



INTERWEAVE
CONNECTING CARE

Cookbook for Regional Interoperability
Detailed Design Paper #002

Data Availability Service

INTERNAL DISCUSSION DRAFT

Version 1.1 – 24th January 2019

Abstract Interoperability Cookbook Anchor Points

Section	Title
3.1.6	PIX/MPI Server
5.3	Interaction with the Regional PIX/MPI server.

Table of Contents

1	Introduction	4
1.1	Purpose of this Document	4
1.2	Intended Users of the This Document	4
2	Dependencies on Other Regional Components	5
3	Service Specification	6
3.1	Authentication	6
3.2	Web socket service	6
3.2.1	Basics.....	6
3.2.2	Authenticate	7
3.2.3	Determine Data Availability	7
3.2.4	Determine Data Availability Sources	8
3.3	REST Service	9
3.3.1	Determine Data Availability	9
3.3.2	Determine Data Availability Sources	9

1 Introduction

1.1 Purpose of this Document

This document is one of a series of design papers which underpin the Abstract Cookbook for Regional Interoperability. These papers, in their totality, describe the technical components and the standards which form the YHCR system of systems. They are intended as a basis for developing or procuring software and so are expressed at a level of precision which is intended to avoid ambiguity but with a consequence that they are focussed towards technical readers.

Design papers are anchored to topics which are discussed in the Abstract Cookbook for Regional Interoperability. They are elaborations of the concepts which were first introduced by the abstract and new content is further detail rather than variations of previously established core principles.

This document (design paper 002 - "Data Availability Service") defines the mechanisms through which data consumers can determine whether the Yorkshire and Humber Care record holds data about a patient.

One key use case for this service is a user interface which structures access to a care record by where the record originated. Such a user interface may offer a sequence of tabs, each representing a data source. If data is available from a particular source for the in-context patient, then the tab is present and otherwise the tab is hidden. Whenever context change to a new patient then this user interface must ask the question of the YHCR 'Do you know of data for this patient?'. This question will be asked for every patient viewed in the user interface and so must be serviced in a highly efficient and low latency manner.

A slight extension to this question is 'Which organisations may have data for this patient?'. This question could allow the user interface to offer more precise visual clues as to where there are sources of data. Knowing where data resides also permits a direct peer relationship with a data provider: the data consumer could invoke the FHIR endpoint of the data provider directly, bypassing the regional FHIR service bus when accessing data from a particular targeted source.

1.2 Intended Users of the This Document

This service specification will inform the development of regional services and consumer software which is interacting with these services.

2 Dependencies on Other Regional Components

The data availability service uses data held in the regional FHIR store. The service uses regional Patient resources and Linkage resources to local Patients to determine whether a patient is known to the region and to a particular data provider.

Regional patient resources and Linkages are constructed by the regional PIX server. The PIX server accepts an HL7 message feed from data providers which it informs it of local patient registrations. A patient registration in a locality, subject to constraints on the PDS traced nature of the NHS number, results in a regional Patient resource, possibly, being created and a Linkage resource being created to the local version.

In most cases it is envisaged that data consumers will be interested in data which is available from outside their own home location and the question 'Do you know of data for this patient?' will imply data that is not already known to me.

Data consumers are registered with the regional Identity and Access Management (IAM) service and the registration associates the consumer with one or more local data providers. The availability service uses this information to exclude these sources from its response. This behaviour can be overridden through parameters provided in the service invocation.

3 Service Specification

The availability service is implemented using two alternative technologies: web socket and REST services. The options are functionally equivalent. The web socket implementation is a highly efficient, low latency technology, and is the preferred invocation mechanism for high frequency usage. Systems needing to access the data availability service at sustained rates of 1 request per second or greater are considered high frequency users.

The REST implementation is a convenient service for low volume usage and for direct interaction from web clients (although use of web sockets from a web client is not precluded).

3.1 Authentication

Both implementations require access to the service to be authenticated using a bearer token obtained from the regional Identity and Access Management (IAM) service. The bearer token can be either:

- A user token obtained by registering an individual authenticated by the data consumer.
- A system token obtained by an IAM registered application.

The architecture of the data consumer application will determine the most appropriate option. An application server implementing a small number of threads, each holding a persistent web socket connection to the availability service and which are independent from a user session are best implemented using a system token. Connections directly from a user's browser or via an application server made in the context of a session can re-use a bearer-token already obtained for the session.

Note that through the authentication, IAM is able to locate the home organisation (or organisations) of the connecting systems. By default, the availability service will disregard patient data which is located at the home organisations' when responding to availability requests. This behaviour is consistent with FHIR searches performed on the regional FHIR bus and allows a system to determine whether the YHCR knows about data for a patient which isn't already known to the home organisation.

The registration of home organisation(s) associated with a data consumer is covered in a separate design paper.

3.2 Web socket service

3.2.1 Basics

The protocol is based on a request-response paradigm. Data is formatted as JSON objects.

A request object is issued by a client on an open web socket connection. A request object specifies an action and has additional properties which are specific to the action being requested.

A single response object is returned by the availability service for each request. The response object has `responseCode` and `responseText` properties. Response codes are derived from, and have the same interpretation as the HTTP protocol. Other response object properties are specific to the action requested.

Multiple requests and responses can be exchanged on an open web socket connection.

3.2.2 Authenticate

A system must authenticate before any other requests are issued over the web socket connection. Authentication only needs to be performed once per connection. Authentication must use an active bearer-token which has not expired. The expiry timestamp of the bearer token is only relevant at the point of authentication. A bearer-token which expires after authentication will not result in the web socket connection being dropped.

A client may authenticate more than one in the course of a single connection.

Authenticate request:

```
{
  "action": "authenticate",
  "bearerToken": "pwwbkvv7abqzonnvztpea91ich7vprwdorbt4w4m"
}
```

Authenticate response:

```
{
  "responseCode": 200,
  "responseText": "OK"
}
```

Possible response codes include:

Code	Interpretation
200	Authenticated successfully.
400	Bad request – the JSON object is badly formed or required properties are not present.
401	Authentication failed – the bearer token is invalid or has expired
500	Internal server error.
503	Service temporarily not available.

3.2.3 Determine Data Availability

A data availability request determines whether data is known to the YHCR. By default, the service excludes data which is available from the home organisation of the requesting system.

Availability request example 1:

```
{
  "action": "request",
  "nhsNumber": "4625400066"
}
```

Availability response example 2:

```
{
  "responseCode": 200,
  "responseText": "OK",
  "availabilityStatus": 1
}
```

Possible response codes include:

Code	Interpretation
200	Request was serviced correctly. The availability of data is indicated by the "availabilityStatus" of the response.
400	Bad request – the JSON object is badly formed or required properties are not present.
500	Internal server error.
503	Service temporarily not available.

Availability request example 2:

```
{
  "action": "request",
  "nhsNumber": "4625400066"
  "allSources": 1
}
```

The allSources property instructs the availability service to ignore the invoking system's home organisation when determining data availability.

Availability response example 2:

```
{
  "responseCode": 200,
  "responseText": "OK",
  "availabilityStatus": 1
}
```

Response codes are as for example 1.

3.2.4 Determine Data Availability Sources

A data source request determines where data is known to the YHCR. By default, the service excludes data which is available from the home organisation of the requesting system.

Availability source request example:

```
{
  "action": "detail",
  "nhsNumber": "4625400066"
}
```

Response:

```
{
  "responseCode": 200,
  "responseText": "OK"
  "detail": [
    {
      "organisationCode": "ODS",
      "organisationName": "Organisation 1",
      "endpointaddress": "https:org.nhs.uh/fhir/"
    },
    {
```



```

        "organisationCode": "ODS2",
        organisationName": "Organisation 2",
        "endpointaddress": "https:org2.nhs.uh/fhir/"
    } ]
}

```

Response codes are as follows:

Code	Interpretation
200	Request was serviced correctly. The "detail" property of the response contains an array of organisations and their endpoints from which data may be available. An empty array indicates that data is not available from the YHCR. HTTP possible response codes are as for the equivalent web socket service.
400	Bad request – the JSON object is badly formed or required properties are not present.
500	Internal server error.
503	Service temporarily not available.

The "allSources" property can be included in the request to include the home organisation(s) in the possible sources of data.

3.3 REST Service

The REST service mirrors the functionality of the web sockets implementation but in a stateless manner: authentication is performed for each service invocation. Response content is in JSON format and requests must include the HTTP header:

```
Accept: application/json
```

Response objects mirror those defined for the WebSocket service but without responseCode and responseText properties (which are returned in the HTTP response)/

3.3.1 Determine Data Availability

Request:

```
GET /DataAvailability?nhsNumber=4625400066
```

Response:

```

{
  "availabilityStatus": 1
}

```

The allSources parameter can be added to the search string to request that data availability should be assessed including the home organisation(s) of the requestor i.e.:

```
GET /DataAvailability?nhsNumber=4625400066&allSources=1
```

3.3.2 Determine Data Availability Sources

Request:

```
GET /DataAvailability?nhsNumber=4625400066&detail=1
```

Response:

```
{
  "detail": [
    {
      "organisationCode": "ODS",
      "organisationName": "Organisation 1",
      "endpointaddress": "https:org.nhs.uh/fhir/"
    },
    {
      "organisationCode": "ODS2",
      "organisationName": "Organisation 2",
      "endpointaddress": "https:org2.nhs.uh/fhir/"
    }
  ]
}
```

HTT