

The functionality required

Technology, Information Technology



Curricular Information System Software Requirements Document (Draft 1 – August, 2000) Roberta Crumrine

1. Introduction

1. 1 Purpose This document details the functionality required for the new Curricular Information System (CIS).

1. 2 Document Conventions Although this document is intended as a set of Requirements, not a design, some technical information has been included with the requirements description. Priority for all functionality is ‘One’ except where noted.

1. 3 Intended Audience and Reading Suggestions The primary audience is Academic Services, especially the Office of Curriculum and Faculty Support, and the Student Services Information Technology staff. Ultimately, all members of the CIS Project Team (Ref 3) are the intended audience.

1. 4 Product Scope The Curricular Information System is intended initially to replace existing Clipper and Web proposal systems with a new, integrated, web-based system having the features detailed in this document. It provides a foundation that facilitates orderly growth of future enhancements. Certain enhancements have been included in this document as Priority 1 features.

Other enhancements are listed as Priority 2 – not essential for the initial release.

1. 5 Reference (Note: documents available on the web are so indicated. Others are available upon request.) Source Documents for the CIS Requirements Specification

- 1) Curricular Information System Vision and Scope <http://web.mit.edu/ssit/cis/CISVisionScope.html>
- 2) HTML/Print Generator and Document Management Sub-Project Vision and Scope
- 3) CIS Project Participants <http://web.mit.edu/ssit/cis/CISpeople.html>
- 4) Catalog Production Schedule Timeline
- 5) CSB and Registration Timeline

) Old Catalog Data description (Clipper system, web proposal system, MITSIS) 7)

Scheduling system Class Schedule Book (CSB) File layout and description 8)

Old Catalog Functions (spread sheet) 9) CSB and Registration Program

documentation 10) Formal Department and Academic

ServicesInterviewDocuments (detailed list available) 11) Final Exam

Recommendation Draft Document Companion CIS Requirements Documents

1) CISData. doc 2) CISMigrationRefresh. doc 3) CISWorkflow. doc 4)

CIS Process Model Diagrams (These require the SilverRun BPM viewer))

Scheduling system Class Schedule Book (CSB) File layout and description 2.

Overall Description 2. 1 Product Perspective The CIS provides: A department

coordinator ' portal' for Academic Services including: ? Complete

proposal development functionality (similar to IAP proposals, but more

extensive) ? A home for broadcast and individual communications

between Academic Services and departments ? Links to related

Academic Services web applications. Centralized Academic Services

administrative control functions Centralized Programmer control functions 2.

2 Product Functions

The overall description of CIS functionality was given in the Project Vision

and Scope (Ref 1): 1. Provide facilities to enhance the exchange of

information among faculty and staff during curriculum development. Do so

by enabling distribution of official information with ancillary discussion

among authorized faculty members, staff, and faculty committees during all

phases of subject proposal development and review, including prior to

proposal submission to the COC/CGSP. 2. Preserve a record of these

decisions and their context. 3. Support versioning and workflow

management of the information that it maintains. . Replace the current

<https://assignbuster.com/the-functionality-required/>

catalog production system, in which departments submit subject listing changes both electronically and on paper and curricular changes on paper, with a fully electronic system. (However, printed listings will still be obtainable upon request.) 5. Enable updating of catalog data throughout the year. Do so for more than one term/year simultaneously. 6. Provide up-to-date information about subjects, schedules, and instructors to the MIT community (faculty, academic staff, students, alumni, and prospective students). 7. Provide easy-to-use, on-demand print and on-line publishing.

Non-subject data now printed in the MIT bulletin will be integrated via the web with subject data for integrated publishing. 8. Enable links from the basic on-line subject postings to auxiliary information such as teaching tools and instructional material, syllabi, class schedules, official subject evaluations, instructor information, supplemental material like that provided in the HASS Guide, and comments from students and alumni who have previously taken the subject. Through the web this information will be available to students at MIT partner universities and to prospective students. . Provide web search capabilities of on-line bulletins and auxiliary information. 10. Provide an infrastructure for future enhancements (as described below). 11. Eventually, offer students and advisors a graphic tool for planning the student's course work over several years, considering Institute requirements, department requirements, and information about what subjects are offered when and with what prerequisites. Students could consider ' what if' situations, e. g. the impact of changing majors. (A paper version of this tool existed in the 1980's). 12.

Eventually provide these future enhancements: student portals, advisor access to broader advisee information, software to suggest courses to students based upon their academic record and outstanding degree requirements, and direct submission of grades to the Registrar's office by departments. 13. Eventually, pending faculty approval, allow on-line registration. 14. Eventually support the development of a dynamic personal scheduling system for students, faculty, and staff (to replace the current student scheduling component), and options for users to select with whom to share their schedule.

In the long range, the new system must have an infrastructure that empowers students with a decision-making network for planning their academic careers, including: ? Ability to communicate with faculty and advisors over course selection, assignments, etc. ? On-line scheduling facilities ? These need to be permanent, not temporary as in the current on-line catalog ? Students need to be able to add/delete activities from their schedule ? Students need to be able to schedule meetings with faculty or advisors ? Personal portals with secure access to information about subjects they are enrolled in: Access to syllabi and other course materials ? Facilities to track course assignments ? Access to chat rooms, bulletin boards for subjects ? Access to tutorials for courses ? Facilities to communicate directly with faculty and advisers Technically, the CIS project employs: A configurable toolkit of functions including: ? Ability to define new fields to capture for certain types of data (extensibility) ? Ability to configure fields, their sequencing, and formatting (i. e. style tags) for downloads so that any type of publication

(print or web) can be downloaded without specialized programming.

Flexible form generation including user-configurable field layout, text descriptors. Reusable components for most functionality. Use of Java and the new SSIT Internet platform, and when appropriate, XML 2. 3 User Classes and Characteristics Academic Services personnel Responsible for the overall tracking and publishing of the MIT catalog. Support the development of new and changed subject proposals. Support COC and CGSP review of new and changed proposals. Pre-register and register students. Manage Add and Drop requests. Schedule classrooms, students, and finals. Manage and report on pre-requisites, co-requisites.

Audit student degrees (GIR) Department Coordinators Responsible for helping faculty develop MIT catalog and related information for their department. Monitor departmental roadmaps Help develop room schedules for subjects and exams Audit department degrees COC and CGSP Review subject proposals Other Administrative Offices The HASS Office, PSB, Communications Office review and support the development of the MIT catalog and supplemental bulletins. Run student lotteries. Submit grades. Faculty Plan and teach curricula Use many reports provided by Academic Services: class lists, etc. Students

Use catalog and related information to plan course work. Use the on-line planning worksheet, lottery submittal, and pre-registration functions. 2. 4 Operating Environment CIS is developed for use on the Unix system: student Under the Netscape web server: enterprise using SSL and personal certificates Accessing the Oracle database (currently on system sisjajp, but soon to be

on a new database server) New code will be developed in Java. The existing General Table Maintenance (GTM) facility may be used for certain features.

2. 5 Design and Implementation Constraints 2. 6 Assumptions and Dependencies 3. External Interface Requirements 3. User Interfaces 3. 2 Hardware Interfaces 3. 3 Software Interfaces 3. 4 Communications Interface 4. System Features See also the Process Model diagrams 4. 1 Authorization Features 4. 1. 1 Description Privacy and Security of data is assured by the authorization of system users. Proposal authorizations depend upon department, certain attributes of the proposal (e. g. HASS), and status of the proposal (in-work, in-review, approved). The authorization scheme must manage, among others, the following situations: Department Administrators (plus faculty and other Administrative offices) need to view and modify their own departments' listing.

They need to view listings for equivalent subjects and joint subjects in other departments. They need to view changes to subjects in other departments that they list as pre-requisites to their own subjects and/or as degree path requirements. THE HOC (HASS Overview Committee) needs view and comment authority for any new/changed subject flagged as HASS-D. HOC approval is required for HASS-D subjects that change – this needs to be included in the workflow scheme. (These subjects are reviewed every three years even when they don't change, but the system won't manage such periodic reviews.)

THE HOC (HASS Overview Committee) needs view and comment authority for any new/changed subject flagged as a HASS Elective. Only HE subjects

outside of the School of HASS need approval by the HOC. This needs to be included in the workflow scheme. HASS Electives inside of the School of HASS are not reviewed by HOC even when new, so they are NOT included in the workflow scheme. Department coordinators (at least the primary coordinator) should be able to add/delete authorizations for her department. System users need to be able to change their passwords. This is an Athena function, outside of this system. Some more specific authorization rules:

Proposals Anyone can see (after approval or during development?) Seeing proposals during development can help departments collaborate on joint subjects – reducing need for departmental systems. Only the master subject department can modify or change status. Comments on Proposals (from department members) Entered and seen by master and joint departments only. Comments on Proposals (made by COC) Seen by COC, Academic Services, and relevant departments only. Note: maybe COC comments should be of two types – those that are restricted to other COC members only and those that are viewable to the departments as well.

4. 1. Functions

Add/change/delete user authorizations: Use the General Table Maintenance (GTM) facility. (This function is used mainly by Academic Services.)

Add/change/delete authorization function codes used for CIS. (Use GTM).

Validate a user's authorization to perform a specified function given the proposal's department, status code, and attributes. (Expansion of existing functionality). Answer: Is user ' X' allowed to perform operation ' Y' on object 'Z'? Produce list of allowable functions against a given proposal given the user's authorizations and the proposal's department, status code, and attributes.

4. 1. 3 Data

The existing stv_roles_auth and stv_roles_desc tables are used for user authorizations. New tables hold: ? Authorization names allowed for system (used to validate updates to stv_role_auth) ? System functions the authorization is used for ? Proposal attributes referenced by authorization check 4. 2 Web / Print Publishing 4. 2. 1Description A major feature of the catalog system is the ability to download selected subject data from the database to html or print. A generic facility is planned that will satisfy most or all reporting capabilities for catalog and other downloads.

Technical Discussion:

The emerging XML and XSL technologies are the desired mechanism for producing downloaded data so that the new system is compatible with state-of-the-art technologies. In brief, XML enables data to be downloaded into a hierarchical tree of elements and attributes with each data element tagged with its name. Relational data in the database should map easily to XML. (If necessary, the elements can be defined using DTD for transfer to another program, but this should not initially be necessary for CIS.) The XML standard will eventually include an XML Query feature that may eventually apply to the Search functionality, but probably not to QBF.

XSL is a language that enables transformations and formatting of XML data. Among other things, XSL is capable of filtering out unwanted data (both rows and columns, in database terms) and of sorting (reordering) data. By using XML and XSL, one simple data download from the database into XML format can feed numerous report formats written in XSL. Practically speaking, several varieties of XML downloads are more efficient with selection and

sorting is done as part of the XML download instead of the XSL style sheet.

The selection and sorting functionality is also usable for QBF.

The XSL documents can be written using an XSL Editor (that needs to be selected.) A well-chosen XSL editor should be usable by non-technical users. Only IE5 and Netscape 5 browsers can format XML documents, so an XML to HTML converter is needed to convert XML documents on the server, producing HTML files that can be served over the Internet. XML documents can be converted dynamically on the server to deliver over the Internet, but for catalog downloads, a converter that produces html documents to store on the server is more useful.

The likely program is ' XT' by James Clark. More research is needed. Reports initially proposed for the XML technology are: ? Bulletin download (web and print) – see ' Catalog Download' ? Miscellaneous downloads (Faculty report, UROP, Thesis subjects, HASS, etc.) – see ' Catalog Download' ? Department coordinator lists (web and print) – see ' Contact List Maintenance' ? Master schedules and sub-schedules (web and print) – see ' Unified Calendar Maintenance' ? QBF reports (this function needs dynamic formatting from XML to HTML) Degree Path downloads – see ' Track and Display Departmental Degree Path Information' (print – chapter 7 of Bulletin and web) (Priority 2) ? Subject Evaluation Forms – see ' Process Subject Evaluation Forms' (Priority 2) ? Miscellaneous text for memos, descriptive information in various bulletins – see ' Define auxiliary fields for subject/department/other use' (Priority 2) More detail is available on the above reports. The basic proposed publishing architecture is as

follows using Bulletin downloads as example: 4. 2. 2 Functions Select data from database, sort, and format as XML.

This could be a generic program if selection criteria are simple enough. This functionality is also used for the QBF functionality. Oracle also has products for downloading database data to XML. Apply an XSL style sheet to an XML file and produce an HTML file. This is a probably a product such as 'XT'. 4. 3 Generic Menu / Form Templates 4. 3. 1 Description Templates are used to control ? Menu contents and layout (sequence and position on page) ? Form field contents and layout By using templates instead of hard-coding content details into programs, changes are easier to make.

For efficiency, it is not desirable to interpret templates each time a menu or form is produced. Consequently, a 'compiler' is used to produce the form or menu from the template specification. 4. 3. 2 Functions Add/change/delete menu, form, download templates (using GTM). 'Compile' the menu or form into HTML from the template specification. An XML product such as 'XT' may be used for this purpose. 4. 3. 3 Data Database tables hold ? Form/Menu ID's and descriptions ? Field descriptions ? Links between the form and field descriptors including position on page. 4. 4 Update Academic Year (Term) 4. 4. Description In the old web and Clipper systems, at a particular point in the year, Academic Services sets the academic year for which proposals apply. Only one year's proposals can be worked on at any given time. The new system allows proposal updates for any year and term that is current or future (within limits.) One proposal can be in work for the current year and term; current year, future term; next year, any term. One proposal

can be in work for more than one such year and term simultaneously. For example: One subject may have different titles or topics each term it is taught and it is taught both fall and spring.

BUT IS THIS FEATURE REALLY NECESSARY? THIS IS AN OPEN QUESTION. 4. 4. 2 Functions Set minimum and maximum term for which CIS updates allowed. Set term code in MITSIS (this will still be done.) Will the timing change? 4. 4. 3 Data The CIS configuration file holds this information. 4. 5 Generic Code Table Maintenance for CIS 4. 5. 1 Description Numerous code tables must be maintained for the Curricular Information System. Examples: Status codes, Status code transitions Examples mentioned under other functional components: Menu Text, Form Text, and Authorizations Authorizations needed to update these tables vary.

Roles with differing authorizations: Programmer, CIS administrator, Department Administrator Note that many current MITSIS tables are used as a source of valid data. Existing MITSIS update methods will continue to be used to maintain such data. 4. 5. 2 Functions Add/Modify/Delete items in code tables (Using GTM) Check user's authorizations for updating code tables. 4. 5. 3 Data * signifies data for CIS GTM maintenance Departments (Valid departments are maintained elsewhere in MITSIS) Audit codes (Valid audit codes are maintained elsewhere in MITSIS) *CIS Status codes (s/b a programmer function only) CIS Revision codes (s/b a programmer function only) Term codes (Maintained elsewhere in MITSIS) Instructor (not used - new MITSIS WTW system has replaced this) Department Contact categories (each department may have different categories) Department Contacts - see

Contact List Maintenance Other tables as detailed design warrants 4. 6

Generic Workflow Mechanism 4. 6. 1Description See ‘ CISWorkflow. doc’ for more information. 4. 7 Catalog Download 4. 7. 1Description The new system will provide for the downloading to either print or web of MIT subject listings, now produced from the Clipper system and Ventura.

Downloads may include all data for the Bulletin (Chapter 8) or some subset of data – for example: all subjects for just one department, group, or just one subject. Print downloads can be used for ‘ tear sheet’, ‘ galley’, and ‘ final page proof’ versions. Many aspects of the Bulletin downloads are controlled by data within the subjects themselves (e. g. suppress-listing flag, group-within-department key, etc.). The selection of subjects and fields to display, the order of presentation, the type and style of the download can be controlled separately from the data. See the ‘ Web / Print Publishing’ functionality. . 7. 2 Functions Select catalog data and format as XML. (Although filtering and sorting are provided by XSL, it is more practical to offer these functions in the selection program.) Produce style sheets (as XSL) for the display of data as HTML or one of several print publishing formats. Style sheets can be prepared using a visual editor (to be selected.) Special Features of the on-line catalog Multiple terms of a subject available for students on web: Need ability to view catalog from prior years and/or terms. (How many in all?) Students may wish to see “ prior course, current and projected versions”. . 7. 3 Data See the CISData. doc document for a detailed list of catalog data fields and validity checking. See ‘ CISMigrationRefresh. doc ‘ for migration/refresh information. Department data including: ? Department URL ? Department

Administrators ? Department subject evaluations link. For example:
<http://tute.mit.edu:8001/acadinfo/sse/courses/course8.html> Subject data including: ? Schedule data for each subject being taught that semester. ? URL to additional subject info, if available ? Catalog link to any 'meets with' courses, pre/co-requisite courses Link to campus map where classroom scheduled Issue: which map version should be used? Old, friendly, fast, but out-of-date: <http://student.mit.edu/map/mitmap.cgi?xxxx> Newer, horribly slow and insulting, but IS 'approved': <http://whereis.mit.edu/bin/map?xxx> Special Auxiliary Information in the on-line catalog ? Access from on-line catalog listing to auxiliary information about subjects. Several types of information are possible – HASS-D supplements, etc. ? Section information available in catalog, where applicable (where sections differ in content.) ? Final Exam/Paper/Oral Exam If a schedule has changed for a subject from what was previously published, a note should be added to subject listing. (Similar to IAP?) ? Listing titles currently show that subject has changed since previous year (?) E. g. " Revised Content and Units" ? If subject can be registered for, a button is used to link to the 'add-to-schedule' function (which allows student to pre-register for subject) ? Currently, HASS-D lottery subjects have stmt: " You must enter the HASS-D lottery to take this subject. " The new system uses a flag to track this information. Similarly, Sloan has special icon linked to sloanbid.mit.edu. It is shown with stmt: " You must pre-register and participate in Sloan's Prioritization process to take this subject. " A flag is needed to support this. ? Subdivisions of courses: by degree paths? (This should provide small enough files.)

line Help 4. 8. 1 Description Technical user documentation for the system is needed, but with good user interface design, only minimal explanations are necessary for system features. Help types planned: ? Pop-up 'What-is-this?' definitions and 'How-to' explanations available on menus and forms. ? Glossary of terms (including the definitions also in the Pop-up's) ? Index of brief articles that can be selected from appropriate web pages ? The brief articles themselves include: o Overviews for new users o Concepts explained and o 'How-to' explanations (including the explanations also in the Pop-up's). ? A new user tutorial would be nice but probably there are no resources for this. 4. 8. 2 Functions Support pop-up help texts, index-to-article links. 4. 8. 3 Data A collection of html files covering the articles described. . 9 Contact List Maintenance 4. 9. 1 Description Academic Services keeps a list of department catalog coordinators that is updated at least once a year. They also keep lists of other department contacts. These lists include name, phone, email, office location, and perhaps other information (e. g. number of copies of CSB and addenda to send to department). Other information needed: 'flags' to indicate that person is catalog coordinator, IAP coordinator, department undergrad administrator, and/or department graduate administrator, schedule coordinator, transfer credit examiner, etc. The roles are often subdivided by program, not merely by department. Note: each department may have own method of breaking down roles so they need a way to specify categories as well as roles. 4. 9. 2 Functions Add/modify/delete contact categories (name and description) by department. (Using GTM). Add/modify/delete contact names by category for a department. (Using

GTM). Download catalog coordinator ID's by department to the authorization table (stv_roles_auth). See the ' Person Locator Feature'. New Program (for use by Department coordinator or Academic Services): Select list of people by any table attributes and/or functional role(s).

Specify which fields of information are selected and in which order. Feature can be used to produce: ? Email lists ? Contact lists for print or html Alternately download all contacts to XML format and use various XSL style sheets to format. 4. 9. 3 Data Data tables for contact categories and contact names. 4. 10 Unified Calendar Maintenance The Unified Calendar holds all scheduled activities (the academic calendar, catalog production calendar, dates and times/sequences of activities around registration, etc.)

Modification of the table for a new year can be done quickly using the General Table Maintenance subsystem.

A more sophisticated method (Priority 2) involves computing the new table data from a generic date table where dates are expressed as computable (e. g. ' REG day + 1'). When the computable method is used, the exact date table can be generated. Other database date tables now in use can be generated from this one. From the master schedule table, printed (e. g. rtf) or html schedules for all purposes can be downloaded. For example, the catalog production schedule can be generated. Download contents are controlled by selecting date ranges, activity categories, and participant groups.

Download formats can vary – controlled separately from data and from download program. 4. 10. 2 Functions Add/change/delete date entries from

the calendar table. (Using GTM). New Program (for Department Coordinator or Academic Services): Select a subset of calendar dates by any table attributes. Specify which fields of information are selected and in what order. (Feature can be used to produce calendars for print or html publication.) Alternately, select everything (to XML format) and let the XSL style sheet determine which items to select, which fields to display, and in what order.

New program (Priority 2): Compute actual dates from computable date specifications (like a spreadsheet) – e. g. ‘ REG day + 1’. When this method used, the date entries are entered as computable fields and the computation is done afterwards. Once the computations are set up, only a few dates need to be specified each year to generate a complete new schedule. New

Program (for Academic Services) (Priority 2): Update various MITSIS calendar tables from the master calendar table. 4. 10. 3 Data Files now done manually that can be produced automatically: File: L:

publiccataloguecoordmeetcatalogaddprogs. doc nd individual school versions of this. Web: registrar/www/calendar. html (academic calendar for current term) Data for the unified calendar master schedule database table includes:

(Diagram this in RDM.) ? Academic year ? Activity unique key ?

Activity name (short description) ? Description of activity (long

description) ? Category of activity - several flags: department or

program code other grouping – e. g. school schedule category – catalog

production, pre-registration ? Date (computable: e. g REGday + 1) ?

Date (exact) – computed from the above date Time (optional) ?

Sequence number within date (optional) ? (Additional possibilities): 1)

names of programs to run to satisfy this activity 2) check-off fields that

indicate the scheduled item is complete. 4. 11 Set User Selection Preference (QBF) 4. 11. 1Description Users are authorized to work on subject proposals for their departments. They may also view in-work proposals from other departments that are joint/swe/meets with proposals or proposals with certain criteria (for instance, the HASS Office may view all HASS-D and HASS Elective subjects).

System users can also always view approved subject proposals from any department. Within that large set of proposals, users may wish to restrict the set of proposals shown in the proposal menu by specifying attributes of proposals. A search-like selection can be set up and saved for future use. This facility satisfies the Clipper system's ' QBF' facility. Note that many queries can be set up and saved for various users so that not everyone will need to learn how to set up preferences. Queries need to be able to plug in user id, user's department, and other data without this being hard-coded into the query itself.

Some suggested attributes: ? All proposals in user's department(s) - this is the default for department coordinators ? All proposals in a specified department (the default Academic Services administrator option) ? All proposals this user has worked on. ? Program/degree path sub-category (this information is new to proposal system) ? Undergraduate, Graduate (G), High graduate (H) ? Particular status code (in-work, in-review, etc.) ? Master subjects which are: Joint/swe/' meets with' subjects ? ' Slave' subjects which are: Joint/swe/' meets with' subjects Subjects with a particular attribute (HASS, GIR, ROTC, Thesis etc.) ? Subjects with

pre-requisites or co-requisites ? Units arranged subjects ? Subjects that specify a particular subject ID as a pre-requisite or co-requisite ?

Subjects with specified field names containing specified values or ranges of values (This the main QBF functionality – it effectively sets up a database query but looks like QBF.) A set of proposals can also be established by using a search word or phrase that applies to the titles or descriptions for any proposal.

Other fields too? (Priority 2) 4. 11. 2 Functions Select a set of proposals by specifying attributes. Any field in proposal listing can be queried. Need to be able to query fields with no data (null or blank). Need to be able to set up compound queries ‘ this and that’, ‘ this or that’, etc. Add or delete items from the set. (Priority 2?) Save the set of records into the database (i. e. their keys, not their data and not the selection criteria.) Attach a set name, query name, user id, and date. Set name by default is same as query name plus user id and date.

Alternately, download the set of records – both keys and data - into XML format for publishing. See the ‘ Web / Print Publishing’ discussion. Save the preferences (the selection criteria that produced the set) in the database. (This is especially useful for future use when using compound selection criteria.) Attach a query name, query description, user id, and date. 4. 11. 3

Data A user selection criteria table, including: ? Query Name ?

Query Description: Free-form text to hold the selection criteria (where clause) ? User id Date A user set table, including: ? Set

name ? Query Name ? User id ? Date 4. 12 Display a List of

Subject Proposals that can be worked on in proposal system 4. 12.

1Description Display a list of subject numbers (and titles) that the user can work on. An icon representing the proposal's status may be included. Within the set of all proposals the user may work on a subset may be established using the 'selection preferences' function discussed under 'Set User Selection Preference (QBF)'. 4. 12. 2 Functions

Retrieve the selection criteria and execute it to produce a set of proposals to display. See 'Set User Selection Preference (QBF)'. Request 'edit' mode or 'print preview' mode for the subsequent detail display. Edit mode shows all the control information (as described under 'Display the detail for one proposal. ') Print preview mode leaves off the control information, simulating the way the description looks when printed in the bulletin. Control the order of the display (Priority 2): Some suggested orders: ? Subject ID (default) ? Cluster number ? Equivalent number

The default order is subject code, subject number. Display the set of subject proposals retrieved as a menu of subject ID's with status icons. Each item in the list can be clicked on to view its detail. 4. 13 Select one Proposal to Work on 4. 13. 1Description Users need several methods for selecting a proposal to work on. When a proposal is selected, the most recent version of the subject proposal is displayed. If more than one version is available, the alternate possible versions are listed in a dropdown list at the top of the detail subject proposal display.

The user can see the detail for an alternate version by clicking on one of the alternate versions in the list. 4. 13. 2 Functions Select a proposal to work on

by clicking on one proposal listed in the menu of subject proposals in the user's current set as produced from the current selection criteria. Type in a subject number (for an existing proposal – not a new proposal). ' Select Next' / ' Select Prior' from one subject proposal detail page. The ' Next' or ' Prior' proposal is the next or prior one in sequence in the set of proposals in the subject proposal menu.

This feature is not available when the subject ID was typed in. Select an alternate version of the current proposal. Alternate versions are displayed with: Term Code, Status Code, User ID, Date, and identifying brief description (which the user entered when saving). 4. 14 Display the Detail for One Proposal (edit mode or print mode) 4. 14. 1Description Once a particular proposal and version is selected, the detail for this proposal is displayed on the screen for user viewing and editing or for printing. The user selects edit or print mode on the preferences page, but can toggle between the two modes from the detail page.

For edit mode: ? The listing itself is shown in the middle of the display in approximately the same format as it will be published in the Bulletin. ?

Control information and any error messages are shown on the left. ?

Edit/status change options are shown on the right. ? Auxiliary information about the proposal that is not printed in the Bulletin but which may be used in other publications (e. g. HASS supplemental descriptions) can be displayed below the main proposal listing (in a different color) so that it is not confused with the Bulletin information.

Distinctions need to be made between data reviewed by COC vs. not. For print mode: ? Only the proposal listing is shown, not the control information or edit options. Is auxiliary information shown in print mode? Do we need a print mode at all? Why not just use the download feature? 4. 14. 2 Functions Check user's authorizations. Retrieve all proposal and auxiliary data from the proposal database (or from persistent store) for the current version of the proposal (Key: Subject code, subject number, effective term, and version). Format individual fields (according to the template, if possible).

Based upon error codes for the version, highlight fields in error and prepare error messages to display. Determine editing and status change options according to the user's authorizations. Prepare the display of subject proposal data, auxiliary data, control information, and editing and status change options. By default the most recent version of a proposal is compared to the most recently approved version, if any. Allow the user to select different versions to compare from a dropdown list. Highlight differences using different colors or symbols. (THE SYMBOLS TO USE? – OPEN QUESTION. See the function ' Side-by-side Comparison of Text Fields' for more information. Optionally, format data only when it changes (when form data is submitted) and save the formatted listing for later display instead of reformatting each time. This can speed the process. 4. 15 Cancel / Temporarily Remove a Subject / Delete a Proposal 4. 15. 1Description Delete means the proposal was started in error and has never been approved. When a proposal is deleted, eventually (though not immediately) its data is removed. Cancelled subjects are those that were once taught but are being permanently retired.

They can later be reinstated. Their data never goes away. Temporarily Removed subjects are those that will not be taught for at least a year but which will eventually be taught again. (Is the category needed? The system tracks when subjects are not being offered for a year or more so that they are not be printed Bulletin. Also, there needs to be a format mechanism to bring these back from their temporarily removed status.) 4. 15. 2 Functions Verify a user's authorizations to perform the function. Verify that the subject's / proposal's status is such that the operation is allowed.

Change the proposal's status to the new one. 4. 15. 3 Data The proposal's status code, activity date, and user id are changed. 4. 16 Reinstated a Cancelled Proposal 4. 16. 1Description When a subject is cancelled, it must first be reinstated before any work can be done on it. Only cancelled subjects can be reinstated. When reinstated, subjects are assigned the status of ' in-work proposal'. 4. 16. 2 Functions Verify a user's authorizations to perform the function. Verify that the proposal's status is such that the operation is allowed. Change the proposal's status to the new one. . 16. 3 Data The proposal's status code, activity date, and user id are changed. 4. 17 Re-number a Subject 4. 17. 1Description Why are subjects renumbered? Is renumbering subjects still necessary? When renumbered, subjects are assigned the status of ' in-work proposal'. 4. 17. 2 Functions Change subject number in database. Track former subject number in database. Find all references to this number (joint subjects, equivalentents) and modify them. Some details need to be worked out about exactly what needs to be done to references. 4. 17. 3 Data Proposal field ' old number'. . 18 Add New Subject Proposal 4. 18. 1Description Add new subject proposal (new number and

code). 4. 18. 2 Functions Type in the id (code and number) for a new subject proposal and verify that the subject id is available (has not been used for at least 5 years.) See also the ' Grid of Subject Numbers'. Subject ID (code and number) cannot have been used in last five years. 4. 18. 3 Data All data is the same as for a modified subject proposal. 4. 19 Set Default Parameters for New Proposals 4. 19. 1Description Default grading types, units, suppress flags, etc. can be assigned to new subject proposals based upon criteria to be decided. Examples: ? HASS-D subjects: enrollment is limited to 25 (18 for CI subjects); final exam required vs. not required vs. optional (and so specified later.) ? Freshman seminars: 6 units of credit, P/D/F grading, etc. 4. 19. 2 Functions TBD. 4. 19. 3 Data TBD. 4. 20 Save Modifications to a Proposal 4. 20. 1Description When the user submits proposal form input, a new version of the proposal is saved into the database. 4. 20. 2 Functions Check user's authorizations to submit proposal changes. Retrieve form input. Validate the form input data, preparing a set of error codes to save for the version. Note that some validation will be possible in the client. (See ' Proposal Form Input Validation' for more information.) Retrieve all proposal and auxiliary data from the proposal database (or from persistent store) for the current version of the proposal (Key: Subject code, subject number, effective term, and version). Compare the new version to the latest approved version to set revision codes. (See ' Automatic Setting of Proposal Revision Codes'.) Store the new proposal version into the database.

Redisplay the proposal as described in ' Display the detail for one proposal'. Display error messages (to left of detail display). Highlight fields in error. 4.

20. 3 Data All proposal data is involved. 4. 21 Automatic Setting of Proposal Revision Codes 4. 21. 1Description Revision codes are set for each proposal field that may change. Currently, only fields that appear in the bulletin are compared. In the future, all fields can be compared. The exact set of fields to track with revision codes is maintained in the proposal control data table.

The default two versions to compare are:

Left: the most recently approved version (from MITSIS catwwdat) and

Right: the most recent proposal version The current system compares the

versions described above. The new system allows users to select which two

versions to compare. Each version is identified more explicitly, e. g. ' MITSIS

approved version effective 1998FA' (the approved version of subject now in

catalog), ' Working copy saved 11-21-2000 by ssmith for: Prof Jones

revisions' (an in-work version of a proposal), ' Proposal submitted on 12-05-

2000' (for a proposal that has been submitted electronically to the COC).

The explanatory titles come from information that users enter upon saving

the proposal. The short fields can be compared in order to set revision codes.

The description field cannot be set up automatically – some description

changes are minor and some major. The system can flag that a change to

content has occurred and Academic Services sets the flag for major vs.

minor. (See the ' Side-by-side Comparison of Text Fields' feature.) Although

revision codes can be calculated when needed, they are stored in the

database for efficiency and so that proposals can easily be queried by

revision code.

The revision codes to store are those calculated by comparing the latest version to the latest approved version. If the proposal is new (has never been approved) no revision codes are stored.

4. 21. 2 Functions Compare values for all fields requiring revision codes. For the description field, use the ' Side-by-side Comparison of Text Fields' facility.

4. 21. 3 Data Proposal data field control table (indicating which proposal fields are subject to revision codes.) This table has a one-character flag for each field compared indicating if data different or not: Values: ' Y' / ' N'. See the document: CISData. doc for more detail on revision codes.)

4. 22 Side-by-side Comparison of Text Fields

4. 22. 1 Description Two text fields can be displayed side-by-side with additions/deletions/changes indicated by using different colors. When comparing descriptions or other multi-word texts, the text should be compared word-by-word since comparison of whole paragraphs is not useful. The selection of the two text fields to compare is done outside of this function.

4. 22. 2 Functions Word-by-word comparisons of text areas can be done as a special algorithm:) Separate a description into individual words

2) Do the comparison (Unix ' diff')

3) Put the description back together for display.

4. 22. 3 Data The comparison needs to track: ? Field being compared ? Difference type (used to display differences with color-codes): present on left, not right; present on right not left identical values; different values

4. 23 Proposal Form Input Validation

4. 23. 1 Description Whenever a proposal changes, the data entered for each field is validated. The CIS Data Dictionary contains the detailed data checking for each field plus the ' cross-field' data checks. Refer to the Clipper documentation where questions remain.

4. 23. 2 Functions Generic validation functions: compare a

field value to the specification of its valid values: ? Numeric /
 Alphanumeric / Alpha / etc. ? An enumerated list indicated as a code
 table reference ? A range of values

4. 23. 3 Data A one-byte flag is kept for each field in the proposal that may have errors. Each flag may have these values: ' Required field missing'. ' Value not in allowed range. Other values are unique to the field but are defined in a table of error codes and messages.

4. 24 Change Status of a Proposal 4. 24. 1Description Proposals go through several status codes from ' New In-work' to ' Submitted' to ' Approved'. The status code of ' Submitted' functions in the new system the way printing and getting a signature for hardcopy submission functioned in the old. With the new system, Academic Services no longer needs: ?

Printed forms with hardcopy signatures ? Manual checking of signed hardcopy proposals and/or galley proofs against Web proposal database data

Logs of signed hardcopy proposals and returned galley proofs ? Logs of marked up galley proofs (Is a ' page proof reviewed' status code needed?)

Once approved, a proposal is kept permanently, but new versions can be undertaken that, when approved, supplant the currently approved version. The exact set of status codes and allowed transitions is detailed in CISWorkflow. doc. Status changes are effected by explicit request of the user and are subject to authorization checks. The system displays only allowed status changes to the user.

The old catalog systems combined ' true' status changes such as new, reinstated, cancelled, deleted, and temporarily removed with editorial change flags such as title change, renumbered, editorial change, etc. The CIS tracks editorial changes not as status code changes but as revision code

changes (most of which are automatically determined by comparing the current version with the prior approved version) – see ‘ Automatic Setting of Proposal Revision Codes’. 4. 24. 2 Functions Determine users authority to perform requested proposal status change. If allowed, change the status code in the proposal database.

If not a valid status change, display an error message and do not change the status. After changing a status code, the user’s authorizations, menus, etc. must be recalculated to take into consideration the new proposal status. 4.

24. 3 Data Retain the ID of the user who changed the status code. Also retain comments made at the time of status code change. 4. 25 Input Auxiliary Subject Information 4. 25. 1Description This feature enables departments to specify auxiliary information about subjects. Auxiliary information is NOT reviewed by COC. It can be entered at any time for the subject (when approved or still a proposal).

The information may be from a standard system table and used for all departments and subjects. Alternately, information may be for user-defined data. (Setting up user-defined data definitions is a Priority 2 function – see ‘ Define auxiliary fields for subject/department/other use’.) Some of the information is specific to a department, others to department programs, others to specific subjects or subjects by term. Examples: ? URL’s for departments. These are displayed on the web page: <http://web.mit.edu/academics.html>. These URL’s don’t often change. Some subjects’ final exam/paper/oral status is optional – When it is optional, the department coordinator can specify which type of final is going to be held for a particular

term (this information is displayed in the catalog and aids the final exam scheduling process.) ? Auxiliary subject descriptive content (especially used for HASS subjects). Note that the facility to define new data fields for auxiliary input is a Priority 2 function (see ‘ Define auxiliary fields for subject/department/other use’), so the ability to enter data for such fields is also Priority 2. URL’s for additional information about a subject. The current method of obtaining this data is for departments to email the information to[(mailto:protected@mit.edu)]edu where someone then enters it onto web page: <http://web.mit.edu/acs/acaduses2.html>. The current web catalog download parses this html file for the data to publish. In the future, CIS obtains the information directly. The Academic Computing folks can download their URL’s from the new CIS system. ? Lottery information – which subjects have a lottery of what type in what term. ? In which term(s) the subject will be taught. 4. 25. 2 Functions

Generic programs to: ? Generate form input for auxiliary data ?
 Validate auxiliary data from form input ? Store data in auxiliary
 tables ? Download to XML format ? Specify in XSL editors 4. 25. 3
 Data Tables to hold auxiliary department-level data. Examples: ? URL to
 department ? Number of copies of class lists to produce (if needed –
 department can print their own) ? Number of final exam lists to produce
 (if needed – department can print their own) Tables to hold auxiliary subject-
 level data by term. Examples: ? Is subject taught this term? Lottery
 status (and lottery type) ? Subject’s final exam/oral/paper
 requirement ? URL for auxiliary subject information 4. 26 Email proposal,
 auxiliary information, and comments to selected persons 4. 26. 1Description

Sometimes department coordinators need to email a proposal to one or more people with attached comments. In the future, faculty can view proposals on the web under access control, so emailing the proposal itself will not be needed as often (the URL can be mailed instead). Comments, however, need to be kept. The sender of the email also needs to keep a copy of the comments. . 26. 2 Functions See the ‘ Person Locator Feature’.

Process the html form used for the email. Format the subject proposal to attach to email (can be html or simple text file). Save the email detail in the database (save comments in the comments table). Send the email. 4. 26. 3

Data The database holds: ? A record that an email was sent by subject code and number, with flags for what was sent and who it was sent to. ?

Comments are saved like other comments The form includes: ? A select list of likely recipients (see ‘ Person Locator Feature’) ? Comments to include in email A checkbox (initially checked ‘ on’) that says ‘ cc sender’ ? Checkboxes for which parts of proposal to send: formal

proposal only, auxiliary comments, control data, etc. ? Optionally, only a URL for the proposal could be emailed so the recipient can view the proposal

on-line. 4. 27 Conversion of Character data for form entry, storage, display

(This is a Technical topic) 4. 27. 1Description Certain data used in the catalog description, names, and titles requires foreign language codes (e. g. umlauts, accent grave) and math or physics symbols. These codes lie outside of the ASCII 7-bit character set.

The ASCII 8-bit character set includes most European language special characters and other common symbols, but data entry on web forms is difficult for many of these codes. How data is going to be stored in the new

Oracle 8 database is not yet decided. The current student database expects ASCII 7-bit characters but it is possible to store ASCII 8-bit characters (it is being done for IAP.) Such codes are easy to move in and out of the database and to display in html and most print formats. However, SQL queries and PC downloads do not always maintain the character values accurately. Using special codes for extended ASCII characters (e. . <`>) works for HTML downloads but not print downloads. More information on storing and displaying foreign languages, math, and scientific notation is available at:

<http://web.mit.edu/acs/WebGuide/technical-foreign.html>. 4. 27. 2 Functions

Provide a list of special codes (when asked) for use in the catalog

descriptions. Possibilities: ? A palette of special codes to copy and paste

into forms (extended ASCII codes) ? Html extended codes (e. g. <`>) to

copy and paste into forms ? Other? Convert special codes as entered on

forms into the approved database format.

Note: more than one input format may be supported but all will be converted to the approved database format before storage. Convert special codes as saved in the database into the approved display format for forms. Although multiple formats are supported for input, only one will be used for re-display on forms – ideally, codes as printable () will be displayed instead of their numeric representation. 4. 28 Review Submitted Proposals 4. 28.

1Description After a department submits a proposal, it is reviewed by COC or CGSP. Other departments also need to review proposals – the HASS Office, etc.

The system needs to keep comments from the review and make them available to the appropriate people. Finally the system needs to ‘log’ proposals that have been reviewed and the review outcome – approval of the proposal or its return to the department for modifications or for answers to questions. 4. 28. 2 Functions Put a proposal into review, approve a proposal, or ‘reject’ a proposal (i. e. send it back to a department with questions):

These are status code changes as discussed elsewhere. Display list of proposals in review for appropriate review group: COC or CGSP, HASS, etc.

Identifying who is authorized to access what is handled by the authorization subsystem. The list and the subsequent detail display are similar to what is described above. Add review comments to a proposal. Comments can be added at any stage. Monitor who can view the review comments and make them available as appropriate. This again is an authorization function. 4. 28.

3 Data Review comments. 4. 29 Person Locator Feature (for use on web forms) 4. 29. 1Description This program feature is adapted from WTW. Currently that implementation uses Java Script. The CIS project will use Java.

Feature: Provide names of department members on select list (hidden key is Kerberos ID). If the user wants a name not on the list, find it in the database and add it to the list. This feature can be used for any MIT person (student, administrative staff, faculty). 4. 29. 2 Functions Produce drop-down list of ‘likely department candidates’ from saved data in database. Find new person at looking in the table `sfbre_current_person` by last name and optionally first name, MIT ID, or possibly even phone number. Add the new person to the ‘likely department candidates’ list. 4. 29. 3 Data

A table of department 'likely candidates' to use on drop-down list. 4. 30
Refresh MITSIS tables with Approved Proposal data 4. 30. 1 Description The proposal system resides in the same database with MITSIS data and all approved proposals are theoretically available for MITSIS use as soon as approved. In reality, Academic Services may wish to control the availability of new/changed subjects in MITSIS applications. Premature availability may cause problems in the MITSIS banner applications. A refresh feature controls the timing of MITSIS data integration. ACS (not the departments) can run the refresh whenever desired.

MITSIS is updated only with approved subject changes for chosen terms. 4. 30. 2 Functions The user enters a term code and all approved proposals for that term that have changed since the prior refresh are updated in MITSIS. See the CISData.doc and 'CISMigrationRefresh.doc' for more information. Included in the refresh: ? The logic for cluster subjects needs to be improved – all cluster subjects should 'march in lockstep' as far as any data changes – i. e. they should all have the same term code ranges. 4. 30. 3 Data The MITSIS tables to refresh with proposal data are those currently maintained by form SCASUBJI.

Although SCASUBJI will not be retired, it should not be needed in the future since the refresh program should correctly populate the MITSIS tables.

SCASUBJI should not be used because any changes made using it will not be reflected back into the proposal system. MITSIS tables: SCBSU_KEY

SCRSU_VAR SCRSU_HGN SCREQIV SCRATTR SCRTRM SCRGMOD Also

SCRSU_DESC (for IAP) 4. 31 Maintain Class Schedule Book Planning data - in

the database 4. 31. 1Description The current CSB is an 80-column card image file that is used in various scheduling functions – the Classroom Schedule Pre-planning, the Class List, etc.

Without rewriting the scheduling algorithm provided by GASP, many now-manual CSB data maintenance functions could be automated and put onto the web. Doing so benefits not only the Scheduling Office, but also the department coordinators who now need to manually provide scheduling requests to the Scheduling Office. Various scheduling programs require CSB input: ? The Class Schedule Book requires the CSB as MITSIS:

GASP\$DATA: CSBFILE. DAT This data is filtered then sorted in preliminary steps of SSPRECSB. COM to select out only desired subjects. (See the

program for details.) Pre-Planning Report requires the CSB as MITSIS:

GASP\$DATA: CSBFILE. DAT This data is filtered then sorted in preliminary steps of SSPREPLN. COM to select out only desired subjects. (See the

program for details.) ? MITSIS SSPREVNT – Load Classroom schedule

from CSB and find errors (eliminate?) ? MITSIS SFPRECLU – Cluster

Report (eliminate?) ? GASP SFPRKGSS – Simulated Schedule ? GASP

SSPGSFA1 and SSPGSFA2 - Scheduling 4. 31. 2 Functions Notification

Reminders to review and submit classroom-scheduling requests can be put onto the departmental ‘ portal’ via a command sent by the Scheduling Office.

See the ‘ Electronic Notification’ function – a Priority 2 feature.) The

reminder stays on portal until department submits classroom schedule

requests. (This function obviates the manually prepared memos sent via

inter-office mail.) Add/Update/Delete CSB data in the database -- Used for

Classroom Planning: The web front-end for department administrators and

the Scheduling Office offers a list of subjects and sections slated to be taught for the selected term. The list shows the time and room that applied in the prior term when the subject was taught.

This is the same information the Department Coordinators now get as hard-copy, however, it will be more complete since it will contain subjects taught this year but not in the prior year (without room, of course). The listings may be subdivided by department program to make them more manageable.

Department Coordinators can change the time and room pattern for the subject and section. If the department 'owns' a room, they can specify it by room number, otherwise they can only specify a room by its pattern.

Available patterns and specific rooms, where allowed, are offered as dropdowns.

Existing MITSIS tables with MIT and non-MIT rooms will be used, augmented with new tables to supply the valid time and room patterns. The user can also add free-form comments if needed. When the department administrator is done with planning, the information can be submitted to the Scheduling Office and so is no longer in the hands of the Department. Since all subject-scheduling requests are needed before they can be processed, the plans are not submitted subject by subject, but as a group for the department as a whole. Display Room Requests by Day/Time/Room Pattern

This feature would be used by the Scheduling Office to examine scheduling requests in order to resolve them. Uses same functionality as the 'Display Room/Subject Schedule Grids'. Provide CSB data to scheduling programs: Download the 80-column CSB file for SSPREPLN and SSPRECSB and other

uses. Need to work out details. It might be better to filter and sort the data that these programs need in the download instead of in the subsequent programs. (See MITSIS program ‘ Download Continuing Subjects’, sfprespl. pc, executed by SFPRESUB. COM, for an example.) Some needed parameters: include freshman advising seminars? . 31. 3 Data One (or more) table to hold the current CSB data: Unique Key: subject code, subject number, section number (type and sequence) MIT Room (building and room number) – validated against MITSIS room table Non-MIT Room – validated against a MITSIS EVENT scheduling table. Room Pattern (used for requesting a room) – validated against a room patterns table. Room patterns include: where on campus, room size, facilities available in the room, etc Time Pattern – validated and translated by time pattern table Time Comment – may be used with or instead of time pattern - from ‘ standard pattern’ table?

Data from other CIS data tables: Final Exam comment Meets with comment Maximum enrollment Linked subject/section data (derived from meets with information) Various flags for GASP (see CSB layout description) Comments to the Scheduling office (not part of CSB) from the department administrator about the scheduling request. 4. 32 Final Exam Scheduling 4. 32.

1Description The Final Exam Planning procedure is similar to the Class Schedule Book planning. Once subject registration figures are known, final exam planning can begin. Some subjects are required to have final exams, others are required NOT to have finals.

Before Final Exam Planning can begin, Department Administrators must specify which of the remaining subjects will have finals. (See the function: ‘

Input Auxiliary Subject Information’.) 4. 32. 2 Functions Notification

Reminders to review and submit final exam scheduling requests can be put onto the departmental ‘ portal’ via a command sent by the Scheduling Office. (See the ‘ Electronic Notification’ function – a priority 2 feature.) Reminder stays on portal until department submits final exam schedule requests. (This function obviates the manually prepared memos sent via inter-office mail.)

Update the Final Exam Planning Report Department Administrators view the final exam planning report on the web for their subjects. (This web report replaces the hard copy planning report that they currently receive and return with markups to the Scheduling Office). On the electronic planning report they view and edit final exam scheduling info, make all the corrections that they currently make by hand. When they are satisfied, they submit the info electronically to the Scheduling Office. System tracks which departments have submitted requests and which have not, also whether all subjects have been submitted or only some.

Note that only subjects that will have a final exam appear on the report. This information is tracked by the system. Review of final exam requests Uses same functionality as ‘ Display Room/Subject Schedule Grids’. When all (or most) requests are in, the Scheduling Office can display and edit requests by: ? Subjects having exams (similar to final exam document that the Scheduling Office produces manually today from marked up planning copies) ? Time and room requests with subject numbers filled in (similar to manual ‘ Room Chart’ that the Scheduling Office uses today)

Produce interface files to/from the final exam scheduling programs Inputs to MSDOS scheduling program can be produced from the database. Outputs of scheduling can be put into database (does this include room assignments?) (Room assignments can be produced from the database from department requests.)