
MIR DICOM Software

Central Test Node

Release Notes

Revision 3.0.6
March 11, 2003

Copyright 2003 Washington University

Version 3.0.6: March 11, 2003

1. Ran some experiments with the *archive_server* and JPEG compressed images (transfer syntax 1.2.840.10008.1.2.4.50). We were able to accept images and retrieve images using this transfer syntax and the *archive_server*. We expect that other encapsulated transfer syntaxes will work, but they have not yet been tested.
2. Modified *archive_server* to return the proper error count when images are not transferred with a C-Move request. In some cases, we would return errors in the “warnings” category in the final C-Move response.

Version 3.0.5 December 13, 2002

1. Updated the DUL facility when receiving large association request PDU's. We have a hard coded constant that limits the size of the A-Associate RQ PDU that we can process. The previous limit was exceeded by recent OFFIS applications. We have increased the limit. A future repair will make this buffer length dynamic.
2. Updated the TBL insert functions for the PostgreSQL and SQL Server databases. They now handle input strings with single quote (') characters. We need to update all functions for all databases in the next release.
3. Updated all PostgreSQL functions to handle single (') characters. This is the continuation of item 2. Still need to complete the process for SQL Server. This means that names like O'Neil should now work in our database.
4. Repair the VR in our data dictionary for 0040 0253 Performed Procedure Step ID.
5. Updated GUI applications to compile properly. We had declared some new data types and needed to update the include files for the applications that use Motif.
6. Modified the *archive_server* application. When it receives an ill-formatted query, it now logs a message to the log file explaining the error and returns a status response of 0xC000. The previous behavior was to return a success status with 0 matches and provide no logging information about the error.
7. Modified the *archive_server* application to print delta times for C-Move operations. These get printed to stdout unless you are using the silent mode (-s).
8. Modified the *archive_server* application to give better output messages when it rejects an image that is missing a field. For example, if the image is missing the patient ID, we reject the image. Prior versions merely indicated a field was missing; now we explicitly list fields that are missing.
9. Updated the DCM function *DCM_ImportStream*. We now use the same logic used in *DCM_OpenFile* so that streams are parsed the same way files are parsed.

Version 3.0.4 May 17, 2002

1. Updated data dictionary with some attributes that are found in MG objects. Some attributes are probably still not in the dictionary and need to be added.
2. Modified the application installation files to no longer include *dbfill*. This application is not used and should not be part of the installation.

3. Modified application installation files to include *dcm_add_fragments*.
4. Minor repair to *dcm_add_fragments*. Add a call to close the object after we write to disk.
5. Updated DCM_CloseObject to delete the memory allocated for fragments. Previous versions of the code allocated memory for each fragment but failed to release that memory.

Version 3.0.3 April 17, 2002

1. Corrected an error introduced into the DCM library when we add support for encapsulated pixel data. A random number of extra bytes were being exported after the pixels were exported (whether the pixels were encapsulated or not).
2. Modified the function DCM_AddFragment. It now returns an error if the caller tries to add a fragment with uneven length. If you need to pad a fragment, your application will have to take that responsibility.
3. Corrected an error when exporting pixel data with fragments. The delimiter at the end should be (and now is) FFFF E0DD.
4. Corrected the encoding of attributes with VR UN when using explicit transfer syntaxes. We were not exporting the length properly. The software did read the proper encodings but did not produce/export the proper encoding.
5. Changed behavior of DCM_OpenFile (and other parsing functions) when reading data encoded using explicit VR. The change was for data types OB. We used to change the VR in our internal representation to something other than OB. We now leave it as OB.
6. Modified the function that dumps elements in ASCII (and therefore *dcm_dump_file*). Data of type VR or OB is now dumped in hexadecimal format. Pixel data is still not dumped.
7. Removed the VR 'UK' that served as a placeholder in our code. This was implemented before WG 6 defined the VR 'UN'. No objects should be created with elements of VR UK.
8. Updated our data dictionary with attributes required for General Purpose Worklist.
9. Updated CTN/MESA Test Tools document. Corrected errors in *cmove* description. Updated formatting so all applications appear in Table of Contents.

Version 3.0.2 February 27, 2002

1. Updated the DCM library to correctly parse objects using encapsulated pixel data. Earlier work had been done to support explicit VR; this work added support for the explicit VRs used when the transfer syntax incorporates encapsulated pixel data.
2. Updated *simple_storage*, *send_image* and the *archive_server* applications to properly handle more than just the IVRLE transfer syntax. We now support the transfer syntaxes with encapsulated pixels. Users will have to be careful about configuring these applications and about using the new *archive_server* with an existing database of images.
3. Updated the *archive_server* application to take a -C switch to set the configuration file. Users who want the new features of other transfer syntaxes will need to use this command line switch.
4. Updated the *archive_server* application in its actions for C-Move requests. When the application requests an association with another system, it now requests the transfer syntax used to store the object. This is better behavior in the context of objects with encapsulated

- pixels. We might want some more behavior with non-encapsulated pixels to ask for 2-3 transfer syntaxes (IVRLE, EVRLE, EVRBE). That second step is not implemented.
5. Modified *simple_storage* to not care about the Calling AE title unless the user specifies a title with the *-c* switch. In other words, if you specify no AE title when you start *simple_storage*, it no longer checks to see that the client application uses a specific AE title.
 6. Updated the application *dcm_dump_compressed* to better handle fragments. The old version did not dump fragments properly if a fragment was larger than our assumed size. We know handle large fragment properly.
 7. Added application *dcm_add_fragments*. This application adds encapsulated one or more encapsulated pixel fragments to a composite object.
 8. Added a constant (DCM_ENCAPSULATEDPIXELS) to the DCM facility to tell the export functions to export composite objects with the proper transfer syntax (and encapsulated pixels).
 9. Updated DCM facility to better handle encapsulated pixels. Added a function *DCM_AddFragment* that allows the user to add one fragment to a composite object. Multiple calls to the function are used to add more fragments.
 10. Add function *UID_IsStorageClass* to UID facility. This function returns a flag indicating if a SOP class is a DICOM storage class (storage of composite objects).
 11. Add function *UTL_FileSize* to UTL facility. Allows user to find the size of a file.
 12. Modified function *UTL_ReadConfigFile*. If user had specified a configuration file of "", treat that as if the user had specified a NULL pointer. That is, we go read the default configuration file.
 13. Added function *SRV_RegisterSOPClassXfer* that allows a client application to request/register a SOP class with a specific transfer syntax.

Version 3.0.1 December 10, 2001

1. Corrected a memory allocation error introduced in 3.0.0 in the UTL facility. We were allocating space for a string that was one character too short.
2. Updated makefiles to get the software to compile under Linux. We had made some changes in the system makefiles for other operating systems that caused problems for Linux.
3. Updated *facilities/utility.c* to allow it to compile under Red Hat Linux 7.1.
4. Added an option to the DCM facility that opens DICOM objects to allow repeated attributes by throwing away all repeated attributes. Repeated attributes are a violation of the standard, but we found an implementation that sends them. With this option, we can open those files (by ignoring the repeated attributes). Added command-line switches to *dcm_dump_file* and *send_image* to enable this option.
5. Update *send_image* application to work on directories as well as files. If you ask the program to send a directory, it will scan all files in that directory.
6. Update the SQL Server implementation of the TBL facility. The code that strips spaces off the end of character strings read from the database was not always stripping spaces properly. In some cases, we would not strip any spaces.
7. Update *dcm_modify_object* to correctly read pixel files in Windows environment. Previous versions did not read the binary file properly.
8. Updated *simple_storage* so that the *-x* and *-n* switches can be used at the same time. There was a bug in the code that prevented these two options from functioning when used together.

- Updated this application by adding new directives to the naming convention file; the new directives are P, S, T (see documentation).
9. Updated DCM function DCM_GetString. This function now returns a string representation for attributes that do not have string VR's (such as short or long values).
 10. Updated javactn DICOM.DICOMWrapper class by adding methods getVR, getDescription, getLength.
 11. Updated documentation for utility applications; merged this document with utilities for MESA tools.

Version 3.0.0 February 19, 2001

1. This version should be considered a beta. It is the first version that directly supports the uncompressed, explicit VR transfer syntaxes for network operations. We have tested send_image and simple_storage, but have not updated the archive_server. This version is mainly for people who want to use those tools or who write their own applications with the CTN libraries. The next release will be tested to make sure the archive_server application is functional.
2. Updated send_image to support recursion through directories.
3. Fixed an error in the DUL facility introduced in the 2.12.0 release. Servers that accepted association using the DUL_NOBLOCK parameter exited with a memory exception. Error in the DUL facility is corrected.
4. Fixed the TBL_NextUnique function in the SQL Server implementation. It encountered "invalid cursor state" problems. This means the ttunique test program now operates in the Win32 environment (MS SQL Server 7.0).
5. Add the dcm_vr_patterns application. This application creates DICOM files for testing transmission with implicit and explicit VRs.
6. Add a runtime configuration file for controlling transfer syntaxes and storage of composite objects that are received. Refer to CTN Runtime Configuration manual.

Version 2.12.0 November 16, 2000

1. Added support for MySQL database. This was contributed by one of our users. We have done minimal testing on Linux.
2. Repaired memory leaks and socket handle in DUL facility.
3. Repaired some errors concerning threaded applications. The COND facility had a section of code that was not thread safe. Also, the Thread facility itself had a resource leak in the W32 environment.
4. Updated DCM library to support Unlimited Text VR.
5. Added function DCM_GroupPresent. Tests to see if a group is found in a DCM object.
6. Added entries in group 0x0070 to help with Greyscale Presentation State objects.
7. Added support for storage of Greyscale Presentation State objects in the SRV facility, send_image, simple_storage, image archive.
8. Added support for storage of SR objects (Basic Text, Enhanced, Comprehensive) in SRV facility, send_image, simple_storage, image archive.
9. Added entries in the UID dictionary for Greyscale Presentation State storage class.

10. Updated DCM library and some applications for better support of VR Unlimited Text.
11. Updated application `dcm_modify_object` to handle the string `####` to signify an empty sequence.
12. Cleaned up some of the warning messages in the facilities. Many were caught by MSVC++.
13. Updated the MSVC++ project files to build Release versions of libraries/applications. Added more of the applications to the MSVC++ project file.
14. Add some file operations to UTL facility. These include a function to indicate if a path names a directory and a function to scan a directory and return the list of contents.
15. Corrected a memory reference error in `SRV_NCreateRequest`. We were releasing the memory for a structure and then tried to reference it.
16. Updated TBL functions which support SQL Server. The software uses an environment variable to generate login information for the ODBC connection. In previous versions, if the environment variable (`SQL_ACCESS`) was not set, we returned an error. Now we use a default value for making the connection.
17. Updated `dcm_dump_file` to search recursively through directories when dumping files. With the prior implementation, if the user specified a directory, the program tried to open it as a DICOM file and died.
18. Added the application `ctn_version` which prints the current version information.
19. Repaired the PostgreSQL configuration scripts for the image archive. There was an error in the table for patient level attributes.
20. Updated the Installation Guide. Moved much of the detailed information to the Appendix section. Simplified the installation procedure (we hope).

Version 2.11.3 May 12, 2000

1. Changed and documented new install procedure for Windows/MSVC++ 6.0.
2. Fix a problem in the dulprotocol library. For simple_storage and other server applications, we were not correctly closing sockets in the windows environment.

Version 2.11.2: March 17, 2000

1. Fixed a bug in dcm_dump_element when the output file was opened. The file is now opened in binary mode (which is important for those binary attributes like rows, columns, pixels).
2. Added environment and makefiles for Linux (Redhat 6.0) using lesstif as a Motif clone (www.lesstif.org).
3. Updated data dictionary according to Supplement 23. Added a few definitions from Supplement 33, but dictionary is incomplete.
4. Updated dcm_create_object to support the Date/Time VR.
5. Add -f switch to dcm_dump_file. This formatting switch indents sequences according to the level of nesting.

Version 2.11.1: January 26, 2000

1. Removed the definition of DCM_PATSSN from facilities/objects/dicom_objects.h. This was a mistake as DICOM never defined an attribute for Social Security Number.
2. Changed the creation of links in the facilities to libsrc directory from absolute path to relative path. This should make it easier to move the source tree.
3. Add support in parsing routines for DICOM VR of UN (unknown).
4. Add function DCM_MergeObject.
5. Update data dictionary with values that were useful in IHE Year 1.
6. Added a short delay in DUL_AcknowledgeRelease for Linux systems. They were closing the socket too soon. (Really means we need to take another look at our state machine.)
7. In the parse functions for N-Event Responses, removed our (incorrect) requirement that the peer application include the Affected SOP Instance UID.
8. In facilities/services, added support for MPPS and MWL classes.
9. In the application dcm_create_object, add support for a symbol to allow the creation of empty sequences.
10. Add two command line switches to send_image. The -p flag indicates that we create pixels (for a file without its own pixels); the -u flag allows us to replace the SOP Instance UID in the image.
11. In simple_storage, add support for Digital X-Ray for Presentation SOP Class.
12. Corrected errors in environment files for Linux/sybase and Linux/mSQL.
13. Added support for LessTif in the Linux environment. Created new set of environment files for the psql/LessTif combination. They have lesstif in the name.
14. Fixed a bug in facilities/xutl which caused a core dump when X-based applications could not load a font.
15. Added more documentation on PostgreSQL notes in the *Installation Guide*.

Version 2.11.0: January 7, 2000

1. Added `blg.h` to the distribution. This is not used in the standard release, but shows up in some of the files and is caught by `MSVC++`. It is easier to include the file in the release than to get `MSVC++` to ignore it.
2. Added entries in `services/get.c` which describe estimated images sizes for other modalities (e.g., PET).
3. Minor change to protocol of `TBL_Select`. The second argument of the callback function is now a long (rather than an int).
4. Added the test images to the `ctn` ftp site as described in the *Installation Guide*.
5. Corrected the default database name in the `storage_commit` application.
6. Added a compile-time flag to the linux master configuration file (`TIMEOFDAYARGS`).
7. Add support for PET images in the services facility. Add some estimates for image sizes for Ultrasound in the services facility.
8. Add facility numbers for IDBMB (Multibyte IDB), TBLMB (Multibyte TBL), and CHR (character manipulation).
9. General cleanup of arguments in functions. Lots of things that were declared `char*` and now `const char*`.
10. Added support in IDB to allow databases to be opened multiple times.
11. Added support for the PostgreSQL database (www.postgresql.org). Usage terms are more favorable than `mSQL`.
12. Added extra checks in IDB for meta-characters in search strings to support PostgreSQL. PostgreSQL handles the meta-characters and the like operator differently than Sybase and `miniSQL`.
13. Add a function `DCM_GetString` which can be used as a convenience function to get `ASCIZ` strings from `DCM_OBJECTS`.
14. Update the DCM parsing function. When resolving differences in VR, take the value from the input when they say OW (explicit xfer syntax).
15. Changed the Makefiles so the default installation does not build the GUIs. This has caused problems for the Linux users (who don't have Motif by default).
16. Correct memory leaks in the `archive_server` when using threaded or fork mode. Repair a problem that did not allow proper implementation of the fork model.
17. Changed default behavior of `archive_server` so that it deletes files that are not stored properly in the database. Previous code just left the file sitting in a directory with no database pointer to it.
18. Updated the storage service class (in facility SRV) to support all storage classes listed in PS 3.4, Table B.5-1 (1999). Likewise, updated `simple_storage` and `archive_server` to support those storage classes.
19. Added a new facility XUTL which is used to display text strings in X-Windows applications.
20. Changed the mechanism for building the CTN libraries. We now build one large library rather than one for each facility. Please refer to *Installation Guide*.
21. Separated the Motif applications from other applications in the build process. Please refer to *Installation Guide*.
22. Provide better support for Linux installation. We now build and test with Red Hat 6.0.