



**HIPS Release 4.1.0**

**Document Production Services  
Technical and Functional Specification**

17 February 2014

Approved for external use

**National E-Health Transition Authority Ltd**

Level 25  
56 Pitt Street  
Sydney, NSW, 2000  
Australia  
[www.nehta.gov.au](http://www.nehta.gov.au)

**Disclaimer**

The National E-Health Transition Authority Ltd (NEHTA) makes the information and other material ('Information') in this document available in good faith but without any representation or warranty as to its accuracy or completeness. NEHTA cannot accept any responsibility for the consequences of any use of the Information. As the Information is of a general nature only, it is up to any person using or relying on the Information to ensure that it is accurate, complete and suitable for the circumstances of its use.

**Document control**

This document is maintained in electronic form and is uncontrolled in printed form. It is the responsibility of the user to verify that this copy is the latest revision.

**Copyright © 2014 National E-Health Transition Authority Ltd**

This document contains information which is protected by copyright. All Rights Reserved. No part of this work may be reproduced or used in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems—without the permission of NEHTA. All copies of this document must include the copyright and other information contained on this page.

# Document Information

## Key Information

**Owner** Head of Strategy, Architecture and Informatics

**Contact for enquiries** NEHTA Help Centre  
t: 1300 901 001  
e: [help@nehta.gov.au](mailto:help@nehta.gov.au)

## Document Version History

---

Version	Date	Release comments
4.1.0	17/2/2014	Release 4.1.0

---

# Table of Contents

<b>1. Purpose</b>	<b>5</b>
<b>2. Background</b>	<b>5</b>
<b>3. Service Overview</b>	<b>5</b>
<b>4. PCEHR Participation and Authorisation Model</b>	<b>6</b>
<b>5. PCEHR Data Store Model</b>	<b>7</b>
5.1 Patient Data Model	7
5.2 Queuing Data Model	8
5.3 Clinical Document Data Model	10
<b>6. Database Loader Service</b>	<b>13</b>
6.1 Functional Logic	13
6.2 Technical Structure	13
<b>7. Get Validated IHI</b>	<b>14</b>
7.1 Functional Logic	14
7.2 Technical Structure	15
<b>8. Check Whether PCEHR is Advertised</b>	<b>17</b>
8.1 Functional Logic	17
8.2 Technical Structure	18
<b>9. Consent and Participation</b>	<b>19</b>
9.1 Functional Logic	19
9.2 Technical Structure	20
<b>10. Upload or Supersede Document to PCEHR</b>	<b>26</b>
10.1 Functional Logic	26
10.2 Technical Structure	32
<b>11. Remove Document from PCEHR</b>	<b>34</b>
11.1 Functional Logic	34
11.2 Technical Structure	35
<b>12. Get Operation Status</b>	<b>37</b>
12.1 Functional Logic	37
12.2 Technical Structure	37
<b>13. Get Queued Operation List</b>	<b>43</b>
13.1 Functional Logic	43
13.2 Technical Structure	43
<b>14. Reference Services</b>	<b>47</b>
14.1 Functional Logic	47
14.2 Technical Structure	48
<b>15. Common Schemas</b>	<b>52</b>
15.1 User Details	52
15.2 Patient Identifier	53
15.3 HIPS Response	56
<b>Appendix A HIPS Architecture</b>	<b>59</b>

## 1. Purpose

The purpose of this document is to specify the functional logic and the technical interfaces for the Document Production services provided in the HIPS Production Release 4.0, so that the Healthcare Identifier and PCEHR communications solution developed by SA Health can be integrated for production use by other healthcare groups.

## 2. Background

NEHTA has a strategic aim to support and facilitate the adoption of national eHealth foundations across the health sector. A key element of delivering on this objective is supporting the implementation of solutions that integrate with the national Healthcare Identifiers Service and the PCEHR system. NEHTA also seeks to collaborate through our implementations projects and, wherever possible, use the lessons learnt and knowledge gained to support the uptake of the national systems, and make that information available to a broader audience, where possible.

The eHealth Integration Sample Code provides implementers and software vendors with a sample implementation of a communications solution that enables patient administration systems and clinical information systems to interact with the Healthcare Identifiers Service and the personally controlled eHealth record system (PCEHR).

## 3. Service Overview

The Document Production functions provided by HIPS include the following web services:

- Database Loader – for processing HL7 messages from PAS
- Get Validated IHI – for interacting with Healthcare Identifier (HI) Service
- Check Whether a PCEHR is Advertised
- Consent and Participation Services
- Upload or Supersede Document to PCEHR
- Remove Document from PCEHR

## 4. PCEHR Participation and Authorisation Model

Each hospital facility within the implementer's group, that intends to upload discharge summaries to the PCEHR system, must be associated with a participating Healthcare Provider Organisation (HPO), identified by an HPI-O. Multiple hospitals may be associated with same HPO. For example:

- Some or all hospitals within a group may be associated to the seed HPO for the group.
- Some or all hospitals within a Local Health Network (LHN) may be associated to the network HPO for that LHN.
- Some or all hospitals may themselves be a network HPO.

For any patient record to which HIPS has assigned a valid IHI, the creation of a new episode of care in the Patient Administration System (PAS) at the hospital will trigger HIPS to send a message to the PCEHR system, asking if the consumer has a PCEHR. The message will include:

1. The consumer's IHI;
2. The hospital's associated healthcare provider organisation's HPI-O; and
3. The local system identifier of the authorised employee for the hospital or the interactive user.

Once the PCEHR system receives this message, the PCEHR system will proceed to verify the right of the HPO to access the PCEHR system, and determine whether the HPO is authorised to know whether the consumer has a PCEHR. The system then advises the HPO whether the consumer has a PCEHR, with the answer being "no" if the HPO is not authorised to know.

HIPS will store this advice separately for each participating HPO. In the case where the consumer has elected to hide the existence of his/her PCEHR, it is possible that one participating HPO has gained access and therefore "knows" the consumer has a PCEHR, while another HPO has not gained access and therefore does not "know" that the same consumer has a PCEHR. The implementer should respect this separation of information for privacy reasons.

When the discharge summary for the episode of care is distributed, if the PCEHR system's advice to the discharging hospital's HPO indicated that the consumer has a PCEHR, the default setting is that the discharge summary record will be uploaded, unless the consumer requests that it is not uploaded (via withdrawal of consent), or the clinical information system user requests that it is not uploaded.

Otherwise, if the advice indicated the consumer does not have a PCEHR, the default setting is that the discharge summary record will not be uploaded, unless the consumer disclosed the existence of the PCEHR.

## 5. PCEHR Data Store Model

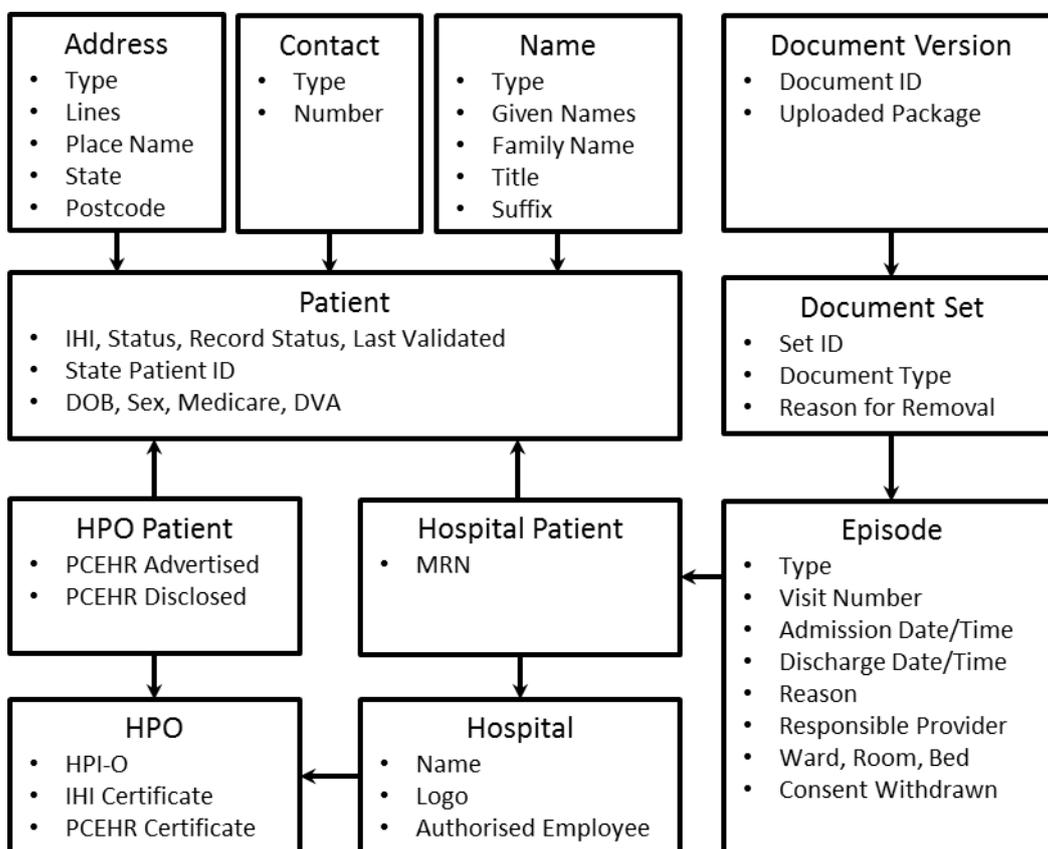
### 5.1 Patient Data Model

The PCEHR Data Store contains certain information about hospital patients that is relevant and required for the PCEHR connectivity in HIPS.

The key requirements for this design were:

- Each hospital belongs to a Health Provider Organisation (HPO) with an HPI-O.
- A disclosure of PCEHR can be stored for each HPO that a patient visits.
- A withdrawal of consent to upload documents can be stored for each episode.

An overview of the data model is given below:



A future release of HIPS, after CCA testing, will contain complete documentation of the database tables. For this evaluation release, an example is provided below.

#### 5.1.1 HealthProviderOrganisationPatient Table

After HIPS calls the PCEHR's "doesPCEHRExist" service method, the result is stored in the HPO Patient table, "HealthProviderOrganisationPatient".

The HealthProviderOrganisationPatient table will have the following columns:

Column	Type	Description
HealthProviderOrganisationId	int	The HPO to which this record relates.
PatientMasterId	int	The patient to which this record relates.
PcehrAdvertised	nullable bit	The value that the PCEHR System last indicated to this HPO as to whether the PCEHR for this patient exists.
AccessCodeRequiredId	int	The value that the PCEHR System last indicated to this HPO as to whether access to this patient's PCEHR is granted and whether a code is required for this HPO to gain access: <ul style="list-style-type: none"> <li>-1: Unknown</li> <li>0: With Code</li> <li>1: Without Code</li> <li>2: Access Granted</li> </ul>
PcehrDisclosed	bit	Whether the patient has disclosed the existence of his/her PCEHR to this HPO. The patient is known to be participating in the PCEHR System if the patient has disclosed the existence of a PCEHR.
DateCreated	datetime	The date and time when the record for this HPO and patient was created.
UserCreated	varchar(256)	The domain and login of the user identified by the source system as responsible for the action that triggered the creation of this record.
DateModified	datetime	The date and time when the record was last modified.
UserModified	varchar(256)	The domain and login of the user identified by the source system as responsible for the action that last modified this record.

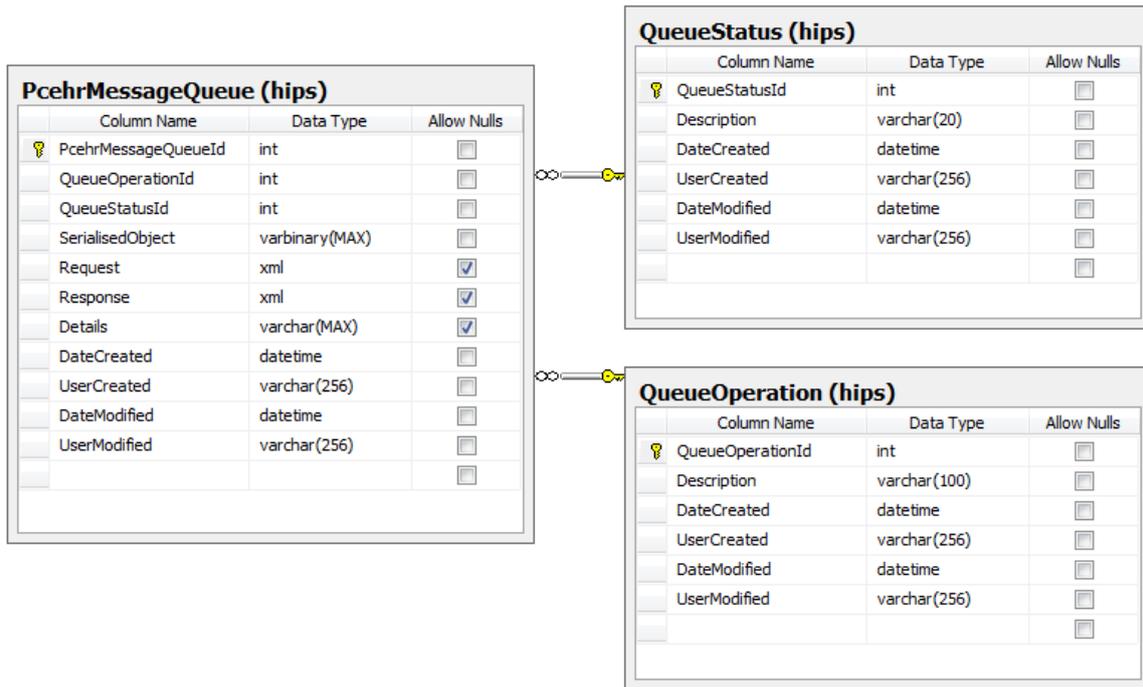
## 5.2 Queuing Data Model

After each request to UploadOrSupersedeDocument or Remove has been validated, a record of the queued operation with "Pending" status will be added to the "PcehrMessageQueue" table in the PCEHR Data Store as illustrated below, if and only if the operation is handed off to the Microsoft Message Queue (MSMQ).

Normal processing is to delete these records of the queued operations once the operation has successfully completed, and so the table will contain only pending operations and failures. We have added a configuration item to allow successful operations to be retained for testing purposes.

The configuration item "DeleteQueuedItemOnSuccess" will default to true. If this configuration item is true then the record of the queued item will be deleted after successful processing.

If the queued operation is unsuccessful, or the above configuration item is set to false, then the QueueStatusId column will be updated to reflect the Success or Failure as appropriate, the SOAP request and response will be populated into the Request and Response columns, and any additional error information including a HIPS exception or stack trace will be populated into the Details column.



### 5.2.1 PcehrMessageQueue Table

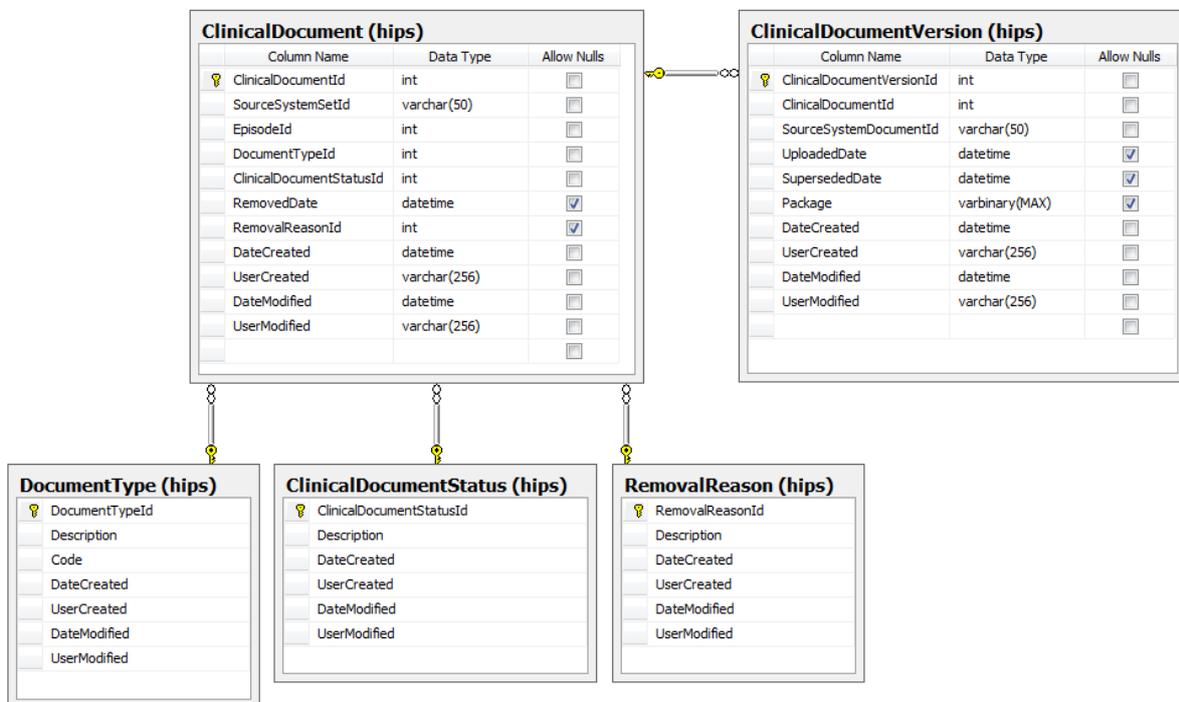
The PcehrMessageQueue table will have the following columns:

Column	Type	Description
PcehrMessageQueueId	int	Auto-incrementing primary key
QueueOperationId	int	The operation type that has been queued: <ul style="list-style-type: none"> <li>• UploadOrSupersede</li> <li>• Remove</li> </ul>
QueueStatusId	int	The status of the queued operation: <ul style="list-style-type: none"> <li>• Pending</li> <li>• Success (unused by default)</li> <li>• Failure</li> </ul>
SerialisedObject	varbinary(max)	The .NET serialised object that has been added to the MSMQ.
Request	XML	The SOAP request sent to the PCEHR B2B Gateway. Only populated after the queued operation has been processed.
Response	XML	The SOAP response received from the PCEHR B2B Gateway. Only populated after the queued operation has been processed.
Details	varchar(max)	The exception message and stack trace from HIPS showing where a failure in processing the queued operation occurred.
DateCreated	datetime	The date and time when the queued operation was created.

Column	Type	Description
UserCreated	varchar(256)	The domain and login of the user identified by the source system as responsible for the action.
DateModified	datetime	The date and time when the queued operation was last modified.
UserModified	varchar(256)	The domain and login of the user identified by the source system as responsible for the action.

### 5.3 Clinical Document Data Model

Documents that are *successfully* uploaded, superseded or removed will be stored within the PCEHR Data Store in the data model illustrated below.



All versions of a document that are uploaded to PCEHR by HIPS will be associated with an episode, which is associated with a patient in a certain hospital, which is associated with the patient master record and the hospital record.

The PCEHR Data Store model makes this hierarchy explicit, and assigns identifiers at each level:

- Clinical Document Version (source system document ID)
  - Clinical Document (source system set ID)
    - Episode (visit ID, admission date/time)
      - Hospital Patient (MRN)
        - Patient Master (enterprise patient ID, IHI)
        - Hospital (Hospital Code)

- Health Provider Organisation (HPI-O)

### 5.3.1 ClinicalDocument Table

The "ClinicalDocument" table stores the information that relates to the overall document.

Column	Type	Description
ClinicalDocumentId	int	Auto-incrementing primary key
SourceSystemSetId	varchar(50)	The source system's unique identifier of the overall document, which must not change between versions of the same document. This is populated from the "root" and "extension" attributes of the "setId" element of the CDA document, separated by ^ (caret).
EpisodeId	int	The episode to which this document relates. The hospital and patient are identified via this link to the episode.
DocumentTypeId	int	The type of document, such as discharge summary or event summary. This is populated from the "code" element of the CDA document.
ClinicalDocumentStatusId	int	The overall document status: <ul style="list-style-type: none"> <li>• 1: Active</li> <li>• 2: Removed</li> </ul>
RemovedDate	datetime (nullable)	If <i>currently</i> removed, the date and time when the document was last removed. This will be reset to null if a new version of a removed document is uploaded.
RemovalReasonId	int	If <i>currently</i> removed, the reason for removal: <ul style="list-style-type: none"> <li>• -1: Not Removed</li> <li>• 1: Withdrawn</li> <li>• 2: Elect to Remove</li> <li>• 3: Incorrect Identity</li> </ul> This will be reset to -1 if a new version of a removed document is uploaded.
DateCreated	datetime	The date and time when the clinical document was first uploaded.
UserCreated	varchar(256)	The domain and login of the user identified by the source system as responsible for the upload.
DateModified	datetime	The date and time when the clinical document was last uploaded or removed.
UserModified	varchar(256)	The domain and login of the user identified by the source system as responsible for the action.

### 5.3.2 ClinicalDocumentVersion Table

The "ClinicalDocumentVersion" table stores the information that relates to an individual version of the document.

Column	Type	Description
ClinicalDocumentVersionId	int	Auto-incrementing primary key
ClinicalDocumentId	int	The clinical document that this is a version of.
SourceSystemDocumentId	varchar(50)	The source system's unique identifier of the document instance, which will change between versions of the same document. This is populated from the "root" and "extension" attributes of the "id" element of the CDA document, separated by ^ (caret).
UploadedDate	datetime	The date and time when this version was uploaded.
SupersededDate	datetime (null)	If this version has been superseded by a later version, the date and time when this version was superseded.
Package	varbinary(max)	The CDA package ZIP file that was uploaded.
DateCreated	datetime	The date and time when this clinical document version was uploaded.
UserCreated	varchar(256)	The domain and login of the user identified by the source system as responsible for the upload.
DateModified	datetime	The date and time when this clinical document version was uploaded or superseded.
UserModified	varchar(256)	The domain and login of the user identified by the source system as responsible for the most recent upload or supersede action on this version.

## 6. Database Loader Service

### 6.1 Functional Logic

#### 6.1.1 Description

This service is designed to accept messages from patient administration systems via a message broker. This message broker should transform the messages to comply with the format expected by HIPS, which is based on the international HL7 standard, version 2.3.1. Refer to the document "HIPS HL7 Interface Specification" for details of this format.

This service stores the patient and episode information into the PCEHR Data Store, and triggers the automatic IHI lookup and check for advertised PCEHR.

A failure to store the message, patient or episode will result in a negative acknowledgement being returned. A failure to obtain an IHI or check the PCEHR status will not result in a negative acknowledgement. The IHI is not returned in the acknowledgement, but stored into the PCEHR Data Store.

An example of the minimal input for registering a patient is as follows:

```
MSH|^~\&|App|Facility||DateTime|03V1|ADT^A28|MsgID|P|2.3.1|||AL|NE|AU|ASCII|EN
EVN|A28|20120716011454|||Operator
PID|||MRN^^^Facility^MR||Surname^First Name^Middle Names^^^L^A||DOB|Sex|||Address Line
1^Address Line 2^Suburb^State^Postcode^^H
```

### 6.2 Technical Structure

#### 6.2.1 Classes and Enumerations

The "DatabaseLoaderService" class in the "HIPS.AppServer.ServiceHost" project implements this web service.

The various "Loader" classes in the "HIPS.HibIntegration" project implement the business logic.

#### 6.2.2 Methods

##### 6.2.2.1 *NotifyPasEvent*

This service method takes the following parameters:

Input Parameter	Data Type	Description
messageForm	string	An HL7 message conforming to the HIPS HL7 Interface Specification.
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action

Return Parameter	Data Type	Description
response	string	HL7 acknowledgement (ADT^ACK) as a string

## 7. Get Validated IHI

### 7.1 Functional Logic

This function retrieves the IHI information that must be inserted in a CDA clinical document to identify the patient and to allow the receiver of the document to re-validate the IHI.

#### 7.1.1 Business Rules / Functional Business Logic

HIPS will first attempt to locate an existing patient record using the given patient identifier. With that patient record:

1. If the stored date of birth does not match that which is specified in the service call, then an InvalidDateOfBirth error is returned and no further action is taken.
2. If no IHI has been obtained for the patient record, then HIPS will attempt to obtain the IHI using the current demographic information and assign the IHI to the local patient record, possibly creating an exception alert in the process. An example of an exception alert is when the IHI status changes from Active to another status, or the IHI number is already assigned to another patient record from the same hospital.
3. If an IHI is obtained or had already been obtained for the specified patient record, and there is an outstanding exception alert on the IHI, such as a suspected duplicate or replica, then HIPS will not return the IHI to the caller.
4. If the IHI was obtained or last validated outside the time period that has been configured for this purpose, then HIPS will attempt to validate the IHI information with the HI Service, and will only return the IHI to the caller if the validation was successful. However if the HI Service is unavailable, the IHI will be returned with a warning that validation must still occur before the IHI can be trusted. This will occur when the document with the IHI embedded is passed back to HIPS for upload.
5. If the IHI was obtained or last validated within the configured valid time period, then the IHI will be returned immediately without triggering another validation.

#### 7.1.2 Usage Notes

This method is intended for use after a patient is registered in the PAS, to extract the IHI that was obtained from the HI Service, for use in a clinical document that will be distributed to an external health provider, to a shared repository or the PCEHR system.

One model for usage is where the clinical system is enhanced to handle IHI and CDA directly:

- A clinical system user finalises a discharge summary for a patient who has a PCEHR
- The system makes a call to an ESB to find or validate the IHI for the patient
- The ESB calls this method of HIPS and returns the validated IHI to the clinical system
- The clinical system produces a CDA discharge summary document with the validated IHI
- The system makes a call to the ESB to upload the discharge summary to the PCEHR

- The ESB sends the CDA document to HIPS for upload to the PCEHR system
- However, in SA Health’s implementation, the clinical system is not aware of IHIs, and there is custom-developed middleware handling conversion of discharge summaries to CDA format:
- A clinical system user finalises a discharge summary for a patient who has a PCEHR
  - The clinical system delivers the discharge summary data to the ESB for upload to PCEHR, in the form of an HL7 message
  - The ESB calls this method of HIPS to obtain the IHI for the patient
  - The ESB embeds the IHI into the HL7 message and delivers it to the CDA conversion middleware
  - The middleware converts the discharge summary from HL7 to CDA format and returns it to ESB
  - The ESB sends the CDA document to HIPS for upload to the PCEHR

## 7.2 Technical Structure

### 7.2.1 Classes and Enumerations

The “IHIService” class in the “HIPS.AppServer.ServiceHost” project implements this web service.

The “PatientIhiValidation” class in the “HIPS.PcehrHiBusinessLogic” project implements the business logic.

#### 7.2.1.1 ValidatedIhiResponse

This class is used for the response from the IHI validation methods.

Property	Data Type	Description
HipsResponse	HipsResponse	Contains an indicator of success or failure, and in case of failure, information about why the request failed.
Mrn	Mrn	Medical Record Number, a patient identifier allocated by the hospital’s Patient Administration System (PAS).
StatePatientId	StatePatientId	State/territory/group Patient ID, a patient identifier allocated by the group’s Enterprise Master Patient Index (EMPI).
ValidatedIhi	ValidatedIhi	Individual Healthcare Identifier, a patient identifier allocated by the national healthcare identifier service operated by Medicare Australia. This object encapsulates the 16-digit IHI number, the IHI status, IHI record status, date and time when the IHI was last validated, and the date of birth, sex, given names and family name used for validation. The clinical system should ensure that each CDA document uploaded to the PCEHR contains the same date of birth, sex, given names and family name as were used for IHI validation, so that the receiving system will be able to validate the IHI upon receiving the document.

## 7.2.2 Methods

### 7.2.2.1 *GetValidatedIhi*

This service retrieves a valid IHI and registered demographics for use in a clinical document. This service will immediately return the IHI information from the last validation if the IHI was obtained or validated within the configured period (typically 24 hours). Otherwise, the IHI will be validated with the HI Service before the information is returned. If there are outstanding alerts on the IHI (such as potential duplicate or replica records) then the IHI will not be returned because the alert must be resolved before the IHI is used in a clinical document.

This service takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
patientIdentifier	PatientIdentifierBase	An identifier of the patient	Required
dateOfBirth	DateTime	The patient's date of birth (as an additional identity check)	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	ValidatedIhiResponse	The validated IHI information and information about success or failure of the operation.

## 8. Check Whether PCEHR is Advertised

### 8.1 Functional Logic

#### 8.1.1 Description

A user-facing system may call this service method to obtain information necessary to inform the user about whether a patient has registered for a PCEHR and wants that PCEHR to be visible to the provider organisations participating in his/her healthcare.

This method wraps the PCEHR B2B Gateway service that NEHTA calls "Does PCEHR Exist". That name is misleading because the result is not strictly whether a PCEHR exists for the consumer, but is affected by whether the consumer has chosen to advertise the existence of his/her PCEHR. The result is also affected by whether the HPI-O is on the provider access list of the consumer's PCEHR, and if so what read access permission the consumer has given the HPI-O.

The result is also used internally by HIPS to drive the related function to determine whether a patient wants the discharge summary to be uploaded to his/her PCEHR. The latter service is described in the Patient Participation section. The main difference is that Patient Participation is also affected by the disclosure of the existence of a PCEHR. If the patient has disclosed the existence to the HPO then they are considered to be participating regardless of whether the PCEHR is hidden.

The response contains a property `AccessCodeRequired`, which provides some information as to whether an access code is required in order to access the patient's PCEHR to list or view documents:

- Null – thus meaning that the patient has not registered for PCEHR, or has chosen to hide the existence of his/her PCEHR. The patient may still give advice of PCEHR existence and may or may not provide an access code.
- With Code – thus meaning that the provider may not gain access to the PCEHR unless their patient provides an access code.
- Without Code – thus meaning that access is open and no Record Code is required. The patient may still advise of a Document Code to grant restricted access.
- Access Granted – thus meaning that access has been granted and no code is required unless the patient advises of a change to access level and provides an access code.

#### 8.1.2 Usage Note

In the SA Health implementation, this method is not used at all. It has proven unnecessary because:

- HIPS automatically calls `DoesPCEHRExist` immediately after obtaining an IHI from the HI Service
- HIPS automatically calls `DoesPCEHRExist` after creating a new episode for an existing patient

This method is a mechanism to manually trigger calls to `DoesPCEHRExist`. It is primarily of use where:

- The implementer does not provide a feed of HL7 messages from the PAS into HIPS, therefore this is the main mechanism to check whether a patient has a PCEHR or not.
- The clinical system requires the latest information about the access status for the PCEHR, such as immediately before or after a call to `Gain Access`.

## 8.2 Technical Structure

### 8.2.1 Classes and Enumerations

The "PCEHRService" class in the "HIPS.AppServer.ServiceHost" project implements this web service.

The "DoesPcehrExist" class in the "HIPS.PcehrBusinessLogic" project implements the business logic.

#### 8.2.1.1 DoesPcehrExistResponse

Property	Type	Description
HipsResponse	HipsResponse	An indicator of success or failure of the operation, and in case of failure, information about why the operation failed.
PcehrExists	bool	Whether or not a PCEHR is advertised for this patient. If false, then the patient may not have registered for a PCEHR, or may have chosen to hide the existence of the PCEHR.
AccessCodeRequired	enum AccessCodeRequired	The PCEHR access status indicator, one of the following: <ul style="list-style-type: none"> <li>Unknown – The access status is unknown.</li> <li>WithCode – Except in an emergency, the patient must provide his/her Record Access Code for the healthcare provider organisation to gain access to this PCEHR.</li> <li>WithoutCode – No code is required for the health provider organisation to gain access to this PCEHR.</li> <li>AccessGranted – The health provider organisation has already gained access to this PCEHR.</li> </ul>

### 8.2.2 Methods

#### 8.2.2.1 IsPcehrAdvertised

For a given local patient record or IHI, this service checks whether the patient has advertised the existence of the PCEHR, whether access has been gained, and whether access can be gained without a code.

This service takes the following parameters:

Input Parameter	Data Type	Description	Required/Optional
patientIdentifier	PatientIdentifierBase	An identifier of the patient (MRN, State ID, PatientMasterId or validated IHI)	Required
dateOfBirth	DateTime	The patient's date of birth (PatientMaster.DateOfBirth)	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

Return Parameter	Data Type	Description
response	DoesPcehrExistResponse	Response information

## 9. Consent and Participation

### 9.1 Functional Logic

#### 9.1.1 Consent to Upload

Under the PCEHR consent model, consent to upload a clinical document to the PCEHR is assumed, because in the process of registering for PCEHR a blanket consent is extended to all providers.

Although this is expected to be rare, it is a NEHTA requirement that the system support the ability for the patient to withdraw this assumed consent at any time. HIPS will store a flag against each patient episode that indicates whether the patient has withdrawn consent to upload documents for that episode.

HIPS will provide a web service method "RecordConsent" that can be used in two situations:

1. **Withdrawal of Consent:** When the patient has advised the provider that he/she does not want the discharge summary to be uploaded.
2. **Rescind Withdrawal:** when such a withdrawal of consent was recorded in error.

Whenever a clinical system requests to upload or supersede a clinical document, HIPS will first check the flag for the episode to which the clinical document relates, to find whether consent has been withdrawn. If consent has been withdrawn then HIPS will refuse to upload the document.

A consumer cannot withdraw consent after a document has been uploaded to the PCEHR. So long as the first version was uploaded while the consumer was consenting, then any later versions of that document can be uploaded. Therefore the correct process is to avoid recording a withdrawal of consent if there are documents already uploaded.

However, if the consumer withdrew consent but the information was not entered into the system in time and the document was uploaded in error, then the correct process is to first remove the document from the PCEHR and then enter the withdrawal of consent to prevent it being uploaded again.

#### 9.1.2 Participation Status

In most cases when a patient has registered for a PCEHR, HIPS will find out that the PCEHR exists when it calls the PCEHR B2B Gateway method "doesPCEHRExist". At a minimum, this call is triggered once at the creation of an episode of care.

However for those rare cases when a patient has chosen to hide the existence of his/her PCEHR, but the patient wishes the provider organisation to upload the document despite this, HIPS will provide a web service method "RecordDisclosure". This service can be used in two situations:

1. **Disclose PCEHR:** when the patient has advised the provider organisation of the existence of his/her PCEHR (and by inference, consented to have his/her discharge summary uploaded), even though the PCEHR is not advertised (or may later become not advertised).
2. **Rescind Disclosure:** when such a disclosure was recorded in error.

HIPS will record the disclosure for each patient at each health provider organisation (HPO). If an implementer using HIPS consists of multiple HPOs, then the patient's disclosure to one HPO will not automatically apply to that patient at any other HPO.

When a clinical system requests the participation status for a certain patient, or requests a list of patients who have changed participation status since a certain

date, then HIPS will indicate that the patient is participating in PCEHR if the patient has disclosed the existence of a PCEHR to the health provider organisation, otherwise it will indicate whether the existence of the PCEHR is currently advertised or not.

## 9.2 Technical Structure

### 9.2.1 Classes and Enumerations

The "ConsentService" class in the "HIPS.AppServer.ServiceHost" project implements the following web services:

- RecordConsent
- RecordDisclosure
- CheckConsent
- CheckDisclosure
- GetPatientParticipationStatus
- GetRecentPatientParticipationStatus

The "RecordUploadConsent" class in the "HIPS.ConsentBusinessLogic" project implements the business logic for recording both positive and negative consent against an episode.

The "CheckUploadConsent" class in the "HIPS.ConsentBusinessLogic" project implements the business logic for finding the current value of the consent for an episode.

The "CheckPatientParticipation" class in the "HIPS.ConsentBusinessLogic" project implements the business logic for determining the participation status for a specified patient or for all changes in participation status since a specified date.

#### 9.2.1.1 *EpisodeConsentResponse*

This class is the response from HIPS for a request to get the consent status for a single episode of care.

Property	Type	Description
Response	HipsResponse	An indicator of success or failure and in case of failure, the reason for the failure.

Property	Type	Description
ConsentStatus	enum EpisodeConsentStatus <ul style="list-style-type: none"> <li>• Unknown</li> <li>• NoConsentRecorded</li> <li>• RequestedUpload</li> <li>• WithdrewConsent</li> </ul>	The current consent status for the specified episode: <ul style="list-style-type: none"> <li>• Unknown: the hospital, patient or episode record was not found</li> <li>• NoConsentRecorded: the patient has not given any explicit instructions as to whether documents related to the specified episode should be uploaded to his/her PCEHR.</li> <li>• RequestedUpload: the patient has explicitly requested that documents related to the specified episode are uploaded to his/her PCEHR.</li> <li>• WithdrewConsent: the patient has withdrawn consent to upload documents related to the specified episode.</li> </ul>

### 9.2.1.2 *PatientParticipationResponse*

This class is the response from HIPS for a request to get the PCEHR participation status, either for a single patient or for all patients with recent changes to their records.

Property	Type	Description
Response	HipsResponse	An indicator of success or failure and in case of failure, the reason for the failure.
PatientParticipationList	Array of PatientParticipationStatus	The information required for the calling system to identify one or more patient records and their current PCEHR participation status.

### 9.2.1.3 *PatientParticipationStatus*

The information in this class represents the current PCEHR participation status for a patient. It holds all the information required for the calling system to identify the patient record, including the MRN, the state patient identifier and the validated IHI.

Property	Type	Description
Mrn	string	A patient identifier unique within the hospital, typically allocated by the hospital's PAS or PMI
StatePatientId	string	A patient identifier unique within the state or territory and typically allocated by the enterprise master patient index.
ValidatedIhi	ValidatedIhi	An Individual Healthcare Identifier allocated by the national healthcare identifier service, and the information required to check the validity of the IHI.

Property	Type	Description
HospitalCode	string	A code in the in the hospital code system that was specified in the service request, which identifies the hospital that allocated the MRN.
ParticipationStatus	enum ParticipationStatus <ul style="list-style-type: none"> <li>NoValidIhi</li> <li>PcehrNotAdvertised</li> <li>PcehrAdvertised</li> <li>RequestedUpload</li> </ul>	<p>The current participation status of the patient:</p> <ul style="list-style-type: none"> <li>NoValidIhi: HIPS was not able to obtain or validate the IHI for this patient. HIPS will not be able to upload the discharge summary for this patient until this situation is resolved.</li> <li>PcehrNotAdvertised: This patient may not have registered for PCEHR, or may have chosen to hide the existence of his/her PCEHR. The clinical system should not send the discharge summary, unless the patient explicitly requests the provider to do so.</li> <li>PcehrAdvertised: This patient has chosen to advertise the existence of his/her PCEHR, so the clinical system should send the discharge summary.</li> <li>RequestedUpload: This patient disclosed the existence of a PCEHR to the health provider organisation of this hospital, so the clinical system should send the discharge summary, regardless of whether the PCEHR is advertised.</li> </ul>

## 9.2.2 Methods

### 9.2.2.1 *RecordConsent*

For a specified episode of care, this service records a withdrawal of consent to upload documents, or the reinstatement of consent.

This service takes the following parameters:

Input Parameter	Data Type	Description	Required/Optional
consent	bool	<p>If true, records that the patient withdrew consent to upload documents related to the specified episode of care.</p> <p>If false, records that the patient reinstated consent to upload documents related to the specified episode of care.</p>	Required
patientIdentifier	PatientIdentifierBase	An identifier of the patient (MRN, State ID, PatientMasterId or validated IHI), containing also an identifier of the hospital.	Required

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
admissionDate	DateTime	The date and time when the episode of care started	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required
auditInformation	binary	Additional audit information to store, such as a copy of a form that the patient signed.	Optional

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	HipsResponse	Response information

#### 9.2.2.2 *RecordDisclosure*

For a specified patient and hospital, this service records when the patient has disclosed the existence of a PCEHR or the reversal of that disclosure.

This service takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
pcehrDisclosed	bool	If true, records that the patient disclosed the existence of a PCEHR to the health provider organisation. If false, reverses a previous disclosure.	Required
patientIdentifier	PatientIdentifierBase	An identifier of the patient (MRN, State ID, PatientMasterId or validated IHI), containing also an identifier of the hospital.	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required
auditInformation	binary	Additional audit information to store, such as a copy of a form that the patient signed.	Optional

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	HipsResponse	Response information

### 9.2.2.3 *CheckConsent*

This service method returns the current consent status for a specified episode of care.

This method takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
patientIdentifier	PatientIdentifierBase	An identifier of the patient (MRN, State ID, PatientMasterId or validated IHI)	Required
admissionDate	DateTime	The date and time when the episode of care started	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	EpisodeConsentResponse	Response status and episode consent status

### 9.2.2.4 *GetPatientParticipationStatus*

This service method returns the current participation status for a specified patient in a specified hospital.

This method takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
patientIdentifier	PatientIdentifierBase	An identifier of the patient (MRN, State ID, PatientMasterId or validated IHI)	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	PatientParticipationResponse	Response status and patient participation status

### 9.2.2.5 *GetRecentPatientParticipationStatus*

This service method returns the current participation status for all patients who have at least one episode record and whose HealthProviderOrganisationPatient (HPOP) record has been updated since the specified date and time. The HPOP record is updated in the following situations:

- When the result of DoesPCEHRExist is recorded, which occurs:
  - When an IHI is obtained from the HI Service
  - When an IHI is revalidated with the HI Service
  - When a new episode is created via the PAS loader
  - When a client system calls the HIPS method IsPcehrAdvertised
- When a patient disclosure is recorded (RecordDisclosure).

Note that patients without an episode record are not included. This decision was made because these are typically the stubs left over after an MRN merge operation.

This method writes an informational message to the system error log table, which records the number of hospital patients returned.

This method takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
since	DateTime	The date and time after which a change in a patient's HPOP record must have taken place in order for the patient record to be included in the response.	Required
hospitalCodeSystem	string	Code that identifies which type of hospital codes should be returned. This has the effect of filtering which patients should be included in the response.	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action.	Required

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	PatientParticipationResponse	Response status and patient participation status

## 10. Upload or Supersede Document to PCEHR

### 10.1 Functional Logic

This service implements a “fire and forget” pattern that adds a document instance to the queue for uploading to the appropriate repository for the document type. The service will return as soon as the item is added to the queue.

#### 10.1.1 Patient Matching

The `patientIdentifier` parameter is used to look up a patient record. If the patient identifier is of type `Mrn`, `StatePatientId` or `PatientMasterId` then the patient must already exist in the PCEHR Data Store, otherwise an “InvalidPatient” error will be returned. However, if the patient identifier is of type `ValidatedIhi`, then the patient need not exist in the PCEHR Data Store; if they do not exist, a minimal stub record will automatically be created.

#### 10.1.2 IHI Validation

If the IHI has an unresolved data-quality alert, then an “UnresolvedIhiAlert” error will be returned and the document will not be placed on the queue for upload.

If the IHI was last validated more than the configured period of time in the past, the IHI will be revalidated with the HI Service. If validation returns no records found with the IHI and the stored demographic information, an “InvalidIhi” error will be returned and the document will not be placed on the queue for upload. However, if the HI Service is unavailable then the document will be placed on the queue with a stale IHI, which will be revalidated when the item is taken off the queue to be processed.

Each time an upload operation is taken off the queue to be processed, HIPS will check that the IHI was validated within the configured period. If there is an outage that stops the upload for longer than the configured period, then HIPS will automatically revalidate the IHI before attempting the upload again.

This has a critical impact on implementations where HIPS is not connected to the HI Service. Without a connection to the HI Service, HIPS cannot upload a document it remains on the queue beyond the configured period. Therefore, the configured period should be long enough that HIPS will not need to revalidate the IHI. Otherwise, documents will fail to upload and need to be resent with a newly validated IHI.

#### 10.1.3 Document Validation

HIPS will not run a full CDA validation on documents before uploading them to the PCEHR, because that would duplicate the work that the PCEHR system does itself. If documents fail validation that implies there is a deficiency in the software that generated the CDA document. The document should be resent from the source system after the deficiency has been corrected.

HIPS will extract and validate the following items from the document:

- The Document Type is extracted from the `<code>` element. The code must match to a code that is configured in the `DocumentType` table. Each document type is associated with a repository. HIPS will connect to the associated repository (PCEHR) for uploading and removing documents.
- The Document ID is extracted from the `<id>` element. The root must be an OID or a UUID. The extension is optional. Note that the PCEHR system requires that the root is unique, so when using an extension to show a

user-friendly numeric ID, it is necessary to repeat the extension inside the root.

- The Set ID is extracted from the <setId> element. The root must be an OID or a UUID. The extension is optional. Although the Set ID is optional in the CDA implementation guide, it is mandatory for HIPS.
- The IHI is extracted from the <id> element whose assigningAuthorityName is "IHI". The IHI must match the IHI assigned to the patient in the HIPS database, otherwise an "InvalidIhi" error is returned and the document is not uploaded.

If the CDA document is not valid XML, or HIPS is unable to extract any of these items, an "InvalidDocument" error is returned.

#### 10.1.4 Document Format Codes (Template Package IDs)

The document format code, also known as a template package ID, is used to specify which validation rules the PCEHR system will apply to the document when it is uploaded. There is a different format code for each conformance level of each document type. Also, as the PCEHR system is upgraded over time, there are new document format codes that can be used. These new format codes allow for changes to be made in the document validation rules over time, without affecting systems that upload documents developed under the older rules.

Each of the document format codes that HIPS will use for uploading documents must be configured in the DocumentFormat table. If the specified format code is not found in the DocumentFormat table, an "InvalidDocument" error is returned.

For implementers that upload only one conformance level of one document type, there is no need to include the parameter "documentFormatCode". When this parameter is omitted or null, HIPS will use the format code that is configured as "DefaultDocumentFormatCode" in the web.config file. Otherwise, specify the format code in the parameter "documentFormatCode" of each upload request.

The HPI-I Relaxed templates are restricted to implementers who have been granted permission from the PCEHR system operator, for the document author's identifier to be a local system identifier instead of an HPI-I. This permission may be time-limited, after which implementers will need to transition to the HPI-I Enforced templates.

As of HIPS 4.1, the format codes for supported document types are:

Document Type	Version	HPI-I	Conformance Level	Document Format Code
Discharge Summary	R2	Relaxed	1A	1.2.36.1.2001.1006.1.20000.12
			1B	1.2.36.1.2001.1006.1.20000.9
			2	1.2.36.1.2001.1006.1.20000.10
			3A	1.2.36.1.2001.1006.1.20000.11
	R3	Relaxed	1A	1.2.36.1.2001.1006.1.20000.13
			1B	1.2.36.1.2001.1006.1.20000.14
			2	1.2.36.1.2001.1006.1.20000.15
			3A	1.2.36.1.2001.1006.1.20000.16
			3B	1.2.36.1.2001.1006.1.20000.17
	R4	Relaxed	1A	1.2.36.1.2001.1006.1.20000.18

Document Type	Version	HPI-I	Conformance Level	Document Format Code
			1B	1.2.36.1.2001.1006.1.20000.19
			2	1.2.36.1.2001.1006.1.20000.20
			3A	1.2.36.1.2001.1006.1.20000.21
			3B	1.2.36.1.2001.1006.1.20000.22
		Enforced	1A	1.2.36.1.2001.1006.1.20000.23
			1B	1.2.36.1.2001.1006.1.20000.24
			2	1.2.36.1.2001.1006.1.20000.25
			3A	1.2.36.1.2001.1006.1.20000.26
			3B	1.2.36.1.2001.1006.1.20000.27
PCEHR Prescription Record	R4	Relaxed	3A	1.2.36.1.2001.1006.1.170.2
		Enforced	3A	1.2.36.1.2001.1006.1.170.3
PCEHR Dispense Record	R4	Relaxed	3A	1.2.36.1.2001.1006.1.171.2
		Enforced	3A	1.2.36.1.2001.1006.1.171.3
Event Summary	R4	Relaxed	3A	1.2.36.1.2001.1006.1.16473.9
			3B	1.2.36.1.2001.1006.1.16473.8
		Enforced	3A	1.2.36.1.2001.1006.1.16473.10
			3B	1.2.36.1.2001.1006.1.16473.11
Shared Health Summary	R4	Relaxed	3A	1.2.36.1.2001.1006.1.16575.4
			3B	1.2.36.1.2001.1006.1.16575.5
		Enforced	3A	1.2.36.1.2001.1006.1.16575.6
			3B	1.2.36.1.2001.1006.1.16575.7
Specialist Letter	R4	Relaxed	1A	1.2.36.1.2001.1006.1.16615.13
			1B	1.2.36.1.2001.1006.1.16615.14
			2	1.2.36.1.2001.1006.1.16615.15
			3A	1.2.36.1.2001.1006.1.16615.16
			3B	1.2.36.1.2001.1006.1.16615.17
		Enforced	1A	1.2.36.1.2001.1006.1.16615.18
			1B	1.2.36.1.2001.1006.1.16615.19
			2	1.2.36.1.2001.1006.1.16615.20
			3A	1.2.36.1.2001.1006.1.16615.21
			3B	1.2.36.1.2001.1006.1.16615.22

### 10.1.5 Episode Matching

Every clinical document uploaded by HIPS must be attached to an episode, which can be an inpatient admission, an emergency visit, an outpatient appointment or another type configured in EpisodeType.

There are two ways to create and manage episodes. One is by sending HL7 messages (such as A01 or A08) to the HIPS PAS Loader, while the other is to use the ValidatedIhi type of patient identifier and allow HIPS to create and manage episode stub records for uploaded documents.

HIPS will look for a previous episode stub for the upload of a document with the same Set ID as this document instance. If one is found then this episode stub is used and updated to the given admission date/time. Otherwise, the admission date and time are used to match an Episode record.

If there is no episode matched, and the patient identifier is of type ValidatedIhi, then an episode stub record will be created. The stub record will have the document Set ID as its source system episode ID, so that future supersedes and removals of this document can proceed even if the admission date/time has changed. For other patient identifier types, if there is no episode matched, or there is more than one episode with the same admission date and time (within one minute), then an "InvalidEpisode" error will be returned.

### 10.1.6 Consent Checking

The ability to handle a patient's withdrawal of consent is a core requirement for Clinical Information Systems connecting to the PCEHR System. Jurisdictions using HIPS can make use of the HIPS consent management services or ignore this functionality and handle consent in other systems. In line with the PCEHR consent model, when an episode is created, the default value for UploadConsent is true, that the patient has given consent to upload documents. This can be changed using the RecordConsent service.

Once an episode is matched, HIPS will check the UploadConsent flag. If the flag is set to false, this means the patient withdrew consent to upload this document. HIPS will return a "ConsentWithdrawn" error and prevent the upload of the document.

### 10.1.7 Age Checking

Some implementers may have a policy that they will disable uploading documents for children under a certain age. This age can be configured for each hospital in the Hospital table. This feature can also be disabled by setting the value 0.

For consistency, HIPS will calculate the age of the patient at the time of admission, rather than the time of discharge or the time of the upload.

If the value of "UploadDocumentMinimumAge" configured for the hospital is non-zero, and the patient was under the configured age at the time of admission, then HIPS will return a "PatientUnderAge" error and the document will not be added to the queue.

### 10.1.8 Attachment Validation

HIPS will ensure that each attachment is under the maximum size for attachments in the PCEHR system (10 megabytes) and that each attachment belongs to one of the supported file types. If either of these checks fails, an "InvalidDocument" error will be returned and the document will not be added to the queue.

The types of attachment files supported by the PCEHR are:

---

<b>MIME type</b>	<b>File Extensions</b>	<b>Description</b>
image/gif	.gif	Graphics Interchange Format
image/jpeg	.jpg, .jpeg	Joint Photographic Experts Group
image/tiff	.tif, .tiff	Tagged Image File Format

---

<b>MIME type</b>	<b>File Extensions</b>	<b>Description</b>
image/png	.png	Portable Network Graphics
application/pdf	.pdf	Portable Document Format

### 10.1.9 CDA Packaging

An electronic signature is created for the CDA document. The signature asserts the author's name and either the author's local system identifier or HPI-I that is extracted from the CDA document. The signature is created using the NASH certificate for the HPI-O that is in use by the hospital. This certificate has the serial number indicated by the PcehrCertSerial value in the HealthProviderOrganisation record in the database.

An "InvalidDocument" error is returned if these elements cannot be extracted from the CDA document, or the logo exceeds the maximum dimensions of 400 x 100 pixels.

An XDM package (ZIP file) is created, consisting of:

- The XML provided in the cdaDocument parameter as CDA\_ROOT.XML. The XML is unchanged except that integrity check attributes are added to the logo reference if HIPS is including an organisational logo from the hospital configuration.
- The created electronic signature as CDA\_SIGN.XML
- If a logo file is configured for the hospital, and not provided as an attachment, the logo file from the hospital configuration is included with the file name specified in the document.
- Any additional attachments provided in the attachments parameter.

After the consent check, signing and packaging, the upload request is added to the upload queue and the service returns to the caller. After the service has returned, the status of the document upload may be determined using the GetOperationStatus method.

### 10.1.10 Cancellation of a Queued Upload Operation

It may be necessary to manually cancel a queued upload operation, if the PCEHR system returns an error message that is documented to have the meaning of system temporarily unavailable, but is actually due to an error in the structure of the uploaded document or metadata. This can be done by setting the queue status in the PcehrMessageQueue record to 3 (failed).

Each time an upload operation is taken off the queue for processing, HIPS checks the current value of the QueueStatusId in the PcehrMessageQueue record. If the queue status is no longer set to 1 (pending) then HIPS will stop processing this upload operation, and the queue processor will move to the next message in the queue.

### 10.1.11 Ordering of Queued Operations

If the PCEHR system is temporarily unavailable, the queued upload operation will be placed on a retry queue. The retry queue is standard functionality of the MSMQ binding in the web.config file. When a message has reached the receiveRetryLimit (set to 3) it is then paused to wait for a period of time set by the retryCycleDelay (set to 5 minutes). The maxRetryCycles is set to 6000 so that the PCEHR can be completely offline for a minimum of 20 days (excluding connection timeout delays for each retry) before the MSMQ is placed into a halted state.

While one or more operations are on the retry queue, other upload requests can arrive. HIPS will attempt to process these operations immediately as the PCEHR system may now have sufficient capacity to process these requests.

However it is important to ensure that requests that relate to the same document set are processed in the correct order. Therefore, each time an upload operation is taken off the queue for processing, HIPS checks whether there are any earlier pending operations for the same document set. If there are any, then HIPS will roll back the queue transaction, so that the upload operation will be retried after the earlier request is completed.

#### 10.1.12 Determination of Request Type

When an upload operation is taken off the queue for processing, the Set ID and Document ID are used to locate existing records in the database that related to the document that is being uploaded.

If the Document ID exists in the ClinicalDocumentVersion table, then the queued operation will be marked as failed because the document has already been uploaded. HIPS will not attempt to upload the document again.

If the Document ID does not exist, but the Set ID exists in the ClinicalDocument table, then a request is generated with the type of replacement (supersede) of the most recently uploaded Document ID in the ClinicalDocumentVersion table that is recorded against the matched record in the ClinicalDocument table. This is the mechanism by which HIPS ensures that it only replaces documents that it uploaded itself.

If neither the Document ID nor the Set ID exists in the database, then the request type is an upload of a new document.

#### 10.1.13 Auditing of Request and Response

HIPS will upload or supersede the clinical document to the PCEHR National Repository, and write an audit record into the PcehrAudit table. This audit record contains the full SOAP request and response, and is vital for troubleshooting when documents have failed to upload.

#### 10.1.14 Response Classification

Using the definitions of the PCEHR error messages in the Technical Service Specification, each response from the PCEHR is classified as one of:

- Success
- Warning (e.g. operation successful but persisted as unstructured document)
- Duplicate Document ID
- System Temporarily Unavailable
- Unrecoverable Error

In the case of a success, warning or duplicate document ID message, the queued operation is deleted and records of the document are stored into the ClinicalDocument and ClinicalDocumentVersion tables.

In the PCEHR model, superseding a removed document changes its status back to active; all versions, including the one that was removed, are once again visible to both consumers and providers. Accordingly, in the HIPS ClinicalDocument table the document status will be reset to "Uploaded" (even if it was previously "Removed"). The removal date and removal reason will be reset to null and -1 (unknown) respectively.

The reason why duplicates are treated the same as success at this point is because the document was missing from the HIPS database even though it was already on

the PCEHR system. This can happen in the case when the HIPS database is restored from backups following the disaster recovery process, and documents that were uploaded after the recovery point are resent to ensure they are recorded correctly in the restored HIPS database.

In the case of a message classified as System Temporarily Unavailable, HIPS will roll back the queue transaction, so that the upload operation will be retried. The retry behaviour is controlled by the MSMQ settings in the web.config file. See the HIPS Installation Guide for details.

In the case of an unrecoverable error, such as an invalid document structure, the queued operation is marked as failed, and the queue processor will move to the next message in the queue.

## 10.2 Technical Structure

### 10.2.1 Classes and Enumerations

The "PCEHRService" class in the "HIPS.AppServer.ServiceHost" project implements this web service.

The "DocumentUploadBeforeQueue", "DocumentUpload" and "DocumentUploadInvoker" classes in the "HIPS.PcehrBusinessLogic" project implement the business logic.

#### 10.2.1.1 Attachment

Class "Attachment" is located in the "HIPS.PcehrSchemas" project. This object represents a file that is attached to a document that is uploaded or downloaded from a PCEHR repository.

The Attachment class contains the following properties:

Property	Data Type	Description
FileName	string	The name of the file, e.g. "XRAY.PNG" or "DOCUMENT.PDF"
Contents	binary	File Contents

### 10.2.2 Methods

#### 10.2.2.1 UploadOrSupersedeDocument

This service takes the following parameters:

Parameter	Data Type	Description	Required/ Optional
cdaDocument	binary	The CDA root XML document	Required
patientIdentifier	PatientIdentifierBase	An identifier of the patient (MRN, State ID, PatientMasterId or validated IHI)	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

<b>Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/ Optional</b>
attachments	array of Attachment	The attached files (GIF, JPEG, PNG, TIFF or PDF) that are referenced by the CDA document. Note that it is not necessary to include organisation logo file in the array because HIPS can automatically add the logo that is configured in the Hospital table of the PCEHR Data Store.	Optional
admissionDate	DateTime	The date and time when the hospital episode started. This is used to match with data from the PAS feed and the consent recording service. If consent is withdrawn from the episode then the document will not be uploaded.	Required
documentFormat Code	string	The document format code is used to specify which validation rules the PCEHR system will apply to the document when it is uploaded. As the PCEHR system is upgraded over time, there are new document format codes that can be used. Any new document format codes must first be configured in the DocumentFormat table. If this parameter is not supplied, HIPS will default to the value specified in the configuration parameter "DefaultDocumentFormatCode".	Optional
<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>	
response	HipsResponse	An indicator of whether the request was queued or failed to be queued and the reason why.	

## 11. Remove Document from PCEHR

### 11.1 Functional Logic

This service implements a “fire and forget” pattern that adds a request to a queue for removing a document from the associated repository for the document type. The service will return as soon as the item is added to the queue.

#### 11.1.1 Business Rules / Functional Business Logic

The remove document operation will have identical patient matching, IHI validation and episode matching logic to the upload or supersede document operation.

#### 11.1.2 Cancellation of a Queued Remove Operation

It may be necessary to manually cancel a queued remove operation, if the PCEHR system returns an error message that is documented to have the meaning of system temporarily unavailable, but is actually due to an error in the removal request. This can be done by setting the queue status in the PcehrMessageQueue record to 3 (failed).

Each time a remove operation is taken off the queue for processing, HIPS checks the current value of the QueueStatusId in the PcehrMessageQueue record. If the queue status is no longer set to 1 (pending) then HIPS will stop processing this operation, and the queue processor will move to the next operation in the queue.

#### 11.1.3 Ordering of Queued Operations

If the PCEHR system is temporarily unavailable, the queued remove operation will be placed on a retry queue. The retry queue is standard functionality of the MSMQ binding in the web.config file. When a message has reached the receiveRetryLimit (set to 3) it is then paused to wait for a period of time set by the retryCycleDelay (set to 5 minutes). The maxRetryCycles is set to 6000 so that the PCEHR can be completely offline for a minimum of 20 days (excluding connection timeout delays for each retry) before the MSMQ is placed into a halted state.

While one or more operations are on the retry queue, other remove requests can arrive. HIPS will attempt to process these operations immediately as the PCEHR system may now have sufficient capacity to process these requests.

However it is important to ensure that requests that relate to the same document set are processed in the correct order. Therefore, each time a remove operation is taken off the queue for processing, HIPS checks whether there are any earlier pending operations for the same document set. If there are any, then HIPS will roll back the queue transaction, so that the remove operation will be retried after the earlier request is completed.

#### 11.1.4 Auditing of Document Removal

The additional audit information provided in the “auditInformation” parameter will be stored in the RemoveAudit table.

#### 11.1.5 Auditing of Request and Response

After sending the remove document request to the PCEHR system, HIPS will write an audit record into the PcehrAudit table. This audit record contains the full SOAP request and response.

### 11.1.6 Response Classification

Using the definitions of the PCEHR error messages in the Technical Service Specification, each response from the PCEHR remove document operation will be classified as one of:

- Success
- System Temporarily Unavailable
- Unrecoverable Error

In the case of a success message, the queued operation will be deleted and the ClinicalDocument record will be updated. HIPS will set the DocumentStatus to removed, set the RemovalDate to the current time, and set the RemovalReason to the value provided in the "reason" parameter.

In the case of a message classified as System Temporarily Unavailable, HIPS will roll back the queue transaction, so that the remove operation will be retried. The retry behaviour is controlled by the MSMQ settings in the web.config file. See the HIPS Installation Guide for details.

In the case of an unrecoverable error, the queued operation is marked as failed, and the queue processor will move to the next message in the queue.

## 11.2 Technical Structure

### 11.2.1 Classes and Enumerations

The "PCEHRService" class in the "HIPS.AppServer.ServiceHost" project implements this web service.

The "DocumentRemovalBeforeQueue", "DocumentRemoval" and "DocumentRemovalInvoker" classes in the "HIPS.PcehrHiBusinessLogic" project implement the business logic.

### 11.2.2 Methods

#### 11.2.2.1 *Remove*

This service method takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/ Optional</b>
patientIdentifier	PatientIdentifierBase	An identifier of the patient	Required
admissionDate	DateTime	The date and time of the start of the episode of care to which this document relates.	Required
documentSetId	string	The identifier of the document, which remains the same across multiple versions of the document. This is formed from the root and extension attributes of the <setId> element of the CDA document. If there is both a root and extension attribute then the two values are separated by a ^ (caret), otherwise if there is only a root attribute then the value is that of the root attribute without a caret.	Required

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
reason	enum RemovalReason <ul style="list-style-type: none"> <li>Withdrawn</li> <li>IncorrectIdentity</li> </ul>	<p>The reason why the document is to be removed from the PCEHR.</p> <ul style="list-style-type: none"> <li>“Withdrawn” means that the authoring organisation has chosen to withdraw the document.</li> <li>“IncorrectIdentity” means that the document was found to be uploaded against the incorrect consumer’s eHealth record.</li> </ul> <p>Note that the value "ElectToRemove" is not permitted because that reason is defined for the consumer’s use only.</p>	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required
auditInformation	Binary	Arbitrary information that the clinical system wishes HIPS to store in the RemoveAudit table. Examples may include a link to a service request in a service management system, or a scanned image of a request form signed by the requesting clinician.	Required
<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>	
response	HipsResponse	An indicator of whether the request was queued or failed to be queued and the reason why.	

## 12. Get Operation Status

### 12.1 Functional Logic

This service is used to return the list of pending and failed Queued Operations, Uploaded Documents and Document Versions for a particular patient for a specific episode.

#### 12.1.1 Business Rules / Functional Business Logic

This service is intended for system analysis of pending or failed operation requests from the HIPS operational message queue.

### 12.2 Technical Structure

#### 12.2.1 Classes and Enumerations

The "PCEHRService" class in the "HIPS.AppServer.ServiceHost" project implements this web service.

The "OperationQuery" class in the "HIPS.PcehrHiBusinessLogic" project implement the business logic.

##### 12.2.1.1 *OperationStatus*

Class "OperationStatus" is located in the "HIPS.PcehrSchemas" project. This object represents the response from the GetOperationStatus method.

The OperationStatus class contains the following properties:

Property	Data Type	Description
QueuedOperations	List of PcehrMessageQueue	List of pending and failed queue operations. If the upload or remove request is not yet started or still in progress, it will be in this list with status of pending. If the upload or remove request failed, it will be in this list with a status of failed. All successful uploads are removed from the queued list and thus not displayed.
UploadedDocuments	List of ClinicalDocument	List of uploaded clinical documents. If the upload or remove request succeeded, the document will be in this list and the document status will show Uploaded or Removed as appropriate.
UploadedDocumentVersions	List of ClinicalDocumentVersion	List of individual versions of uploaded clinical documents. This list contains the complete package contents of each version uploaded, and the date and time when it was uploaded.
Response	HipsResponse	Indicator of success or failure of this request and the reason for failure.

### 12.2.1.2 *IndividualOperationStatus*

Class "IndividualOperationStatus" is located in the "HIPS.PcehrSchemas" project. This object represents the response from the GetIndividualOperationStatus method. This is for PcehrMessageQueue items that are still within the PcehrMessageQueue table.

The IndividualOperationStatus class contains the following properties:

Property	Data Type	Description
QueuedOperation	PcehrMessageQueue	The individual pending or failed queue operation. If the upload or remove request is not yet started or still in progress, it will be displayed with a status of pending. If the upload or remove request failed, it will be displayed with a status of failed. All successful uploads are removed from the queued list and thus not displayed.
ClinicalDocument	ClinicalDocument	Attempted Upload/Supersede or Removal of a clinical document, which is not as yet successful
DecodedPackage	binary	(for Upload and Supersede only) The CDA package (ZIP file) that is being attempted to uploaded to the PCEHR system. This contains the CDA document XML file, the signature XML file, the hospital logo image file and any other attached images or PDFs.
Response	HipsResponse	Indicator of success or failure of this request and the reason for failure.

### 12.2.1.3 *PcehrMessageQueue*

Class "PcehrMessageQueue" is located in the "HIPS.PcehrDataStore.Schemas" project. This object represents a queued upload or remove operation.

The PcehrMessageQueue class contains the following properties:

Property	Data Type	Description
PcehrMessageQueueId	int	The database primary key for the queued operation.
QueueOperationId	int	The database primary key for the operation type. These primary keys are fixed because they are referenced by a HIPS enumeration: UploadOrSupersede = 1 Remove = 2
QueueOperationName	string	The description of the queued operation type, from the QueueOperation reference table.

Property	Data Type	Description
QueueStatusId	int	The database primary key for the operation status. These primary keys are fixed because they are referenced by a HIPS enumeration: Pending = 1 Success = 2 Failure = 3
QueueStatusName	string	The description of the queued operation status, from the QueueStatus reference table.
EpisodeId	int	The database primary key of the patient episode.
SourceSystemSetId	string	The ID assigned to the document set by the system that created the CDA document. This is extracted from the <setId> element in the CDA document.
SourceSystemDocumentId	string	The ID assigned to the document version by the system that created the CDA document. This is extracted from the <id> element in the CDA document.
SerializedObject	binary	Operation-specific details. In the case of an upload operation, this includes the packaged document that is to be uploaded.
Request	string	The SOAP request that HIPS sent to the PCEHR system. This is only populated if the queued operation has failed.
Response	string	The SOAP response that HIPS received from the PCEHR system. This is only populated if the queued operation has failed.
Details	string	Details of the error that occurred. This is only populated if the queued operation has failed.

#### 12.2.1.4 *ClinicalDocument*

Class "ClinicalDocument" is located in the "HIPS.PcehrDataStore.Schemas" project. This object represents an uploaded clinical document "set". It contain high-level information that is common across all versions of a document.

The ClinicalDocument class contains the following properties:

Property	Data Type	Description
ClinicalDocumentId	int	The database primary key for the uploaded document.
SourceSystemSetId	string	The ID assigned to the document set by the system that created the CDA document. This is extracted from the <setId> element in the CDA document. If there is an extension attribute, its value is appended to the value of the root attribute, separated by a caret ("^"). Otherwise this is just the value of the root attribute.

Property	Data Type	Description
EpisodeId	int	The database primary key of the patient episode.
DocumentTypeId	int	The database primary key of the document type. The document type is determined using the <code> element in the CDA document. The value of the code element must match a code in the DocumentType reference table.
DocumentTypeDescription	string	The description of the document type, from the DocumentType reference table.
DocumentTypeCode	string	The code for the document type, from the DocumentType reference table.
RemovedDate	DateTime (nullable)	The date and time when the document was removed from the PCEHR system. If the document is currently active on the PCEHR system then this value is null.
RemovalReasonId	int	The database primary key of the reason for removal. This is a fixed set of values because it is referenced by a HIPS enumeration: Withdrawn = 0 IncorrectIdentity = 2
RemovalReasonDescription	string	The description of the removal reason, from the RemovalReason reference table.
ClinicalDocumentStatusId	int	The database primary key for the clinical document status. This is a fixed set of values because it is referenced by a HIPS enumeration: Active = 1 Removed = 2

#### 12.2.1.5 *ClinicalDocumentVersion*

Class "ClinicalDocumentVersion" is located in the "HIPS.PcehrDataStore.Schemas" project. This object represents an uploaded clinical document "version". It contains low-level information that is specific to a single version of a document. The version object is related to the document object via the ClinicalDocumentId.

The ClinicalDocumentVersion class contains the following properties:

Property	Data Type	Description
ClinicalDocumentVersionId	int	The database primary key for the uploaded clinical document version.
ClinicalDocumentId	int	The database primary key for the uploaded document.

Property	Data Type	Description
SourceSystemDocumentId	string	The ID assigned to the document version by the system that created the CDA document. This is extracted from the <id> element in the CDA document. If there is an extension attribute, its value is appended to the value of the root attribute, separated by a caret ("^"). Otherwise this is just the value of the root attribute.
EpisodeId	int	The database primary key of the episode.
UploadedDate	Date/Time	The date and time when the document version was uploaded to the PCEHR system.
SupersededDate	Date/Time (nullable)	The date and time when this document version was superseded by another version of the same document. If this version is the latest version uploaded, then this value is null.
Package	binary	The CDA package (ZIP file) that was uploaded to the PCEHR system. This contains the CDA document XML file, the signature XML file, the hospital logo image file and any other attached images or PDFs.

## 12.2.2 Methods

### 12.2.2.1 *GetOperationStatus*

This method is used to retrieve the pending and failed queue operations for a specific patient and admission date.

This service method takes the following parameters:

Input Parameter	Data Type	Description	Required/Optional
patientIdentifier	PatientIdentifierBase	An identifier of the patient	Required
admissionDate	DateTime	The date and time of the start of the episode of care for which to retrieve document status	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

Return Parameter	Data Type	Description
response	OperationStatus	List of queued operations and documents.

### 12.2.2.2 *GetIndividualOperationStatus*

This method is used to retrieve the pending and failed queue operations for a unique PcehrMessageQueueId (this can be initially derived from the GetQueuedOperationList).

This service method takes the following parameters:

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
PcehrMessageQueueId	int	The key for the queued operation.	Required
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	IndividualOperationStatus	List of queued operations and documents.

If the PcehrMessageQueueId cannot be found then an empty operation status will be returned.

## 13. Get Queued Operation List

### 13.1 Functional Logic

This service is used to return a list of active pending and failed Queued Operations.

#### 13.1.1 Business Rules / Functional Business Logic

This service is intended as a simplistic list of pending or failed operation requests from the HIPS operational message queue. Results returned can then be used to perform a request on the more detailed Get Operation Status (either with the "GetOperationStatus" or the "GetIndividualOperationStatus") service for more detailed analysis.

### 13.2 Technical Structure

#### 13.2.1 Classes and Enumerations

The "PCEHRService" class in the "HIPS.AppServer.ServiceHost" project implements this web service.

The "OperationQuery" class in the "HIPS.PcehrHiBusinessLogic" project implement the business logic.

##### 13.2.1.1 *QueuedOperationStatus*

Class "QueuedOperation" is located in the "HIPS.PcehrSchemas" project. This object represents the response from the GetQueuedOperationList method.

The QueuedOperation class contains the following properties:

---

<b>Property</b>	<b>Data Type</b>	<b>Description</b>
QueuedOperations	List of MessageQueueItem	List of pending and failed queue operations. If the upload or remove request is not yet started or still in progress, it will be in this list with status of pending. If the upload or remove request failed, it will be in this list with a status of failed. All successful uploads are removed from the queued list and thus not displayed.
Response	HipsResponse	Indicator of success or failure of this request and the reason for failure.

---

### 13.2.1.2 *MessageQueueItem*

Class "MessageQueueItem" is located in the "HIPS.PcehrSchemas" project.

This object is a contracted display of a queued message operation. The MessageQueueItem class contains the following properties:

<b>Property</b>	<b>Data Type</b>	<b>Description</b>
PcehrMessageQueueId	int	The database primary key for the queued operation.
QueueItemDateCreated	datetime	Date and time that the queue item was created
QueueOperationId	int	The database primary key for the operation type. These primary keys are fixed because they are referenced by a HIPS enumeration: UploadOrSupersede = 1 Remove = 2
QueueOperationName	string	The description of the queued operation type, from the QueueOperation reference table.
QueueStatusId	int	The database primary key for the operation status. These primary keys are fixed because they are referenced by a HIPS enumeration: Pending = 1 Success = 2 Failure = 3
QueueStatusName	string	The description of the queued operation status, from the QueueStatus reference table.
FacilityId	string	The database primary key of the facility that the operation occurred for.
FacilityName	string	Name of the facility that the operation occurred for.
PatientId	int	The database primary key of the HospitalPatient record for the patient.
IHI	string	IHI of the patient
EpisodeId	Int	The database primary key of the patient episode.
AdmissionDateTime	datetime	(If applicable) The date and time of the admission.
SourceSystemSetId	string	The ID assigned to the document set by the system that created the CDA document.
RequestSize	Int64	The approximate size (in bytes) of the SOAP request that HIPS sent to the PCEHR system. This is only populated if the queued operation has failed.
CDAPackageSize	Int64	(for Upload and Supersede only) The byte size of the CDA package (ZIP file) that is to be uploaded to the PCEHR system. This contains the CDA document XML file, the signature XML file, the hospital logo image file and any other attached images or PDFs.

Property	Data Type	Description
DocumentTypeId	int	(for Upload and Supersede only) The database primary key of the document type. The document type is determined using the <code> element in the CDA document. The value of the code element must match a code in the DocumentType reference table.
DocumentTypeDescription	string	(for Upload and Supersede only) The description of the document type, from the DocumentType reference table.
DocumentTypeCode	string	(for Upload and Supersede only) The code for the document type, from the DocumentType reference table.
DocumentFormatDescription	string	(for Upload and Supersede only) Document Format Description For example, the format description "PCEHR Release 3 Discharge Summary Level 3A".
DocumentFormatCode	string	(for Upload and Supersede only) Document Format Codes / Template IDs For example, the format code "1.2.36.1.2001.1006.1.20000.16" represents the validation rules for "PCEHR Release 3 Discharge Summary Level 3A", with the relaxation of display name for mode of separation.
Details	string	Details of the error that occurred. This is only populated if the queued operation has failed.

## 13.2.2 Methods

### 13.2.2.1 *GetQueuedOperationList*

This service method takes the following parameters:

Input Parameter	Data Type	Description	Required/Optional
DateTimeFrom	DateTime	Datetime for the start of the created queue items	Required
DateTimeTo	DateTime	Datetime for the end of the created queue items	Required
QueueOperation	IList <QueueOperation> /enum	Collection of the Queue Operations that are to be returned. If empty then all items will be returned.	Optional
QueueStatus	IList <QueueStatus> /enum	Collection of the Queue Status that are to be returned. If empty then all items will be returned.	Optional
patientIdentifier	PatientIdentifierBase	An identifier of the patient to filter by. If empty then all patients will be returned.	Optional

---

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/Optional</b>
User	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

---

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	IList <MessageQueueItem>	List of queued operations and details.

---

## 14. Reference Services

### 14.1 Functional Logic

The following items in the HIPS database are considered reference data and are cached in memory for faster access.

Schema Item	Tables	Information Represented
Hospital	Hospital HealthProviderOrganisation HospitalAddress, Address HospitalContact, Contact HospitalCode, CodeSystem	Name, Authorised Employee, Logo Image HPI-O, Certificate Serial Numbers Hospital Addresses Hospital Contact Methods (e.g. Phone, Fax) Hospital Codes
Title	Title	Name Titles (Dr, Ms, Mr, etc.)
Suffix	Suffix	Name Suffixes (Senior, Junior, etc.)
Sex	Sex	Sex Codes and Descriptions
Country	Country	Country Codes and Descriptions
State	State	Australian State Codes and Descriptions
EpisodeType	EpisodeType	Episode Type Codes and Descriptions
CodeSystem	CodeSystem	Coding Systems and Namespaces
HospitalCode	HospitalCode CodeSystem	Hospital Codes These codes are used to look up a hospital.
AddressType	AddressType	Address Types (Home, Temporary, Business, Mailing, etc.)
EpisodeLifecycle	EpisodeLifecycle	Episode Lifecycle Statuses (Pre-admit, Admitted, Discharged, Cancelled, etc.)
DocumentType	DocumentType	Document Types (Discharge Summary, etc.)
DocumentFormat	DocumentFormat CodeSystem	Document Format Codes / Template IDs For example, the format code "1.2.36.1.2001.1006.1.20000.16" represents the validation rules for "PCEHR Release 3 Discharge Summary Level 3A", with the relaxation of display name for mode of separation.
MedicareExclusion	MedicareExclusion	Certain values that are populated by PAS systems in the MedicareNumber field, which indicate that the person is ineligible for Medicare benefits or that the Medicare number is unknown.

### 14.1.1 Reload Reference Data

This method instructs HIPS to reload all reference data from the database. The system administrator can invoke this method after making a change to reference data in the HIPS database, instead of restarting the HIPS application server.

### 14.1.2 Get Hospital Details

This method returns a set of information about a specified hospital, which is useful for the generation of a CDA document from that hospital.

## 14.2 Technical Structure

### 14.2.1 Classes and Enumerations

The "ReferenceService" class in the "HIPS.Reference.ServiceHost" project implements this web service.

The "ListSingleton" and "HospitalSingleton" classes in the "HIPS.CommonBusinessLogic" project implement the business logic.

#### 14.2.1.1 HospitalDetailResponse

Class "HospitalDetailResponse" is located in the "HIPS.BusinessEntities" project. This object represents the response from the GetHospitalDetails method.

The HospitalDetailResponse class contains the following properties:

Property	Data Type	Description
ShortDescription	string	A short but user-friendly description of the hospital, for example "TQEH". This may or may not be the same as the hospital code.
FullName	string	The full name of the hospital, for example "The Queen Elizabeth Hospital".
Contacts	List of Contact	Electronic contact methods (such as phone and fax numbers) for the hospital.
Addresses	List of Address	Physical addresses (such as street and mailing address) for the hospital.
HpiO	string	The HPI-O of the health provider organisation that operates this hospital. This may go into the CDA document as the HPI-O of the authoring organisation, encounter facility, custodian, etc.
ProviderIdentifierRoot	string	The root OID to use for a local identifier allocated by the hospital PAS for the responsible health professional at time of discharge.
PatientIdentifierRoot	string	The root OID to use for the MRN allocated by the hospital PAS for the patient.

<b>Property</b>	<b>Data Type</b>	<b>Description</b>
HospitalIdentifierRoot	string	The root OID to use for a local identifier for the hospital itself.
HospitalIdentifierExtension	string	The extension to use for a local identifier for the hospital itself.

#### 14.2.1.2 *Contact*

Class "Contact" is located in the "HIPS.PcehrDataStore.Schemas" project. This object represents an electronic contact method.

The Contact class contains the following properties:

<b>Property</b>	<b>Data Type</b>	<b>Description</b>
ContactId	int	The database primary key of the contact.
HospitalId	int	The database primary key of the hospital.
Detail	string	The actual telecommunications number or address, e.g. "(08) 8111 2222".
ContactMethodId	int	The database primary key of the contact method type. These primary keys are fixed because HIPS has a built-in enumeration: HomePhone = 1, HomeFax = 2, PersonalMobile = 3, PersonalEmail = 4, Pager = 5, WorkPhone = 6, WorkFax = 7, WorkMobile = 8, WorkEmail = 9.
ContactMethodDescription	string	The description of the contact method, e.g. "Work Phone" or "Work Fax".
TelecommunicationType	string	The HL7 telecommunication type for the contact method (e.g. H, MC, PG, WP, MC).
CdaType	string	The URL scheme for the contact method (e.g. tel, fax, mailto).

### 14.2.1.3 Address

Class "Address" is located in the "HIPS.PcehrDataStore.Schemas" project. This object represents an physical address.

The Address class contains the following properties:

Property	Data Type	Description
AddressId	int	The database primary key of the address.
HospitalId	int	The database primary key of the hospital.
AddressLine1	string	The first line of the unstructured address.
AddressLine2	string	The second line of the unstructured address.
PlaceName	string	The name of the suburb, town or locality.
AustralianStateId	int	The database primary key for the Australian state or territory.
AustralianStateCode	string	The code for the Australian state or territory.
AustralianStateName	string	The name of the Australian state or territory.
InternationalStateCode	string	The code for the international state, province or territory.
PostCode	string	The post code (ZIP) for national and international addresses.
CountryId	int	The database primary key for the country.
CountryName	string	The name of the country.
AddressTypeId	int	The database primary key of the address type.
AddressTypeDescription	string	The description of the address type, e.g. "Home", "Business", "Temporary" or "Mailing".

## 14.2.2 Methods

### 14.2.2.1 ReloadReferenceData

This service method takes the following parameters:

Input Parameter	Data Type	Description	Required/Optional
user	UserDetails	The interactive user, authorised employee or health provider individual responsible for this action	Required

Return Parameter	Data Type	Description
response	HipsResponse	An indicator of whether the request was successful or failed and the reason why.

#### 14.2.2.2 *GetHospitalDetails*

This service method takes the following parameters:

---

<b>Input Parameter</b>	<b>Data Type</b>	<b>Description</b>	<b>Required/ Optional</b>
hospitalCode	string	A code that identifies the target hospital	Required
hospitalCodeSystem	string	A code that identifies which code system the previous code is a member of	Required

---

---

<b>Return Parameter</b>	<b>Data Type</b>	<b>Description</b>
response	HospitalDetailResponse	Details of the hospital facility.

---

## 15. Common Schemas

### 15.1 User Details

#### 15.1.1 Functional Logic

The user is included as a parameter on all calls to HIPS in order to assert the authorisation role under which any calls to the HI Service or PCEHR System should take place.

#### 15.1.2 UserDetails

The class consists of the following properties:

Property	Data Type	Description	Required/Optional
Role	enum UserRole <ul style="list-style-type: none"> <li>• ProviderIndividual</li> <li>• InteractiveUser</li> <li>• AuthorisedEmployee</li> </ul>	The authorisation role for any B2B web service invocations triggered by this action. One of the following: <ul style="list-style-type: none"> <li>• ProviderIndividual – Access by Individual Healthcare Provider identified by an HPI-I</li> <li>• InteractiveUser - Interactive access by clinician or patient administration clerk etc.</li> <li>• AuthorisedEmployee – Authorised employee responsible for non-interactive access by batch or background processes</li> </ul>	Required
HpiI	string	The HPI-I of the health provider who is logged in.	Required if Role is ProviderIndividual otherwise optional.
Login	string	The locally-issued user ID of the interactive user, such as their Active Directory account name.	Required if Role is InteractiveUser otherwise optional.
Domain	string	The type of the locally-issued user ID, such as the Active Directory domain.	Required if Role is InteractiveUser otherwise optional.
Name	string	The name of the person logged in, or the name of the Authorised Employee. This is not sent to Medicare but is sent to PCEHR and will be audited.	Required if Role is ProviderIndividual or InteractiveUser. Optional if Role is AuthorisedEmployee. If omitted then the name will come from the hospital table.

Property	Data Type	Description	Required/ Optional
Authorised Employee UserId	string	The user ID that represents the Authorised Employee for non-interactive requests. This may be omitted and will be populated from the hospital configuration in the database.	Optional if Role is AuthorisedEmployee. If omitted then the ID will come from the hospital table. Otherwise prohibited.
Is Contracted Service Provider	bool	A flag indicating whether the HIPS server is being operated by a contracted service provider (CSP) on behalf of a healthcare provider organisation. In calls to HI Service, the HPI-O is only included if the call is made by a CSP on behalf of a healthcare provider organisation.  Note that HIPS has not been tested in a scenario where it is operated by a CSP, only as part of a distributed CIS. Some assumptions, particularly around which certificate can be used for signing CDA documents as opposed to connecting to the PCEHR, may need to change for this to be successful.	Required

## 15.2 Patient Identifier

### 15.2.1 Functional Logic

Each HIPS service that is designed to act upon a single patient record will contain a parameter "patientIdentifier". The Patient Identifier object will be used to identify which hospital and patient to operate upon. The Patient Identifier can be either:

- Medical Record Number (MRN), scoped within a specified hospital,
- State Patient Identifier, scoped within the group,
- Validated Individual Healthcare Identifier (IHI), scoped nationally, or
- Patient Master Identifier, internal to the PCEHR Data Store.

### 15.2.2 PatientIdentifierBase

This class is an abstract base class. Service parameters of this type will accept an object from any one of its subclasses. The base class contains two properties that are used to identify the hospital, and an optional alternative name for the health provider organisation that will be supplied in the PCEHR headers and is intended to appear on the PCEHR audit log.

This class consists of the following properties.

Property	Data Type	Description	Required/Optional
HospitalCode	string	The hospital code. This code, qualified by the hospital code system, is used to determine: <ul style="list-style-type: none"> <li>Which hospital allocated the MRN, if the Mrn subclass is used,</li> <li>Which hospital the episode occurred in, for consent, upload and remove calls,</li> <li>Which authorised employee is identified, if the user Role is Authorised Employee, and</li> <li>Which health provider organisation (HPI-O) is making the calls to HI Service and PCEHR B2B Gateway.</li> </ul>	Required, except where concrete type is PatientMasterId and the HospitalId has been provided.
HospitalCodeSystem	string	The hospital code system. This is necessary because different systems may use different codes to identify the same hospital. For example, "NOA" or "NHS" for Noarlunga Health Service, where "NOA" is the LAB Facility Code and "NHS" is the PAS Facility Code.	Required, except where concrete type is PatientMasterId and the HospitalId has been provided.
AlternateOrganisationName	string	The alternate organisation name. As defined in "PCEHR R12 Implementation Guide", the accessing organisation may choose to disclose a different organisation name on the PCEHR Audit Log by populating the alternateOrganisationName in the PCEHR header.	Optional

### 15.2.3 Mrn

This class represents a Medical Record Number (MRN) that identifies a patient at a hospital. It is usually allocated by the hospital PAS or PMI. This value is stored in the Mrn column of the HospitalPatient table in the PCEHR Data Store.

This class consists of the following properties.

Property	Data Type	Description	Required/Optional
Value	string	The value of the MRN	Required

### 15.2.4 StatePatientId

This class represents a number or code that identifies a patient within a group. It is usually allocated by the EMPI. This value is stored in the StatePatientId column of the PatientMaster table in the PCEHR Data Store.

This class consists of the following properties.

Property	Data Type	Description	Required/Optional
Value	string	The value of the state patient identifier	Required

### 15.2.5 ValidatedIhi

This class represents the set of information that makes up a validated IHI. The information includes everything that is required to determine whether the IHI remains valid, and to revalidate the IHI. These values are stored in the PatientMaster and PatientMasterIhi tables in the PCEHR Data Store.

When a validated IHI is used to identify a patient in a HIPS service call, the HIPS will create or update the patient record as necessary, taking the incoming information as authoritative. This makes it possible to operate HIPS in a distributed system where HIPS does not receive a feed of PAS messages and does not make its own connections to the HI Service.

Note that this class does not contain properties for Medicare Card Number or DVA File Number, because this information is not required for IHI validation, but only when the IHI is first retrieved.

This class consists of the following properties.

Property	Data Type	Description	Required/Optional
Ihi	string	The value of the IHI	Required
IhiStatus	enum IhiStatus <ul style="list-style-type: none"> <li>• Active</li> <li>• Deceased</li> <li>• Retired</li> <li>• Expired</li> <li>• Resolved</li> </ul>	The IHI status	Required
IhiRecordStatus	enum IhiRecordStatus <ul style="list-style-type: none"> <li>• Verified</li> <li>• Unverified</li> <li>• Provisional</li> </ul>	The IHI record status Note: In this release HIPS only supports Verified IHIs. It is an error to provide an IHI with any record status apart from Verified.	Required
IhiLastValidated	DateTime	The date and time when the IHI was last validated with the HI Service. If this value is within the configured period then HIPS will not attempt to revalidate the IHI before use in a PCEHR service call.	Required
FamilyName	string	The family name that was used to obtain or validate the IHI with the HI Service	Required
GivenName	string	The given name that was used to obtain or validate the IHI with the HI Service	Required

Property	Data Type	Description	Required/Optional
DateOfBirth	DateTime	The date of birth that was used to obtain or validate the IHI with the HI Service	Required
Sex	enum SexEnumerator <ul style="list-style-type: none"> <li>• Male</li> <li>• Female</li> <li>• IntersexOrIndeterminate</li> <li>• NotStatedOrInadequatelyDescribed</li> </ul>	The sex that was used to obtain or validate the IHI with the HI Service	Required

### 15.2.6 PatientMasterId

This class represents the internal primary key of the PatientMaster table in the PCEHR Data Store. This option is made available for applications that share use of the PCEHR Data Store with HIPS and hence have direct knowledge of the database keys.

This class consists of the following properties.

Property	Data Type	Description	Required/Optional
Value	int	The value of the PatientMaster primary key	Required
HospitalId	nullable int	The value of the Hospital primary key. This is an alternative to providing the hospital code and code system in the patient identifier base.	Optional

## 15.3 HIPS Response

### 15.3.1 Functional Logic

This object is used to wrap up a response status indicator, error code, description and details that are returned from HIPS to calling systems.

### 15.3.2 HipsResponse

This class packages a HIPS response status indicator, along with any error codes and messages that are returned from the national infrastructure services including the HI Service and the PCEHR B2B Gateway.

This class consists of the following properties.

Property	Data Type	Description
ResponseStatus	enum HipsResponseIndicator <ul style="list-style-type: none"> <li>• OK</li> <li>• ConsentWithdrawn</li> <li>• HiServiceError</li> <li>• PcehrServiceError</li> <li>• SystemError</li> <li>• InvalidPatient</li> <li>• InvalidHospital</li> <li>• InvalidEpisode</li> <li>• InvalidDocument</li> <li>• InvalidUser</li> <li>• InvalidIhi</li> <li>• CouldNotAddToQueue</li> <li>• InvalidAccessCode</li> <li>• DemographicMismatchWarning</li> <li>• UnresolvedIhiAlert</li> <li>• DatabaseError</li> <li>• InvalidDateOfBirth</li> <li>• PcehrServiceWarning</li> <li>• PcehrServiceUnavailable</li> <li>• PatientUnderAge</li> <li>• CouldNotFindQueueItem</li> </ul>	Status of the operation: <ul style="list-style-type: none"> <li>• OK: The operation completed successfully. In case of Upload or Remove, the operation was added to a queue and will be processed later.</li> <li>• ConsentWithdrawn: The document was not uploaded because the patient withdrew consent to upload it.</li> <li>• HiServiceError: A call to HI Service was initiated but the response indicated the operation was not successful</li> <li>• PcehrServiceError: A call to PCEHR B2B Gateway was initiated but the response indicated the operation was not successful</li> <li>• SystemError: HIPS has detected an internal error or misconfiguration</li> <li>• Invalid[Item]: No outbound call was initiated because the validation of [Item] failed</li> <li>• CouldNotAddToQueue: The system could not add the operation (upload or remove) to the queue.</li> <li>• InvalidAccessCode: The Access Code is invalid.</li> <li>• DemographicMismatchWarning: The downloaded document had different patient information to the local system.</li> <li>• UnresolvedIhiAlert: The IHI for this patient cannot be used to connect to the PCEHR System because there are one or more unresolved alerts (such as potential duplicates or replicas) against it.</li> <li>• DatabaseError: The operation could not be completed because the local SQL Server database is unavailable.</li> <li>• InvalidDateOfBirth: The date of birth stored for the patient does not match the date of birth that was specified in the request.</li> <li>• PcehrServiceWarning : The action was successful but returned one or more warnings that should be investigated.</li> <li>• PcehrServiceUnavailable: The PCEHR Service is temporarily</li> </ul>

Property	Data Type	Description
		<p>unavailable. The action should be tried again later. In the case of an upload or remove operation, the queue transaction will be rolled back so that the MSMQ will handle retrying.</p> <ul style="list-style-type: none"> <li>• PatientUnderAge : The document was not uploaded because the patient was under the configured minimum age at the time of admission.</li> <li>• CouldNotFindQueueItem: This queue item could not be found - it may have been removed from the list or the ID is invalid</li> </ul>
ResponseCode	String	The response status code that was returned from one of the national infrastructure services.
ResponseCodeDescription	String	The description associated with the response status code that was returned from one of the national infrastructure services.
ResponseCodeDetails	String	Additional context or service messages returned from one of the national infrastructure services.
HipsErrorMessage	String	Additional context explaining the reason for the validation, consent or system error.

## Appendix A HIPS Architecture

Rapid Integration to the PCEHR  
 HIPS 4.1.0 Jurisdictional Release  
 February 2014

