

The **McCammon Group Document Server**: Installation & Customization of the CERN Document Server Software (CDSware)

Cameron Mura*

November 12, 2003^{†,‡}

Overview

This document is a detailed guide to the installation and configuration of the CERN Document Server software suite (CDSware, v0.0.9) in the McCammon Group at UCSD (<http://mccammon.ucsd.edu>). These notes are adapted from a running log of the installation, configuration, and troubleshooting/debugging efforts with this software suite on one of the group PCs (chemcca42); as such, the descriptions provided here are only loosely structured or organized – *i.e.*, this is not meant as an official CDSware manual. This document is primarily intended to serve as a step-by-step reference for installation, customization, and troubleshooting issues, particularly for someone unfamiliar with MySQL, PHP, WML, or the other software/languages/packages upon which CDSware relies. Except for any bugs explicitly noted herein, the CDSware documentation files (README, INSTALL, *etc.*) should be considered as superceding anything described herein.

#####

Pre-pre-installation notes:

#####

- Create a relatively large /var partition (~2 Gb) when installing Red Hat 9 on chemcca42, because that is where all of the MySQL database stuff will eventually reside. . .
- For convenience, you may add your RSA and DSA public SSH keys (*e.g.*, ~cmura/.ssh/id_rsa.pub, id_dsa.pub) to root's authorized keys on chemcca42 (/root/.ssh/authorized_keys2). This circumvents password-prompting (among other things) . . .
- Create a user account 'cdsware' (passwd '*****'), belonging to groups 'users' and 'cdsware' on chemcca42, with standard default login parameters (/home/cdsware, bash login shell, etc. . . .).

#####

UN-INSTALL / UPDATE / INSTALL stuff:

#####

If a current version of CDSware is installed, un-install it, and upgrade to newer versions of the underlying support software (MySQL, Apache httpd, PHP, etc. . . .)

*cmura@mccammon.ucsd.edu

[†]Last compiled: January 7, 2013 (cmura)

[‡]This document is available at http://mccammon.ucsd.edu/~cmura/HOWTO/cdsware_notes.pdf

[NOTE: see the CDSware UNINSTALL file if you want to simply uninstall CDSware alone, and not upgrade all the other stuff...]:

1. Clean-up the MySQL database by deleting 'cdsware' database and revoking privileges (see CDSware INSTALL notes or general MySQL help):

```
# mysql -h chemcca42 -u root -p mysql
      (passwd for mysql root user was previously set to '#%$^&!*', so enter this)
mysql> DROP DATABASE cdsware;
mysql> REVOKE ALL PRIVILEGES ON cdsware.* FROM cdsware@chemcca42.ucsd.edu;
mysql> \q
```

NOTE: You can check that this actually deletes the /var/lib/mysql/cdsware/ sub-directory...

2. Stop current CDSware-related server daemon processes:

```
MySQL: # /etc/init.d/mysql stop
Apache: # /etc/init.d/httpd stop
```

3. Uninstall just the CDSware component, and clean-up directories:

```
# cd /usr/local/src/cdsware-0.0.9/
# [env LANG=C] gmake uninstall      UTF-8 font problems are mentioned below;
                                   you may want to set LANG=C

# cd ..
# rm -rf cdsware-0.0.9 cdsware-0.0.9.tar.gz
# rm -rf /var/www/htdocs/CDSware /var/www/cgi-bin/CDSware
```

4. Remove all currently installed packages whose names contain 'mysql', 'php', and 'httpd' (and several Apache-related files):

```
# rpm -qa | egrep -i mysql | xargs rpm -e
# rpm -qa | egrep -i php | xargs rpm -e
      (note: old conf file saved as /etc/php.ini.rpmsave)
# rpm -qa | egrep -i httpd | xargs rpm -e mod_perl-1.99_07-5 \
      mod_auth_pgsq1-0.9.12-14 mod_python-3.0.1-3 \
      mod_ssl-2.0.40-21.3 webalizer-2.01_10-11 mailman-2.1-8 \
      ...and-any-others-that-were-installed ... ..
      (note: old conf file saved as /etc/httpd/conf/httpd.conf.rpmsave)
```

If actual RPMs are stored in subdirectories of /usr/local/RPMS/for_*/. (e.g., for_mysql/, for_apache/, etc.), then this is an alternative method to remove those which are installed:

```
# ls for_*/rpm | xargs rpm -qp 2>/dev/null | xargs rpm -e
      (i.e., redirect any ldconfig error messages to /dev/null)
```

NOTE: It may be necessary to explicitly 'rm -rf /var/lib/mysql' directory after doing 'rpm -e' ...

5. Make sure the following 9+4+4 required packages still exist from previous installation; they're mostly related to Perl and database utilities, and are stored in `/home/cdsware/for_CDSware_misc/`:

```
perl-DBD-Pg-1.22-1
perl-DB_File-1.804-88.3
perl-DBI-1.37-1
perl-Digest-HMAC-1.01-11
perl-Digest-SHA1-2.01-15
perl-Net-DNS-0.31-3
perl-Time-HiRes-1.38-4
postgresql-7.3.4-2
unixODBC-2.2.5-2
```

If necessary, install the following packages as well:

```
ncurses-devel-5.3-4 (header files and libraries for ncurses)
db4-devel-4.0.14-20 (header files and libraries for Berkeley DB)
gdbm-devel-1.8.0-20 (header files and libraries for GNU db system)
mx-2.0.3-8 (mx extensions for Python software tools)

libungif-progs-4.1.0-15 (programs for manipulating GIF format image files)
webalizer-2.01_10-11 (weblog stuff)
mailman-2.1.1-4 (mailing list software)
MySQL-python-0.9.2-1 (Python interface to MySQL; http://sourceforge.net/projects/mysql-python)
```

6. Install the most recent, stable version of MySQL (or the unstable v4.1.0-0 :)), *via* the following RPMs (downloaded from <http://www.mysql.com/downloads/mysql-4.0.html> and stored in `/home/cdsware/for_MySQL-4.1.0-0/`):

NOTE: This newer version of MySQL supposedly fixes problems with the UTF-8 fontset that occurred in an earlier CDSware install; but in my experience it doesn't cooperate very well with this version of CDSware.

[...But first you need to create these links and reinitialize shared libs:

```
# cd /lib/
# ln -s libcrypto.so.0.9.7a libcrypto.so.0.9.6
# ln -s libssl.so.0.9.7a libssl.so.0.9.6
# ls -lsart libcrypto.so.0.9.6 libssl.so.0.9.6 (check this!)
# ldconfig
]
```

```
MySQL-bench-4.1.0-0.i386.rpm
MySQL-devel-4.1.0-0.i386.rpm
MySQL-server-4.1.0-0.i386.rpm
MySQL-client-4.1.0-0.i386.rpm
MySQL-embedded-4.1.0-0.i386.rpm
MySQL-shared-compat-4.1.0-0.i386.rpm
    (version of the shared libs that's compatible w/ pre-existing MySQL installation)
# rpm -Uvh /home/cdsware/for_MySQL-4.1.0-0/MySQL*.rpm [-nodeps]
```

(use the nodeps option if errors still occur despite linking the 2 libs above!)

Check that the startup script (/etc/init.d/mysql) was installed, and added to boot-up runlevels:

```
# /sbin/chkconfig --list mysql
```

If necessary, activate the mysql daemon for runlevels 3,4,5:

```
# /sbin/chkconfig --levels 345 mysql on
```

7. Install/update the Apache web server v.2.0.43 and related packages from these 10 RPMs (in /home/cdsware/for_Apache-2.0.43/):

```
httpd-2.0.43-1.i386.rpm
httpd-devel-2.0.43-1.i386.rpm
httpd-manual-2.0.43-1.i386.rpm
mod_auth_mysql-1.11-12.i386.rpm
mod_auth_pgsq1-0.9.12-14.i386.rpm
mod_perl-1.99_07-5.i386.rpm
mod_python-3.0.1-3.i386.rpm
    (according to the CDSware team, "mod_python 3.x is recommended due to
    better Apache2 and MySQL support")
mod_ssl-2.8.15-1.i386.rpm
openssl096b-0.9.6b-3.i386.rpm
redhat-config-httpd-1.1.0-3.noarch.rpm
```

8. Install/update PHP v.4.3.1-1 with these 8 RPMs (in /home/cdsware/for_PHP-4.3.1-1/):

```
compat-db-4.0.14-2.i386.rpm
mm-1.1.3-6.i386.rpm
php-4.3.1-1.i386.rpm
php-imap-4.3.1-1.i386.rpm
php-ldap-4.3.1-1.i386.rpm
php-mysql-4.3.1-1.i386.rpm
php-odbc-4.3.1-1.i386.rpm
php-pgsq1-4.3.1-1.i386.rpm
```

NOTE: NO — actually, go back to PHP v4.2.2-17.2, because this newer version does not work perfectly (there are problems with shared libraries, *etc.*). These RPMs are in /home/cdsware/for_PHP-4.2.2-17.2/.

...In doing so, note that you may need to force re-install the db4 libraries (which conflict slightly with the compat-db-4* package) by doing this:

```
# rpm -Uvh /net/linux/RPM/9-include/db4-4.0.14-20.i386.rpm \
    /net/linux/RPM/9-include/db4-utils-4.0.14-20.i386.rpm --force
# rpm -Uvh /home/cdsware/for_CDSware_misc/db4-devel-4.0.14-20
```

IMPORTANT NOTE: *If the above packages are not properly installed and configured, then WML (and therefore CDSware) may not successfully install!...*

9. Check that a recent version of Python is installed (\geq v2.2):

```
#rpm -qa | egrep python    (we have the python-2.2.2-26 RPM installed)
```

10. Install the following 3 RPMs to add programs for CDSware's advanced text-processing features (in /home/cdsware/for_CDSware_misc/):

```
antiword-0.32-2.i386.rpm
html2txt-1.2.4-217.i586.rpm
xlhtml-0.5.1-3.i686.rpm
```

*** Up to this point, check that the 43 packages referred to above have been installed:

```
# ls for_PHP-4.2.2-17.2/* for_CDSware_misc/* for_Apache-2.0.43/* \
  for_MySQL-4.1.0-0/* | wc -l  $\Rightarrow$  43 (should result)
# ls for_PHP-4.2.2-17.2/* for_CDSware_misc/* for_Apache-2.0.43/* \
  for_MySQL-4.1.0-0/* | xargs rpm -qp | wc -l  $\Rightarrow$  43 (should result)
```

*** **FINAL conclusion:** the 42 RPMs in /home/cdsware/LATEST_CDSware_RPMs/ are required for successful installation of CDSware (or the required WML software). This is the breakdown:

```
6 MySQL 4.0.15 RPMs..... (see ~cdsware/for_MySQL-4.0.15-0/)
1 other MySQL-related RPMs... (see ~cdsware/for_CDSware_misc/MySQL-python-0.9.1-6.i386.rpm)
6 PHP 4.2.2 RPMs ..... (see ~cdsware/for_PHP-4.2.2-17.2/)
9 Apache httpd 2.0.40-related ..... (see ~cdsware/for_Apache-2.0.40-21.5)
2 other web-related RPMs ..... (webalizer-2.01_10-11.i386.rpm and mailman-2.1.1-4.i386.rpm)
7 Perl-related RPMs ..... (see ~cdsware/for_CDSware_misc/perl-*)
3 text-processing-related ..... (see antiword, html2txt, xlhtml in ~cdsware/for_CDSware_misc/)
8 other miscellaneous RPMs ... (see db4-devel, gdbm-devel, unixODBC, etc. in ~cdsware/for_CDSware_misc/)
```

```
#####
Sidenote 1: INSTALL WML (Website MetaLanguage):
#####
```

There are 2 methods to install WML:

- (i) compile/build/install directly from the source, or
- (ii) install from the RPM (/home/cdsware/wml-2.0.9-1.asp.i386.rpm)

The former approach seems to work better on chemcca42, so here is a description of how that was done (also see instructions in WML INSTALL file):

```
* download the WML 2.0.9 tarball from http://thewml.org/, save as
  /home/cdsware/wml-2.0.9.tar.gz
* cp /home/cdsware/wml-2.0.9.tar.gz /usr/local/src/.
* cd /usr/local/src/.
* tar zxvf wml-2.0.9.tar.gz
* cd wml-2.0.9
* ./configure --prefix=/usr/local/src/wml-2.0.9
* make          (⇒ may need to repeat this 2-3x for some quirky reason!)
* make test     (⇒ should pass all tests ok)
* make install  (⇒ apparently ok)
* finally, link the wml-2.0.9/bin/wm* executables to /usr/bin/:

  # ln -s /usr/local/src/wml-2.0.9/bin/* /usr/bin/.
```

NOTE: In earlier attempts, I tried installing wml-2.0.8.tar.gz and wml-2.1a2.tar.gz tarballs (because, for instance, Frank Sudholt said they used v2.0.8); however, these attempts failed for one unclear reason or another...

```
#####
Sidenote 2: INSTALL jpegsize:
#####
```

Download source for the 'jpegsize' application (from <http://www.squarebox.co.uk/download/>), and then compile jpegsize.c in /usr/local/src/jpegsize/. Place the output binary in /usr/local/bin:

```
# gcc -W -pedantic -o jpegsize jpegsize.c
# cp jpegsize /usr/local/bin/.
```

Test it on sample.jpg:

```
# jpegsize sample.jpg (⇒ returns '471 200')
```

```
#####
Configure the MySQL server and set-up an initial CDSware database:
#####
```

- When installing MySQL from the v4.1 RPMs, set some server-specific and global options by copying this default configuration file for large servers (note: this may be unnecessary):

```
# cp /usr/share/mysql/my-large.cnf /var/lib/mysql/my.cnf
# cp /usr/share/mysql/my-large.cnf /etc/my.cnf
```

NOTE: They are currently disabled on chemcca42 by renaming them `/etc/my.cnf.DISABLE`, *etc.*

- Restart the MySQL server with these options:

```
# /etc/init.d/mysql restart
```

- Set a passwd for MySQL 'root' user on localhost ('localhost' is the default server IP):

Initially, set it *via* `"# /usr/bin/mysqladmin -u root password '*****'"`

Thereafter, change it *via* `"# /usr/bin/mysqladmin -u root password 'new-password' -p"`

- Here is a transcript of how access was enabled for root@localhost, root@chemcca42(.ucsd.edu), root@132.239.16.202: [NOTE: these are not all the same in MySQL-speak!]:

```
[root@chemcca42 temp]# mysql -p
```

```
Enter password:
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
Your MySQL connection id is 12 to server version: 4.1.0-alpha-log
```

```
Type 'help;' or '\h' for help.  Type ';' to clear the buffer.
```

```
mysql> grant all privileges on * to root@chemcca42.ucsd.edu identified by '*****';
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> \q
```

```
Bye
```

```
[root@chemcca42 temp]# /etc/init.d/mysql restart
```

Repeat a similar procedure as needed for root@localhost, root@chemcca42, whatever. There is apparently no need to do it again for root@132.239.16.202 because it is properly dealt with by mysql's ip-translation modules...

```
***** NOTE *****
```

If things get really screwed-up with the MySQL privilege tables, passwords, *etc.*, try re-initializing things:

```
# /usr/bin/mysql_fix_privilege_tables -password '*****' (logged in as root)
```

```
# /usr/bin/mysql_install_db
```

```
# /etc/init.d/mysql restart
```

Now you should be able to log in again as root (`# mysql -p`)...

```
*****
```

- Finally, use the mysql client to configure the MySQL server for CDSware:

```
# mysql -h 132.239.16.202 -u root -p mysql
```

NOTE 1: You may have to omit the '-h 132.239.16.202' portion and just use the default 'localhost' on chemcca42...

NOTE 2: On cm1.mura.net, had to make changes in iptables and router config to allow MySQL server on port 3306

...and then:

```
mysql> CREATE DATABASE cdsware;
Query OK, 1 row affected (0.08 sec)

mysql>
mysql> GRANT ALL PRIVILEGES ON cdsware.* TO cdsware@chemcca42.ucsd.edu
IDENTIFIED BY '*****'
-> ;
Query OK, 0 rows affected (0.01 sec)
```

- Now you can see that the changes have taken effect:

```
... ..
[cdsware@chemcca42 cdsware]$ mysql -u cdsware -h chemcca42 -p cdsware
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9 to server version: 4.1.0-alpha-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+-----+
| Database |
+-----+
| cdsware  |
| test     |
+-----+
2 rows in set (0.00 sec)

mysql> \q
Bye
[cdsware@chemcca42 cdsware]$
... ..
```

- Another simple test of database accessibilities (chemcca42, chemcca42.ucsd.edu, 132.239.16.202 work equally well here):


```
[root@chemcca42 cdsware]# mysqlshow -h 132.239.16.202 -u root -p
Enter password:
+-----+
| Databases |
+-----+
| cdsware   |
| mysql     |
| test      |
+-----+
[root@chemcca42 cdsware]# mysqlshow -h 132.239.16.202 -u cdsware -p
Enter password:
+-----+
| Databases |
+-----+
| cdsware   |
| test      |
+-----+
[root@chemcca42 cdsware]#
```

- **Sidenote (unrelated to MySQL):** If you want to proceed with the user ‘cdsware’ being able to write to some of the required directories, do this:

```
as root:      # chgrp cdsware /var/www/htdocs /var/www/cgi-bin
(one time)   # chmod g+w /var/www/htdocs /var/www/cgi-bin
as cdsware:  # mkdir /var/www/htdocs/CDSware /var/www/cgi-bin/CDSware
(thereafter)
```

But, if you are doing this as a non-root user who does not have write permissions for /var/www/htdocs and /var/www/cgi-bin, then after the mkdir have a root user do this:

```
change groups: # chgrp -R cdsware /var/www/htdocs/CDSware /var/www/cgi-bin/CDSware
...and permissions: # chmod g+w /var/www/htdocs/CDSware /var/www/cgi-bin/CDSware
```

```
#####
```

Apache and PHP configuration and restart:

```
#####
```

- Modify `/etc/httpd/conf/httpd.conf` as instructed in the CDSware INSTALL file, or restore the old `httpd.conf.rpmsave` one (if Apache was updated *via* RPM installation after establishment of an edited conf file with which you were happy!). Here are the additions:

1. # CM: added these 2 lines for CDSware (09/15/2003):

```
AddType application/x-httpd-php .php .shtml
AddType application/x-httpd-php-source .phps .shtmls
```
2. `DirectoryIndex index.shtml index.php index.html index.html.var`
 #CM: added "index.shtml index.php" to above lines for CDSware...

- Modify the PHP config file (`/etc/php.ini`) as instructed by the CDSware INSTALL file; save the default one as `/etc/php.ini.OLD`. Here are some settings:

```
session.save_handler = user
session.auto_start = 0
session.name = SESSIONID
session.gc_probability = 0
session.gc_maxlifetime = 60000
session.cookie_lifetime = 0
log_errors = On
display_errors = Off
expose_php = Off
max_execution_time = 160
*** register_globals = On
*** register_argc_argv = On
```

*** **IMPORTANT NOTE:** `register_globals` and `register_argc_argv` must be set to 'On' in order for CDSware's PHP scripts (`bibupload`, `bibformat`) to work! *More generally, there seem to be many CDSware←→PHP interactions which may cause trouble...*

- Restart the httpd daemon ("`# /etc/init.d/httpd restart`"). At this point, you may find it necessary to downgrade Apache to `httpd v.2.0.40-21.5` for various module incompatibility issues with what's already installed. In chemcca42's case, this was done with the RPM files in `/home/cdsware/for_Apache-2.0.40-21.5/`

[NOTE: The `/home/cdsware/DO_NOT_USE_for_*` files (PHP and Apache) are versions of these softwares which were not used (successfully) in the end...]

```
#####
FINALLY, install CDSware in /usr/local/src/ (or /home/cdsware/cdsware-0.0.9/):
#####
```

- Some preliminaries: get the source, extract it, set some environment variables:

```
# cd /home/cdsware
# wget http://cdsware.cern.ch/download/cdsware-0.0.9.tar.gz
# tar zxvf cdsware-0.0.9.tar.gz
# cd /home/cdsware/cdsware-0.0.9
```

[Remember that if you didn't link the WML executables to `/usr/bin/` earlier, then you need to explicitly add them to `$PATH`:

```
# export PATH=$PATH:/usr/local/src/wml-2.0.9/bin
```

]

- The following **must be done** (in `/usr/local/src/cdsware`, `/home/cdsware/cdsware-0.0.9/`, wherever):

```
# cd where-ever-you-are-installing/
# mv .wmlrc .wmlrc.DEFAULT
# export WMLOPTS="-I'pwd'/config"    ('pwd' = /where-ever-you-are-installing/)
```

[This is because we do NOT want the WMLOPTS environment variable set to the abbreviated, non-full-path '-Iconfig', which is what happens by default because of the `.wmlrc` file that is extracted into the `cdsware-0.0.9/` directory when the `cdsware` tarball is unpacked. See WML documentation: <http://thewml.org/docs/>, particularly the WMLOPTS section in <http://thewml.org/docs/frontend/wml.html>]

- Finally, run the configuration and makefile steps in two stages:

```
./configure --with-htdocsdir=/var/www/htdocs/CDSware/      \
--with-cgibindir=/var/www/cgi-bin/CDSware/                 \
--with-dbutildir=/usr/bin/                                  \
--with-htdocsurl=http://132.239.16.202/CDSware/            \
--with-cgibinurl=http://132.239.16.202/cgi-bin/CDSware/   \
--with-dbhost=132.239.16.202                               \
--with-dbname=cdsware                                      \
--with-dbuser=cdsware                                      \
--with-dbpass=#$%^&*^*
```

NOTES for 1st stage:

1. You must include the `'/'` after the above `'htdocsurl'`, because the default CDSware configuration is buggy in its behavior – some sub-directories will properly work (*i.e.*, assume the forward slash – *e.g.*, <http://chemcca42.ucsd.edu/CDSware/admin>), whereas others will not work (*e.g.*, clicking on 'docs' sections leads to the junk URL <http://chemcca42.ucsd.edu/CDSwaredocs> ... instead of <http://chemcca42.ucsd.edu/CDSware/docs>).
2. In what follows, you must include the `'env LANG=C'` bit, otherwise UTF-8 fontset problems may ensue...

3. **Sidenote:** *customize before proceeding?*:

Now do some customization in the first part of the `config/config.wml` file — *e.g.*, prepend `'/var/'` to `RUNTIMELOGDIR`, giving it a value of `/var/log/cdsware/cdsdemoruntimelogs` and change its directory permissions to `a+rwx`, or else copy over already customized files from previous installs (in `~cdsware/cdsware-0.0.9/MCCAMMON_SPECIFIC_CDSware/`):

```
# mv config/config.wml config/config.wml.DEFAULT
# mv htdocs/img/cds.css.wml htdocs/img/cds.css.wml.DEFAULT
# cp ~cdsware/MCCAMMON_SPECIFIC_CDSware/config.wml.MCCAMMON      config/config.wml
# cp ~cdsware/MCCAMMON_SPECIFIC_CDSware/cds.css.wml.MCCAMMON    htdocs/img/cds.css.wml
```

Now, get on with the 2nd part:

```
#          env LANG=C gmake
#          env LANG=C gmake create-tables
# sudo /bin/env LANG=C /usr/bin/gmake install
# sudo /bin/env LANG=C /usr/bin/gmake load-sample-data
# sudo /bin/env LANG=C /usr/bin/gmake clean-sample-data
```

NOTES for 2nd stage:

1. The server should be up and running (*i.e.*, you should be able to point your browser to `http://132.239.16.202/CDSware`) after the `'gmake install'`. Do not `'clean-sample-data'` until you have at least checked that the server seems to be functioning properly.
2. FINALLY, after all this screwing around, it may be a **very good idea** to restart the Apache and MySQL servers (doing this may fix some problems you encounter due to caching of pages, *etc.*):

```
# /etc/init.d/httpd restart
# /etc/init.d/mysql restart
```

- **DONE!**

```
#####
```

Some postmortem...

```
#####
```

- `'egrep -R "ABOUT THIS SITE" *'` while in `/usr/local/src/cdsware-0.0.9/` (or `/home/cdsware/cdsware-0.0.9/` or wherever) to see where this "ABOUT THIS SITE" portal description (or navbar descriptions, or whatever else) line comes from. See below for a more detailed description of how to rationally alter portal boxes *via* the MySQL client...
- To keep just 1 document from each collection category instead of the total of 74 docs, `'gmake clean-sample-data'`, and then load a smaller set of sample data for sake of illustrating each collection:

```
# cd /home/cdsware/cdsware-0.0.9
# mv dbutil/samplebibdata.xml dbutil/samplebibdata.xml.DEFAULT
# cp ~cdsware/MCCAMMON_SPECIFIC_CDSware/samplebibdata.xml.MCCAMMON dbutil/samplebibdata.xml
```

NOTE: The smaller sample data file (`~cdsware/MCCAMMON_SPECIFIC_CDSware/samplebibdata.xml.MCCAMMON`) was created by `'vi samplebibdata.xml'` and then editing it to remove 68 entries, thus leaving 6 behind (1 per default category)...:

- × Articles - 11 (58,57,9,8,7,6,5,4,3,2,1) (NOTE- subfield codes may be cross-referenced)
- × Preprints - 32 (67,....,59, 10,....,32)
- × Books - 14 (54,53,52,....,41)
- × Theses - 8 (40,....,33)
- × Reports - 2 (56,55)
- × Pictures - 7 (74...68)

- After changing the number of records like this, you may like to update the Makefile (although it doesn't seem necessary): `'vi /home/cdsware/cdsware-0.0.9/Makefile'` and change the `'load-sample-data'` target so that it contains the line `./dbutil/bibwords add 1-6` (not 1-74!). This seems unnecessary because $6 < 74$...
- And finally, after this further layer of screwing-around, re-run `'sudo /bin/env LANG=C /usr/bin/gmake load-sample-data'` to actually load the new data into the SQL database, and then run the `'env LANG=C create_collection_pages'` Python script to update the web-space (this script may be found in `/home/cdsware/cdsware-0.0.9/cgibin/` and `/var/www/cgi-bin/CDSware/`). Or, better yet, just run the `~cdsware/bin/update_collections.sh` script to update (it takes care of the WMLOPTS and LANG environment vars)...
- On the issue of updates *via* CVS, see <http://cdsware.cern.ch/download/>. Being inexperienced with CVS, here's one naive way to go about it(?): pull the cvs into `/usr/local/src/cdsware` *via* `'cvs co cdsware'`, and then `'cp -u -r cdsware/* cdsware-0.0.9/.'` (the `-u` switch ensures that only the more recent version of corresponding filenames are copied over)... This issue could use some work.

```
#####
RE-DOing the installation as a non-root user:
#####
```

Because of future security/administration concerns, you may wish to re-do this installation as the non-root user 'cdsware'. Here's how that may differ from what has been already described...

1. First create the 'cdsware' group and then add the users 'cdsware' and 'root' to this group. Then make 'cdsware' the primary group for user 'cdsware' (rather than the 'users' group, which is the current primary). All of this may be done very easily by using, *e.g.*, the `redhat-config-users` gui tool...
2. Allow the user 'cdsware' to write-out the 'dbexec' and 'bib*' executables to `/usr/bin/`. Accomplish this by either:

(a) directly specifying `'dbutildir=/usr/bin/'` and then do the `'gmake install'` step *via* 'sudo' (or as root!)

or...

(b) specifying `'dbutildir=/home/cdsware/bin/'` in the config step (or whatever dir 'cdsware' can write to), so then you can `'gmake install'` without having to sudo. [Note: if this dir is not `$HOME/bin/`, then make sure to update `$PATH` in `~cdsware/.bash_profile`.] THEN, `sudo cp` the 5 executables (`bibconvert`, `bibupload`, `bibformat`, `bibwords`, `dbexec`) in `/home/cdsware/bin/` to `/usr/bin/`.

***** NOTE:** It is not good enough to simply write the executables to `/home/cdsware/bin/` (or wherever) and then use root or sudo to `'ln -s /home/cdsware/bin/* /usr/bin/'` — I found that troubles may arise when you try to submit documents later on! Also, there are other troubles and concerns with option (b) which I haven't tried to figure-out. Therefore, go with option (a)!

Execute option (a) by including the following lines in `/etc/sudoers` (which is edited *via* 'visudo'):

```
cdsware chemcca42=(root) /bin/env LANG=C /usr/bin/gmake install
cdsware chemcca42=(root) /bin/env LANG=C /usr/bin/gmake load-sample-data
cdsware chemcca42=(root) /bin/env LANG=C /usr/bin/gmake clean-sample-data
cdsware chemcca42=(root) /bin/env LANG=C /var/www/cgi-bin/CDSware/create_collection_pages +
```

...and then execute actual commands like so:

```
# sudo /bin/env LANG=C /usr/bin/gmake install
etc...
```

Execute option (b) by including following lines in `/etc/sudoers` ('visudo'):

```
cdsware chemcca42=(root) /bin/cp /home/cdsware/bin/bib* /usr/bin/.
cdsware chemcca42=(root) /bin/cp /home/cdsware/bin/dbexec /usr/bin/.
```

```
#####
```

CDSware post-installation customization:

```
#####
```

- For general administration, go to `http://66.75.19.109/CDSware/admin/`, login as the CDSware admin (*e.g.*, `jam_cdsware@mccammon.ucsd.edu`), set passwd to `'@#%^&*'` (Note: this shouldn't be sensitive, as it is not encrypted in SQL database). Create/register user accounts such as `jmccammon@ucsd.edu` (passwd `'~%$%^$#'`)...
- User note: You must explicitly press the "logout" button to logout of CDSware – it's very sensitive about this... Also, you can't have more than one independent login – if you login on one browser window and spawn any new ones, then you're automatically logged-in as the same instantiation of the original...
- After making changes to the CDSware configuration, implement them (*i.e.*, write-out new `/var/www/htdocs/CDSware/index.shtml`, *etc.* files) *via*

```
# sudo /bin/env LANG=C /var/www/cgi-bin/CDSware/create_collection_pages +
    (the '+' makes it recurse)
```

or, equivalently,

```
# sudo /bin/env LANG=C /var/www/cgi-bin/CDSware/create_collection_pages ""+
    (the "" can be left out as above)
```

When doing this, make sure `WMLOPTS` is set to `/home/cdsware/cdsware-0.0.9/config`, as described in the installation instructions above. Also, do it with `'env LANG=C'` to avoid "Malformed UTF-8 character" errors (in other words, you could just run the `~/cdsware/bin/update_collections.sh` script mentioned earlier!).

- As alluded to above, you may want to edit these 2 files during installation (after `'./configure'`, before `'gmake'`):

```
config/config.wml (original version saved as config.wml.DEFAULT) –
    ⇒ add link to McCammon group page, edit email addresses, etc.
htdocs/img/cds.css.wml (original saved as cds.css.wml.DEFAULT) –
    ⇒ change colors to complement of default (i.e., dark backgrnd with light txt)
```

The following files are saved for future use in `/home/cdsware/MCCAMMON_SPECIFIC_CDSware/`:

```
cds.css.wml.MCCAMMON
config.wml.MCCAMMON
```

In general, when changing any files (*e.g.*, "whatever.wml") from the CDSware source tarball, **preserve the original by appending ".DEFAULT"** to a cp of the original (*e.g.*, "whatever.wml.DEFAULT").

- Other `.wml` files eventually need editing (of font-colors, primarily) to work with the new CSS color motif (dark background/light foreground). This can be achieved using the two homemade scripts `"invert_html_colors.pl"` and `"invert_many_wmls.bash"`, as shown in following example:

(1) Say we want to fix the colors for all files within the following hierarchy:
 McCammon Group Document Server > Administrator's Area > WebSubmit Admin > what-ever-section...

(2) `# cd /home/cdsware/cdsware-0.0.9/htdocs/admin/websubmit/`

(3) run `'#env LANG=C invert_many_wmls.bash'` within this directory, and answer 'y' to whichever *.wml files you would like to be edited (or hit Enter to skip, or enter 'a' initially to do them all *en masse*)...

(4) descend into any subdirectories which also contain *.wml files and run the script there too (so `'# cd docs/english'; '# ./invert_many_wmls.bash'`)

(5) after this, all the original *.wml files should be saved as *.wml.DEFAULT, and the new ones (with corrected color tags) replace the original *.wml

[NOTE 1: Use the script `~/bin/rename_DEFAULT_files.bash` to mv all original `./*.wml.DEFAULT` files back to `./*.wml`]

[NOTE 2: Other examples are within the `htdocs/admin/webaccess/` hierarchy... (`superusers.wml`, *etc.*)]

[NOTE 3: In some instances, the HTML color specification is present in the *.wml file, but in an irregular/un-predictable format – for instance, in the `htdocs/admin/bibformat/BIBREFORMAT_tree.shtml` file, at ~line 128, where the color depends on row # (even or odd). In such cases, manually change it... (until the `invert_html_color.pl` script is eventually improved!)]

[NOTE 4: The modified forms of some of the *.wml files (those which require more effort to edit) are saved in `~/cdsware/MCCAMMON_SPECIFIC_CDSware/.`]

After these changes, make sure to re-run 'gmake' and 'gmake install' in `~/cdsware/cdsware-0.0.9/` base dir, in addition to the usual `update_collections.sh`.

- Set-up automatic update of the Document Server collection pages, front page, *etc.* by adding the script `'/home/cdsware/bin/update_collections.sh'` to root's crontab:

```
# crontab -e
```

and then enter this:

```
# Update the McCammon Group Document Server collections every 10
# minutes:
*/10 * * * * /home/cdsware/bin/update_collections.sh >/dev/null 2>&1
```

To verify this rigamarole: `'# crontab -l'`


```
#####  
Some random MySQL stuff:  
#####
```

- Inspecting the sbmDOCTYPE table:

```
show databases;  
use cdsware      (select cdsware database)  
show tables like "sbm%";  
select * from sbmDOCTYPE;  
describe sbmDOCTYPE;
```

- See who has created user accounts and describe the 'user' table:

```
# mysql -h chemcca42 -u cdsware -p  
    show databases;  
    use cdsware  
    show tables;  
    select * from user;  
    describe user;  
    \q
```

- Another example: playing around with the 'portalbox' table from the 'cdsware' database:

```
# mysql -h chemcca42 -u cdsware -p  
then, in mysql:  
    use cdsware  
    show tables;  
    select * from portalbox;      (to show this table in its entirety)  
    describe portalbox;         (to describe the structure of this table)  
    select * from portalbox where title = "test";  
                                (to show the entire row that has title = 'test')  
    select id from portalbox ORDER BY id DESC;  
                                (show all rows of portalbox, sorted by 'id' column in descending order)
```


Change user and admin privileges in the WebAccess Admin area:
 #####

- Go to <http://chemcca42.ucsd.edu/CDSware/admin/> and login as `jam_cdsware@mccammon.ucsd.edu`
- Click on the WebAccess Admin, and then, if necessary, go in a shell to `~cdsware/cdsware-0.0.9/htdocs/admin/webaccess/` and customize all of the `.wml` files by running `'invert_many_wmls.bash'` (answer 'y' to each); run `'gmake', 'sudo /usr/bin/gmake install', etc.`, as described above...
- Manually set the following privileges and permissions in the WebAccess Admin:

⇒ **Superusers:** + `jam_cdsware@mccammon.ucsd.edu`

⇒ Authorizations on **Submit** (for the following doctypes):

	viewers	users
all catalogues	+ everyone	+ jam_cdsware@mccammon.ucsd.edu
picture document	+ everyone	+ jam_cdsware
picture doc (refereed)	+ everyone	+ jam_cdsware
text document	+ everyone	+ jam_cdsware, + jmccammon@ucsd.edu
text doc (refereed)	+ everyone	+ jam_cdsware, + jmccammon@ucsd.edu
library book	+ everyone	+ jam_cdsware, + jmccammon@ucsd.edu

NOTE 1: see below for instructions on how the "library book" doctype was created

NOTE 2: all of the 'users' additions were authorized for 'All actions' (*i.e.*, Approve a Document, MBI, SBI, SRV, FTT)

⇒ Authorizations on **WebSubmit Admin** (for the following doctypes):

```
all catalogues:      + jam_cdsware
picture document:   + jam_cdsware
picture doc (refereed): + jam_cdsware
text document:      + jam_cdsware
text doc (refereed): + jam_cdsware
library book:       + jam_cdsware
```

⇒ Authorizations on **WebSearch Admin:** + `jam_cdsware`

⇒ Authorizations on **WebAccess Admin:** + `jam_cdsware`

⇒ Authorizations on **BibData Admin:** + `jam_cdsware`

⇒ Authorizations on **BibFormat Admin:** + `jam_cdsware`

#####

How the 'LIBB' doctype was created:

#####

(This was largely an exercise in reverse-engineering from the CDSware source.)

- Login as admin, go to **WebSubmit Admin**, click New Doctype
- Fill in these details:

Document Type Name: library book
 Document Type ID: LIBB
 Doctype Description: A book or thesis from the McCammon Group library
 ⇒ clone it from the TEXT doctype

- Ignore the errors about not being able to create links to `go.shtml`, and do this instead (as root):

```
# cd /var/www/htdocs/CDSware/submit/access/protected
# ln -s /var/www/htdocs/CDSware/submit/access/protected/go.shtml TFULIBB.shtml
# ln -s /var/www/htdocs/CDSware/submit/access/protected/go.shtml SRVLIBB.shtml
# ln -s /var/www/htdocs/CDSware/submit/access/protected/go.shtml SBILIBB.shtml
# ln -s /var/www/htdocs/CDSware/submit/access/protected/go.shtml MBILIBB.shtml
# ln -s /var/www/htdocs/CDSware/submit/access/protected/go.shtml FTTLIBB.shtml
```

These errors are due to a missing 'submit' in the directory path above, and should be easy to fix in the CDSware source code (later!). You need to make these symlinks for the various actions to work (SBI, MBI are particularly important at this stage!)

- If not already there, point the web interface to → WebSubmit Admin → Details of the LIBB document type, and then:

- * VIEW CATEGORIES ⇒ delete everything except for BOOK and THESIS
- * SIMPLE APPROVAL REFEREES ⇒ add `jam_cdsware@mccammon.ucsd.edu`
 [sidenote: if it hasn't been done already, go to the WebAccess Admin area and add `jam_cdsware@mccammon.ucsd.edu` and `jmccammon@ucsd.edu` for LIBB, as for TEXT doctype]
- * Delete the SRV and FTT actions from the list (keep SBI, MBI)
- * Edit the SBI action by adding functions ('Edit Functions') to get this configuration:

function	score	step

Create_Base_Files	10	1
Report_Number_Generation	20	1
Make_Weplib_Record	30	1
Format_Record	40	1
Insert_Record	50	1
Mail_Submitter	70	1
Print_Success_FTT	90	1
Move_to_Done	110	1

NOTE: In order for this non-FTT version of SBI to work, it is *essential* that all the steps be '1' and scores increase in this order!!

- * Also, edit some of the functions as follows:

1. edit the "Create_Base_Files" function so that the param 'base' has value 'BOOK'
2. edit the "Report_Number_Generation" function so that: param 'counterpath' has value 'lastid_test_LIBB'; param 'rnformat' has value 'TEST-LIBB-`<PA>yy</PA>`'; param 'rnin' has value 'comboLIBB'
3. edit the "Make_Weblib_Record" function so that the 'mysqlInsert' and 'source-Submit' params have values of 'EDSLIBBmysql.tpl' and 'EDSLIBB.tpl', respectively

* In `/var/www/htdocs/CDSware/submit/bibconvert/config`, as 'root', do this:

```
# cp EDSTEXT.tpl EDSLIBB.tpl
# cp EDSTEXTmysql.tpl EDSLIBBmysql.tpl
```

Edit these two files (*e.g.*, delete 'linkmanager' line), and place copies in `~cdsware/MCCAMMON_SPECIFIC_CDSware/`

- Go to WebSubmit Admin → Organize Main Page and make changes to Catalog names (*e.g.*, change from "Catalogues" to "McCammon Group Catalogs", add the "library book" catalog as an option,...)

- Go to WebSubmit Admin → Available Element Descriptions and do this:

⇒ Create the "Check_out" Form Element (check that MARC code 689__* isn't already in use):

```
Element Type:  Select Box
Element:       Check_out
Aleph Code:    <empty>
Marc Code:     689__z
Item Description:  <select name="Check_out">
                   <option value="NO">NO
                   <option value="YES">YES
                   </select>
Cookie: No
```

⇒ Create the "Who_check_out" Form Element:

```
Element Type:  Text Input
Element:       Who_check_out
Aleph Code:    <empty>
Marc Code:     689__y
Size:          30
Maxlength:     50
Value:         <empty>
Cookie: No
```

- Edit the Submission Pages for the SBI action of the LIBB doctype too, removing the default page 3 (abstract), adding fields for the 'Check_out' and 'Who_check_out' form elements, *etc.* For example, on SBILIBB Page 3 have this set of 5 "Elements on this Page":

Item#	Name	Page#	Element Label	Level	Short Desc.	Check	Creat.Date	Mod.Date
1	NO	3	<i>Optional</i>...	0	Comments		2003-11-10	2003-11-10
2	Rep	3	 <i>Option...	0	Report Number(s)		2003-11-10	2003-11-10
3	Check_out	3	 Is this book...	M	check-out status		2003-11-10	2003-11-10
4	Who_check_out	3	 If it is...	0	who checked-out		2003-11-10	2003-11-10
5	EndButton	3	 	0	etc... etc...			

- Make the new library book catalog available for browsing and searching by going to the WebSearch Admin frontpage (McCammon → Administrator's Area → WebSearch Admin) and "ADD NEW COLLECTION"; then play with hierarchy, *etc.* to achieve this overall structure {"narrow search" appearance}:

<long name>	<associated collection id>		
Group Publications	ARTICLE		
Suggested Reading	S_READING	←	"clone from" 'Articles'
Library Books	BOOK	}	
Preprints	PREPRINT	}	← these 5 + 'Group Publications'/ARTICLE are
Theses	THESIS	}	unaltered from the installation's <code>load-sample-data</code>
Reports	REPORT	}	step, except for <longname>s
Pictures	PICTURE	}	(<i>e.g.</i> , 'Articles' → 'Group Publications')

- Go to each of the TEXT, RTEXT, PICT, RPICT doctypes in the Administrator's Area → Web-Submit Admin, then into the "VIEW CATEGORIES", and then add/remove/edit names to reflect changes, as described earlier for the new LIBB book doctype — *e.g.*, for the simple TEXT type, have this set-up:

Category ID	Category Description	
ARTICLE	Group Publications	
PREPRINT	Preprint	(in other words, removed the BOOK and THESIS Categ.IDs,
REPORT	Report	because these are now associated with the LIBB doctype...)
S_READING	Suggested Reading	

- Finally, make the required changes in the Administrator's Area → BibFormat Admin:

Extraction Rules:

Add the \$689 datafield tag, with subfields z (for check-out status) and y (for Who_check_out). End-up with this:

Varname	Mapping Tag	Allow Multiple values
689	<datafield tag="689" ind1="" ind2="">	Yes

with the subfields SF Name 'y' and 'z', with Mapping Tags <subfield code="y"> and <subfield code="z">

Formats:

Edit the code for the DEFAULT_HTML_BRIEF format so that the check-out status (and email id of whoever checked it out) will be displayed when a user browses the Search front-page... Accomplish this by adding the following after the last default line (*i.e.*, after the 'forall(\$8564.u)' bit):

```
# if($689.z="YES")
# {
# forall($689.y)
# {
#   if ($689.y!="")
#   { "<br /><small class=quicknote> [Currently CHECKED-OUT by <a href='mailto:'$689.y''>' $689.y '</a>.]</small>"}
#   else { " <br /><small class=quicknote> [Currently CHECKED-OUT by UNKNOWN]</small> " }
# }
# }
# else { " <br /><small class=quicknote> [Currently AVAILABLE]</small> " }
```

After doing this, go to the Admin Area → BibFormat Admin → Reformat Records → Reformat Collections and then select whichever collection (*e.g.*, "Library Books") to reformat... Then,

after crontab runs the ‘update_collections.sh’ script, everything should be updated...

NOTE: Entries submitted in the future should be taken care of automatically – *i.e.*, formatted according to this new DEFAULT_HTML_BRIEF style.

- One more thing: make entries of the LIBB doctype modifiable by doing a few things:

⇒ In the Admin area → WebSubmit Admin → Available modification texts, create the following new mod. texts, if they do not already exist:

```
AF (for affiliation)    - <br />Please correct the author affiliations:<br />
Check_out              - <br />Please update the check-out status:<br />
Who_check_out          - <br />Please update the email address of the person who has checked-out this item:<br />
Num (for no.pages)    - etc...
```

⇒ Go in to Admin area → WebSubmit Admin → Details of the LIBB document type and edit the MBI action (analogous to what was done with the SBI action earlier):

Edit Submission Pages:

MBILIBB page 2: use the CANChange2 element (not the pre-filled CANChange), make Level ‘Optional’, and add this Element Label:

```
<br /><br />
Please select all the fields that you would like to modify:
<br /><span style="font-size: smaller;">
[NOTE: Due to a current bug, <span style="color: red;">please select ALL of these</span>,
even if you intend to modify only some of them!]
</span><br />
```

NOTE: Actually, this bug was fixed by making the changes to EDS***mysql_Modify.tpl described on pgs. 27-28. Once this bug was fixed, I replaced this element label with the following description:

```
<br /><br />
Please select the fields that you would like to modify:
<br /><span style="font-size: smaller;">
[NOTE: You may <span style="color: red;">select more than one</span> by dragging the
mouse over many choices while pressing the left mouse button.]
</span><br />
```

Edit Functions:

Create_Modify_Interface → change to CANChange2

Create_Base_Files → param base should be ‘BOOK’

Make_Modify_Record → mysqlModify = EDSLIBBmysql_Modify.tpl ; sourceModify = EDSLIBB.tpl

(created the EDSLIBBmysql_Modify.tpl file similarly as for EDSLIBBmysql.tpl, by copying and editing EDSTEXTmysql_Modify.tpl, *etc.* – see the ~cdsware/MCCAMMON_SPECIFIC_CDSware/EDSLIBBmysql_Modify.tpl)

How to add output formats (HTML brief, HTML detailed, *etc.*) for a new collection (*e.g.*, "Suggested Reading"), so as to permit 'reformat collections' type of operations:
 #####

- See the email exchange (11/14/2003) between cmura and Tibor Simko (tibor.simko@cern.ch) for a description of the problem and its solution:

The problem:

```
> (1) I added a new collection (long name: "Suggested Reading") by
> going to the WebSearch Admin and cloning from the "Article" type
> that comes with the CDSware demo installation (i.e. from doing
> "gmake load-sample-data"). Then I gave the "associated id" and
> "associated query" both "S_READING". Now, I'm wondering how I could
> add the "HTML brief", "HTML detailed", etc. output formats to the
> options in the BibFormat page ?? In other words, when I go to the
> BibFormat Admin -> Reformat Records -> Reformat Collections, I see a
> table with the columns for all of the "Collection name" and "Output
> Format Selection", but the new "Suggested Reading" collection does
> not have any of these boxes (I looked around in the BibFormat Admin
> -> Behaviours section, but didn't see anything obvious)... This is
> probably something very simple which I'm missing in my understanding
> ??
```

...and its solution:.....

I assume you are using 0.0.9. The safest way would be to add these output formats manually, i.e. by launching direct SQL commands.

First you look up the collection ID of 'Suggested Reading':

```
mysql> select id,name from collection where name='Suggested Reading';
```

Let's assume it's 9.

Then you look up format IDs:

```
mysql> select * from format;
```

Then you insert the ones you want into 'collection_format' table:

```
mysql> insert into collection_format values \
(9,1,100),(9,2,90),(9,3,80),(9,4,70),(9,5,60);
```

see also the existing content of this table.

After that you should be fine.

#####

How to customize some portal boxes:

#####

- There is not a clear link into this area from the existing CDSware installation, but I found that URLs of the form `http://132.239.16.202/CDSware//admin/websearch/editMenu.shtml?id='Y'` permit customization of many portal/navbar boxes, for the following values of Y:

Y	box description
1	"ABOUT THIS SITE"
2	"SEE ALSO" (for ABOUT THIS SITE, i.e., frontpage)
3	"ABOUT ARTICLES"
4	"SEE ALSO" (for ARTICLES)
5	"ABOUT PREPRINTS"
6	"SEE ALSO" (for PREPRINTS)
7	"ABOUT BOOKS"
8	"SEE ALSO" (for BOOKS)
9	"ABOUT THESES"
10	"SEE ALSO" (for THESES)
11	"ABOUT REPORTS"
12	"TEST portal box" (for testing)
13	"test" (for testing)
14	"ABOUT PICTURES"

- Edit title of the id=13 portal box (*i.e.*, the "test" one) by going into MySQL and changing some fields. Do the following in the MySQL monitor:

```

> use cdsware
> select * from portalbox;
> describe portalbox;
> UPDATE portalbox SET title = "<small>ABOUT SUGGESTED READING</small>" WHERE id = 13;
verify: > select * from portalbox;

```

- As usual, run `'~/cdsware/bin/update_collections.sh'` to update everything immediately afterwards (if you're too impatient to wait for < 10 minutes for cron to do its thing)...


```
#####
```

Changes to the overall McCammon webservice after CDSware is up-and-running:

```
#####
```

- Add a link under the ‘Miscellaneous’ section of the group’s Local pages (/net/home/www/htdocs/local/index.html):

```
<li>The <a href="http://mccammon.ucsd.edu/CDSware">McCammon Group Document Server</a></li>
```

- Add the following 3 lines to the overall/global httpd conf for McCammon webservice (chemcca51:/usr/local/apache/conf/httpd.conf):

```
# Redirection for CDSware document server
RedirectMatch permanent ^/CDSware(.*) http://132.239.16.202/CDSware$1
RedirectMatch permanent ^/cdsware(.*) http://132.239.16.202/CDSware$1
```

- To notify users that server is down for upgrades/etc. on chemcca42, first comment-out the crontab line that runs the ‘update_collections.sh’ script (which writes out a new index.shtml ever 10 minutes) by running ‘crontab -e’, and then do this:

```
# cd /var/www/htdocs/CDSware
# mv index.shtml index.shtml.DEFAULT
# cp /home/cdsware/MCCAMMON_SPECIFIC_CDSware/cdsware_down_now.html index.html
```

or just append this filename (“cdsware_down_now.html”) to the DirectoryIndex line in the httpd.conf file. . .

#####

Miscellaneity, and some random troubleshooting scenarios:

#####

- The UC's Melvyl library catalog system also uses MARC tags for entries that are found as search results. Example? Do an ISI Web of Science search for author Mura C*; click on the 'UC-eLinks' for the entry "Structure and assembly of an augmented..."; click on "See if you campus...Full Record on MELVYL" to pull-up the Melvyl entry; click on Display: Long, then the 'Full Record' link; and then, finally, the 'Display: MARC' link to see the tagged file..
- To manually reset the count (Y) of TEST-XXXX-2003-00Y ← look in the:
/var/www/htdocs/CDSware/submit/counters/lastid_test_XXXX files...

Look in here too:

```
/var/www/htdocs/CDSware/submit/storage/done/modify/LIBB/
/var/www/htdocs/CDSware/submit/storage/done/submit/LIBB/
```

- Note that there are at least two ways to update the file entry XML/Marc21/OAI fields – (i) using the BibFormat Edit tool or (ii) the Submit → Modify → *etc.* tool (assuming any bugs with Modify have been fixed)!
- Copy the 3 EDSLIBB* files that were originally created in /var/www/htdocs/CDSware/submit/bibconvert/config/ (by copying and editing the corresponding EDSTEXT* files) into /home/cdsware/MCCAMMON_SPECIFIC_CDSware/ :

```
EDSLIBB.tpl
EDSLIBBmysql.tpl
EDSLIBBmysql_Modify.tpl
```

(Actually, one could imagine including these files (and their .wml precursors?!) in the proper CDSware install sub-directory (*e.g.*, /home/cdsware/cdsware-0.0.9/htdocs/submit/bibconvert/config/)...

- Due to problems with maximum uploadable filesize, errors occurred when trying to upload a 3.9M PDF file into the Suggested Reading collection (for the entry TEST-S_READING-2003-005). Here is part of an email exchange (11/12/2003) between cmura and Thomas Baron (thomas.baron@cern.ch) regarding how to fix this:

The problem:

```
> (2) When I try to upload a very large (3.9 Mb) PDF file to our server
> through the usual Submit -> text document, the submission fails in an
> unusual way. I go through page: 1, 2, 3, 4 as usual, and then click the
> "finish submission" EndButton. Then I try to upload the file by clicking on
> the link to "Upload your files, if you have local access...", select PDF,
> "Browse" to get the path to the main document file, and then press "finish
> selection". Now the unusual part -- it advances to the next page (i.e., the
> "end of action" tab, where you can enter additional files) for less than 1
```

> second, and then returns to the "page: 1" tab ("Please choose the format of
> your document" page), except that the values I just entered are reset to
> null (the path to main document file is clear)... I thought this might have
> something to do with the Test_File_Size function that is in the FTT action
> of the TEXT doctype; however, when I change the 'maxsize' param of this
> function to a very large value (e.g., 20,000,000), this upload error still
> occurs. The error occurs even if I delete all of the Test_File_* functions
> from the TEXT's FTT action. This problem also occurs with other large PDF
> files, so I don't think a corrupt PDF file is the problem...

...and its solution:.....

The problem might come from your php configuration.
The following variables should be set in your php.ini file:
upload_max_filesize = 100000000
post_max_size = 100000000

So, this problem was solved by:

1. If it has not been done already, increase the value of the maxsize param of the FTT action for the TEXT doctype. How? Go to the Admin Area → WebSubmit Admin → Deatils of the TEXT doctype → Edit Functions for the FTT Action Name, then click on the Test_File_Size function, edit the maxsize param...
 2. Edit the following 2 lines from the /etc/php.ini file:
upload_max_filesize = 2M → upload_max_filesize = 10M
post_max_size = 8M → post_max_size = 10M
 3. Restart the httpd server ('/etc/init.d/httpd restart')
 4. Login to the Document Server as privileged user; go to the frontpage, then Submit → text doc → Suggested Reading (Submit a revised version); enter TEST-S_READING-2003-005, select "new version - old versions are removed" type of revision, and then give path to revised file (*i.e.*, the large, 3.9M PDF file); finally, now force update with ~/bin/update_collections.sh. Check the Suggested Reading collection to verify your changes.
 5. As usual, remember to save copies of customized/modified conf files (/etc/php.ini and /etc/httpd/conf/httpd.conf) in chemcca42:~cdsware/MCCAMMON_SPECIFIC_CDSware/.
- To allow modification of entries, you must fix a bug in the bibconvert EDS***mysql_Modify.tpl configuration files that come with CDSware v0.0.9 (look in /var/www/htdocs/CDSware/submit/bibconvert/config/ for these files). The reason for this has to do with the MINLW() function which is used to output the XML and update the MySQL entry (see the 11/13/2003 email exchange between cmura and thomas.baron@cern.ch). Change the contents of these files as illustrated by this EDSTEXT* example:

OLD (default) EDSTEXTmysql_Modify.tpl file:

```

START---<record>
SN---<controlfield tag="001"><:SN::SN:></controlfield>
RN::MINLW(74)---<datafield tag="088" ind1="" ind2=""><subfield code="a"><:Rep*::Rep:></subfield></datafield>
TI::MINLW(74)---<datafield tag="245" ind1="" ind2=""><subfield code="a"><:TI::TI:></subfield></datafield>
AU::MINLW(105)---<datafield tag="100" ind1="" ind2=""><subfield code="a"><:AU*::AU:></subfield><subfield
code="u"><:AF*::AF:></subfield></datafield>
NUM::MINLW(74)---<datafield tag="300" ind1="" ind2=""><subfield code="a"><:Num::Num:><:Num::Num::IF(, ,p):>
</subfield></datafield>
AB::MINLW(74)---<datafield tag="520" ind1="" ind2=""><subfield code="a"><:AB::AB:></subfield></datafield>
NO---<datafield tag="500" ind1="" ind2=""><subfield code="a"><:NO::NO:></subfield></datafield>
KW::MINLW(74)---<datafield tag="650" ind1="" ind2=""><subfield code="a"><:KW*::KW:></subfield></datafield>
LN::MINLW(74)---<datafield tag="041" ind1="" ind2=""><subfield code="a"><:LANG::LANG::IF(,eng,ORIG):></subfield>
</datafield>
FMT---<datafield tag="FMT" ind1="" ind2=""><subfield code="f">hd</subfield><subfield code="g"></subfield></datafield>
FMT---<datafield tag="FMT" ind1="" ind2=""><subfield code="f">h</subfield><subfield code="g"></subfield></datafield>
END---</record>

```

NEW (modified) EDSTEXTmysql_Modify.tpl file:

```

START---<record>
SN---<controlfield tag="001"><:SN::SN:></controlfield>
RN::MINLW(81)---<datafield tag="088" ind1="" ind2=""><subfield code="a"><:Rep*::Rep:></subfield></datafield>
TI::MINLW(81)---<datafield tag="245" ind1="" ind2=""><subfield code="a"><:TI::TI:></subfield></datafield>
AU::MINLW(111)---<datafield tag="100" ind1="" ind2=""><subfield code="a"><:AU*::AU:></subfield><subfield
code="u"><:AF*::AF:></subfield></datafield>
NUM::MINLW(81)---<datafield tag="300" ind1="" ind2=""><subfield code="a"><:Num::Num:><:Num::Num::IF(, ,p):>
</subfield></datafield>
AB::MINLW(81)---<datafield tag="520" ind1="" ind2=""><subfield code="a"><:AB::AB:></subfield></datafield>
NO::MINLW(81)---<datafield tag="500" ind1="" ind2=""><subfield code="a"><:NO::NO:></subfield></datafield>
KW::MINLW(81)---<datafield tag="650" ind1="" ind2=""><subfield code="a"><:KW*::KW:></subfield></datafield>
LN::MINLW(81)---<datafield tag="041" ind1="" ind2=""><subfield code="a"><:LANG::LANG::IF(,eng,ORIG):></subfield>
</datafield>
FMT---<datafield tag="FMT" ind1="" ind2=""><subfield code="f">hd</subfield><subfield code="g"></subfield></datafield>
FMT---<datafield tag="FMT" ind1="" ind2=""><subfield code="f">h</subfield><subfield code="g"></subfield></datafield>
END---</record>

```

NOTE 1: The most important change here is an increase of the MINLW parameter values, which sets the minimum line length in the XML record that is created. If this value is too low, then an empty line is created:

```
<datafield ...><subfield ...></subfield></datafield>
```

Then, the empty line is inserted into the MySQL database entry, replacing the original value; you can see this by looking at the contents of the "recmysql" executable that gets put into, *e.g.*, `/var/www/htdocs/CDSware/submit/storage/done/modify/LIBB/TEST-BOOK-2003-###.tar.gz` (unpack it someplace and have a look!)

NOTE 2: Any changes made to a `EDS***mysql_Modify.tpl` file that comes with the CDSware installation (*e.g.*, `EDSTEXTmysql_Modify.tpl`, *versus* homemade ones like `EDSLIBBmysql_Modify.tpl`) in `/var/www/htdocs/CDSware/submit/bibconvert/config/` will be ruined the next time a 'gmake'/'gmake install' is done, because these files will be recreated by the install step. To avoid this, either:

(a) make the changes directly to the `EDS***mysql_Modify.wml` source files in the install directory (`~cdsware/cdsware-0.0.9/htdocs/submit/bibconvert/config/`),

... or ...

(b) `cp -f` to force over the modified files from `~cdsware/MCCAMMON_SPECIFIC_CDSware/` after any 'gmake install' run

Option (a) was implemented on chemcca42 by editing the four files:

`EDSTEXTmysql_Modify.wml`, `EDSRTEXTmysql_Modify.wml`

`EDSPICtmysql_Modify.wml`, `EDSRPICtmysql_Modify.wml`

(`EDSLIBBmysql_Modify.tpl` was copied over from `~cdsware/MCCAMMON_SPECIFIC_CDSware/.`)

- **Problem:** someone entered a new TEXT submission to the "Suggested Reading" collection, but pasted in an abstract containing lots of non-standard/meta-characters crap — so the entry (submitted *via* the usual WebSubmit interface) was never incorporated. Why?? Turns out this corrupt abstract caused `bibformat` to output an empty "recmysqlfmt" XML file (presumably in the '`bibformat < recmysql > recmysqlfmt`'), as was discovered by going to the directory (`/var/www/htdocs/CDSware/submit/storage/done/submit/TEXT/`), unpacking the archive (`TEST-S_READING-2003-008_20031208175449.tar.gz`; `cd` there), and running the above `bibformat` command.

Solution? (1) Edit the suspect `recmysql` file to clean-up the abstract; (2) create a fixed `recmysqlfmt` XML file *via* '`bibformat < recmysql > recmysqlfmt`'; (3) upload the new file into the MySQL database ('`bibupload recmysqlfmt`'); (4) run '`bibwords add 1-30`' (if you have ≤ 30 entries) to update the words-index for search results, *etc.*; and (5) update the collections/pages/*etc.* *via* `~cdsware/bin/update_collections.sh` (which runs '`create_collection_pages +`'). Finally, re-archive/compress the new directory in place of the old one (`tar -zcvf TEST-S_READING-2003-008_20031208175449.tar.gz \ TEST-S_READING-2003-008_20031208175449`)...

- **Problem:** A duplicate entry was accidentally added (for example, "Statistical Physics of Macromolecules..." by Grosberg *et al.*). How do I delete it?

Solution? Begin by browsing the collection to which the duplicate entry was submitted, and get the record number ('`id`') of the duplicate entry: click on "Library Books" from the Document Server frontpage. In this example, there is an entry "Statistical Physics of Macromolecules" (TEST-BOOK-2003-005, `id=21`) which was originally entered, and then the duplicate (TEST-BOOK-2003-011, `id=34`), which we want to delete. The '`TEST-BOOK-YYYY-xxx`' tag should be shown on the short displays, and you can see its '`id`' value by mousing over the 'Detailed record' link (the '`id`' will be displayed at the end of the line in the navigator window's toolbar); if it is not displayed, just click on the 'Detailed record' link for the entry you want to delete, and then follow the 'HTML MARC' link (top of page) – the '`id`' for this entry is the number after the `001__` field. Now that you have the '`id`', go to the Administrators Area → BibData Admin → Edit Record → enter the '`id`' value for the duplicate entry ('34' in this example) and click 'Edit record'. Now, finally, click the 'delete' link beside the Trashcan icon in the upper right. The entry should now be deleted, and the document server pages will be updated the next time cron runs `update_collections.sh` (which in turn runs CDSware's '`create_collection_pages`' script); as currently set, this should be within 10 minutes.

- **Problem:** I've deleted an entry but it still appears when browsing the collections. How do I eradicate all trace of these suspect entries – *i.e.*, sanitize the `bibwords` collection that is drawn upon for searching, browsing, *etc.*?? See emails of 04/07→04/15/2004 for more detailed descriptions of the problem and its solution (excerpts below):

...Indeed, what I would like to do is what Tibor referred to as “wipe out the trace of record ID ### fully”, for a v0.0.9 set-up. I can get most of the way there by going to the **Admin Area** → **BibData Admin** → **Edit Record** → enter “28”, click “Edit record” button, and then click the “delete”/trash-can icon to the right. After doing this and updating the collection pages *via* “create_collection_pages” script, everything is fine (even the count of total number of records on the server’s front page), **except** for the blank records showing-up when I browse the collection (as seen in the image files attached to my last email). Somehow it seems like I need to finalize the deletions by manually editing one of the v0.0.9 SQL tables??...

Solution? Use the MySQL “TRUNCATE” command, as described by Tibor in the following:

Apparently this record was not deleted properly from the indexes, which is why it still appears in your search results even if it looks otherwise empty. **BibData Admin** should have launched “bibwords del 29” prior to deleting, but apparently this step somehow failed. Now that your record #29 is empty, it wouldn’t help to launch this command manually anymore, since the old state of the record is lost^[1]. The best solution for CDSware v0.0.9 is therefore to reindex your records:

- Firstly, wipe out all existing indexes:

```
$ mysql -u your-mysql-user -p your-cdsware-database
mysql> TRUNCATE TABLE bibwords1;
mysql> TRUNCATE TABLE bibwords2;
[...]
mysql> TRUNCATE TABLE bibwords9; ## for each bibwords[0-9]+ table
```

- Secondly, force indexing of all your records:

```
$ ./dbutil/bibwords add 1-1000000|}
```

- Thirdly, recreate your collection pages:

```
$ ./cgibin/create_collection_pages ""+
```

After which the things should be back to normal. (If not, please verify your <http://132.239.16.202/cgi-bin/CDSware/search?id=29&of=hm> to check whether there are not any MARC fields left for the record #29 that were not shown in the XML Dublin Core snapshot that you posted.)

Footnotes: ^[1] Note that CDSware development branch releases contain reverse indexes that would be able to take care of this situation, so that the reindexing would not be needed here.



Figure 1: Browsing the Suggested Reading collection of the McCammon Group Document Server; note the “EMPTY RECORD”.

How to submit new entries to the McCammon Group Document Server:
 #####

Here is a step-by-step guide to submitting new entries into the database and modifying current entries; this is all done *via* the web page interfaces. Steps pertaining to the modification of an existing entry – as opposed to the creation of a new entry – are *in italics*.

1. From the server's frontpage (<http://mccammon.ucsd.edu/CDSware>), click on the **Submit** link in the upper left-hand corner.
2. Enter your account info in the lower left-hand corner (Email and Password) and click on the "login" button.
3. Now you should see the "McCammon Group catalogs" in the main body of the page. Click on either:

"text document" — to enter an article into the "Group Publications" or "Suggested Reading" collections and have your new entry incorporated immediately

"text document (with simple refereeing)" — to enter an article into the "Group Publications" or "Suggested Reading", but have it *not* incorporated right away (*i.e.*, it has to be either approved or rejected after review by one of the referees). This may be a better option to follow for the first couple times.

"library book" — to create an entry and provide information for a book or thesis from the group library

"picture document" or "picture document (with simple refereeing)" — do not play with these yet (or do so only at the server's risk)

4. Select a collection category (*e.g.*, if you chose the "library book" catalog, then now you can select either "book" or "thesis") and then click either the "Submit" or "Modify" bar.
5. If this is a new submission, go through all the pages, entering the required information. **Note:** If this is in any of the Article-related categories (*e.g.*, "Group Publications" or "Suggested Reading"), you will need to upload the locally accessible PostScript or PDF file.

If this is a modification, then enter the Report Number of the entry you want to modify. You can find this Report Number by browsing the collection – e.g., click on Library Books from the frontpage to see that the BOOK entry "The Principles of Statistical Mechanics" has a report # equal to 'TEST-BOOK-2003-001'. Then proceed to select the fields you want to modify (you can select more than one by dragging the mouse with left button depressed), make the changes, and then reformat the collections (step 6).

6. *If you created or modified a library book ("BOOK") doctype*, then you also must go to the Admin Area → BibFormat Admin → Reformat Records area and click on the "Reformat Collection" link

in the main page body. You should see a central table with two columns: "Collection name" and "Output Format Selection". If this was a new submission, then check just the "HTML brief" box in the "* Library Books" row; *if this was a modified entry*, click both boxes ("HTML brief" and "HTML detailed"). Then click "Submit query" to start the reformatting. The request status will be shown in a table at the bottom of the page, and clicking the "Refresh" link will update the status of this table.

Either way – whether you just entered a BOOK or an Article ("Suggested Reading" or "Group Publication") – ***your new entry should automatically appear on the server's web pages within 10 minutes...***

7. Make sure to logout of your session (click the logout button in the lower left-hand corner of the left menu panel). This is important – you may cause yourself trouble in future logins if you do not logout of the current session!

Setting-up and maintaining a functional library book “check-out” system on the current CDSware-based Document Server (<http://chemcca42.ucsd.edu/CDSware>).
 #####

A detailed description of the initial set-up of a check-out functionality *via* implementation of a “Modify Checkout Information” (MCI) action is provided elsewhere in this guide (see, *e.g.*, the 01/11/2004 entry on pages 41–45), and additional information may become available on the group wiki (<http://chemcca42.ucsd.edu:8080/mcwiki/DocumentServerInfo>). The MCI action and related check-out functionality was written in early 2004, and still seems to be working fine as of at least August 2007. In order to utilize this feature to track library books.....

Admins need to do the following:

- **For each new user:** You (jam_cdsware@mccammon.ucsd.edu) will be automatically notified each time a new user creates an account (see the 10/13/2004 note near pg. 48). To enable each new user to modify the check-out status of a book, navigate to the “Authorized Users of Library Book” section of the admin area (<http://chemcca42.ucsd.edu/CDSware/admin/web-access/submitusers.shtml?doctype=LIBB>), pick the new user name in the right-hand list, pick the “Modify Check-out Information” action in the pull-down menu, and then click the “ADD” button. You should then see the new user account name (which is typically an email address) appear in the list of users on the left hand side. Now each user should be able to update the check-out status of the library books.
- **Periodically (e.g., once a week?):** Manually re-format the “Library Books” collection in order to have changes to check-out statuses which have occurred since the last time you did this re-formatting step take effect... Note that “take effect” is meant *visually* – in the HTML output seen either as the product of a search of just when browsing the collections (changes take effect instantly in the underlying MySQL db when the check-out field is initially updated by users). Perform this step by going to the Administrators’ Area → BibFormat Admin → Reformat Records and then selecting the buttons for the Library Books, and, finally, click Submit Query.

Users need to do the following:

1. **Register a new account** by visiting <http://chemcca42.ucsd.edu/CDSware/personalize/youraccount.shtml?action=register>; *do not use a sensitive password* (like your Unix one!), as it will not be encrypted in the underlying MySQL database (see elsewhere in this document for more about that).
2. **Identify the “Report Number”** code for the book which you’re either checking-out or returning. This info can easily be found by manually browsing the “Library Books” collection or by searching within it; for instance, the “Nucleic Acids” text by Bloomfield, Crothers, & Tinoco is TEST-BOOK-2005-198.
3. Finally, (1) From the CDSware front page, click the left-hand ‘Submit’ button, (2) Enter your login info in the left-hand form, (3) Navigate back to the ‘Submit’ area if you’ve strayed from there (<http://chemcca42.ucsd.edu/CDSware/submit/>), (4) Under “McCammon Group catalogs”, select ‘Library Book’, (5) On the next pg, check the ‘Book’ radio button, then click

on “Update Check-Out Status”, (6) Enter the “Report Number” and click ‘next page’, (7) Follow the remaining directions through the menus

4. Please direct questions to `jam_{steer,cdsware}@mccammon.ucsd.edu`.

Related information:

- In particular regards to the **Library Books** collection, see pgs 19-22 for an explanation of how this doc type was created, and pgs 41-45 for a description of the MCI action which operates on this collection and forms the basis for the check-out functionality..
- See pgs 32-33 for an explanation of how to submit new entries (books, articles, etc.) to the Doc Server...
- See the last couple pages of this document for other useful references, notes, links, etc...

Running LOG of users and their privileges:
#####

Perform all these administrative tasks by logging in as superuser (`jam_cdsware@mccammon.ucsd.edu`) and going to various Admin Area → WebAccess Admin → ... areas

- 11/10/2003: Set-up user accounts for `cmura@mccammon.ucsd.edu`, `jmccammon@ucsd.edu`, as described above.
- 11/13/2003: Added `lbare@mccammon.ucsd.edu` (passwd '@#\$%^&*') to database and granted her rights on Submit Pages – added her to authorized 'users' for the document types "library book", "text", and "text (with simple refereeing)". Also added `lbare` to all of the "viewers" categories.
- Added `jmccammon@ucsd.edu` and `lbare@mccammon.ucsd.edu` to authorized users of the Bib-Format Admin area (in order to allow them to reformat collections).
- 12/16/2003: Added `lbare@mccammon.ucsd.edu` to authorized users of the BibData Admin area (in order to allow her to delete duplicate entries and such).
- 01/11/2004: Added `cmura@mccammon.ucsd.edu` to authorized users of the MCI action of the library book (LIBB) doctype, which permits updating of check-out status.

```
#####
Running LOG of changes/enhancements to the server:
#####
```

11/17/2003:

Step 1: I edited the "library book" doctype to add optional fields for (1) ISBN # and (2) a scientific category for the type of book (classified at the level of Biology, Chemistry, Computer Science, Mathematics, or Physics)... Note that this is somewhat of a prelude to adding something like a Digital Object Identifier (DOI) tag to the TEXT or RTEXT doctypes in order to enable hyperlinking to remote copies of articles. [**Step 1** was done mostly within the Admin Area → WebSubmit Admin area.]

Step 2: Also, I edited the BibFormat output formats by (1) adding a "_MORE_BOOK_INFO" format and (2) modifying the "DEFAULT_HTML_DETAILED" format to account for whether or not the entry is a BOOK. [**Step 2** was done mostly within the Admin Area → BibFormat Admin area.]

Here's how this was achieved:

Step 1:

- In the web interface, go to the WebSubmit Admin area → Available Element Descriptions CREATE NEW ELEMENT DESCRIPTION: create 2 of them – 'book_ISBN_num' and 'science_categ':

⇒ Create the 'science_categ' Form Element:

```
Element Type:   Select Box
Element:        science_categ
Aleph Code:     <empty>
Marc Code:      687__a <---- NOTE: like the 689__z(y) tags used to describe
                  the check-out status, this MARC code was
                  selected somewhat arbitrarily (after
                  verifying that it is 'unassigned' in the
                  current MARC21 standard at
                  http://www.carl.org/tlc/crs/Bib1468.htm)

Item Description: <SELECT name="science_categ">
                  <OPTION value="Chemistry" selected>Select a category:
                  <option value="Biology">Biology
                  <option value="Chemistry">Chemistry (default)
                  <option value="Computer Science">Computer Science
                  <option value="Mathematics">Mathematics
                  <option value="Physics">Physics
                  </SELECT>
```

Cookie: No

⇒ Create the 'book_ISBN_num' Form Element:

```
Element Type:   Text Input
Element:        book_ISBN_num
Aleph Code:     <empty>
Marc Code:      688__a
Size:          40
Maxlength:     0
Value:         <empty>
Cookie: No
```

⇒ Also edit the homemade ‘CANChange2’ Form Element to add these 2 new descriptors as choices in the SelectBox:

```
<option value="science_categ"> Scientific category
<option value="book_ISBN_num"> ISBN #
set the size of the array = 12, etc.
```

- Then go to the WebSubmit Admin → Available modification texts and add the ‘science_categ’ and ‘book_ISBN_num’ options (bringing the total num of homemade texts to 4, including ‘Check_out’ and ‘Who_check_out’):

```
science_categ: <br /><br />Please update the Scientific Category of the book:<br />
book_ISBN_num: <br /><br />Please update the ISBN # of the book:<br />
```

- Now actually incorporate these additions by going to WebSubmit Admin → Details of the LIBB document type and editing the SBI (Submit Bibliographic Information) Submission Pages:

SBILIBB Page 1: Add ‘science_categ’ as (Optional) Item No 3 (of 4), with following Element Label value:

```
<br /><br />Optionally, classify the book into one of the following scientific categories:
```

SBILIBB Page 2: Add ‘book_ISBN_num’ as (Optional) Item No 4 (of 5), with following Element Label value:

```
<br /><br />Optionally, enter an ISBN # (<i>e.g.</i>, <tt>1-56396-071-0</tt>)
```

While I’m here, also modified the Element Label for AU (author) to make it a bit clearer:

```
<BR><BR>ALL author(s) of book:<br />Only 1 per line, in the form <tt>Name, Initial(s)</tt> --
<span style="color: red;"><i>e.g.</i>, <tt>McCammon, JA</tt></span><br />
```

[Note – there is no need to ‘Edit Functions’ or ‘Edit Submissions’ within SBI, and no need to edit any of the MBI (modify) action.]

Step 2:

- Go to the Admin Area → BibFormat Admin → Extraction Rules and ‘Add new Extraction Rule’ (use form at bottom of page) for the 687__a (science_categ) and 688__a (book_ISBN_num) MARC tags which were just described. Por ejemplo, fill-in the following:

```
Type = DEFAULT
Variable name = 687
Field Type = datafield
Attribute TAG value = 687
Attribute IND1 and IND2 values – leave them empty!
```

Then scroll-up to the new 687 and 688 rules and add subfield ‘a’ (or ‘A’, as it is displayed for some reason!)

- Now incorporate these extracted MARC tags into the HTML (both brief and detailed) that gets output by BibFormat. That is, go to BibFormat Admin → Formats and do the following:

⇒ Add new FORMAT (bottom of pg) with:

```
Format Name = _MORE_BOOK_INFO
```

```
Format documentation: HTML of more BOOK info (#pages,ISBN#,check-out status,
scientific category,etc.)
```

```
EL Code – what follows is the actual working code:
```

```

if($300.a!="")
{
  forall($300.a)
  { "<span style='font-weight: bold;'">Number of pages:</span> "$300.a" <br />" }
}

if($687.a!="")
{
  forall($688.a)
  { "<span style='font-weight: bold;'">Scientific category:</span>  "$687.a" <br />" }
}

if($688.a!="")
{
  forall($688.a)
  { "<span style='font-weight: bold;'">ISBN #:</span>  "$688.a" <br />" }
}

if($689.z="YES")
{
  forall($689.y)
  {
    if ($689.y!="")
    { "<span style='font-weight: bold;'">Check-out status:</span>Currently CHECKED-OUT
      by <a href='mailto:"$689.y"'>" $689.y "</a>."}
    else { "<span style='font-weight: bold;'">Check-out status:</span>Currently CHECKED-OUT
      by UNKNOWN <br />" }
  }
}
else { "<span style='font-weight: bold;'">Check-out status:</span>  Currently AVAILABLE <br />" }

```

NOTE 1: CDSware's BibFormat tool uses a special 'Evaluation Language' (EL), which you can read all about at <http://132.239.16.202/CDSware/admin/bibformat/BibFormatAdminGuide.shtml>. In actual EL code, keep all the HTML-encoding strings on a single line; long strings are line-wrapped here (*e.g.*, after the 'CHECKED-OUT' bit) only for the sake of fitting within the page width!

NOTE 2: The \$300.a bit is for the number of pages; the \$687.a bit is for the science_cat; the \$688.a bit is for the book_ISBN_num; the \$689.z and \$689.y bits store the check-out status and the email of the person who checked out, respectively.

⇒ Check the DEFAULT_HTML_BRIEF format to make sure it has the following EL Code (*i.e.*, the check-out status is displayed) after the _DEFAULT_URL (\$8564.u) section:

```

if($689.z="YES" && $980.a="BOOK")
{
  forall($689.y)
  {
    if ($689.y!="")
    { " <br /><small class=quicknote> [Currently CHECKED-OUT
      by <a href='mailto:"$689.y"'>" $689.y "</a>.]</small> " }
    else { " <br /><small class=quicknote> [Currently CHECKED-OUT
      by UNKNOWN]</small> " }
  }
}
else {
  if ($980.a="BOOK") { " <br /><small class=quicknote> [Currently AVAILABLE]</small> " }
}

```

```
}
```

⇒ Edit/check the `DEFAULT_HTML_DETAILED` format to make sure it has the following EL Code after the `'format("_full_keyword")'` bit. Note that it conditionally handles the doctype so that it adds `"_MORE_BOOK_INFO"` only for the `BOOK` type; otherwise, the stream falls through and creates the default formats specified in the original CDSware EL Code :

```
if($980.a="BOOK")
{
  format("_more_book_info")
  format("_full_citedby")
}
else {
  format("_full_note")
  format("_full_publiinfo")
  "<br>"
  format("_full_url")
  format("_full_citedby")
  "<p>"
  format("_full_references")
}
```

That should do it! — the new HTML Detailed and Brief HTML formats should appear after reformatting the "Library Books" collection (Administrators' Area → BibFormat Admin → Reformat Records) and running `'~cdsware/bin/update_collections.sh'` (either manually or waiting for cron).

12/16/2003:

In order to avoid confirmation emails being sent out after each new library book submission, I temporarily deleted the `"Mail_Submitter"` function from the SBI action for the LIBB doctype (see 12/16/2003 email from cmura). I accomplished this by going to Admin Area → WebSubmit Admin, clicking on "library book" doctype, then "Edit Functions" for the SBI action name — in the list of functions, delete the `"Mail_Submitter"` (score 70, step 1). That's all. See the section "How the 'LIBB' doctype was created" (pg.19) for more info. about working with functions, the SBI action, *etc...*

12/17/2003:

Reinstated a customized form of the `Mail_Submitter` function by doing this (see email exchange of 12/17/2003):

1. temporarily disable the `'jam_cdsware'` mail alias in `chemcca51:/etc/aliases`; it currently consists of `jmccammo, lbare, cmura, cwong, rhenchma`
2. add back the `Mail_Submitter` function to the SBI action of the LIBB doctype by going to Admin Area → WebSubmit Admin → Details of the LIBB doctype → edit functions for SBI → add `Mail_Submitter` with `score=70, step=1`
3. `cd /home/cdsware/cdsware-0.0.9/htdocs/submit/MESS/functions/` and modify the source file for this function (`Mail_Submitter.wml`; default was previously saved to `Mail_Submitter.wml.DEFAULT`) by replacing the `ADMIN_EMAIL` target of `"bcc: "` as shown here:

```
###
[root@chemcca42 functions]# diff -uN Mail_Submitter.wml.DEFAULT Mail_Submitter.wml
```



```

--- Mail_Submitter.wml.DEFAULT 2002-08-01 00:29:47.000000000 -0700
+++ Mail_Submitter.wml 2003-12-17 11:31:30.000000000 -0800
@@ -110,7 +110,7 @@
     $email_txt = $email_txt . "Thank you for using CDS webSubmit.\n"
     . "The Electronic Submission team.";

-     mail("$m_recipient", "$fullrn: Document Received", $email_txt, "from: ".SUPPORTEMAIL."\nbcc: ".ADMIN_EMAIL);
+     mail("$m_recipient", "$fullrn: Document Received", $email_txt, "from: ".SUPPORTEMAIL."\nbcc: cmura@ucsd.edu");
     Request_Print("S", "Email Sent to " . $m_recipient);
}
</protect>
###

```

4. login as `cdsware@chemcca42`; `cd ~cdsware/cdsware-0.0.9/`; recompile and reinstall CDSware as usual:

```

export WMLOPTS="-I/home/cdsware/cdsware-0.0.9/config"
env LANG=C gmake
sudo /bin/env LANG=C /usr/bin/gmake install

```

5. Re-activate the ‘`jam_cdsware`’ alias (edit `chemcca51:/etc/aliases`; `/usr/bin/newaliases`); note that this will also serve as a negative control based on what emails I receive upon submission.
6. For test purposes, login to the document server as `jam_cdsware`, go to the **WebAccess Admin**, add user ‘`cmura@mccammon.ucsd.edu`’ to authorized users for library book. Then make a test submission to the Library Books collection while logged-in as `cmura@mccammon.ucsd.edu`.
7. Verified that proper emails are sent – should receive one email to `cmura@ucsd.edu` and one email to `cmura@mccammon.ucsd.edu` (that’s what I was logged-in as for submission), both from `jam_cdsware`. Then, `cp` the modified `Mail_Submitter.wml` to usual place (`~cdsware/MCCAMMON_SPECIFIC_CDSware/`) for posterity.

01/11/2004:

Completed a new enhancement – now ‘ordinary’ users can update the check-out status of a library book (LIBB) via a new action (‘MCI’ = **Modify Check-out Information**) which is analogous to SBI, MBI, *etc.*, and which was created explicitly for this purpose. Basic idea: create a new action (MCI) in which only the `Check_out` and `Who_check_out` fields can be modified (create this by copying the regular modify (MBI) action and editing it), and then grant permission for this MCI action to all users (whereas the default MBI modify action is reserved for privileged users). See email exchange of 01/08/2004 with Thomas Baron. Here is how this was accomplished:

1. As usual, temporarily disable the `jam_cdsware` mail alias to avoid annoying emails
2. Logged-in as `cdsware`, go to `chemcca42:/home/cdsware/cdsware-0.0.9/htdocs/submit/MESS/functions/` and copy two MBI-related WML source files to be used by the upcoming MCI action:

```

# cp -a Send_Modify_Mail.wml Send_MCI_Mail.wml
# cp -a Create_Modify_Interface.wml Create_MCI_Interface.wml

```

Edit these files (primarily by replacing instances of ‘MBI’ with the new action name, ‘MCI’) to get following differences:

```

[cdsware@chemcca42 functions]$ diff -uN Send_Modify_Mail.wml Send_MCI_Mail.wml
--- Send_Modify_Mail.wml      2003-11-11 10:38:47.000000000 -0800
+++ Send_MCI_Mail.wml       2004-01-08 18:09:13.000000000 -0800
@@ -1,4 +1,4 @@
-## $Id: Send_Modify_Mail.wml,v 1.2 2002/08/01 07:29:47 thomas Exp $
+## $Id: Send_MCI_Mail.wml,v 1.2 2002/08/01 07:29:47 thomas Exp $

## This file is part of the CERN Document Server Software (CDSware).

```

```

## Copyright (C) 2002 CERN.
@@ -18,29 +18,29 @@
## 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.
<?
<protect>
-## $Id: Send_Modify_Mail.wml,v 1.2 2002/08/01 07:29:47 thomas Exp $
+## $Id: Send_MCI_Mail.wml,v 1.2 2002/08/01 07:29:47 thomas Exp $
## DO NOT EDIT THIS FILE! IT WAS AUTOMATICALLY GENERATED FROM CDSware WML SOURCES.

- ## Description:    function Send_Modify_Mail
+ ## Description:    function Send_MCI_Mail
##                  This function sends an email saying the document has been
##                  correctly updated
## Author:          T.Baron
## PARAMETERS:     addressesMBI: email addresses to which the mail is sent
- ##                  fieldnameMBI: name of the file containing the modified
+ ##                  fieldnameMCI: name of the file containing the modified
##                  fields
##                  sourceDoc: name of the type of document
##                  emailFile: name of the file containing the email of the
##                  user

- function Send_Modify_Mail ($param_array,$MAINPATH)
+ function Send_MCI_Mail ($param_array,$MAINPATH)
{
    global $ACCESSURL;

    $sub = join("",file("$MAINPATH/".$param_array[emailFile]));
    $sub = ereg_replace ("\n","", $sub);
-   $m_fields = $param_array[fieldnameMBI];
+   $m_fields = $param_array[fieldnameMCI];
    $type = $param_array[sourceDoc];

    $doctype = $GLOBALS[VARs]->get_doctype();
@@ -70,4 +70,4 @@
    , $email_txt, "from: ".SUPPORTEMAIL."\nbcc:
$param_array[addressesMBI]");
}
</protect>
-?>
\ No newline at end of file
+?>
[cdsware@chemcca42 functions]$

```

AND...

```

[cdsware@chemcca42 functions]$ diff -uN Create_Modify_Interface.wml Create_MCI_Interface.wml
--- Create_Modify_Interface.wml 2003-11-11 10:38:47.000000000 -0800
+++ Create_MCI_Interface.wml    2004-01-08 18:14:12.000000000 -0800
@@ -1,4 +1,4 @@
-## $Id: Create_Modify_Interface.wml,v 1.2 2002/06/20 15:05:15 thomas Exp $
+## $Id: Create_MCI_Interface.wml,v 1.2 2002/06/20 15:05:15 thomas Exp $

## This file is part of the CERN Document Server Software (CDSware).
## Copyright (C) 2002 CERN.

```

```

@@ -18,23 +18,23 @@
## 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.
<?
<protect>
-## $Id: Create_Modify_Interface.wml,v 1.2 2002/06/20 15:05:15 thomas Exp $
+## $Id: Create_MCI_Interface.wml,v 1.2 2002/06/20 15:05:15 thomas Exp $
## DO NOT EDIT THIS FILE! IT WAS AUTOMATICALLY GENERATED FROM CDSware WML SOURCES.

- ## Description:    function Create_Modify_Interface
+ ## Description:    function Create_MCI_Interface
##
##                This function creates the html form allowing the user to
##                some bibliographic fields
## Author:          T.Baron
- ## PARAMETERS:    fieldnameMBI: name of the file containing the
+ ## PARAMETERS:    fieldnameMCI: name of the file containing the
##
##                "+"-separated list of fields to modify

require_once ("${HTDOCSDIR}submit/MESS/functions/Retrieve_Data.shtml");

- function Create_Modify_Interface($param_array,$MAINPATH)
+ function Create_MCI_Interface($param_array,$MAINPATH)
{
    // variables declaration
-   $fieldname = "$param_array[fieldnameMBI]";
+   $fieldname = "$param_array[fieldnameMCI]";
   $username = "$param_array[username]";
   $password = "$param_array[password]";
   $modifybase = "$param_array[modifybase]";
@@ -63,7 +63,7 @@
    $sth = mysql_query("
SELECT modd
FROM sbmMODIFY
-WHERE modn='$fieldname') or outError ("Create_Modify_Interface: Error in MySQL"
+WHERE modn='$fieldname') or outError ("Create_MCI_Interface: Error in MySQL "
        . "query: ".mysql_error());
    $array = mysql_fetch_array($sth);
    Request_Print("A", "<small>$array[0]</small> </FONT>\n");
@@ -82,7 +82,7 @@
    $sth = mysql_query("
SELECT *
FROM sbmFIELDDDESC
-WHERE name = '$fieldname') or outError("Create_Modify_Interface: Error in MySQL"
+WHERE name = '$fieldname') or outError("Create_MCI_Interface: Error in MySQL "
        . "query: ".mysql_error());
    $array = mysql_fetch_array($sth);
    $type = "$array[type]";
[cdsware@chemcca42 functions]$

```

3. Also create a new pair of images for the MCI action by using Gimp to edit the 'modify_button.gif'/'highlight_modify_button.gif' pair that is used in the MBI action (replace the "Modify" text string with "Update Check-out Status"). These new images – 'mci_button.gif' and 'highlight_mci_button.gif' – are stored in chemcca42:/home/cdsware/cdsware-0.0.9/htdocs/img/.
4. As usual, save a cp -a of these four customized files — 'Send_MCI_Mail.wml', 'Create_MCI_Interface.wml', 'mci_button.gif', 'highlight_mci_button.gif' — to chemcca42:/home/cdsware/MCCAMMON_SPECIFIC_CDSware/. And implement these changes to the source *via* the usual 'export WMLOPTS', gmake, and gmake install steps...

5. Now create the MCI action through several modifications made *via* the WebSubmit Admin web interface:
 [Note: much of this is very similar to the first half of the “How the ‘LIBB’ doctype was created” major section.]

- McCammon Group Document Server → Admin Area → WebSubmit Admin → Available Actions ...
 ADD NEW ACTION: Model it after the default MBI action. Call it ‘MCI’, with following final details:

```
Action Code:      MCI
Action Description:  Modify Check-out Information
dir:      modify_checkout  (eventually you should see that this dir is in fact created in
                           chemcca42:/var/www/htdocs/CDSware/submit/storage/.)
actionbutton:  mci_button.gif
statustext:    Modify Check-out Status of a Book
```

You should then see it as ‘MCI: Modify Check-out Information’ in the list of Available Actions

- Then go to the Admin Area → WebSubmit Admin → Available Functions interface and ADD two new functions:
 (View/Edit Function Details shows the following:)

```
Create_MCI_Interface:
  description: this function is specialized for the MCI (Modify Check-out Information) action
  Parameters:  Field:  'fieldnameMCI'
Send_MCI_Mail:
  description: function to send Modification of Check-out Status emails
  Parameters:  Fields:  sourceDoc
                  fieldnameMCI
                  emailFile
                  addressesMBI
```

- Then go to the Admin Area → WebSubmit Admin → Available Element Descriptions and CREATE NEW ELEMENT DESCRIPTION. Call it ‘CANChange3’. Make it identical to CANChange2 except that only the two check-out-related fields are given in the Item Description:

```
<SELECT multiple name="CANChange3[]" size=2>
<option value="Check_out">      Check-out status
<option value="Who_check_out">  Who checked-out
</SELECT>
```

This is necessary because the CDSware software is not configurable in such a way that only a portion of CANChange2 (corresponding to the two check-out-related fields) can be used in the new MCI action. Instead, a whole new CANChange3 element must be created.

- Final step: Go to the main configuration page for the LIBB doctype (Admin Area → WebSubmit Admin → library book) and ADD A NEW ACTION – the ‘MCI’ action of course, with the following settings:

```
On Submission Page:      Y
No. Pages:                2
Button Order:            3
level / score / stpage  0 / 0 / 1
```

```
Edit Submission Pages:  2 pages, identical to the MBI action, except that Item no. 1 on
                        MCILIBB Page 2 is the CANChange3 element (not CANChange2)
```

```
Edit Functions:        Copy the MBI list of functions, parameters, etc., but make the following
                        changes:
```

- replace `Create_Modify_Interface` with `Create_MCI_Interface` -- should have param = 'fieldnameMCI' / value = 'CANChange3'
- replace `Send_Modify_Mail` with `Send_MCI_Mail` -- should have following set of param / values:

<code>addressesMBI</code>	<code>x</code>
<code>emailFile</code>	<code>SuE</code>
<code>fieldnameMCI</code>	<code>CANChange3</code>
<code>sourceDoc</code>	<code>BOOK</code>
- verify that `Make_Modify_Record` is using the `EDSLIBB.tpl` / `EDSLIBBmysql_Modify.tpl` files, as for `LIBBMBI` action

6. Upon initial creation of the MCI action, you may get errors about not being able to create links to `go.shtml` (see the above section describing creation of the 'LIBB' doctype). You must create the link manually as follows:

```
# cd /var/www/htdocs/CDSware/submit/access/protected
# ln -s /var/www/htdocs/CDSware/submit/access/protected/go.shtml  MCILIBB.shtml
```

7. Enable access by regular users to the MCI action for the library book doctype *via* the WebAccess Admin interface: Admin Area → WebAccess Admin → Authorizations on Submit ('Rights on Submit Pages') → click on 'users' for the library book doctype. Then, 'Add' the 'Modify Check-out Information' action for a regular user (e.g., `cmura@mccammon.ucsd.edu`).
8. Usage note: Remember, you still need to use the BibFormat Admin → Reformat Records web interface to reformat the Library Books collection and create updated HTML-detailed and HTML-brief, etc. after modifications of fields (such as the Check_out status) as usual...

03/02/2004:

Configured the TEXT and RTEXT (refereed text) doctypes to include a new "journal_title" element description which will be used to display the journal title in both the brief (HTML brief) and "Detailed record" views of collections which contain (R)TEXT items (e.g., "Suggested Reading" and "Group Publications"). This was accomplished in basically the same way as for addition of the ISBN # ('book_ISBN_num') and scientific category ('science_categ') fields (see 11/17/2003). What follows are the main differences between the previous modifications and this one:

1. Changes had to be made twice – once for TEXT and once for RTEXT (rather than just once, as in case of LIBB)
2. Called the new element description "journal_title" and used a free MARC tag (189__j was arbitrarily chosen)...
3. Modified the default CANChange element which is used for (R)TEXT submissions...
4. Added a new modification text for the new journal_title field (WebSubmit Admin → Available modification texts)
5. Edit the submission pages of the SBI action for the (R)TEXT doctypes by adding this new journal_title element to SBITEXT Page 3.

6. Make changes in the 'Extraction Rules' and 'Formats' areas within Admin area → BibFormat Admin. Namely, add the extraction rule for \$189_j and edit the DEFAULT_HTML_BRIEF and DEFAULT_HTML_DETAILED output formats by adding some EL code to take care of this new field. For example, this was added to the DEFAULT_HTML_DETAILED immediately before the \$088.a loop:

```
if ($189.j!="")
{ " <br /><small class=journal_title>" $189.j "</small><br /> " }
```

Sidenote: defined the journal_title style class as follows in /home/cdsware/cdsware-0.0.9/htdocs/img/cds.css.wml:

```
.journal_title {
  color: #dfc;
  background: transparent;
  font-style: italic;
  font-weight: bold;
}
```

7. Finally, go to the CDSware installation subdirectory containing the EDS*TEXT*wml source files (/home/cdsware/cdsware-0.0.9/htdocs/submit/bibconvert/config/), make backup copies of these (*.DEFAULT), and edit them to include the new journal_title field. This is basically what was done for the LIBB doctype in the actual/final directory (/var/www/htdocs/CDSware/submit/bibconvert/config/), except that for these (R)TEXT files it's important to make changes to the WML source (otherwise any changes made in /var/www/blah-blah will be over-written next time cdsware is compiled!). Now go to /home/cdsware/cdsware-0.0.9/, set WMLOPTS, and compile as usual.
8. To implement modifications just made to any journal article entries, go to the BibFormat admin → Reformat Records → Reformat Collections area and check the 'Group Publications' and 'Suggested Reading' collections. Then run /home/cdsware/bin/update_collections.sh (or wait for cron to do it), and changes should be visible.

04/07/2004:

Made four cyan-colored (not default navy blue) single- and double-arrow forward and reverse button icons by editing the original files (s?.gif) using ImageMagik's "display". The original and modified files are also copied to chemcca42:/home/cdsware/MCCAMMON_SPECIFIC_CDSware/. Moved the modified ones to ~cdsware/cdsware-0.0.9/htdocs/img/ and then did the usual re-compilation of the software (set WMLOPTS, gmake, 'sudo /bin/env LANG=C /usr/bin/gmake install'). Verified changes by browsing, e.g., Suggested Reading collection...

09/02/2004:

Not a change to the CDSware Document Server per say, but: (1) implemented an email archiving system for the 'jam_cdsware' list and (2) updated Document Server back-up strategy to include local daily backups to chemcca15. Here's how that was done:

1. Implemented email archiving system for the 'jam_cdsware' list as is done with the other jam_steer, jam_group, etc. aliases. This was done via method described in an email exchange with Steve Bond (sdbond@uiuc.edu; below). The following procedure is also described in ~cmura/LINUX_chingis/sendmail_and_email_archiving_tips.txt):
 - Created a 'cdsware_arc' user account on c16 by something like the following:

```
$] mkdir -p /net/home/archives/cdsware\_arc
$] useradd -c "JAM CDSware Mail Archiver" -d /net/home/archives/cdsware\_arc \
    -n -g 900 -s /bin/csh cdsware\_arc
```

(‘redhat-config-users’ GUI tool is useless for this, as it complains about punctuation in the name!)... and then made c51 and the other ypclient machines on mccammon network aware of this account by rebuilding the NIS maps (‘make -C’) and restarting ypserver as per usual when make changes to these types of things. Also go to c51 and set-up the ‘cdsware_arc’ username there too (see Steve’s email). Note that at no point in any of this is a password or other authentication token set for the new cdsware_arc user account!

- Check that permissions and other stuff (.forward, maybe .procmail if want filtering, *etc.*) are set-up properly in /net/home/archives/cdsware_arc; usermod, chgrp, chown, chmod commands are useful here. May also have to edit ‘cdsware_arc’ line in /etc/shadow and /etc/passwd (which is not writeable by default, even by root) on c51 (which is the group’s mail and web server) and/or c16 (file server) in order to unlock the passwd field (denoted by a ‘!’ vs ‘*’) and make everything work OK. If end up having to do all this, then rebuild the NIS/yp maps and restart ypserver as usual.
- Make sure to create the /net/home/www/htdocs/local/hypermail/jam_cdsware/ dir, and then go to c51 and update /etc/cron.d/hypermailer by adding following bit for jam_cdsware archive:

```
cd /net/home/www/htdocs/local/hypermail/jam_cdsware
/usr/local/bin/mhonarc -sort -reverse -tsort -treverse \
    -add /net/home/archives/cdsware_arc/archive > mhonarc.log
chmod 644 *.*
```

- Go through the mail2web, hypermail, MHonArch, Namazu, *etc.* steps to set-up the actual processing of the archived emails (which end up in the file /net/home/archives/cdsware_arc/archive as a result of all the previous steps) and auto-generation of HTML pages, daily cron jobs to archive, *etc.*, exactly as per Steve’s instructions in /net/home/www/htdocs/local/hypermail/README. Here’s the relevant excerpt of that file (note that mhonarc runs on c51!):

```
(3) Here’s how I (cmura) initialized the jam_cdsware archive:
/usr/local/bin/mhonarc -title "McCammon CDSware Mailing List" \
    -tttitle "McCammon CDSware Mailing List (Threads)" \
    -sort -reverse -tsort -treverse /net/home/archives/cdsware_arc/archive
chmod 644 /net/home/www/htdocs/local/hypermail/jam_cdsware/*
```

- Update relevant parts of the website, specifically the index.html files in /net/home/www/htdocs/local/ and /net/home/www/htdocs/local/hypermail/. This is for the traditional local webpages, as well as the McWiki ones (<http://chemcca38.ucsd.edu:8080/mcwiki/DocumentServerInfo>).
- Here’s the email explanation referred to above:

```
Date: Thu, 02 Sep 2004 16:52:06 -0700
From: Cameron Mura <cmura@mccammon.ucsd.edu>
To: jam_cdsware@mccammon.ucsd.edu
CC: Stephen Bond <sbond@uiuc.edu>
Subject: jam_cdsware mail archive
```

Hi,

This probably should’ve been done a long time ago, but I’ve now set it up so that all "jam_cdsware" email will be archived in the same way as the usual "jam_group" and "jam_steer" lists. As with these other lists, archived mail will be searchable or browseable by threads from both the Wiki (<http://chemcca38.ucsd.edu:8080/mcwiki/LocalWebpages>) and non-Wiki (<https://mccammon.ucsd.edu/local/>) versions of the local webpages.

Many thanks to Steve for his usual helpfulness in all this (email below).

Cam

=== S Bond wrote (on 09/02/2004 11:26 AM): ===

>Hi Cam-
>
>You could probably call it cdsware_arc, and follow all the steps
>that you outlined. I believe the reasoning behind having it in
>/etc/passwd on both machines, rather than just on chemcca16 was
>that we didn't have to server it over NIS. If you look in
>/var/yp/ on chemcca16 you will see a Makefile. In that Makefile
>there is a place to specify the minimum group and user ID number
>for export over NIS. This is currently set to 800 which is below
>the 900 range being used for the archive accounts. Since the
>archive accounts are locked out from login there is probably no
>risk here. The only way to become steer_arc is to su from root.
>If you did want the new archive account to show up on other
>machines (numbers will resolve to names when someone does an
>ls on that directory), you will need to do a "make" in the
>/var/yp directory and possibly do a /etc/init.d/ypserv restart
>on chemcca16 after that.
>
>If you want the mail2web feature to work:
>
>(1) The daily cron job is the script
>/etc/cron.d/hypermailer (on chemcca51)
>
>(2) Instructions for initializing the mail archive is in
>/net/home/www/htdocs/local/hypermailer/README
>
>Our local mail searching feature (namazu) will work as
>long as the archive is in the same subdirectory as the
>others (/net/home/www/htdocs/local/hypermailer).
>
>I have been doing to segmentation (by years) by hand for
>the active archives at the start of every new year. At
>some point I should automate that. I may do that at the
>start of next year.
>
>Feel free to unify, simplify, document any of these
>instructions and/or features. As a general rule things
>run faster on chemcca16, so perhaps the Mhonarc program
>should run there as well. Then again we don't get
>that much new mail per day so it runs pretty fast. On
>the other hand, running namazu on chemcca16 is absolutely
>essential if you want it to finish in under 12 hours.
>
>-Steve
>
>---- Original message ----
>
>
>>Date: Wed, 01 Sep 2004 18:04:31 -0700
>>From: Cameron Mura <cmura@mccammon.ucsd.edu>
>>Subject: Re: switch to mailman? McArchiving
>>To: S Bond <sdbond@uiuc.edu>
>>
>>Would the proper procedure for setting-up archiving of messages to the
>>"jam_cdsware" (document server) list alias be to: (1) create an ordinary
>>user account "jamdocs_arc" akin to steer_arc, uhbd_arc, jam_arc, etc.
>>with its \$HOME being in /net/home/archives/ (group 900) and then (2) add
>>this new user to the jam_cdsware alias in c51's /etc/aliases, rebuild
>>the database w/ 'newaliases', etc. as per usual?
>>
>>I noticed that *_arc usernames are in /etc/passwd on c51 as well as the
>>usual c16; what's that all about?
>>
>> Cam

2. Updated Document Server back-up strategy to include local backups to chemcca15 daily, in addition to the ongoing offsite ones (to home, every other day). See chemcca15:/etc/cron.daily/zsync-chemcca42 and updated version of chemcca42:/root/cron_mysql_backup.bash for how this is done. Idea is for c15 to use rsync-based /etc/cron.daily/zsync-chemcca42 to pull the three relevant non-MySQL directories from c42 (/home/cdsware/, /var/www/htdocs/CDSware/, and /var/www/cgi-bin/CDSware/) into subdirs of c15:/home/c42/. Following lines added to the rsync-based 'zsync-chemcca42' script to achieve this:

```
# Next 3 rsync's take care of the non-MySQL portion of the c42 Document Server.
# The MySQL bit is handled via 'mysqldump' in /root/cron_mysql_backup.bash from the
# c42 side (every other day).
echo "--- PULL chemcca42:/home/cdsware/. to /home/c42/cdsware/" >> $LOCALLOG
$RSYNC $RSYNCFLAGS \
  chemcca42:/home/cdsware/. /home/c42/cdsware/ >> $LOCALLOG 2>&1

echo "--- PULL chemcca42:/var/www/htdocs/. to /home/c42/var/www/htdocs/" >> $LOCALLOG
$RSYNC $RSYNCFLAGS \
  chemcca42:/var/www/htdocs/. /home/c42/var/www/htdocs/ >> $LOCALLOG 2>&1

echo "--- PULL chemcca42:/var/www/cgi-bin/. to /home/c42/var/www/cgi-bin/" >> $LOCALLOG
$RSYNC $RSYNCFLAGS \
  chemcca42:/var/www/cgi-bin/. /home/c42/var/www/cgi-bin/ >> $LOCALLOG 2>&1
```

The MySQL bit is handled differently, *via* "mysqldump" as well as rsync (/root/cron_mysql_backup.bash). The following table provides an overview of all Document Server-related backups. Except as noted, all scripts are in /root/ dir of given machine, and the four chemcca42 dirs that are listed at left contain everything related to c42's CDSware installation and operation:

	LOCAL (-> chemcca15)	OFFSITE (-> home)
chemcca42:		
1) /home/cdsware	--> pulled by c15:/etc/cron.daily/zsync-chemcca42 (DAILY, rsync)	pushed by c42's cron_backup.bash (EVERY 2 DAYS, rsync)
2) /var/www/htdocs/CDSware/	--> pulled by c15:/etc/cron.daily/zsync-chemcca42 (DAILY, rsync)	pushed by c42's cron_backup.bash (EVERY 2 DAYS, rsync)
3) /var/www/cgi-bin/CDSware/	--> pulled by c15:/etc/cron.daily/zsync-chemcca42 (DAILY, rsync)	pushed by c42's cron_backup.bash (EVERY 2 DAYS, rsync)
4) /var/lib/mysql	--> pushed by c42's cron_mysql_backup.bash (EVERY 2 DAYS, rsync & mysqldump+scp)	pushed by c42's cron_mysql_backup.bash (EVERY 2 DAYS, rsync & mysqldump+scp)

The cron_mysql_backup.bash includes calls to the c42's cron_backup.bash and cron_backup_c15.bash scripts, and both of these – cron_backup.bash and cron_mysql_backup.bash (and therefore cron_backup_c15.bash) – are scheduled to run at ~3:00 AM by root's crontab on c42.

10/13/2004:

Modified the new user registration function (performRegister()) in the source tree file htdocs/personalize/yourfunc.inc.wml by adding a mail() function call, thereby automatically sending email to jam_cdsware@mccammon.ucsd.edu upon each new account registration (see emails of 10/13/2004). The motivation for customizing this function is that CDSware admins can then go ahead and add such things as check-out (MCI) privileges to new user accounts without waiting for users to gripe about it (;-)). This was achieved by the following simple edit:

```
###
[cdsware@chemcca42 cdsware-0.0.9]$ diff -uNr htdocs/personalize/yourfunc.inc.wml.DEFAULT \
      htdocs/personalize/yourfunc.inc.wml
--- htdocs/personalize/yourfunc.inc.wml.DEFAULT 2002-06-17 08:38:41.000000000 -0700
+++ htdocs/personalize/yourfunc.inc.wml 2004-10-13 18:01:01.000000000 -0700
@@ -2152,6 +2152,7 @@
     mysql_query($squery,$SESS_DBH);
     $uid;
     $uid_email=getEmail($uid);
+   mail("jam_cdsware@mccammon.ucsd.edu","New Document Server account notification", \
        "A new Document Server account '$p_email' has just been created.", "from: \
        jam_cdsware@mccammon.ucsd.edu \nbcc: cmura@ucsd.edu");
   }
   mysql_free_result($res);
} else {
###
```

After making the above change, the usual procedure was followed: (1) saved the original source file as *.DEFAULT and copied ('cp -a -parents') the modified file to the usual place (~cdsware/MCCAMMON_SPECIFIC_CDSware/), and then (2) recompiled/reinstalled CDSware as usual (gmake; gmake install targets); *etc.*, as described above for the Mail_Submitter() function (12/17/2003 entry). Note that there was no need to temporarily disable the 'jam_cdsware' mail alias. The new feature was tested by creating a dummy account (cmura@ucsd.edu) and confirming that email was sent to jam_cdsware.

Some useful references for further exploration:
#####

Of course, the CDSware web pages:

<http://cdsware.cern.ch/> – main CERN Document Server Software page (downloads, mailing list archives, etc.)
<http://cds.cern.ch/> – an actual implementation of CDSware at CERN
<http://cdsweb.cern.ch/> – implementation of a more recent version of CDSware at CERN [v.0.0.9+]
<http://ett-div.web.cern.ch/ett-div/DH/cds/> – CERN’s document handling service, including
links to personnel
<http://cdsware.cern.ch:8000/hacking/> – CDSware hackers how-to

Some of the underlying software and unix-related stuff:

<http://thewml.org/> – homepage of WML, the WebSite Meta Language
<http://www.mysql.com/doc/en/index.html> – the MySQL Reference Manual
<http://www.courtesan.com/sudo/> – homepage of sudo (‘superuser do’)

Database-related standards, protocols, etc.:

<http://www.loc.gov/standards/marcxml/> – the Library of Congress’ MARC-XML schema (this is
the bibliographic data structure upon which CDSware is built!)
<http://www.carl.org/tlc/crs/bib0001.htm> – more info. about the MARC-21 bibliographic data format
(such as the MARC tags!)
<http://www.openarchives.org> – homepage of the Open Archives Initiative, including OAI’s Protocol
for Metadata Harvesting (OAI-PMH)
<http://www.doi.org> – homepage of the Digital Object Identifier (DOI) System
<http://dx.doi.org> – server to resolve DOIs
<http://help.sciencedirect.com/robo/projects/sdhelp/doi.htm> – creating DOI links from ScienceDirect

Potential improvements, future plans, things-to-do:
#####

Some possible future enhancements:

- Add available DOI links for articles, rather than storing local copies?
- Incorporate Progress Reports, *etc.* under this document server umbrella?

THE END