

IdmsQrServer

DICOM Query/Retrieve Server for Accuray IDMS

User & Administrator Manual

Version 1.0

TC Consulting

For clinical-support use on a trusted network — not a medical device

Important Notice

IdmsQrServer is an unofficial, third-party utility. It is not produced, endorsed, or supported by Accuray. It is not an FDA-cleared medical device and must not be used for primary diagnostic interpretation.

The server provides read-only access to images and RT objects already stored by the Accuray IDMS. It does not modify the IDMS software or the CDMS database. Even so, install and operate it only with the approval of your clinical physics and IT security staff, and only on a trusted clinical network.

This software is provided “as is” without warranty of any kind. The author is not liable for any damages arising from its use. Verify every retrieval against the source system before any clinical use.

Table of Contents

Important Notice	2
1 Overview & Purpose	5
2 How It Works.....	5
3 Requirements & Permissions	6
3.1 Software	6
3.2 Account permissions	6
4 Installation & Running.....	6
4.1 Console mode (interactive).....	6
4.2 Windows service (production)	7
4.3 Command reference	7
5 Configuration Reference	7
6 Connecting a DICOM Client.....	8
6.1 C-MOVE destinations	8
7 Firewall Configuration.....	8
7.1 Windows host firewall (on the IDMS)	9
7.2 Cisco ASA.....	9
7.3 Juniper SRX.....	10
7.4 Firewall troubleshooting	10
8 What the Server Exposes	12
8.1 Objects served.....	12
8.2 How DICOM query keys map to CDMS	12
8.3 Name matching	12
9 Verifying Operation.....	12
10 Logging & Auditing	13
11 Troubleshooting.....	13
11.1 Server accepts connections but never responds (console mode)	13
11.2 A retrieve reports “no negotiated storage context”	13
11.3 C-MOVE returns nothing / status 0xA801.....	13
11.4 Expected series are missing from a study	14
11.5 Blank or unexpected modality	14

11.6 Storage-vault (SMB/CIFS) files fail to retrieve	14
11.7 Reading the log	14
12 Security Considerations	14
Appendix A Supported SOP Classes & Transfer Syntaxes	15
A.1 Query/Retrieve & Verification SOP classes.....	15
A.2 Transfer syntaxes	15
Appendix B DIMSE Status Codes	15
Appendix C Quick Reference.....	15
C.1 Everyday commands	15
C.2 Firewall (IDMS at 192.168.193.3, TCP 10401)	16

1 Overview & Purpose

The Accuray IDMS (Integrated Data Management Server) stores treatment images and plan objects in a Microsoft SQL Server database named CDMS, with the image and plan files kept on disk. The IDMS itself does not offer a DICOM Query/Retrieve service, so standard DICOM clients cannot search for a patient and pull their images.

IdmsQrServer fills that gap. It is a small, self-contained Windows service that runs alongside the IDMS and answers standard DICOM Query/Retrieve requests by reading the CDMS database directly and serving the files the IDMS already stores.

Supported DICOM services:

- **C-ECHO** verification (“DICOM ping”).
- **C-FIND** Patient-Root and Study-Root queries at Patient, Study, Series and Image levels.
- **C-GET** returns matching instances on the same connection.
- **C-MOVE** pushes matching instances to a destination defined in the IDMS WORKSTATION table.

What it deliberately does not do:

- It never writes to CDMS or to the IDMS file store — access is read-only.
- It exposes only active (non-archived) patients by default.
- It is not a replacement for the IDMS and does not require any IDMS configuration change.

2 How It Works

IdmsQrServer listens on a TCP port (10401 by default). When a client connects, the server negotiates a DICOM association and then:

- For C-FIND, translates the query into SQL against CDMS (PATIENT, SERIES, DICOM_FILE, PLAN_RT_SERIES) and returns matching records.
- For C-GET / C-MOVE, resolves the matching files on disk (D:\cdms\files or the storage vault) and sends them to the requesting client or to the C-MOVE destination.

Files are sent in their stored (native) transfer syntax without re-compression, so every stored encoding is supported. When a client negotiates a different uncompressed little-endian encoding than the file on disk, the server transcodes between Explicit and Implicit VR Little Endian on the fly.

At a glance

DICOM client ↔ IdmsQrServer (port 10401) ↔ CDMS SQL database + IDMS file store. The server is the only new component; the IDMS and CDMS are untouched.

3 Requirements & Permissions

3.1 Software

- Windows Server (the IDMS host) with the .NET Framework 4.0 runtime (present on all supported IDMS builds).
- Network reachability from client workstations to the IDMS on the DICOM port (see Section 7, Firewall Configuration).

3.2 Account permissions

The identity that runs the server (the Windows service account, or the interactive user in console mode) must have:

- Read access to the CDMS database on the local SQL Server instance.
- Read access to the IDMS file store (typically D:\cdms\files).
- Read access to the storage vault, if archived-to-vault files (file_status = 3) are to be served (see the note below for SMB/CIFS vaults).

Storage vault on an SMB/CIFS share

When a file's storage location is a UNC path (\\host\share\...) rather than a local drive letter, the server automatically maps that share to a free drive letter before copying, using the built-in Accuray vault account and selecting the correct password by CDMS build (DB_BUILD). No manual drive mapping is required. If your vault uses different credentials, override them with VaultUser / VaultPassword in the configuration. Mapped drives are released when the service stops. The service account must be permitted to run net use and to reach the vault host over SMB (TCP 445).

Tip

The simplest working configuration uses Windows Integrated Security in the connection string and runs the service under an account that already has read access to CDMS. Avoid embedding SQL passwords where you can use integrated security instead.

4 Installation & Running

IdmsQrServer is a single executable, IdmsQrServer.exe, with a plain-text configuration file beside it. There is nothing to install into the operating system beyond the optional Windows service registration.

4.1 Console mode (interactive)

Run the executable directly to operate it in a console window. This is ideal for first-time setup and for watching live activity. Press Ctrl+C to stop.

IdmsQrServer.exe	(double-click, or run with no arguments)
IdmsQrServer.exe console	(force console mode)

4.2 Windows service (production)

For unattended operation, install it as a Windows service from an elevated (Administrator) command prompt:

```
IdmsQrServer.exe install
net start IdmsQrServer
```

To stop or remove the service:

```
net stop IdmsQrServer
IdmsQrServer.exe uninstall
```

4.3 Command reference

Command	Action
(no argument)	Console mode if interactive, otherwise runs as a service
console	Force console mode (Ctrl+C to stop)
install	Register the Windows service (run as Administrator)
uninstall	Remove the Windows service (run as Administrator)
test	Test the CDMS database connection and exit
selftest	Run offline internal self-checks and exit (no database needed)
help	Show usage help

Upgrading

To deploy a new build: net stop IdmsQrServer, copy the new IdmsQrServer.exe over the installed one, then net start IdmsQrServer. The configuration file and logs are preserved.

5 Configuration Reference

Settings are read from IdmsQrServer.ini, located in the same folder as the executable. The file is created automatically on first run with default values. Restart the server after editing it.

Setting	Default	Description
Port	10401	TCP port the server listens on
AETitle	IDMS_QR	The server's DICOM Application Entity title
ConnectionString	Server=localhost; Database=CDMS; Integrated Security=True; ...	ADO.NET connection string to the CDMS database
FilesRoot	(blank)	Override for the local file root. Blank = auto-detect D:\cdms\files or C:\cdms\files from the database build
LogDir	(blank)	Log folder. Blank = a 'logs' folder beside the exe
Debug	false	Set true for verbose logging
AllowedCallingAE	(blank)	Comma-separated allow-list of client AE titles. Blank = accept any client
VaultUser	(blank)	SMB user for a UNC storage vault. Blank = <vaulthost>\CDS

VaultPassword	(blank)	SMB password for the vault. Blank = selected automatically by DB_BUILD
----------------------	---------	--

Example IdmsQrServer.ini:

```
Port=10401
AETitle=IDMS_QR
ConnectionString=Server=localhost;Database=CDMS;Integrated
Security=True;TrustServerCertificate=True;
FilesRoot=
LogDir=
Debug=false
AllowedCallingAE=
VaultUser=
VaultPassword=
```

6 Connecting a DICOM Client

A client needs three things to talk to the server:

Parameter	Value
Host / IP	The IDMS server address (this manual uses 192.168.193.3)
Port	10401 (or your configured Port)
Called AE Title	IDMS_QR (or your configured AETitle)

The client's own Calling AE title may be anything, unless you have set AllowedCallingAE, in which case it must be on that list.

6.1 C-MOVE destinations

C-MOVE asks the server to send images to a named destination AE rather than back over the same connection. The server looks that destination up in the IDMS WORKSTATION table, so the destination must be registered there (ws_type = 1) with a matching AE title, IP address and port. If it is not, the server returns status 0xA801 ("move destination unknown").

C-GET vs C-MOVE

C-GET is self-contained — the client needs no listening port and no registration. C-MOVE needs the destination registered in WORKSTATION and a firewall path from the IDMS to that destination's DICOM port. When in doubt, use C-GET.

7 Firewall Configuration

For clients on other network segments to reach the server, inbound TCP traffic to the IDMS on the DICOM port must be permitted at two places: the Windows host firewall on the IDMS itself, and any network firewall between the client and the IDMS.

Throughout this section the IDMS server address is 192.168.193.3 and the DICOM port is TCP 10401. The Cisco ASA and Juniper SRX examples use the actual interface, zone, object and policy names from this site's CoreFW firewall configuration; adjust them if your device differs.

Stateful rule of thumb

DICOM Query/Retrieve rides on a single client-initiated TCP connection. On a stateful firewall you only need to permit the inbound connection to port 10401; return traffic is allowed automatically. The exception is C-MOVE, where the IDMS initiates a new outbound connection to the move destination's DICOM port — that path must be open too.

7.1 Windows host firewall (on the IDMS)

Add an inbound allow rule for TCP 10401. From an elevated command prompt:

```
netsh advfirewall firewall add rule name="DICOM QR (IDMS_QR) TCP 10401" dir=in action=allow
protocol=TCP localport=10401
```

Or with PowerShell:

```
New-NetFirewallRule -DisplayName "DICOM QR (IDMS_QR) TCP 10401" -Direction Inbound -Action
Allow -Protocol TCP -LocalPort 10401
```

You can scope the rule to your client subnet for tighter control, e.g. add remoteip=192.168.1.0/24 to the netsh rule.

7.2 Cisco ASA

On this site's ASA (named CoreFW), the interface that faces the clients is outside and the IDMS sits behind the inside interface. The IDMS is already defined as the object IDMS (host 192.168.193.3), and the service group object-group service DICOM already lists TCP 10401 among the DICOM ports.

The inbound access rule is already present too — it permits the DICOM source group to reach the IDMS on the DICOM ports and is applied inbound on the outside interface:

```
access-list outside_access_in extended permit tcp object-group DICOM_Servers object IDMS
object-group DICOM
access-group outside_access_in in interface outside
```

Two things to check or add:

- Source address — make sure the client's IP is included in the permitted source group (object-group network DICOM_Servers); add it if it is not.
- Static NAT for 10401 — the other IDMS DICOM ports are individually port-forwarded on the outside interface, but 10401 is not. Add it to match the existing pattern:

```
object network IDMS-10401
host 192.168.193.3
nat (inside,outside) static interface service tcp 10401 10401
```

```
!  
write memory
```

The reverse path for C-MOVE (the IDMS initiating a connection out to an external node) is already permitted by access-list `inside_access_out` for the DICOM service group.

Clients on the treatment network

If the querying client is on the inside (treatment) side with the IDMS — the 192.168.192.0/24 through 192.168.194.0/24 networks — it reaches the IDMS directly and no ASA rule or NAT is required. The rule and NAT above are only for clients arriving through the outside interface.

7.3 Juniper SRX

On this site's SRX (named CoreFW), clients arrive in the untrust zone (interface `ge-0/0/0`, "outside") and the IDMS is in the trust zone (interface `ge-0/0/1`, "inside"), where it is defined as the address IDMS (192.168.193.3/32).

DICOM Query/Retrieve on 10401 is already provisioned in three places:

- Application — application-set DICOM includes application 10401 (protocol tcp, destination-port 10401).
- Policy — from-zone untrust to-zone trust policy DICOM1 permits the DICOM source addresses to reach destination-address IDMS with application DICOM.
- Static NAT — rule DICOM10401 maps the untrust address :10401 to 192.168.193.3:10401.

In most cases, therefore, no change is required — verify with the commands in Section 7.4. The outbound path for C-MOVE is likewise already permitted by policy DICOM0 (from-zone trust to-zone untrust). If you ever need to expose an additional Q/R port, define the application and add it to the DICOM set, then commit:

```
set applications application 10402 protocol tcp destination-port 10402  
set applications application-set DICOM application 10402  
commit
```

Also add a NAT and confirm the source

A new inbound port also needs a matching static-nat rule (modeled on DICOM10401) and the client's address present in the policy's source — the DICOM_Server or Customer_MP address sets.

7.4 Firewall troubleshooting

Work outward from the host. Confirm each hop before moving to the next.

Step 1 — Is the server listening on the IDMS?

On the IDMS, confirm the port is open locally:

```
netstat -ano | findstr :10401
```

You should see a LISTENING entry on 0.0.0.0:10401. If not, the server is not running or is configured for a different port — check the log and IdmsQrServer.ini.

Step 2 — Can a client reach the port?

From a client workstation (PowerShell):

```
Test-NetConnection -ComputerName 192.168.193.3 -Port 10401
```

TcpTestSucceeded : True means the whole path — host firewall and every network firewall — is open. If it is False, a firewall or route is blocking; continue with the device-specific checks below.

Cisco ASA checks

- Confirm 10401 is in the DICOM service group: show run object-group id DICOM
- Verify the inbound rule and its hit count: show access-list outside_access_in | include 10401
- Confirm the 10401 static NAT exists: show nat detail | include 10401
- Trace a client's packet through the ASA — the destination is the outside interface address (10.230.13.140), which is port-forwarded to the IDMS; expect a final ALLOW:

```
packet-tracer input outside tcp 10.230.13.200 12345 10.230.13.140 10401
```

- See active connections: show conn address 192.168.193.3 port 10401
- Watch drops in real time: show logging | include 192.168.193.3

Juniper SRX checks

- Confirm which policy a flow matches (expect DICOMI). Use the IDMS real address as the destination — policy is evaluated after destination NAT:

```
show security match-policies from-zone untrust to-zone trust source-ip 10.210.117.200  
destination-ip 192.168.193.3 source-port 12345 destination-port 10401 protocol tcp
```

- Verify the policy and its hit count: show security policies from-zone untrust to-zone trust policy-name DICOMI detail
- Confirm the destination NAT: show security nat static rule DICOM10401
- See active sessions: show security flow session destination-port 10401
- Confirm the interface-to-zone mapping: show security zones

Asymmetric routing

A common cause of “connects but hangs” is the IDMS returning replies by a different path than the firewall expects. Ensure the IDMS default gateway routes client-bound traffic back through the same firewall that permitted the inbound connection.

Step 3 — Does DICOM itself respond?

Once the port is reachable, verify the DICOM layer with a C-ECHO from the client (see Section 9). A successful C-ECHO confirms firewall, server and database are all in order for queries.

8 What the Server Exposes

The server presents the IDMS's stored data as standard DICOM. Only active patients (archive_id IS NULL) are exposed by default.

8.1 Objects served

- **Image series** CT, MR and PET series (from SERIES / DICOM_FILE), including planning CT, daily kVCT and reconstructed MVCT ("CTtrue").
- **RT objects** RT Structure Set, RT Dose, RT Plan and RT Record (from PLAN_RT_SERIES).

8.2 How DICOM query keys map to CDMS

DICOM attribute	CDMS source
Patient ID	PATIENT.med_id
Patient Name	PATIENT.last_name ^ first_name
Patient Birth Date / Sex	PATIENT.dob / gender
Study Instance UID	SERIES.study_uid / PLAN_RT_SERIES.study_uid
Series Instance UID	SERIES.series_uid / PLAN_RT_SERIES.series_uid
Modality	SERIES.modality_desc, falling back to the modality code (1=CT, 4=MR); RT objects from the plan modality
Study Date / Series Date	SERIES.scan_dte

Modality of kVCT / MVCT

Some Accuray series (daily kVCT, reconstructed MVCT) leave modality_desc empty. The server reports these as CT based on the numeric modality code, so they appear correctly in queries and are retrievable.

8.3 Name matching

Patient-name queries are case-insensitive and match on a partial (substring) basis against either the family or given name, so a fragment of a first or last name returns all matching patients. Explicit DICOM wildcards (* and ?) are also honored.

9 Verifying Operation

Use these checks, in order, after installing or changing configuration.

1. Database: run `IdmsQrServer.exe test` — it should report the active patient count. A failure here means the connection string or SQL permissions need attention.
2. Internals: run `IdmsQrServer.exe selftest` — all checks should pass. This needs no database and validates the DICOM encoding internally.

3. Connectivity: from a client, confirm TCP reachability (Section 7.4, Step 2).
4. DICOM ping: perform a C-ECHO from the client to AE IDMS_QR at 192.168.193.3:10401 — a success status confirms end-to-end operation.
5. Query: perform a Study-level C-FIND (for example by patient name) and confirm the expected studies are returned.

10 Logging & Auditing

The server writes a dated log file to the log folder (a 'logs' folder beside the executable unless LogDir is set), named idmsqr-YYYYMMDD.log, one per day.

Each association and request is logged. Retrievals additionally produce an audit line identifying the patient whose data was transferred — useful for access records:

```
C-GET AUDIT: PatientID='MED12345' PatientName='DOE^JOHN' level=STUDY StudyUID=1.2.840...  
requestedBy=DICOMRTVIEWER  
C-GET complete for PatientID 'MED12345': 465 ok, 0 failed, 0 warning
```

For C-MOVE, the audit line also records the destination AE. Set Debug=true in the configuration for additional detail while diagnosing an issue.

11 Troubleshooting

11.1 Server accepts connections but never responds (console mode)

Older behavior: clicking inside the console window activates Windows “QuickEdit” selection, which freezes console output and could stall the server. Current builds disable QuickEdit automatically and log on a background thread, so this no longer occurs. Running as a service avoids console effects entirely and is recommended for production.

11.2 A retrieve reports “no negotiated storage context”

This means the client did not offer to receive that image type in the encoding it is stored in. The server sends files as stored; for uncompressed data it will transcode between Explicit and Implicit VR Little Endian, but it cannot decompress compressed pixel data (for example JPEG Lossless). Resolve it in one of two ways:

- Have the client negotiate the stored transfer syntax (including JPEG Lossless) and decode it locally;
or
- Use C-MOVE to a destination that accepts that transfer syntax.

11.3 C-MOVE returns nothing / status 0xA801

The destination AE is not recognized. Register it in the IDMS WORKSTATION table (ws_type = 1) with the correct AE title, IP and port, and ensure the IDMS can reach that destination’s DICOM port through the firewall (Section 7).

11.4 Expected series are missing from a study

Retrieving a whole study returns every series it contains. If a client only shows a study-level summary, individual series (such as each daily kVCT) may not be visible even though they transfer. Use a client that supports Series-level query and retrieve to list and pull individual series; the server supports Series-level C-FIND, C-GET and C-MOVE.

11.5 Blank or unexpected modality

Series with an empty modality_desc are reported by their numeric modality code (CT or MR). If a modality still looks wrong, verify the underlying SERIES row in CDMS.

11.6 Storage-vault (SMB/CIFS) files fail to retrieve

If files that live on the storage vault (file_status = 3, a \\host\share path) fail while local files succeed, the share mapping is the likely cause. The log records the mapping attempt and any net use error. Check:

- The service account may run net use and can reach the vault host over SMB (TCP 445).
- The vault credentials are correct — override VaultUser / VaultPassword in the configuration if the automatic (DB_BUILD-based) defaults do not match your vault.
- A free drive letter is available for the server to map the share to.

11.7 Reading the log

The daily log is the first place to look. Key lines: association accept, the C-FIND / C-GET / C-MOVE request with its level, the AUDIT line, and any WARN/ERROR entries. Include the relevant lines when reporting a problem.

12 Security Considerations

- The server provides read-only access to patient data and must be treated as sensitive clinical infrastructure.
- It does not use TLS; deploy it only on a trusted, segmented clinical network, never exposed to the internet.
- Restrict which clients may connect using AllowedCallingAE and/or firewall scoping (Section 7).
- Retain and review the audit logs in line with your institution's access-logging policy.

Appendix A Supported SOP Classes & Transfer Syntaxes

A.1 Query/Retrieve & Verification SOP classes

Role	SOP Class	UID
SCP	Verification	1.2.840.10008.1.1
SCP	Patient Root Q/R – FIND	1.2.840.10008.5.1.4.1.2.1.1
SCP	Patient Root Q/R – MOVE	1.2.840.10008.5.1.4.1.2.1.2
SCP	Patient Root Q/R – GET	1.2.840.10008.5.1.4.1.2.1.3
SCP	Study Root Q/R – FIND	1.2.840.10008.5.1.4.1.2.2.1
SCP	Study Root Q/R – MOVE	1.2.840.10008.5.1.4.1.2.2.2
SCP	Study Root Q/R – GET	1.2.840.10008.5.1.4.1.2.2.3

A.2 Transfer syntaxes

- Query/Retrieve associations: Implicit VR Little Endian (1.2.840.10008.1.2) and Explicit VR Little Endian (1.2.840.10008.1.2.1).
- Storage (C-GET / C-MOVE): files are sent verbatim in whatever transfer syntax they are stored in — all syntaxes are supported for transfer.
- On C-GET, the server transcodes between Explicit and Implicit VR Little Endian when the client negotiated a different uncompressed encoding than the stored file. Compressed syntaxes are transferred but not decompressed.

Appendix B DIMSE Status Codes

Status	Meaning
0x0000	Success
0xFF00	Pending (a C-FIND match, or a retrieve sub-operation in progress)
0xB000	Sub-operations complete — one or more failures (warning)
0xA702	Refused — unable to perform sub-operations (all failed)
0xA801	Refused — move destination unknown
0xC000	Unable to process (for example a database error)
0x0122	SOP class not supported

Appendix C Quick Reference

C.1 Everyday commands

```
IdmsQrServer.exe install      # register the service (as Administrator)
net start IdmsQrServer        # start it
net stop IdmsQrServer         # stop it
IdmsQrServer.exe uninstall    # remove the service
IdmsQrServer.exe test         # test the CDMS connection
IdmsQrServer.exe selftest     # offline self-check
```

C.2 Firewall (IDMS at 192.168.193.3, TCP 10401)

Windows host:

```
netsh advfirewall firewall add rule name="DICOM QR (IDMS_QR) TCP 10401" dir=in action=allow  
protocol=TCP localport=10401
```

Cisco ASA — the DICOM group and outside ACL already cover 10401; add the one missing port-forward:

```
object network IDMS-10401  
  host 192.168.193.3  
  nat (inside,outside) static interface service tcp 10401 10401  
write memory
```

Juniper SRX — already provisioned (application-set DICOM has 10401, policy DICOMI permits it, static NAT DICOM10401 maps it); verify with:

```
show security match-policies from-zone untrust to-zone trust source-ip <client> destination-  
ip 192.168.193.3 source-port 12345 destination-port 10401 protocol tcp  
show security nat static rule DICOM10401
```