Image Access Everywhere
DICOM Web Services

James F Philbin, PhD
Johns Hopkins University
Co-Director, Center for Biomedical & Imaging Informatics
Baltimore, Maryland, USA

Co-Chair WG-27; Member WG-06
Co-Authors: Tim Culp, Tim Dawson, Jonathan Whitby
Introduction

Web Access for DICOM Objects
- WADO-URI
- WADO-WS
- WADO-RS

Store Over the Web

Query by IDs for DICOM Objects

Putting it all together
Why Do We Use DICOM?

- A good standard that continues to evolve
- Rich, Structured Information Model
- Compact representation
- Language independent
- Widely supported
Because no specific Ad Hoc Group on Biomedical Imaging will be set up in ISO / TC215, new works on medical imaging must be done in DICOM (with a Category A Liaison Group between both)

More and more it will be important that DICOM makes recommendations on the medical imaging aspects within non «pure» DICOM protocols

*Courtesy of Emmanuel Cordonnier*
A Brief History

- **WADO(-URI)**
  1999 - 2003
  - IHE XDS-l.a

- **WADO-WS**
  2008-2011
  - IHE XDS-l.b

- **MINT**
  2010 – 2011 v1.0
  2011 – 2013 v2.0

- **WADO-RS**
  2011 - 2013

- **STOW-RS**
  2011 – in progress

- **QIDO-RS**
  2012 – in progress
Web Access for (Persistent) DICOM Objects

WADO
WADO: Web Access to DICOM (persistent) Objects

1. **WADO-URI**
   HTTP Universal Resource Identifier w/ parameters

2. **WADO-WS**
   SOAP & WS-* family of specifications

3. **WADO-RS**
   RESTful services
WADO-URI

WADO Using

Universal Resource Identifiers
WADO-URI

- HTTP Universal Resource Identifier
- Parameters
- Supports DICOM & Rendered images
- Used in XDS-I.a (deprecated)
- GET one image at a time
Syntax of the WADO HTTP GET method

- Syntax defined by the RFC2396 (URI)
  - `http://<authority><path>?<query>`
  - e.g:
    - `http://www.hosp.fr/dicom/wado.asp?studyUID=1...`

- The « Web Access to DICOM Persistent Object » standard defines only the `<query>`
WADO-URI Use Cases

- Embedding images in reports or other documents by URL
- Browser based viewing of DICOM images for clinical use (one image at a time)
WADO-WS

WADO Using WS-* Services
• SOAP + WS-*  (*Only HTTP POST*)
• Support DICOM & Rendered images
  ▪ Retrieve DICOM objects (studies, series, instances) referenced by an XDS-I Imaging Document Set
  ▪ Retrieve metadata (XML) for specified instances
  ▪ Either DICOM or Rendered objects
  ▪ Filtered metadata using XPath parameter
What is SOAP?
Simple Object Access Protocol

- Invented by Microsoft in 1998
- RPC for Client / Server
- Any transport protocol
  e.g. HTTP, SMTP, TCP, JMS
- XML based messages
- WSDL: Web Services Description Language
- **WS-**: WS-Addressing, WS-Policy, WS-Security, WS-Federation,
  WS-ReliableMessaging, WS-Coordination,
  WS-AtomicTransaction, WS-RemotePortlets...
Structured XML SOAP envelope

```xml
<env:Envelope
xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <!-- Header information here -->
  </env:Header>
  <env:Body>
    <!-- Body or "Payload" here, Fault if error -->
  </env:Body>
</env:Envelope>
```
WADO-WS Use Cases

- Accessing images and reports from another clinical application such as an EMR
- Embedding images in reports or other documents by URL
- Providing access to anonymized DICOM images for research and teaching.
- Providing access to images & reports for remote diagnostic workflows using the Web.
WADO-RS

WADO Using RESTful Services
What is REST?

- Representational State Transfer\(^1\)
- RESTful services follow certain principles
  - Addressable Resources (URI)
  - Uniform, Constrained Interface (HTTP)
  - Representation oriented (content negotiation)
  - Stateless
  - Hypermedia*

1. Coined by Roy Fielding, one of the principal authors of the HTTP, in his PhD thesis.
SOAP vs. REST

SOAP

- Structured XML SOAP envelope

```xml
<env:Envelope
xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <!-- Header information here -->
  </env:Header>
  <env:Body>
    <!-- Body or "Payload" here, Fault if error -->
  </env:Body>
</env:Envelope>
```

REST

- Supports hypermedia, multiple MIME types
SOAP vs. REST

**SOAP**
- Structured XML SOAP envelope
- WSDL
- Transport agnostic
- Exception/fault in payload
- Steep learning curve, tools required
- Heavy weight; requires significant client and server logic

**REST**
- Supports hypermedia, multiple MIME types
- Manual documentation
- HTTP specific
- HTTP error codes
- Easy to develop, no tools required
- Lightweight; accessible via JavaScript and the simplest web frameworks
- Just HTTP 1.1 with Resources, 
  http:{service}/studies/{UID}
- Multiple objects per study
- **No support** for rendered images
- Metadata can be retrieved separately
- Headers specify transfer syntax & encodings:
  - DICOM Binary Instances
  - XML Metadata
  - Pixel data for Instances or Frames
A set of one or more DICOM objects at a time
An XML metadata
A set of bulk data as a multi-part response

Coming soon?... *Normalized metadata*
## WADO-RS API

<table>
<thead>
<tr>
<th>Name</th>
<th>Operator</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>RetrieveStudy</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}</td>
</tr>
<tr>
<td>RetrieveSeries</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}</td>
</tr>
<tr>
<td>RetrieveInstance</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}</td>
</tr>
<tr>
<td>RetrieveFrames</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames/{FrameList}</td>
</tr>
<tr>
<td>RetrieveBulkData</td>
<td>GET</td>
<td>{BulkDataURL}</td>
</tr>
<tr>
<td>RetrieveMetadata</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/metadata</td>
</tr>
</tbody>
</table>

All API are prefixed by `{SERVICE}/studies`, e.g. [http://10.0.0.10:8001/studies](http://10.0.0.10:8001/studies), or [http://somehost/archive/studies](http://somehost/archive/studies)
The response format depends on the Accept header of the request

- **Accept: multipart/related; type=image/dicom+xml**
  Specifies that the response should be DICOM XML metadata

- **Accept: multipart/related; type=application/dicom**
  Specifies that the response should be DICOM Instances encoded in PS3.10 format.

- **Accept: multipart/related; type=application/octet-stream**
  Specifies that the response can include uncompressed bulk or pixel data.

- **Accept: multipart/related; type=image/***
  Specifies that the response can include be compressed pixel data.

- **Accept: multipart/related; type=video/***
  Specifies that the response can include video-encoded pixel data.
WADO-RS Use Cases

- Retrieve DICOM (whole studies, entire series, or instances) by UID
- Retrieve metadata (whole studies, entire series or instances) in one set
- Retrieve pixel and other bulk data in one message regardless of whether the original study used multi-frame or multiple single-frame instances
STOW-RS
STOW-RS Capabilities

- Upload DICOM objects to the server (whole studies, append series / individual instances)
- Update/replace existing DICOM objects on server
<table>
<thead>
<tr>
<th>Name</th>
<th>Operator</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>StoreInstances</td>
<td>POST</td>
<td>/studies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/studies/{StudyInstanceUID}</td>
</tr>
</tbody>
</table>

All API are prefixed by {SERVICE}/studies, e.g. [http://10.0.0.10:8001/studies](http://10.0.0.10:8001/studies), or [http://somehost/archive/studies](http://somehost/archive/studies)
STOW Formats

- `application/dicom`
  Specifies that the post is DICOM

- `application/dicom+xml`
  Specifies that the post is WADO XML metadata and bulk data
STOW Use Cases

- Viewer adding measurements, GSPS, etc.
- DICOM routers from modalities to PACS, VNA
- Securely push studies from outside firewall (e.g. between institutions)
- Web based QC tools (STOW.v2)
QIDO-RS
Query by IDs for DICOM Objects
QIDO Capabilities

- Look up DICOM objects based on any metadata, to get additional metadata and UIDs
- Pagination – search with offset & limit
<table>
<thead>
<tr>
<th>Name</th>
<th>Operator</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>SearchForStudy</td>
<td>GET</td>
<td>/studies</td>
</tr>
<tr>
<td>SearchForSeries</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series</td>
</tr>
<tr>
<td>SearchForInstance</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/instances</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances</td>
</tr>
</tbody>
</table>

All API are prefixed by {SERVICE}/studies, e.g. [http://10.0.0.10:8001/studies](http://10.0.0.10:8001/studies), or [http://somehost/archive/studies](http://somehost/archive/studies)
QIDO Request Parameters

- `{dicomAttribute} = {keyValue}`
- `includefield = {dicomAttribute}`
- `{dicomAttribute}` can be a Tag or Key Word as defined in DICOM PS3.6 or a sequence of the form `{dicomSequenceAttribute}.{dicomAttribute}`
- Parameters can be included 0-n times
- `limit = {maxResults}`
- `offset = {num of results to skip}`
QIDO Use Cases

- Basic study search (patient, date range, type, description)
- Locate and filter for relevant priors
- Identify any new studies received
Putting It All Together
Putting it together…

- Simple URL-based search using QIDO
- Parse XML or JSON response
- Launch zero footprint viewer for selected study
- Download metadata and subset of images
- Download rest of study in background
- Upload new evidence / artifacts
- Upload entire studies
Use Case – Thin Clients

- Web Client
- Mobile App
- Windows App

QIDO WADO Proxy STOW

C-FIND
C-MOVE
C-STORE

PACS
Use Case – Remote Access

Site 1

Client

QIDO
WADO
STOW
Proxy

QIDO
WADO
STOW
Proxy

C-FIND
C-MOVE
C-STORE

PACS

Site 2

Image Access Everywhere, James Philbin, PhD
Use Case – Cloud Sharing

Client

Referring Portal

- QIDO
- WADO
- Cache
- STOW

Teaching File

Patient Portal

Client
Other Use Cases:

1. Physician referrals
2. ED transfer
3. Cloud Storage
4. Viewer Worklist
5. Integration with FHIR / HL7 RESTful services
Author Contacts

- James F Philbin
  james.philbin@jhmi.edu
  5801 Smith Ave.
  Baltimore, MD 21209, USA
- Tim Culp
  timothy.culp@harris.com
- Tim Dawson
  tdawson@vitalimages.com
- Jonathan Whitby
  jwhitby@vitalimages.com

Thank you for your attention!
Backup Slides
DICOM & RESTful Services

- **WADO-RS**
  Web Access to DICOM Objects by RESTful Services
  Supplement 161 – Final Text

- **STOW**
  STore DICOM Over the Web
  Supplement 163 – Public Comment

- **QIDO**
  Query Interface to DICOM Objects
  Supplement 166 – Work In Progress

- **NADO**
  Notification for Availability of DICOM Objects
  Future – under discussion

- **Other:** Normalized study level XML and JSON
DICOM doesn’t define security mechanism

- Typical HTTP security mechanisms
  - Basic (over SSL)
  - Digest
  - Client certificate
  - Kerberos
  - Windows NT LAN Manager
- IHE-IUA: Internet User Authentication for HTTP profiles
  - Likely to be OAuth 2.0
WADO-RS
Additional Use Cases

- Study Launch
  1. Retrieve Metadata (one request)
  2. Determine images to display
  3. Retrieve initial image(s)
  4. Retrieve remaining images (e.g. by series)
<table>
<thead>
<tr>
<th>Name</th>
<th>Operator</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>RetrieveStudy</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}</td>
</tr>
<tr>
<td>RetrieveSeries</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}</td>
</tr>
<tr>
<td>RetrieveInstance</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}</td>
</tr>
<tr>
<td>RetrieveFrames</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames/{FrameList}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames?calculated=…</td>
</tr>
<tr>
<td>RetrieveBulkData</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/bulkdata/{BulkDataID}?offset={Offset}&amp;length={Length}</td>
</tr>
<tr>
<td>RetrieveMetadata</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/metadata</td>
</tr>
<tr>
<td>Name</td>
<td>Operator</td>
<td>Resource</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>RetrieveStudy</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}</td>
</tr>
<tr>
<td>RetrieveSeries</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}</td>
</tr>
<tr>
<td>RetrieveInstance</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}</td>
</tr>
<tr>
<td>RetrieveFrames</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames/{FrameList}</td>
</tr>
<tr>
<td>RetrieveBulkData</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/bulkdata/{BulkDataID}?offset={Offset}&amp;length={Length}</td>
</tr>
<tr>
<td>RetrieveMetadata</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/metadata</td>
</tr>
<tr>
<td>Name</td>
<td>Operator</td>
<td>Resource</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RetrieveStudy</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}</td>
</tr>
<tr>
<td>RetrieveSeries</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}</td>
</tr>
<tr>
<td>RetrieveInstance</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}</td>
</tr>
<tr>
<td>RetrieveFrames</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames/{FrameList}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames?calculated=…</td>
</tr>
<tr>
<td>RetrieveBulkData</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/bulkdata/{BulkDataID}?offset={Offset}&amp;length={Length}</td>
</tr>
<tr>
<td>RetrieveMetadata</td>
<td>GET</td>
<td>/studies/{StudyInstanceUID}/metadata</td>
</tr>
</tbody>
</table>
## Retrieve Types

<table>
<thead>
<tr>
<th></th>
<th>Study</th>
<th>Series</th>
<th>Instance</th>
<th>Rendered</th>
</tr>
</thead>
<tbody>
<tr>
<td>WADO-URI</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>WADO-WS</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>WADO-RS</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Service Status</td>
<td>DICOM Response</td>
<td>Code</td>
<td>HTTP1.1 Status Codes</td>
<td>STOW-RS Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------</td>
<td>------</td>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Failure</td>
<td>Refused: Out of Resources</td>
<td>A7xx</td>
<td>503 – Busy</td>
<td>This indicates that the STOW-RS Service was unable to store the instances because it was out of resources.</td>
</tr>
<tr>
<td></td>
<td>Error: Data Set does not match SOP Class</td>
<td>A9xx</td>
<td>422 – Unprocessable Entity</td>
<td>This indicates that the STOW-RS Service was unable to store the instances because the Data Set does not match the SOP Class.</td>
</tr>
<tr>
<td></td>
<td>Refused: Not Authorized</td>
<td>C1xx</td>
<td>401 - Unauthorized</td>
<td>This indicates that the STOW-RS Service refused to create or append instances because the client is not authorized.</td>
</tr>
<tr>
<td></td>
<td>Failed</td>
<td>C1xx</td>
<td>403 - Forbidden</td>
<td>This indicates that the STOW-RS Service understood the request, but is refusing to fulfill it (e.g. an authorized user with insufficient privileges).</td>
</tr>
<tr>
<td></td>
<td>Error: Cannot understand</td>
<td>Cxxx</td>
<td>400 – Bad Request</td>
<td>This indicates that the STOW-RS Service was unable to store the instances because it cannot understand certain Data Elements.</td>
</tr>
<tr>
<td>Warning</td>
<td>Coercion of Data Elements</td>
<td>B000</td>
<td>202 – Accepted</td>
<td>This indicates that the STOW-RS Service modified one or more data elements on reception. Image transmission is considered successful.</td>
</tr>
<tr>
<td></td>
<td>Data Set does not match SOP Class</td>
<td>B007</td>
<td>202 – Accepted</td>
<td>This indicates that the STOW-RS Service encountered instances where the Data Set did not match the SOP Class. Image transmission is considered successful.</td>
</tr>
<tr>
<td></td>
<td>Elements Discarded</td>
<td>B006</td>
<td>202 – Accepted</td>
<td>This indicates that the STOW-RS Service discarded some data elements. Image transmission is considered successful.</td>
</tr>
<tr>
<td>Success</td>
<td></td>
<td>0000</td>
<td>200 – OK</td>
<td>This indicates that the STOW-RS Service successfully stored the instances.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0000</td>
<td>201 – Created</td>
<td>This indicates that the STOW-RS Service successfully stored the instances to a new study and the response indicates the URI for the newly created study.</td>
</tr>
</tbody>
</table>
QIDO Response

- **DICOM PS3.19 XML**
  one message part per matching Study / Series / Instance

- **JSON**
  one message part with all matching Studies / Series / Instances

- **For each entry:**
  - response must include all Study Level attributes listed as Required for SCPs in IHE RAD TF Table 4.14-1
  - response should include any attributes used as query keys if used as matching criteria
  - response should include all attributes requested in includefield parameters if possible