" is already in the group "carsprog", proceed to the next step. Otherwise, perform the following:

```
# addtogrp carsprog carsu
```

3. As root, change directory to \$CARSPATH/system/crontabs

```
# cd $CARSPATH/system/crontabs
```

4. Set up the "carsu" crontab file.

If the "carsu" crontab file already exists in the current directory (i.e., system/crontabs), then the cat command will append the new *cron* entries to the already existing file. If the crontab file does not already exist, then the cat command will create a new one.

5. Cat the "carsu" crontab file from the Procedures directory of the API SMO 12615.

**Note:** You may need to substitute "12615-inst" with the actual name of the SMO as it currently exists on your system (i.e., 12615 or 12615-ip, etc.). Also, if this SMO no longer exists on your system, you will need to read in or type the *cron* text into the carsu crontab file. This text is included at the end of this section.

```
# cd $CARSPATH/smo/12615-inst/Procedures
# cat carsu >>$CARSPATH/system/crontabs/carsu
```

6. Set up the appropriate environment in the carsu crontab file.

Edit the carsu crontab file and be sure to replace all of the "XXXXXXXX" values with the values of the indicated environment variables determined in the previous section.

```
# cd $CARSPATH/system/crontabs
# vi carsu
```

**Note:** The crontab commands are initially commented out so that the commands do not begin executing without you starting them arbitrarily. If you are ready at this time to execute these *cron* commands, be sure to uncomment them.

7. Set up the appropriate permissions for the carsu crontab file.

```
# chmod 660 carsu
# chown carsu carsu
# chgrp carsprog carsu
```

8. Exit root.

```
# exit
```

### Setting Up *cron* Manually

Use this step ONLY if you were not successful setting up the crontab file in the preceding steps. If SMO 12615 no longer exists on your system, then you will need to read or type the following text into the crontab file manually. Once this task has been completed, return to step 6 above.

**Note:** Enter the *cron* instructions as indicated, placing them at the end of the "carsu" file. Note that the actual crontab commands must be entered on one continuously wrapping line. The crontab commands as shown below are not continuously wrapping.

```
# cd $CARSPATH/system/crontabs
# vi carsu
```

Enter the following text into the carsu file:

```

#=====
#      Cron Table File (crontab) for user "carsu"
#=====
#
#  Jenzupload process
#
#  Run jenzupload script as often as specified to compare changed data on
#  the CX database to data that has already been uploaded to Jenzabar.com.
#  The -u parameter must be passed to the jenzupload script if the data
#  files that are created are to be automatically sent to Jenzabar.com.
#
#  If you do not pass the -u parameter to the jenzupload script, then
#  the data files that are created are stored in $TXTPATH/jenzabar, but
#  they are not automatically uploaded to Jenzabar.com. The jenzupload
#  script may also be run manually from the CX menu. Refer to
#  the documentation for the Jenzabar-CX API process for more details.
#
#0,30 * * * * HOME=/; USER=carsu; LOGNAME=carsu; CARSDb=XXXX; CARSV=XXXXXXXXXX;
CARSPATH=XXXXXXXXXXXXXXXXXX; INFORMIXDIR=XXXXXXXXXXXXXXXXXX; ONCONFIG=XXXXXXXXXXXXXXXXXX; INFORM
IXSERVER=XXXXXXXXXX; TXTPATH=XXXXXXXXXXXXXXXXXXXXXXXXXX export HOME USER LOGNAME CARSDb
CARSV CARSPATH INFORMIXDIR ONCONFIG INFORMIXSERVER TXTPATH; /bin/csh -c "perl
/opt/carsi/install/scp/regist/jenzupload.scp > /tmp/jenzupd.log"
#
#  Jenzcoupon process
#
#  Run the jenzcoupon script as often as specified to send Registration Code
#  Coupons to faculty and students who have profile_rec.reg_code_sent = "N."
#  The process will then update the reg_code_sent field to "Y" for those
#  persons who were sent coupons. The -s Y parameter indicates that the
#  coupons will be sent; if -s N is passed, only a list of those persons who
#  would receive Registration Code Coupons will be sent to the
#  Jenzabar Administrator, but the coupons would not be sent to students
#  or faculty.
#
#15,45 * * * * HOME=/; USER=carsu; LOGNAME=carsu; CARSDb=XXXX; CARSV=XXXXXXXXXX; C
ARSPATH=XXXXXXXXXXXXXXXXXX; INFORMIXDIR=XXXXXXXXXXXXXXXXXX; ONCONFIG=XXXXXXXXXXXXXXXXXX; INFORM
IXSERVER=XXXXXXXXXX; TXTPATH=XXXXXXXXXXXXXXXXXXXXXXXXXX export HOME USER LOGNAME CARSDb
CARSV CARSPATH INFORMIXDIR ONCONFIG INFORMIXSERVER TXTPATH; /bin/csh -c "perl /o
pt/carsi/install/scp/regist/jenzcoupon.scp -s N > /tmp/jenzcoupon.log"
#
#  Removal of old data files
#
#  The command below can remove old data files from
#  TXTPATH/jenzabar. This will
#  help save disk space since some of the data files can become
#  quite large.
#  The default command will remove files that are 60 days and older. You
#  may choose to increase or decrease this value depending on your system
#  requirements.
#
# find $TXTPATH/jenzabar -mtime +60 -exec rm {} \;

```

## Executing the *cron* Processes for Jenzabar-CX API Data Processing

Once the "carsu" crontab file has been set up according to your institutional requirements and you want to run the *jenzupload* and/or *jenzcoupon* scripts as batch processes, you can run the *cron* jobs by performing these steps:

**CAUTION:** You MUST run the "carsu" crontab file as the user "carsu" and not as any other user, including root.

1. Access the system/crontabs directory and login as user "carsu" from your own account. Do not log in as root.

```

% cd $CARSPATH/system/crontabs
% SU su carsu
Password for user: <password>

```

2. Prepare the *cron* commands to run.

```
# vi carsu
```

Find the *cron* commands for jenzupload and/or jenzcoupon and make sure the lines for these commands are not preceded by the "#" comment character. Delete this comment character on each command line if it exists.

3. Save the changes:

```
:wq
```

4. Start the "carsu" crontab file.

```
% crontab carsu
```

5. Check for errors when the crontab file starts running. If you have errors, correct them, then repeat the above command until all errors are corrected and the crontab file is running successfully. To verify that the *cron* process is running, execute the following command:

```
% crontab -l
```

6. If you should need to stop the *cron* from running, execute this command:

```
% crontab -r
```

7. Exit carsu.

```
% exit
```

8. Be sure to periodically check the files /tmp/jenzupd.log and /tmp/jenzcoupon.log for errors.

```
% view /tmp/jenzupd.log
```

```
% view /tmp/jenzcoupon.log
```

# Setting Up Authentication Files

## Introduction

This section contains information about setting up files for Web authentication.

## Setting Up the DBM File for Web Authentication

Authentication files must be set up for faculty and student access to your Web server. The following `setup_web_dbm` command generates the DBM file used by Apache for Web server authentication. To activate a faculty member or student for Web server access, they must have a `pin#` stored in the `profile_rec.password` column. Faculty, but not students, must also have an entry in the `userid_table`. Faculty members authenticate with their username as it appears in the `userid_table.user_name` column. Students usually authenticate with their ID number as their username.

To set up the DBM file, use the following commands. Make sure you run them for each server (TEST, LIVE, etc.).

```
% SU csh
Password for user:
# setup_web_dbm faculty LIVE
# exit
```

```
% SU csh
Password for user:
# setup_web_dbm faculty TEST
# exit
```

## Using Troubleshooting Tools

### Processing Help Messages

Help and parameter messages are available for most of the jenz\* Perl scripts which support the API process. If you wish to see the help message for any of the scripts, run the script and pass the `-h` parameter. For example, to obtain a help message for the `jenzupload` script, run this command:

```
perl $SCPPATH/regist/jenzupload.scp -h
```

The other scripts with help messages are: `jenzcoupon`, `jenzpost`, and `jenzproc`. The sources for these scripts are located in `$CARSPATH/modules/regist/scripts`.

These help and parameter messages can help you determine how to locally customize the jenz\* scripts to meet your needs, although it is anticipated that the default setup for running these processes from CRON and from the CX menu will be adequate for most needs.

### Running in Debug Mode

You can set the `jenzupload` script to operate in debug mode. To do this, the API Administrator, or any user running the process, must define an environment variable called `API_DEBUG`. To do this, execute the follow command at the shell prompt:

```
% setenv API_DEBUG 1
```

When the `jenzupload` script runs in debug mode, it provides more detailed information about the process and stores this information in files in `/tmp` in the format of `/tmp/jenzupload.$$debug`, where `$$` represents the system process number assigned to the current execution of the script. For example, a new debug file might be created as follows:

```
/tmp/jenzupload.16386.debug
```

The debug file should contain information regarding the execution times of each type of data processed by the script. This information may be useful when you are trying to determine which part of the process might be having some difficulty. Usually, the debug mode will not be necessary, but if you begin having some problems with the process, you may want to activate this mode and examine the debug files for possible problems.



## SECTION 3 – POST-IMPLEMENTATION CONSIDERATIONS

### Overview

#### Introduction

After the data is initially uploaded at the Jenzabar site, subsequent updates to the data at the CX site must be reflected in the Jenzabar database through periodic updates. The data from the CX database must be tracked and processed to correspond to the Jenzabar format. The process requires ASCII dumps to be sent to the Jenzabar server, which then decodes the data and updates its database accordingly. The process will also create a log file.

If an error occurs during the upload, the log file is sent to the administrator at Jenzabar.

#### In This Section

This section explains the steps involved in the subsequent upload of data to the Jenzabar database. It also addresses cascade delete and modification of key fields, and alternate strategies for subsequent uploads of data.

# Subsequent Uploads of Data

## Introduction

After the initial upload of data to the Jenzabar site, the type of changes that can occur to the CX database are:

- A new record gets added.
- An existing record is deleted.
- Data in the Jenzabar non-key fields is modified.
- Data in the Jenzabar key fields is modified.

While adding/deleting records, users must take care to maintain the referential integrity of the data. The order of the addition of records must be from top to bottom (i.e., the master records must be added before the lower level records). When deleting records, if you delete a master record, you must also delete all of its associated lower level (child) records.

An algorithm to prepare a record for subsequent upload would include the following steps:

- Retrieve the modified data from the CX database.
- Append key values to identify the record uniquely in the Jenzabar tables.
- Append the proper Jenzabar action code (A/U/D) in the header.
- Format the data and prepare an ASCII dump.

The Jenzabar CRP table maps to the following tables in the CX database:

- cw\_rec
- secmtg\_rec
- instr\_rec
- crs\_rec
- prog\_enr\_rec

Of these, the instr\_rec and prog\_enr\_rec are used to obtain the key and do not change. Because these tables are used for lookup purposes only, no audit table is maintained for these tables (from a CRP perspective).

# Jenzabar Tables in the CX Database

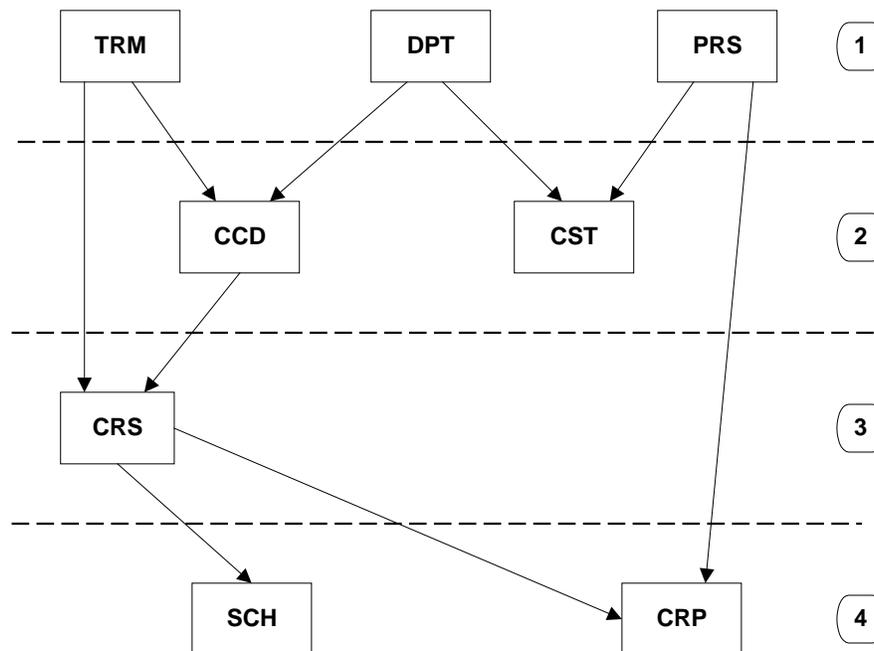
## Jenzabar Tables

Eight tables beginning with *jenz* and ending with one of the following 3-character strings (i.e., *jenztrm*) are used with the API.

- TRM term data
- DPT department data
- PRS person data
- CST college/status data
- CCD course code (course catalog) data
- CRS course (offering) data
- SCH course schedule data
- CRP course/person (registration) data

## Transaction Types

There are four levels of dependence when processing data transactions. The TRM, DPT and PRS records are at the top of the hierarchy. No transaction types on any other level can proceed unless one of these three transactions has taken place. In the following example, DPT has two directly dependent transactions, CCD and CST. Indirectly, CRS, SCH, and CRP are also related to DPT.



## Table Relationships

The following table shows the Transaction Table and the associated CX tables from which data is drawn to compare whether data was added, deleted, or updated.

<b>Transaction Table</b>	<b>Referenced Tables</b>
jenzccd_rec	crs_rec, sec_rec, sess_table, crsabstr_rec
jenzcrp_rec	crs_rec, sec_rec, sess_table, secmtg_rec, instr_rec, mtg_rec
jenzcrs_rec	crs_rec, sec_rec, sess_table
jenzcst_rec	sec_rec, sess_table, prog_enr_rec, profile_rec, stu_acad_rec, fac_rec, instr_rec
jenzdpt_rec	dept_table
jenzprs_rec	sec_rec, sess_table, prog_enr_rec, profile_rec, fac_rec
jenzsch_rec	crs_rec, sec_rec, sess_table, secmtg_rec, mtg_rec, bldg_table, im_table
jenztrm_rec	acad_cal_rec, sess_table

## Advantages to Maintaining a Copy of the Jenzabar Tables

A replica, or copy, of Jenzabar tables is kept in the CX database. Advantages to keeping a copy of the Jenzabar tables include:

- With comparison to other approaches, the system does not need to maintain triggers or an audit trail.
- Complexity involved in interpreting the audit trail not required.
- Ease in maintenance from administrator and developer point of view. Need to maintain only one SQL statement for each of eight entities.

## Maintaining a Copy of Jenzabar Tables

Follow these steps to keep a copy of Jenzabar tables:

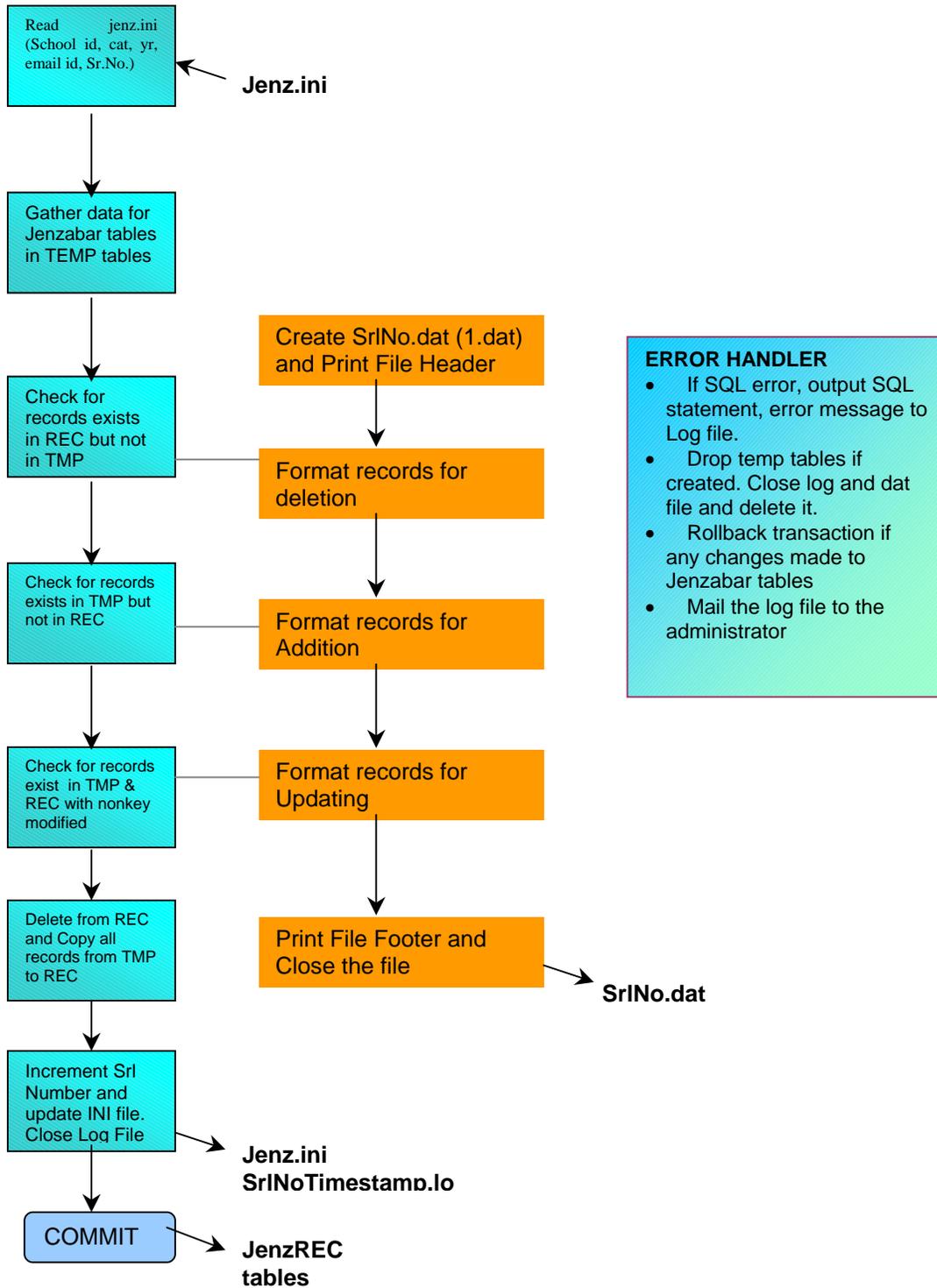
1. Maintain a file to store:
  - School id
  - Catalog for which data is to be selected
  - Year for which data is to be selected
  - Serial number (to ensure that Jenzabar doesn't miss any ASCII dump in between)
  - School Administrator e-mail-id
2. Set up a *cron* job to run on the CX server that will:
  - Initially, gather data related to the 8 Jenzabar entities and store it in Jenzabar tables on the CX side.
  - For subsequent uploads, gather data for a specific catalog and year for which the initial upload was done and store it in temporary tables.
3. For each of the eight entities:
  - Check for records that are in Jenzabar tables but not in temporary tables based on key information. (Use **bottom-up** approach).
  - If found, prepare an ASCII dump for that row with status marked for DELETION (from Jenzabar database).
  - Check for records that are in Temporary tables but not in Jenzabar tables based on key information. (Use **top-down** approach).
  - If found, prepare an ASCII dump for that row with status marked for ADDITION (into jenzabar database).
  - Check for records that are in both Temporary tables and Jenzabar tables based on key information but non-key fields not matching. (Use **top-down** approach).
  - If found, prepare an ASCII dump for that row with status marked for UPDATING (into jenzabar database).
4. Delete data from Jenzabar tables.
  - Insert data from temporary tables into Jenzabar tables.
  - Drop temporary tables.
  - Store the ASCII dump as SrlNo.dat file only if entire process is successful.
  - Log summary information in srlno.log file. In case of failure, mail it to Jenzabar Administrator with the log as an attachment and rollback the entire transaction. If the process concludes successfully, commit the changes.
5. Upon receipt of the message on the Jenzabar side, the process will check for "expected" serial number with the "received" serial number. If it doesn't match, it will send an error message back to the Jenzabar Administrator. If it matches, it will interpret the data and upload it into the Jenzabar database. Upon completion, it will send an e-mail to the Jenzabar Administrator with the Jenzabar log file.



## APPENDIX A: PROGRAM FLOW

The following illustrates the interface program flow.

**Note:** REC indicates Jenzabar tables stored in the CX database. TMP indicates temporary tables created during the process.





## APPENDIX B: MAPPING OF DATA ELEMENTS

### Purpose

The purpose of the following table is to show the relationships between Jenzabar and the CX data elements.

Jenzabar data maps to...					CX data		
Data Row	Field(s)	Length	Req'd?	Description	Table	Column	Length
File header line	School ID	A6	Y	Unique Jenzabar-assigned identifier for all schools	jenzini_rec	school_id	integer
	Transfer time	date	Y	Time transfer was executed	N/A	N/A	N/A
	Format	A10	Y	Name of the data format	N/A	N/A	N/A
	Mode	A10	Y	TEST or LIVE	jenzini_rec	tx_mode	N/A
File footer line	EOF tab	A3	Y	Unique identifier of end of file	N/A	N/A	N/A
	Records	number	N	Number of records transmitted	N/A	N/A	N/A
Data line header information	Trx type	A3	Y	Transaction type code; the first element on each data line	N/A	N/A	N/A
Term table information	TRM	A3	Y	Term table information	N/A	N/A	N/A
	Jenzabar action	A1	N	Add, update, and delete. To delete, you must delete all children records first. Jenzabar does not cascade delete.	N/A	N/A	N/A
	LU time	date	N	Time data was last changed in the host	N/A	N/A	all fields
	Term code	A30	Y	Text code (e.g., Fa99). Not	acad_cal_rec	sess	4 char

Data Row	Field(s)	Length	Req'd?	Description	Table	Column	Length
				updateable. Primary key.			
					acad_cal_rec	yr	smallint
					acad_cal_rec	prog	4 char
	Desc	A30	Y	Text description	sess_table	txt	24 char
					acad_cal_rec	yr	smallint
					prog_table	prog	4 char
	Start date	date	Y	Date term starts	acad_cal_rec	beg_date	date
	End date	date	Y	Date term ends	acad_cal_rec	end_date	date
Department Table Information	DPT	A3	Y	Department table information	N/A	N/A	N/A
	Jenzabar action	A1	N	Add, update, and delete. To delete, you must delete all children records first. Jenzabar does not cascade delete.	N/A	N/A	N/A
	LU time	Date	N	Time data was last changed in the host	N/A	N/A	all fields
	Department code	A6	Y	Internal department identifier code. Hot updateable. Primary key.	dept_table	dept	4 char
	Desc	A60	Y	Department description	dept_table	txt	24 char
Person identity information	PRS	A3	Y	Identifies a single individual			
	Jenzabar action	A1	N	Add, update. Only host password is updateable.	N/A	N/A	N/A
	LU time	Date	N	Time data was last changed in the host	jenzprs_rec	all fields	all fields
	Host ID	A30	R	Unique ID in Host Primary key	N/A	Jenzabar ID	integer

Data Row	Field(s)	Length	Req'd?	Description	Table	Column	Length
					id_rec	id	11 digits
					N/A	reg_code	integer
This will not be passed from CX	Host password	A30	N	Host password: needed if hot swapping. Updateable.	N/A	N/A	N/A
N/A	Registration code	A30	Y	Registration code	N/A	Jenzabar ID	integer
					id_rec	id	11 digits
College Status data	CST	A3	Y	College status data	N/A	N/A	N/A
	Jenzabar action	A1	N	Add (delete) Composite key = (host ID + Dept_code + Status_code) Fields that make up the key are not updateable.	N/A	N/A	N/A
	LU time	Date	N	Time data was last changed in the host	N/A	N/A	all fields
	Host ID	A30	Y	Unique ID in host	N/A	Jenzabar code	integer
					id_rec	id	11 digits
	Dept code	A6	N	Department code	fac_rec	dept	4 char
	Status code	A6	Y	Role in dept/school (faculty-FAC, student-STU, Jenzabar admin-ADM)	fac_rec and prog_enr_rec	no specific column; computed at the time of processing	Usually 3 char for FAC, STU, or ADM
	Degree type	A6	N	Degree (e.g., MS or BA)			
	Graduation date	date	N	Date of graduation			
Course Code	CCD	A3	Y	Course code information	N/A	N/A	N/A
	Jenzabar action	A1	N	Add, update and delete. To delete you must delete all children records first.	N/A	N/A	N/A

Data Row	Field(s)	Length	Req'd?	Description	Table	Column	Length
				Jenzabar does not cascade delete.			
	LU time	Date	N	Time data was last changed in the host	N/A	N/A	all fields
	Course code	A20	Y	Text code (e.g., ENG350). Not updateable. Primary key.	crs_rec	crs_no	12 char
	Title	A150	Y	Course title	crs_rec	cat	4 char
crs_rec					title1	32 char	
					title 2	32 char	
					crs_rec	title 3	32 char
	Description	A1000	Y	Catalog description of the course	crsabstr_rec	abstr	blob – no limit
	Dept code	A6	Y	Department of the course	crs_rec	dept	4 char
Course	CRS	A3	Y	Session/year of course offering	N/A	N/A	N/A
	Jenzabar action	A1	N	Add, update, and delete. To delete you must delete all children records first. Jenzabar does not cascade delete. Composite key = (Term code + Course code + section) Only title is updateable.	N/A	N/A	N/A
	LU time	Date	N	Time data was last changed in the host	N/A	N/A	all fields
	Term code	A30	Y	Text code (e.g., FA99)	sec_rec	sess	4 char
					sec_rec	yr	4 char
					crs_rec	prog	4 char
	Course code	A20	Y	Text code (e.g.,	sec_rec	crs_no	12 char

Data Row	Field(s)	Length	Req'd?	Description	Table	Column	Length
				ENG350)			
	Section	A10	N	Offering identifier	sec_rec	cat	4 char
	Title	A150	N	Title of the course. Updateable.	sec_rec	sec_no	2 char
					sec_rec	title	32 char
Course schedule	SCH	A3	Y	The meeting times of the course	N/A	N/A	N/A
	Jenzabar action	A1	N	Add, delete. All fields are used to make up the key except for SCH, Action and LU time. No updates allowed. To update, you must first delete the old record and add a new record.	N/A	N/A	N/A
	LU time	Date	N	Time data was last changed in the host	N/A	N/A	all fields
	Term code	A30	Y	Term code (e.g., FA99)	sec_rec	sess	4 char
					sec_rec	yr	4 char
					crs_rec	prog	4 char
	Course code	A20	Y	Text code (e.g., ENG350)	sec_rec	crs_no	12 char
					sec_rec	cat	4 char
	Section	A10	N	Offering identifier	sec_rec	sec_no	2 char
	Start date	Date	N	Date schedule starts	mtg_rec	beg_date	10 char
	End date	Date	N	Date schedule ends	mtg_rec	end_date	10 char
	Days	A7	Y	Days class meets (combo of 1,2,3, etc. ???)	mtg_rec	days	7 char
	Start time	Time	Y	Time class starts	mtg_rec	beg_tm	5 char
	End time	Time	Y	Time class	mtg_rec	end_tm	5 char

Data Row	Field(s)	Length	Req'd?	Description	Table	Column	Length
				ends			
	Location	A60	N	Classroom location	mtg_rec	campus	4 char
					bldg_table	txt	24 char
					mtg_rec	room	4 char
					im_table	descr	24 char
Course person data	CRP	A3	Y	Specifies the person's role in the course	N/A	N/A	N/A
	Jenzabar action	A1	N	Add, delete. All fields make up the key except for CRP, Action and LU. Not updateable.	N/A	N/A	N/A
	LU time	Date	N	Time data was last changed in host	N/A	N/A	all fields
	Term code	A20	Y	Text code (e.g., FA99)	cw_rec/ secmtg_rec	sess	4 char
					cw_rec/ secmtg_rec	yr	4 char
					crs_rec	prog	4 char
	Course code	A20	Y	Text code (e.g., ENG350)	cw_rec/ secmtg_rec	crs_no	12 char
	Section	A10	N	Offering identifier	cw_rec/ secmtg_rec	sec	2 char
	Host ID	A30	Y	Unique ID in host	N/A	Jenzabar code	integer
					cw_rec/ instr_rec	id	11 digits
	Status	A6	Y	Role in course	Insert during processing	Insert during processing	Usually 3 char (STU or 1PR)

## \$

\$TXTPATH/jenzabar, 25  
\$TXTPATH/jenzabar directory, 11

.

.dat file, 25

## A

account, for administrator, 8

administrator

    setup of account, 8

algorithm for upload, 36

alternate addresses, 18

API Process

    engage/disengage, 26

ASCII

    data format, 1

    dumps, 35, 39

## C

changes to database, 36

considerations

    for after implementation, 35

    for setup, 7

copying

    Jenzabar tables, 39

create/upload data file, 24

Create/Upload Data File, 24

creating

    Jenzini record, 14

*cron*

    executing Jenzabar processes, 27

Cron file, 29

CRP table, 36

## D

data elements, 43

data transfer, 2

database changes, 36

debug mode, 33

department codes, 18

diagrams

    data flow, 4

directories

    \$TXTPATH/jenzabar, 11

disengage API process option, 13

Disengage API Process option, 26

## E

e-mail addresses, 18

engage API process option, 13

## INDEX

Engage API Process option, 26

environment variables

    for *cron*, 27

environs.scp file, 27

error messages, 19

errors

    during uploads, 35

## F

fields

    mapping, 43

files

    .dat, 25

    Cron, 29

    environs.scp, 27

    in-process, 20

    jenzpurge.ip, 26

    jenzupload.ip, 25

    posting, 19

    uploading, 5, 12

    Web authentication, 32

## H

help messages, 33

host links, 2

## I

implementation

    considerations, 7

    process overview, 2

Informix, 7

in-process files, 20

institution ID, 11

intended audience, 1

interface

    purpose, 1

issues for after implementation, 35

issues for setup, 7

## J

Jenzabar API menu, 12

Jenzabar records, 20

Jenzabar tables, 37

JENZABAR\_MANUAL\_UPDATE\_DATA macro,  
    22, 24

*jenzcoupon*, 27, 28, 30

jenzcoupon script, 23

    running, 30

Jenzini File Upload menu option, 12

Jenzini Record menu option, 12, 14

jenzini\_rec

    creating, 14

- signup sample, 17
- jenzproc script, 26
- jenzpurge.ip file, 26
- jenzupload, 27, 28, 30
  - running, 24
- jenzupload script
  - resetting session and year, 26
  - running, 30
- jenzupload.ip file, 25

## **K**

- key field, 36

## **L**

- log files, 35

## **M**

- macros
  - JENZABAR\_MANUAL\_UPDATE\_DATA, 22, 24
- maintenance submenu, 13
- mapping
  - Jenzabar and CX data, 43
- Mapping
  - Data Elements, 43
- menu options
  - disengage API process, 13
  - engage API process, 13
  - Jenzini File Upload, 12
  - Jenzini Record, 12
  - Process All New Coupons, 12
  - Process Single Coupon, 13
  - Purge All Jenzabar Data, 14
  - reset session and year, 13
- menus
  - Jenzabar API, 12
- messages
  - errors, 19

## **P**

- Perl, 7
- post-implementation, 35
- posting files, 19
- Process All New Coupons
  - enable/disable, 23
  - menu option, 12
- Process Single Coupon
  - menu option, 13
- program flow, 41

- Purge All Jenzabar Data menu option, 14

## **R**

- records
  - Jenzabar, 20
- referential integrity, 36
- Registration Code Coupons
  - functionality, 23
  - signup variables, 23
- relationships of tables, 38
- reset session and year option, 25
- reset the session/year option, 13

## **S**

- school ID, 11
- scripts
  - jenzproc, 26
- sequence number, 21
- session resetting, 13, 25
- setup, 7
- system requirements
  - for interface, 7

## **T**

- tables
  - copying, 39
  - Jenzabar, 37
  - relationships, 38
- transaction types, 37
- troubleshooting, 33

## **U**

- upload process
  - purpose, 1
- uploading files, 5, 12
- uploads, 21, 35
  - algorithms, 36
- URL definition, 8
  - with single login, 10
  - without single login, 8

## **W**

- Web authentication files, 32

## **Y**

- year resetting, 13, 25