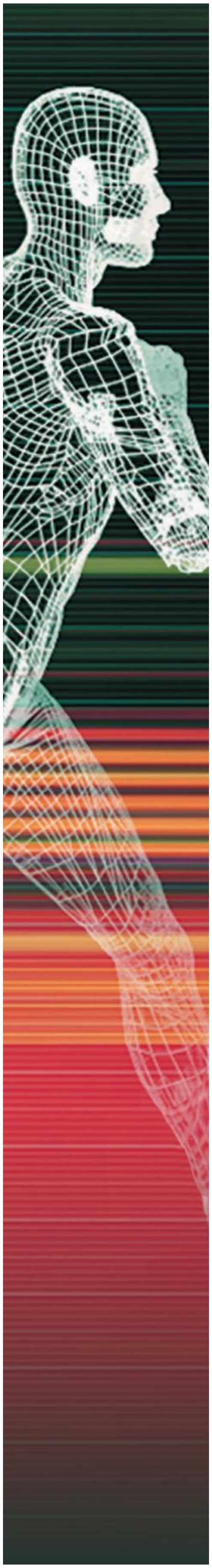
DICOM Connectivity Framework

Software Development Libraries for DICOM



The DICOM® Connectivity Framework (DCF) is an advanced, object-oriented collection of software components for implementing the DICOM v3.0 protocol for medical imaging systems. It includes a wide range of tools and utilities that support DICOM software development and testing. The DCF is the only DICOM solution that provides software developers a consistent API to C++, Java and C# and .Net, as well as support for Windows and Linux platforms.

The DCF also supports the DICONDE, DICOS, and veterinary customizations for utilizing DICOM.

Frustrated with your current DICOM Toolkit?

- Struggling because you don't have managed language support for your code development?
- Exasperated because you don't have comprehensive SOP class support?
- Disappointed that you can't select a different configuration for each association?
- Stressed trying to integrate your database with your toolkit?
- Frustrated trying to selectively insert, delete, edit, re-tag or encrypt DICOM attributes?
- Upset that you don't have real-time connection statistics monitoring and reporting capability?
- Overwhelmed trying to customize DICOM servers and clients in the field?

Solve these frustrations and more. Choose the DCF as your DICOM SDK.

The DCF enables a medical imaging system–modality, printer, scanner, archive, workstation–to communicate with other devices over a network, using the DICOM protocol. It allows an OEM to provide DICOM connectivity for their application or modality with a minimum of effort by providing:

- A robust, portable, high-performance implementation of DICOM protocol services
- A simple yet powerful API for communicating medical imaging information between the OEM's code and the DCF, such that detailed knowledge of the DICOM protocol is not required
- A customizable architecture that enables an OEM to fully integrate the DCF into their application

DCF Architecture

- Advanced, object-oriented, component-based architecture implemented in managed code
- Pluggable common services, such as logging and configuration management
- Multi-threaded implementation
- Component-based logging, debugging, and configuration

DCF Developer Support

- Example DICOM client and server programs for all major SOP classes
- Component-based framework facilitates rapid application development
- Multi-language support: C# .Net, Java, or C++
- Multi-platform support: Windows, Linux or other Unix platform
- Built-in DICOM compression support, optional integration with Accusoft, Aware & other high-performance JPEG compression libraries
- Online API and application documentation
- Web-based service, diagnostic, and configuration interfaces with real-time access and updates
- IOD object generation technology – Java, C++, or C# wrappers for DICOM IODs are auto-generated

V1.30

Comprehensive DICOM SOP Class Support – SCU and SCP support for all SOP classes is provided.

SCP	SCU	SOP Class
☑	☑	Verification (Echo)
☑	☑	Print
☑	☑	Print Job
☑	☑	Image Storage – Single-frame / Multi-frame (for all C-Store SOP classes)
☑	☑	Storage Commitment
☑	☑	Query Retrieve
☑	☑	Worklist (MWL, GPWL)
☑	☑	Scheduled / Performed Procedure Step (MPPS, GPPS)
Supported:		Media Storage (Part-10) – Removable media; roles: FSC, FSR, FSU
Supported:		DICOM compressed transfer syntaxes: RLE, JPEG, JPEG 2000
Supported:		Structured Reporting (SR)
Supported:		Presentation States (GSPS)

DICOM Transfer Syntax Support – Standard ILE, ELE, EBE encoding, plus RLE, JPEG and JPEG2000 support for lossy and lossless image transmission; flexibility: receive in one syntax, write to disk/network in another. Transfer Syntax Codec objects can be installed dynamically, allowing the use of other supported third-party compression libraries such as Accusoft and Aware.

Multiple Platform & Language Support – Includes access to all supported languages and platforms.

- Supports C# .NET, Java, and C++. Java components are pure Java and C# components are managed C#.
- Supports Windows 7 or 10, or Windows Server2008/2012 (32 or 64 bit) and supports .Net 2.0 or later.
- Supports Linux (x86 & x64).

Advanced API and Development Environment – Complements MS Visual Studio or Java Eclipse IDEs to allow rapid application development. Powerful yet simple APIs let you write enterprise class client (SCU) or server (SCP) applications with ease.

Advanced Dataset/DIMSE Message Filtering or Tag Morphing – Create custom filters, use DCF provided classes to perform common image manipulations, or field configurable insertion, deletion, editing, re-tagging, or encrypting of DICOM attributes. Perform text substitutions using regular expression syntax. Easily create or modify complex DICOM sequences on the fly. Automatically insert dynamically generated UID's and timestamps; automatically update the Original Attributes Sequence to track dataset change history.

Dataset Anonymization Support – a collection of C# .Net utilities is provided to assist in the consistent removal of patient identifying information from DICOM datasets; includes non-reversible, reversible, and encrypted anonymization options.

Per-Association Configuration – Dynamically select SCU or SCP settings based on flexible, OEM extensible rules. Configure access control, I/O parameters, filter sets and logging on a per-connection basis.

Pixel Data Streaming – streaming mode transfers allow very large datasets (> 2GB, e.g., multi-slice CT, pathology, cardiology) to be transferred between network and storage devices without allocating large blocks of memory.

Performance Tracking – Extensible DICOM connection and I/O statistics are maintained and can be accessed in real time.

Logging Facilities – The DCF provides advanced logging facilities. Use existing log output classes, or add your own. Troubleshoot low level DICOM connection problems, redirect selected messages to a separate log for HIPAA compliance, or send critical system error messages via email log forwarding, e.g., to a cell phone. Debug or tracing instrumentation is automatically generated for DCF or OEM code.

Configuration Object Data Base – A simple, powerful distributed object database stores configuration information in a hierarchical layout. Observer notifications allow high availability applications to dynamically adapt to configuration changes.

System Manager – Implement your own scheme, or use the dcf_sysmgr and AppControl API's to provide configurable startup and shutdown management for collections of related processes. Run as a normal application or install as a Windows Service.

Web-Based Service/Diagnostic Interface – View and manage log files, start and stop applications, adjust debug settings on running processes, or edit filter sets remotely using your web browser.

Extensive Scripting Support – Users can perform complex tasks quickly using numerous command line applications. A custom DICOM text file representation allows users to easily create or modify DICOM data sets without writing code.

Numerous Example Programs – Includes source code for all example programs, as well as for many other DICOM applications.