

PowerCast Integration Guide

OVERVIEW

FHIRcast is a [proposed HL7 standard](#) for real-time context synchronization between healthcare applications. For example, a radiologist typically works in three disparate applications at the same time (PowerScribe, PACS, EMR) and wants each of these systems to display the same study or patient simultaneously.

In addition to basic synchronization, our implementation builds upon FHIRcast to support real-time structured data exchange between PowerScribe One and other applications such as PACS, EMR, workflow orchestrators, and AI providers.

Our FHIRcast support includes our PowerScribe Connector as a FHIRcast client (and PowerScribe gateway) and of course our FHIRcast Hub server. The total solution is referred to as "PowerCast".

This document is intended for our business partners that choose to use PowerCast to integrate client applications with PowerScribe One.

Terms/Glossary

- The PACS/RIS/EHR vendor client applications are herein referred to as **CLIENT** or **CLIENTS**.
- The PowerCast Hub is herein referred to as **HUB**.
- The PowerCast Connector is herein referred to as **CONNECTOR** and is included collectively with **CLIENTS**.
- The Auth0 OAuth2/OpenID Identity Server is herein referred to as **AUTH0**.

POWERCAST HUB RESPONSIBILITIES

FHIRCast Hub

The PowerCast Hub is a FHIRCast Hub, and responsible for providing the functioned defined in the current specification.

Context Management

The PowerCast Hub keeps track of open resources in a FHIRCast topic session, and makes them available in a REST API. To do so, it requires the following:

1. Clients adhere to a FHIRCast event naming convention as follows: <FHIR Resource>-[open|update|close].
2. Clients will use the -open event to open a new resource which becomes the context (for this FHIR resource type).
3. When a -close event is received, the Hub no longer retains the context for that resource.
4. The -update event message must follow the specification for DiagnosticReport-update documented below. The -update event contains a FHIR Bundle which is a collection of transactions that are applied to the current context.
5. Clients will now receive a response payload in the Subscribe method, containing the current context. For Websocket subscriptions, this will be a change since the Subscribe method previously return a string value with the websocket endpoint URL.

FHIR Server

Finally, the Hub is responsible for storing large resources that would be too large to distribute to every subscriber inline. Currently, PowerCast only caches the Media FHIR resource.

- See the **DiagnosticReport-update** section to see an example of a Media resource sent inline in a context change request
- See **Get Topic** to see how these resources are referenced in context.
- See **Get Resource** which describes the "FHIR Server" API used to retrieve the media resource.

DIAGNOSTIC REPORT-CENTERED WORKFLOW

When reporting applications integrate with PACS and/or RIS applications, the radiologists workflow is centered around the Radiologist's final deliverable: the diagnostic report. While the imaging study (exam) is an integral resource, the report encapsulates one or more of these studies. Also, the diagnostic report contains certain attributes (such as report status), that are useful to the PACS/RIS applications and don't exist in the imaging study. Structured data, usually in the form of an Observation resource can also be encapsulated in the report.

FHIRCast Notification Request Event Fields

F ie ld	O pt io n a l ity	T y pe	Description
hu b. to pic	Re qui red	str ing	The session topic given in the subscription request.
hu b. ev ent	Re qui red	str ing	The event that triggered this notification (see list of supported events below).
co nt ext	Re qui red	ar ray	An array of named FHIR objects corresponding to the user's context after the given event has occurred. The contents of this field will vary depending on the event. However, given the diagnostic report-centered workflow supported by PowerCast, context will usually be a DiagnosticReport FHIR resource, with other FHIR resources associated with that report embedded in its contained field.

PowerCast and PowerScribe One supported events

Event	Description
DiagnosticReport-open	This notification is used to begin a new report. This should be the first event for a topic session.
DiagnosticReport-update	This notification is used to make changes (updates) to the current report. These changes usually include adding/removing imaging studies and/or observations to the current report.
DiagnosticReport-select	This notification is sent to tell subscribers to make one or more images or observations visible (in focus), such as a measurement (or other finding).
DiagnosticReport-close	This notification is used to close the current report. This should be the last event for a topic session.
ImagingStudy-open	This notification can be used by subscribers to open new report in PSOne.
ImagingStudy-close	This notification can be used by subscribers to close the current open report in PSOne.

PowerScribe One affirmative events

Event	Description
DiagnosticReport-opened	Affirmative event, to notify subscribers that report opened successfully in PSOne, in response to DiagnosticReport-open event from a subscriber.
DiagnosticReport-closed	Affirmative event, to notify subscribers that current open report has been closed successfully in PSOne, in response to DiagnosticReport-close event from a subscriber.

The PowerCast Hub is designed to be "agnostic" in the sense that it performs very little data validation. It does not enforce an event catalogue (any event name is valid), nor does it validate the correctness of FHIR resources contained in those events. Since events themselves are not fixed, and any vendor or group of vendors can decide to create and use event names as they like, PowerCast makes no attempt to interpret these events.

PowerCast is currently implemented per the STU1 specifications, as well as subsequent changes not yet documented, but [voted](#) by the DICOM working group 20. PowerCast also supports the [WebSocket Proposal](#) for FHIRcast clients.

DISCOVERY, AUTHORIZATION AND LOGIN

Discovery

Nuance will provide a FHIR Authorization Endpoint and Capabilities Discovery using a Well-Known Uniform Resource Identifiers (URIs) similar to (but not conformant to) the SMART on FHIR specification as defined [here](#). At this time, the host will be the CONNECTOR (on the local desktop, not a cloud server).

In addition to the URIs, the discovery endpoint will return the FHIRCast topic if a user is logged in to PowerScribe. Otherwise the topic value will be null.

NOTE: This operation could take 30 seconds or longer to complete, since the CONNECTOR obtains the information from PowerScribe. If PowerScribe is not running, the CONNECTOR will launch it.

Example request and result:

```
GET /configuration HTTP/1.1
Host: localhost:9292 HTTP/1.1 200 OK
Content-Type: application/json
{
  "authorization_endpoint": "https://nuancehce.auth0.com/oauth/authorize",
  "token_endpoint": "https://nuancehce.auth0.com/oauth/token",
  "hub_endpoint": "https://connect.nuancepowerscribe.com/powercast/api
/hub",
  "topic" : ""
}
```

CORS

We are currently keeping track of whitelisted domains that are allowed to make cross-domain request to Connector endpoints.

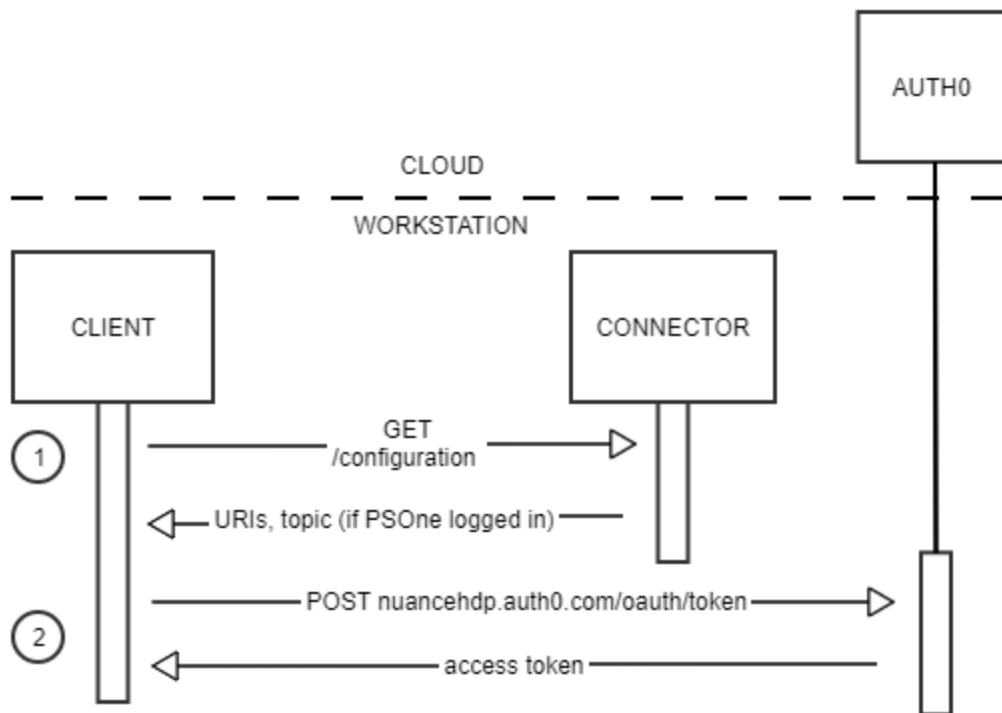
User can add domains to the whitelist through the RadPortal. They can do this through the System tab from the PowerCast section in the Connector CORS Domain(s) text box. Multiple domains can be added by separating them with a semi-colon.

The screenshot shows the Nuance RadPortal interface. The top navigation bar includes tabs for Setup, Accounts, AutoText, Fields, Bridge, Procedures, Sites, Speech, and System. The System tab is selected. Below the navigation bar, there are links for Configuration..., Review Ratings..., Clinical Guidance..., Follow-up Activities..., Structured Findings..., and Audit... The main content area is titled 'Save Changes | Preferences... | Templates... | Import Persons...'. It contains a form with various configuration options. The 'Connector CORS Domain(s)' field is highlighted in yellow and contains the value 'https://www.test-cors.org'.

When a request is made to one of the Connector endpoints (health check, configuration or login) we first check the origin of the request. If the origin is included in the whitelist, then we append the origin to Access-Control-Allow-Origin header and set Access-Control-Allow-Credentials to true in the http listener response. This allows users to access the connector endpoints from across domains when they otherwise wouldn't be able to.

Authorization

The CLIENT obtains an authorization token from the AUTH0 token endpoint retrieved in the discovery (see above). Nuance provides a client ID and secret to each application vendor. This is used as the OAuth2 "Authorization Code" which is subsequently used to obtain the access token from the Nuance Authentication server. This type of OAuth2 authentication is referred to as "Client Credentials Flow". See the AUTH0 explanation here: <https://auth0.com/docs/flows/concepts/client-credentials#how-it-works>



Example request and result:

```

GET /oauth/token HTTP/1.1
Host: https://nuancehdp.auth0.com
{
  { "grant_type", "client_credentials" },
  { "client_id", "your_id" },
  { "client_secret", "your_secret" },
  { "audience", "https://nuancehdp.com/PowerCast/Hub" }
}
HTTP/1.1 200 OK
Content-Type: application/json
{
  { "access_token", "encoded_access_token_string" },
  { "expires_in", "86400" },
  { "token_type", "Bearer" }
}
  
```

Login

Once the CLIENT has completed the authentication process, and obtains the FHIRCast topic it can subscribe to the HUB. If the topic was not returned in the discovery process, that means that a user is not logged in to PowerScribe. The CLIENT has two options:

1. Poll the CONNECTOR continuously on the **configuration** endpoint, waiting for a user to login, or...
2. Log a user into PowerScribe using the **login** endpoint on the CONNECTOR.

Authorization **is** required for the **login** method. The access token is sent in the HTTP Authorization header as "bearer".

CLIENT Issues an HTTP GET to the CONNECTOR endpoint to log in a user on PowerScribe One.

Example request and result:

```
GET/login?username=drXRay HTTP/1.1
Authorization Bearer oiwerualskjf34zx897ldh394...
Host: https://localhost:9292
HTTP/1.1 200 OK
Content-Type: application/json

{
  "authorization_endpoint": "https://nuancehce.auth0.com/oauth/authorize",
  "token_endpoint": "https://nuancehce.auth0.com/oauth/token",
  "hub_endpoint": "https://connect.nuancepowerscribe.com/powercast/api
/hub",
  "topic", "B3668641BF717FF17B39F3A2B48C5.drXRay"
}
```

POWERCASST HUB REST ENDPOINTS

The PowerCast hub supports the following REST endpoints. For more detail on FHIRCast Hub Rest endpoints, see the [complete specification here](#).

Subscribe/Unsubscribe

HTTP Method: GET

Endpoint: base-hub-URL

Returns on Success:

Status Code will be 202. Response content will contain a JSON object with the websocket endpoint (for websocket, not rest-hook subscriptions), and the current context. If there is no current context, the result will be an empty array ("[]").

```
HTTP/1.1 202 Accepted
Content-Type: application/json

{
  "hub.channel.endpoint": "wss://connect2.nuancepowerscribe.com/ws
/aslkdjqweiucxvno",
  "contexts": [
    ... see example contexts in Get Topic below
  ]
}
```

Returns on Failure:

Status Code will be 4xx/5xx. The response content will contain a JSON object containing (minimally) a message. Example:

```
HTTP/1.1 400 BAD REQUEST
Content-Type: application/json

{
  "message": "Reason for failure/error
message."
}
```

Event Notification

HTTP Method: POST

Endpoint: base-hub-URL/topic

Returns on Success:

Status Code will be 200.

Returns on Failure:

Status Code will be 4xx/5xx. The response content will contain a JSON object containing (minimally) a message. Example:

```
HTTP/1.1 400 BAD REQUEST
Content-Type: application/json

{
  "message": "Context is missing the contextual resource of type
DiagnosticReport."
}
```

Get Context

HTTP Method: Get

Endpoint: base-hub-URL/<topic>

Returns: This method returns an object containing the current context of the topic session. The current context is made up of one or more "top-level" contextual resource types (like ImagingStudy or DiagnosticReport. The contextType field identifies how the context was created. For example, a DiagnosticReport-open event will create a new context with contextType=DiagnosticReport. This is considered the "anchor context."

Each resource is listed in Key/Resource pairs to follow the FHIRCast spec for event notifications.

Example: Context of type DiagnosticReport which includes

- ImagingStudy
- Patient (referenced by ImagingStudy)
- DiagnosticReport
- Observation
- Media (reference)

NOTE on Media Reference: The reference will contain the absolute URL of the resource. The client application that wishes to retrieve the resource will obtain the resource from the Hub FHIR service as described in the section **Get Resource** section further on in this document.

```
[
  {
    "contextType": "DiagnosticReport",
    "context": [
      {
        "key": "patient",
        "resource": {
          "resourceType": "Patient",
          "id": "4b6de2b1f1c343888c2fffb751ccfb349",
          "identifier": [
            {
              "system": "urn:oid:1.2.840.114350",
              "value": "185444"
            }
          ]
        }
      ]
    },
    {
      "key": "study",
      "resource": {
        "resourceType": "ImagingStudy",
        "description": "CHEST XRAY",
        "started": "2010-01-30T23:00:00.000Z",
        "status": "available",
        "id": "aa6c6aa3875d41e0b0e7222bfa553b2a",
        "identifier": [
          {

```

```
0203",
    "type": {
      "coding": [
        {
          "system": "http://terminology.hl7.org/CodeSystem/v2-
          "code": "ACSN"
        }
      ]
    },
    "value": "342123458"
  },
  "patient": {
    "reference": "Patient/4b6de2b1f1c343888c2fffb751ccfb349"
  }
},
{
  "key": "Report",
  "resource": {
    "resourceType": "DiagnosticReport",
    "id": "40012366",
    "status": "unknown",
    "meta": {
      "versionId": "0"
    },
    "imagingStudy": [
      {
        "reference": "ImagingStudy/8i7tbu6fby5ftfbku6fniuf"
      }
    ]
  }
},
{
  "resourceType": "Observation",
  "id": "4b6de2b1f1c343888c2fffb751ccfb349",
  "identifier": [
    {
      "system": "dcm:121151",
      "value": "L1"
    }
  ],
  "status": "preliminary",
  "issued": "2001-07-23T06:02:11-04:00",
  "component": [
    {
      "code": {
        "coding": [
          {
            "system": "https://loinc.org",
```

```

        "code": "21889-1",
        "display": "Distance"
      }
    ],
    },
    "valueQuantity": {
      "system": "http://unitsofmeasure.org",
      "value": "30.3134634578934",
      "code": "mm",
      "unit": "mm"
    }
  },
  ],
  "category": {
    "system": "http://terminology.hl7.org/CodeSystem/observation-
category",
    "code": "imaging",
    "display": "Imaging"
  },
  "code": {
    "system": "http://hl7.org/fhir/ValueSet/observation-codes",
    "code": "32449-1",
    "display": "Physical findings of Lung"
  }
},
{
  "key" : "Media"
  "reference" : "https://powercast.nuancepowerscribe.com
/B3668641BF717FF17B39F3A2B48C5.drXRay/resource/Media
/8e7519362aeb4c10885e03a7287fedaf"
}
]
}
]

```

Get Resource

HTTP Method: Get

Endpoint: base-hub-URL/<topic>/resource/<resource type>/<resource id>

Returns: This method returns a cached FHIR Resource.

NOTES:

- Currently, only Media resources are cached.
- FHIR Resources are cached and then referenced in Hub notification events

POWERSCRIBE SUPPORTED FHIRCAST EVENTS

The following event names are supported by the CONNECTOR (and therefor PowerScribe). The HUB has no limitation on events - any event name is allowed.

DiagnosticReport-open

This event will be posted to PowerCast when a new or existing report is opened in PowerScribe One.

The event may also be posted by a PACS/RIS (or other application) in which case, PowerScribe One will open a new report (or existing report as an addendum) using the accession number(s) found in the ImagingStudy FHIR resource.

The **context** field must contain at least three resources: Patient, ImagingStudy and DiagnosticReport. Only ONE DiagnosticReport can be present in the context. The DiagnosticReport will reference one or more ImagingStudy resources. The ImagingStudy resource(s) must reference a single Patient.

PowerScribe One will support both "current" studies as well as "prior" studies (used for comparison). If a study is not to be considered a "current" study being reported on, it must contain the **status field**, with a value of "available".

Example DiagnosticReport-open

```
{
  "timestamp": "2019-09-10T14:58:45.988Z",
  "id": "0d4c9998",
  "event": {
    "hub.topic": "DrXRray",
    "hub.event": "DiagnosticReport-open",
    "context": [
      {
        "key": "patient",
        "resource": {
          "resourceType": "Patient",
          "id": "4b6de2b1f1c343888c2ffb751ccfb349",
          "identifier": [
            {
              "system": "urn:oid:1.2.840.114350",
              "value": "185444"
            }
          ]
        }
      ]
    },
    {
      "key": "study",
      "resource": {
        "resourceType": "ImagingStudy",
        "description": "CHEST XRAY",
        "started": "2010-01-30T23:00:00.000Z",
        "status": "available",
        "id": "aa6c6aa3875d41e0b0e7222bfa553b2a",
        "identifier": [
          {
            "type": {
              "coding": [
                {
                  "system": "http://terminology.hl7.org/CodeSystem/v2-0203",
                  "code": "ACSN"
                }
              ]
            }
          ]
        ]
      }
    ]
  }
}
```

```

        },
        "value": "342123458"
      }
    ],
    "patient": {
      "reference": "Patient/4b6de2b1f1c343888c2fffb751ccfb349"
    }
  },
  {
    "key": "Report",
    "resource": {
      "resourceType": "DiagnosticReport",
      "id": "4b6de2b1f1c343888c2fffb751ccfb349",
      "status": "unknown",
      "imagingStudy": [
        {
          "reference": "ImagingStudy
/aa6c6aa3875d41e0b0e7222bfa553b2a"
        }
      ]
    }
  ]
}

```

DiagnosticReport-update

Once a report has been opened via the DiagnosticReport-open event, one or more updates may occur before the report is closed (saved and/or signed). The updates could include (but are not limited to) any of the following:

- Adding or removing active imaging studies
- Adding or removing comparison imaging studies
- Changing report status or other DiagnosticReport meta-data
- Adding or removing observations and/or media

Since the Hub will need to update the contents of its "retained context resource", the update event message must follow a standard where the context portion of the message will contain two resources:

1. Context resource: for risk mitigation, this would be the current context resource. Note that only the id and versionId are necessary.
2. Bundle of transactions containing updates.

Other implementation notes:

- The transaction methods (PUT, POST, DELETE) are implicitly associated with the current context resource by the FHIRCast Hub. That is, if the event is a DiagnosticReport-update, the transactions are to be applied to the retained DiagnosticReport context resource.
- The Hub will apply the transactions blindly without regard for the contents of the resources. No validation is performed by the hub.
- The Hub will be responsible for incrementing the value of the resource's **versionId** after each update. The versionID will be "0" when the original is retained during the resource-open event.
- When receiving an update event, the Hub will examine the versionId of the inbound contextual resource (the DiagnosticReport) and compare it with the current version maintained by the Hub. If the versions do not match, the message will be rejected.

Supported Resource Types in the Transaction Bundle:

Resource Type	Comments
---------------	----------

DiagnosticReport	PowerScribe only supports a PUT request on the current report (in context). The DiagnosticReport resource id MUST match the current report. The purpose of this operation is to allow other report attributes (such as report status) to be updated.
ImagingStudy	Both current studies and prior studies (for comparison) are supported in the same manner as the DiagnosticReport-open event (see previous section). All request types (PUT, POST, DELETE) are supported.
Observation	All request types (PUT, POST, DELETE) are supported.

Supported Transaction Request Types

Request Type	Operation/Comments
PUT	Replace/update an existing resource
POST	Add a new resource
DELETE	Remove an existing resource

Example Message

Three transaction requests: (a) Prior study added for comparison, (b) measurements added as a result observation, and (c) Media resource containing an image.

```
{
  "timestamp": "2019-09-10T14:58:45.988Z",
  "id": "0d4c9998",
  "event": {
    "hub.topic": "DrXRay",
    "hub.event": "DiagnosticReport-update",
    "context": [
      {
        "key": "report",
        "resource": {
          "resourceType": "DiagnosticReport",
          "id": "8e7519362aeb4c10885e03a7287fedaf",
          "meta": {
            "versionId": "0"
          }
        }
      }
    ],
  },
  {
    "key": "updates",
    "resource": {
      "resourceType": "Bundle",
      "id": "0c5a96f01aa743158823020af641c49b",
      "type": "transaction",
      "entry": [
        {
          "request": {
            "method": "POST"
          },
          "resource": {
            "resourceType": "ImagingStudy",
```

```

        "description": "CHEST XRAY",
        "started": "2010-01-30T23:00:00.000Z",
        "status": "available",
        "id": "4b6de2b1f1c343888c2ffb751ccfb349",
        "identifier": [
            {
                "type": {
                    "coding": [
                        {
                            "system": "http://terminology.hl7.org
/CodeSystem/v2-0203",
                            "code": "ACSN"
                        }
                    ]
                },
                "value": "342123458"
            }
        ],
    },
    {
        "request": {
            "method": "POST"
        },
        "resource": {
            "resourceType": "Media",
            "id": "aa6c6aa3875d41e0b0e7222bfa553b2a",
            "content": {
                "contentType": "image/jpeg",
                "data": "/9j/4...KAP//Z",
                "title": "CT ABDOMEN"
            }
        },
    },
    {
        "request": {
            "method": "POST"
        },
        "resource": {
            "resourceType": "Observation",
            "id": "4b6de2b1f1c343888c2ffb751ccfb349",
            "status": "preliminary",
            "category": {
                "system": "http://terminology.hl7.org/CodeSystem
/observation-category",
                "code": "imaging",
                "display": "Imaging"
            },
            "code": {
                "coding": [

```

```

        {
            "system": "http://hl7.org/fhir/ValueSet
/observation-codes",
            "code": "10193-1",
            "display": "Physical findings of Breasts
Narrative"
        }
    ],
    },
    "issued": "2019-05-05T13:28:17.239-05:00",
    "identifier": [
        {
            "system": "dcm:121151",
            "value": "Lesion-1"
        }
    ],
    "derivedFrom": [{
        "type": "ImagingStudy",
        "reference": "ImagingStudy
/4b6de2b1f1c343888c2ffb751ccfb349"
    }],
    "component": [
        {
            "code": {
                "coding": [
                    {
                        "system": "https://loinc.org",
                        "code": "21889-1",
                        "display": "Size Tumor"
                    }
                ]
            },
            "valueQuantity": {
                "value": "13.3",
                "unit": "mm",
                "system": "http://unitsofmeasure.org",
                "code": "mm"
            }
        },
        {
            "code": {
                "coding": [
                    {
                        "system": "dcm",
                        "code": "121242",
                        "display": "Distance from Nipple"
                    }
                ]
            },
            "valueQuantity": {

```

```

        "value": "60",
        "unit": "mm",
        "system": "http://unitsofmeasure.org",
        "code": "mm"
      }
    ]
  }
}

```

NOTE ON REFERENCING: This example above is a context change request, and therefore includes the full Media resource inline. When the Hub broadcasts this update to the topic subscribers, it will reference the Media like this:

```

{
  "key" : "Media"
  "resource" :
  {
    "reference" : "https://powercast.
nuancepowerscribe.com/resource/Media/8e7519362aeb4c10885e03a7287fedaf"
  }
}

```

DiagnosticReport-select

This notification event is sent by a client indicating one or more FHIR resources that are to be made visible, in focus or otherwise "selected".

In this example, the sender is requesting that the Observation with id **1.2.276.0.45.1.7.4.1097371716104** be selected. NOTE: The context will always include the report resource. While it is assumed that this observation is contained in the current DiagnosticReport context, the Hub DOES NOT provide validation.

```

{
  "timestamp": "2022-05-31T16:42:34.4033329Z",
  "id": "PSOneFHIRProxy-847861d77bf1414b941534157102571b",
  "event": {
    "hub.topic": "Ashok.admin",
    "hub.event": "DiagnosticReport-select",
    "context": [
      {
        "key": "report",
        "resource": {
          "resourceType": "DiagnosticReport",
          "id": "2a951fa3a85f4aea94ea119b97ad9ec8",

```

```

        "meta": {
          "versionId": "1"
        }
      },
      {
        "key": "Observation",
        "resource": {
          "resourceType": "Observation",
          "id": "1.2.276.0.45.1.7.4.1097371716104",
          "status": "preliminary",
          "code": {
            "coding": [
              {
                "system": "http://hl7.org/fhir/ValueSet/observation-
codes",
                "code": "10193-1",
                "display": "Physical findings of Breasts Narrative"
              }
            ],
            "text": "Mammographic distance from nipple"
          },
          "component": [
            {
              "code": {
                "coding": [
                  {
                    "system": "dcm",
                    "code": "121242",
                    "display": "Distance from Nipple"
                  }
                ]
              },
              "valueQuantity": {
                "value": 37.3936004638672,
                "unit": "mm",
                "system": "http://unitsofmeasure.org",
                "code": "mm"
              }
            },
            {
              "code": {
                "coding": [
                  {
                    "system": "http://nuancepowerscribe.com/ai",
                    "code": "ai-finding-disposition",
                    "display": "AI Finding Disposition"
                  }
                ]
              }
            }
          ]
        }
      },
    ]
  },

```

```

    "valueString": "accepted"
  }
]
}
}

```

DiagnosticReport-close

When this event is received, PowerScribe One closes the current report. When a report is saved and signed in PowerScribeOne, it posts this event to PowerCast.

Example (report signed and closed in PowerScribeOne).

```

{
  "timestamp": "2019-09-10T14:58:45.988Z",
  "id": "0d4c9998",
  "event": {
    "hub.topic": "DrXRray",
    "hub.event": "DiagnosticReport-close",
    "context": [
      {
        "key": "Report",
        "resource": {
          "resourceType":
"DiagnosticReport",
          "id": "40012366",
          "status": "final",
          "issued": "2019-09-13T14:58:45"
        }
      }
    ]
  }
}

```

ImagingStudy-open

When this event is received, PSONe opens a new report for the accession, if there is no current open report. If the report is opened successfully, PSONe sends out "DiagnosticReport-open" event.

In case there is already a current open report, PSONe will ignore this event and sends out "sync-error" event.

```

{
  "timestamp": "2022-07-07T22:21:24.4192763Z",
  "id": "TestPowerCast-7b9faaafa1bd4b1b9eac9abfb79c9a26",
  "event": {
    "hub.topic": "Ashok.admin",

```

```
"hub.event": "ImagingStudy-open",
"context.versionId": "0c89a415-8bbb-41a4-a7c4-52594f7fa596",
"context": [
  {
    "key": "patient",
    "resource": {
      "resourceType": "Patient",
      "id": "f78f7295f3e74b248e2d5a78419b70d9",
      "identifier": [
        {
          "use": "usual",
          "type": {
            "coding": [
              {
                "system": "http://terminology.hl7.org/CodeSystem/v2-
0203",
                "code": "MR"
              }
            ]
          },
          "value": "mrn123"
        }
      ]
    },
  },
  {
    "key": "study",
    "resource": {
      "resourceType": "ImagingStudy",
      "id": "f18fe5bbea454e5aa0ab17a1d79a6353",
      "identifier": [
        {
          "use": "usual",
          "type": {
            "coding": [
              {
                "system": "http://terminology.hl7.org/CodeSystem/v2-
0203",
                "code": "ACSN"
              }
            ]
          },
          "value": "acc123"
        }
      ],
      "subject": {
        "reference": "Patient/f78f7295f3e74b248e2d5a78419b70d9"
      },
      "meta": {
        "versionId": "0"
      }
    }
  }
]
```

```

    }
  }
]
}

```

ImagingStudy-close

When this event is received, PSONe closes the current open report. A previously open and in context study is no longer open nor in context. If the report is closed successfully, PSONe sends out "DiagnosticReport-close" event.

```

{
  "timestamp": "2022-07-07T22:27:58.6584709Z",
  "id": "TestPowerCast-e508125fc687434a9a372f1d2278ec7c",
  "event": {
    "hub.topic": "Ashok.admin",
    "hub.event": "ImagingStudy-close",
    "context": [
      {
        "key": "study",
        "resource": {
          "resourceType": "ImagingStudy",
          "id": "f18fe5bbea454e5aa0ab17a1d79a6353"
        }
      }
    ]
  }
}

```

DiagnosticReport-opened

Users can subscribe to this event to get affirmation that report has been opened successfully in PSONe, for the "DiagnosticReport-open" event that they sent.

```

{
  "timestamp": "2022-07-07T22:35:57.8120641Z",
  "id": "PSOneFHIRProxy-e97d7f5647b646a6a9ed2eea8b86bce2",
  "event": {
    "hub.topic": "Ashok.admin",
    "hub.event": "DiagnosticReport-opened",
    "context": [
      {
        "key": "OperationOutcome",
        "resource": {
          "resourceType": "OperationOutcome",
          "issue": [
            {
              "severity": "information",
              "diagnostics": "Report with accession(s) 'acc123' opened."
            }
          ]
        }
      }
    ]
  }
}

```

DiagnosticReport-closed

Users can subscribe to this event to get affirmation that current open report has been closed successfully in PSOne, for the "DiagnosticReport-close" event that they sent.

```
{
  "timestamp": "2022-07-07T22:47:53.0624113Z",
  "id": "PSOneFHIRProxy-4578677040db4d0d8c551a586e84b092",
  "event": {
    "hub.topic": "Ashok.admin",
    "hub.event": "DiagnosticReport-closed",
    "context": [
      {
        "key": "OperationOutcome",
        "resource": {
          "resourceType": "OperationOutcome",
          "issue": [
            {
              "severity": "information",
              "diagnostics": "Report closed"
            }
          ]
        }
      }
    ]
  }
}
```

SyncError

This event can be raised either from PSOne if the incoming event is not process successfully, Or subscribers can notify syncerror to PowerCast Hub which is then broadcasted to the other subscribers of that topic.

When PowerScribe One receive the sync-error notification, it will display message to the user.

It is the responsibility of PSOne (PowerScribeConnector) / vendor applications to subscribe to "syncerror" hub.event, for getting notified when an error is posted to PowerCast hub.

It is the responsibility of the vendors to notify hub about any error that occur during their processing. Below is the sample format of json content, that vendors need to send in the Http request body for Notify API request.

Example of syncerror notification when processing "DiagnosticReport-open" event is unsuccessful:

```
{
  "timestamp": "2022-07-07T22:49:10.4998007Z",
  "id": "PSOneFHIRProxy-d2ab5a67465e48b2850ad58f11a4c319",
  "event": {
    "hub.topic": "Ashok.admin",
    "hub.event": "syncerror",
    "context": [
      {
        "key": "OperationOutcome",
        "resource": {
```

```

    "resourceType": "OperationOutcome",
    "issue": [
      {
        "severity": "error",
        "code": "processing",
        "details": {
          "coding": [
            {
              "system": "https://fhircast.hl7.org/events/syncerror
/eventid",
              "code": "TestPowerCast-
9a1ee6c2d1614837b7273248e85749fa"
            },
            {
              "system": "https://fhircast.hl7.org/events/syncerror
/eventname",
              "code": "DiagnosticReport-open"
            },
            {
              "system": "https://fhircast.hl7.org/events/syncerror
/vendorname",
              "code": "PowerScribe"
            },
            {
              "system": "http://example.com/events/syncerror
/errorcode",
              "code": "Internal Error."
            }
          ]
        },
        "diagnostics": "Failed on DiagnosticReport-open event. A
report with accession 'acc123' is already opened in PSOne, not opening
new report for accession 'acc127'."
      }
    ]
  }
}

```

Example (syncerror notification) message from subscriber:

```

{
  "timestamp": "2018-01-08T01:37:05.14",
  "id": "1cfcd7c017cf472abe341ee4283992db",
  "event": {
    "hub.topic": "7544fe65-ea26-44b5-835d-14287e46390b",
    "hub.event": "syncerror",
    "context": [
      {
        "key": "operationoutcome",
        "resource": {
          "resourceType": "OperationOutcome",
          "issue": [
            {
              "severity": "warning",
              "code": "processing",
              "diagnostics": "Additional information to aid subsequent
investigation or presentation to the end-user.",
              "details": {
                "coding": [
                  {
                    "system": "https://fhircast.hl7.org/events/syncerror
/eventid",
                    "code": "fdb2f928-5546-4f52-87a0-0648e9ded065"
                  },
                  {
                    "system": "https://fhircast.hl7.org/events/syncerror
/eventname",
                    "code": "study-save"
                  },
                  {
                    "system": "https://fhircast.hl7.org/events/syncerror
/vendorname",
                    "code": "eUnity"
                  },
                  {
                    "system": "http://example.com/events/syncerror
/errorcode",
                    "code": "500 - Internal Server Error"
                  }
                ]
              }
            }
          ]
        }
      }
    ]
  }
}

```

Field	Optionality	Type	Description
context	Required	array	An array containing a single FHIR OperationOutcome. And context's key should be operationoutcome. The OperationOutcome shall use a code of processing. The OperationOutcome's details shall contain the id of the event that this error is related to as a code with the system value of https://fhircast.hl7.org/events/syncerror/eventid The name of the relevant event where the issue occurred, with a system value of https://fhircast.hl7.org/events/syncerror/eventname. Other required coding values can be included with different system values so as to include extra information about the syncerror. • The name of the vendor who raised the event with a system value of https://fhircast.hl7.org/events/syncerror/vendorname. • The error info with a system value of https://fhircast.hl7.org/events/syncerror/errorcode. The OperationOutcome's diagnostics element (required) should contain additional information to aid subsequent investigation or presentation to the end-user.
event	Required	object	A JSON object describing the event. See below .
hub.event	Required	string	Shall be the string syncerror .
hub.topic	Required	string	The session topic given in the subscription request.
id	Required	string	Event identifier, which MAY be used to recognize retried notifications. This id SHALL be unique and could be a UUID.
timestamp	Required	string	ISO 8601-2 timestamp in UTC describing the time at which the syncerror event occurred.

User-Logout

When this event is received by PowerScribe One, it will:

1. Close any reports currently open.
2. Unsubscribe from the topic
3. Log off the current user.

EXAMPLES

Obtain Token (C#)

```

private async Task<string> GetToken()
{
    const string CLIENT_ID = "3lFjkSFf9iljs9uW45lFd9uwr1asd90uw";
    const string CLIENT_SECRET = "asldkj309ulaskdj3094asdklj-lskDFJj394lakjdf30szj";
    const string AUDIENCE = "https://nuancehdp.com/PowerCast/Hub";

    string token = null;
    HttpClient client = new HttpClient();
    HttpResponseMessage response;
    UriBuilder urlBuilder = new UriBuilder($"https://nuancehdp.auth0.com/oauth/token");
    HttpRequestMessage request = new HttpRequestMessage(HttpMethod.Post, urlBuilder.Uri);
    Dictionary<string, string> auth = new Dictionary<string, string>
    {
        { "grant_type", "client_credentials" },
        { "client_id", CLIENT_ID },
        { "client_secret", CLIENT_SECRET },
        { "audience", AUDIENCE }
    };
    FormUrlEncodedContent enc = new FormUrlEncodedContent(auth);
    request.Content = enc;
    response = await client.SendAsync(request);
    if (!response.IsSuccessStatusCode)
    {
        Log($"Unable to obtain access token from Auth0: {(int)response.StatusCode} - {response.ReasonPhrase}");
    }
    else
    {
        {
            string s = await response.Content.ReadAsStringAsync();
            var jwt = JObject.Parse(s);
            token = jwt["access_token"].ToString();
        }
        return token;
    }
}

```

Subscribe to PowerCast(C#)

```

private async void Subscribe()
{
    UriBuilder urlBuilder = new UriBuilder($"https://connect2.nuancepowerscribe.com/powercast/api/hub");
    HttpRequestMessage request = new HttpRequestMessage(HttpMethod.Post, urlBuilder.Uri);
    request.Headers.Authorization = new AuthenticationHeaderValue("Bearer", _accessToken); // obtained earlier.
    Dictionary<string, string> hub = new Dictionary<string, string>
    {
        { "hub.channel.type", "websocket" },
        { "hub.callback", "" },
        { "hub.mode", "subscribe" },
        { "hub.topic", "drXRray" },
        { "hub.events", "DiagnosticReport-open,DiagnosticReport-close,DiagnosticReport-update,syncerror" },
        { "hub.secret", "mySeCreT" }
    };
    FormUrlEncodedContent enc = new FormUrlEncodedContent(hub);
    request.Content = enc;
    response = await client.SendAsync(request);
    //
    // the rest of the code to connect to the web socket goes here
    //
}

```

Note: For better user experience and integration between PSOne and subscribers the minimum events that need to be subscribed are: DiagnosticReport-open, DiagnosticReport-close, DiagnosticReport-update, and syncerror. These events are required at the minimum for opening/closing/updating report in PSOne and also to receive error notification in case of error while processing one of the above events.

Useful References

Subject	Link
Obtaining an access token (multiple languages)	https://auth0.com/docs/quickstart/backend/aspnet-core-webapi/02-using#obtaining-an-access-token
OAuth2 Client Flow	https://auth0.com/docs/flows/concepts/client-credentials#how-it-works