Pillar test specification

A full pillar test specification is listen below. Examples of the acceptance test status for different concrete pillars can be found on the right.

TestCase: IdentifyPillarsForGetFileIT

Test 1: goodCaseIdentificationIT

Purpose
Tests the general IdentifyPillarsForGetFile functionality of the pillar for the successful scenario.

Step	Stimuli	Expected result	Result
1	Create and send the identify request message.	Should be received and handled by the pillar.	
2	Retrieve and validate the response getPillarID() the pillar.	The pillar should make a response.	

Test 2: nonExistingFileIdentificationIT

Purpose
Tests the IdentifyPillarsForGetFile functionality of the pillar for a IdentificationForGetFile for a non existing file.

Step	Stimuli	Expected result	Result
1	Create and send the identify request message.	Should be received and handled by the pillar.	
2	Retrieve and validate the response getPillarID() the pillar.	The pillar should make a response.	

TestCase: MultipleCollectionIT

Test 1: fileInOtherCollectionTest

Purpose
Tests that a file is put correctly to a second collection, and that the file can be access with getFile, getChecksums, getFileIDs and can be replaced and deleted correctly.

Step	Stimuli	Expected result	Result
1	Put the file to the second collection	Should complete successfully	
2	Send a getFileIDs for the file in the second collection	The fileID should be retrieved	
3	Send a getFileIDs for the file in the other collections	The file should not be found here	

TestCase: IdentifyPillarsForReplaceFileIT

Test 1: irrelevantCollectionTest

Purpose
Verifies identification works correctly for a collection not defined for the pillar

Step	Stimuli	Expected result	Result
1	Sending a putFile identification with a irrelevant collectionID. eg. the pillar is not part of the collection	The pillar under test should not make a response	

Test 2: missingCollectionIDTest

Purpose
Verifies the a missing collectionID in the request is rejected

Newest reports for different pillar implementations

- Reference pillar
- Checksum pillar
- KB pillar
- SB Tape pillar
- SB Disk pillar

Note: Reports may be incomplete if the lasted test run has errors.

Step	Stimuli	Expected result	Result
1	Sending a request without a collectionID.	The pillar should send a REQUEST_NOT_UNDERSTOOD_FAILURE Response.	

Test 3: otherCollectionTest

Purpose
Verifies identification works correctly for a second collection defined for pillar

Step	Stimuli	Expected result	Result
1	Sending a identify request with a non-default collectionID (not the first collection) the pillar is part of	The pillar under test should make a positive response	

Test 4: fileDoesNotExistsTest

Purpose
Verifies that a request for a non-existing file is handled correctly

Step	Stimuli	Expected result	Result
1	Sending a replaceFile identification for a file not in the pillar.	The pillar under test should send a FILE_NOT_FOUND_FAILURE response.	

Test 5: normalIdentificationTest

Purpose
Verifies the normal behaviour for replaceFile identification

Step	Stimuli	Expected result	Result
1	Sending a replaceFile identification.	The pillar under test should make a response with the correct elements.	

TestCase: GetChecksumTest

Test 1: md5ChecksumsForAllFilesTest

Purpose
Test the pillar support for MD5 type checksums

Step	Stimuli	Expected result	Result
1	Request MD5 checksums for all files on the pillar	A list (at least 2 long) of MD5 checksums should be returned.	
2	Retrieve the first two files and verify that the checksums are correct	Not implemented	

Test 2: md5SaltChecksumsForDefaultTest

Purpose
Test the pillar support for MD5 type checksums with a salt

Step	Stimuli	Expected result	Result
1	Request salted MD5 checksums for the default on the pillar	The correct of SHA1 checksum should be returned (Not checked yet).	

Test 3: sha1ChecksumsForDefaultTest

Purpose
Test the pillar support for SHA1 type checksums

Step	Stimuli	Expected result	Result
1	Request SHA1 checksums for the DefaultFile on the pillar	The SHA1 checksum for the default file should be returned should be returned (Not checked yet).	

Test 4: sha1SaltChecksumsForDefaultTest

Purpose
Test the pillar support for SHA1 type checksums with a salt

Step	Stimuli	Expected result	Result
1	Request salted SHA1 checksums for the default on the pillar	The correct of SHA1 checksum should be returned (Not checked yet).	

TestCase: PutFileRequestIT

Test 1: missingCollectionIDTest

Purpose
Verifies the a missing collectionID in the request is rejected

Step	Stimuli	Expected result	Result
1	Sending a request without a collectionID.	The pillar should send a REQUEST_NOT_UNDERSTOOD_FAILURE Response.	

Test 2: otherCollectionTest

Purpose
Verifies identification works correctly for a second collection defined for pillar

Step	Stimuli	Expected result	Result
1	Sending a identify request with a non-default collectionID (not the first collection) the pillar is part of	The pillar under test should make a positive response	

Test 3: normalPutFileTest

Purpose
Tests a normal PutFile sequence

Step	Stimuli	Expected result	Result
1	Send a putFile request to reference2	The pillar should send a final response with the following elements: 1. 'CollectionID' element corresponding to the supplied value 2. 'CorrelationID' element corresponding to the supplied value 3. 'From' element corresponding to the pillars component ID 4. 'To' element should be set to the value of the 'From' elements in the request 5. 'Destination' element should be set to the value of 'ReplyTo' from the request 6. 'ChecksumDataForExistingFile' element should be null 7. 'ChecksumDataForNewFile' element should be null 8. 'PillarID' element corresponding to the pillars component ID 9. 'FileID' element corresponding to the supplied fileID 10. 'FileAddress' element corresponding to the supplied FileAddress 11. 'ResponseInfo.ResponseCode' element should be OPERATION_COMPLETED	

Test 4: putFileOperationAcceptedProgressTest

Purpose
Tests a that a pillar sends progress response after receiving a putFile request.

Step	Stimuli	Expected result	Result
1	Send a putFile request to reference2	The pillar should generate a progress response with the following elements: 1. 'CollectionID' element corresponding to the value in the request. 2. 'CorrelationID' element corresponding to the value in the request. 3. 'From' element corresponding to the pillars component ID 4. 'To' element should be set to the value of the 'From' elements in the request 5. 'Destination' element should be set to the value of 'ReplyTo' from the request 6. 'PillarID' element corresponding to the pillars component ID 7. 'FileID' element corresponding to the supplied fileID 8. 'FileAddress' element corresponding to the supplied FileAddress 9. 'ResponseInfo.ResponseCode' element should be OPERATION_ACCEPTED_PROGRESS	

Test 5: putFileWithMD5ReturnChecksumTest

Purpose
Tests that the pillar is able to return the default type checksum in the final response

Step	Stimuli	Expected result	Result
1	Send a putFile request to reference2 with the	The pillar should send a final response with the ChecksumRequestForNewFile elemets containing the MD5 checksum for the supplied file.	

TestCase: GetStatusRequestIT

Test 1: normalGetStatusTest

Purpose
Tests the GetStatus functionality of a pillar for the successful scenario.

Step	Stimuli	Expected result	Result
1	Send a GetStatusRequest	The pillar should send a progress response followed by a OK final response.	
2	Receive and validate the final response	Should be sent by the pillar.	

TestCase: IdentifyPillarsForGetFileIDsIT

Test 1: irrelevantCollectionTest

Purpose
Verifies identification works correctly for a collection not defined for the pillar

Step	Stimuli	Expected result	Result
1	Sending a putFile identification with a irrelevant collectionID. eg. the pillar is not part of the collection	The pillar under test should not make a response	

Test 2: missingCollectionIDTest

Purpose
Verifies the a missing collectionID in the request is rejected

Step	Stimuli	Expected result	Result
1	Sending a request without a collectionID.	The pillar should send a REQUEST_NOT_UNDERSTOOD_FAILURE Response.	

Test 3: otherCollectionTest

Purpose
Verifies identification works correctly for a second collection defined for pillar

Step	Stimuli	Expected result	Result
1	Sending a identify request with a non-default collectionID (not the first collection) the pillar is part of	The pillar under test should make a positive response	

Test 4: nonExistingFileTest

Purpose
Tests that the pillar is able to reject a GetFileIDs requests for a file, which it does not have during the identification phase.

Step	Stimuli	Expected result	Result
1	Create and send the identify request message.	Should be received and handled by the pillar.	
2	Retrieve and validate the response getPillarID() the pillar.	The pillar should make a response.	

Test 5: normalIdentificationTest

Purpose
Verifies the normal behaviour for getFileIDs identification

Step	Stimuli	Expected result	Result
1	Sending a identify request.	The pillar under test should make a response with the correct elements.	
2	Create and send the identify request message.	Should be received and handled by the pillar.	
3	Retrieve and validate the response getPillarID() the pillar.	The pillar should make a response.	

TestCase: ReplaceFileRequestIT

Test 1: missingCollectionIDTest

Purpose
Verifies the a missing collectionID in the request is rejected

Step	Stimuli	Expected result	Result
1	Sending a request without a collectionID.	The pillar should send a REQUEST_NOT_UNDERSTOOD_FAILURE Response.	

Test 2: otherCollectionTest

Purpose
Verifies identification works correctly for a second collection defined for pillar

Step	Stimuli	Expected result	Result
1	Sending a identify request with a non-default collectionID (not the first collection) the pillar is part of	The pillar under test should make a positive response	

Test 3: normalReplaceFileTest

Purpose
Tests a normal ReplaceFile sequence

Step	Stimuli	Expected result	Result
1	Send a ReplaceFile request to reference2	The pillar should generate a OPERATION_ACCEPTED_PROGRESS progress response followed by a OPERATION_COMPLETED final response	

TestCase: GetChecksumQueryTest

Test 1: checksumSortingTest

Purpose
Test whether the checksum result is sorted oldest to newest.

Step	Stimuli	Expected result	Result
1	Ensure at least two files are present on the pillar		
2	Retrieve a list of all checksums.	Run through the list and verify each element is older or the same age as the following element	

Test 2: maxNumberOfResultTest

Purpose
Verifies the size of the result set can be limited by setting the maxNumberOfResult parameter.

Step	Stimuli	Expected result	Result
1	Ensure at least two files are present on the pillar		
2	Retrieve a list of all checksums by setting maxNumberOfResult to null.	At least 2 checksums should be returned	
3	Repeat the request checksums, this time with maxNumberOfResult set to one	A checksum result with a single checksum should be returned. The checksum should be the oldest/first checksum in the full list.	

Test 3: maxTimeStampTest

Purpose
Test the pillar support for only retrieving checksums older that a given time. Note that this test assumes there is at least 2 checksums with different timestamps.

Step	Stimuli	Expected result	Result
1	Request default checksums for all files on the pillar	A list with at least 2 different timestamps (it is not the fault of the pillar if this fails, but the test needs this to be satisfied to make sense).	
2	Request checksums with MaxTimeStamp set to the timestamp of the newest checksum	All checksums should be returned.	
3	Request checksums with MaxTimeStamp set to the timestamp of the oldest checksum	Only checksum with the timestamp equal to MaxTimeStamp are returned.	
4	Request checksums with MaxTimeStamp set to the timestamp of the oldest checksum - 10 ms	No checksums are returned.	

Test 4: minTimeStampTest

Purpose
Test the pillar support for only retrieving checksums newer that a given time. Note that this test assumes there is at least 2 checksums with different timestamps.

Step	Stimuli	Expected result	Result
1	Request default checksums for all files on the pillar	A list with at least 2 different timestamps (it is not the fault of the pillar if this fails, but the test needs this to be satisfied to make sense).	
2	Request checksums with MinTimeStamp set to the timestamp of the oldest checksum	All checksums should be returned.	
3	Request checksums with MinTimeStamp set to the timestamp of the newest checksum	Only checksum with the timestamp equal to MinTimeStamp are returned.	
4	Request checksums with MinTimeStamp set to the timestamp of the newest checksum + 10 ms	No checksums are returned.	

TestCase: IdentifyPillarsForPutFileIT

Test 1: irrelevantCollectionTest

Purpose
Verifies identification works correctly for a collection not defined for the pillar

Step	Stimuli	Expected result	Result
1	Sending a putFile identification with a irrelevant collectionID. eg. the pillar is not part of the collection	The pillar under test should not make a response	

Test 2: missingCollectionIDTest

Purpose
Verifies the a missing collectionID in the request is rejected

Step	Stimuli	Expected result	Result
1	Sending a request without a collectionID.	The pillar should send a REQUEST_NOT_UNDERSTOOD_FAILURE Response.	

Test 3: otherCollectionTest

Purpose
Verifies identification works correctly for a second collection defined for pillar

Step	Stimuli	Expected result	Result
1	Sending a identify request with a non-default collectionID (not the first collection) the pillar is part of	The pillar under test should make a positive response	

Test 4: fileExistsTest

Purpose
Verifies the exists of a file with the same ID is handled correctly. This means that a checksum for the existing file is returned, enabling the client to continue with the put operation for the pillars not yet containing the file. The client can easily implement idempotent behaviour based on this response.

Step	Stimuli	Expected result	Result
1	Sending a putFile identification for a file already in the pillar.	The pillar under test should send a DUPLICATE_FILE_FAILURE response with the (default type) checksum of the existing file.	

Test: 5 normalIdentificationTest

Purpose
Verifies the normal behaviour for putFile identification

Step	Stimuli	Expected result	Result
1	Sending a putFile identification request.	The pillar under test should make a response with the following elements: 1. 'CollectionID' element corresponding to the supplied value 2. 'CorrelationID' element corresponding to the supplied value 3. 'From' element corresponding to the pillars component ID 4. 'To' element should be set to the value of the 'From' elements in the request 5. 'Destination' element should be set to the value of 'ReplyTo' from the request 6. 'ChecksumDataForExistingFile' element should be null 7. 'PillarChecksumSpec' element should be null 8. 'PillarID' element corresponding to the pillars component ID 9. 'ResponseInfo.ResponseCode' element should be IDENTIFICATION_POSITIVE	

TestCase: IdentifyContributorsForGetStatusIT

Test 1: normalGetStatusTest

Purpose
Tests the GetStatus functionality of a pillar for the successful scenario.

Step	Stimuli	Expected result	Result
1	Send a IdentifyContributorsForGetStatusRequest.	The pillar should send a IDENTIFICATION_POSITIVE response.	

TestCase: GetFileIDsTest

Test 1: missingCollectionIDTest

Purpose
Verifies the a missing collectionID in the request is rejected

Step	Stimuli	Expected result	Result
1	Sending a request without a collectionID.	The pillar should send a REQUEST_NOT_UNDERSTOOD_FAILURE Response.	

Test 2: otherCollectionTest

Purpose
Verifies identification works correctly for a second collection defined for pillar

Step	Stimuli	Expected result	Result
1	Sending a identify request with a non-default collectionID (not the first collection) the pillar is part of	The pillar under test should make a positive response	

Test 3: pillarGetFileIDsTestBadDeliveryURL

Purpose
Test the case when the delivery URL is unaccessible.

Step	Stimuli	Expected result	Result

Test 4: pillarGetFileIDsTestDeliveryThroughUpload

Purpose
Test the case when the results should be delivered through the message .

Step	Stimuli	Expected result	Result

Test 5: pillarGetFileIDsTestFailedNoSuchFileInOperation

Purpose
Tests that the pillar is able to handle requests for a non existing file correctly during the operation phase.

Step	Stimuli	Expected result	Result
1	Send a GetFileIDs request for a non-existing file.	A FILE_NOT_FOUND_FAILURE response should be generated.	

Test 6: pillarGetFileIDsTestSuccessCase

Purpose
Tests the GetFileIDs functionality of the pillar for the successful scenario.

Step	Stimuli	Expected result	Result
1	Create and send a GetFileIDsRequest to the pillar.	A GetFileIDsProgressResponse should be sent to the client with correct attributes follow by a GetFileIDsFinalResponse.	
2	Retrieve the ProgressResponse for the GetFileIDs request	A GetFileIDs progress response should be sent to the client with correct attributes.	
3	Retrieve the FinalResponse for the GetFileIDs request	The GetFileIDs response should be sent by the pillar.	

TestCase: GetAuditTrailsTest

Test 1: eventSortingTest

Purpose
Test whether the checksum result is sorted oldest to newest.

Step	Stimuli	Expected result	Result
1	Ensure at least two files are present on the pillar.		
2	Retrieve a list of all audit trails. The list should be at least 2 elements long	Run through the list and verify each element sequence number is lower than the following elements.	

Test 2: maxNumberOfResultTest

Purpose
Verifies the size of the result set can be limited by setting the maxNumberOfResult parameter.

Step	Stimuli	Expected result	Result
1	Ensure at least two files are present on the pillar		
2	Retrieve a list of all audittrails by setting maxSequece to null.	At 2 audit trails should be returned	
3	Repeat the audit trail request, this time with maxNumberOfResult set to one	A result with a single audit event should be returned. The event should be the first audit event in the full list.	

Test 3: maxSequenceNumberTest

Purpose
Test the pillar support for only retrieving audit event with SequenceNumbers lower than MaxSequenceNumber.

Step	Stimuli	Expected result	Result
1	Request audit trails for all files on the pillar	A list with at least 2 events is returned.	
2	Request audit events with MaxSequenceNumber set to the SequenceNumber of the last event checksum	The full list of audit events should be returned.	
3	Request audit trail with MaxSequenceNumber set to the SequenceNumber of the first event	Only the first event is returned.	

Test 4: minSequenceNumberTest

Purpose
Test the pillar support for only retrieving events with sequence number higher than the provided MinSequenceNumber. Note that this test assumes there is at least 2 audit event.

Step	Stimuli	Expected result	Result
1	Request audit trails for all files on the pillar	A list with at least 2 events is returned.	
2	Request audit events with MinSequenceNumber set to the SequenceNumber of the first event checksum	The full list of audit events should be returned.	
3	Request audit trail with MinSequenceNumber set to the SequenceNumber of the last event	Only the last event is returned.	
4	Request audit trail with MinSequenceNumber set to the SequenceNumber of the last event + 1	No events are returned.	

TestCase: GetFileIDsQueryTest

Test 1: fileidsSortingTest

Purpose
Test whether the file id result is sorted oldest to newest.

Step	Stimuli	Expected result	Result
1	Ensure at least two files are present on the pillar		
2	Retrieve a list of all file ids.	Run through the list and verify each element is older or the same age as the following element	

Test 2: maxNumberOfResultTest

Purpose
Verifies the size of the result set can be limited by setting the maxNumberOfResult parameter.

Step	Stimuli	Expected result	Result
1	Ensure at least two files are present on the pillar		
2	Retrieve a list of all file ids by setting maxNumberOfResult to null.	At least 2 file ids should be returned	
3	Repeat the request file ids, this time with maxNumberOfResult set to one	A file id result with a single file id should be returned. The file id should be the oldest/first file id in the full list.	

Test 3: maxTimeStampTest

Purpose
Test the pillar support for only retrieving file ids older that a given time. Note that this test assumes there is at least 2 file ids with different timestamps.

Step	Stimuli	Expected result	Result
1	Request default file ids for all files on the pillar	A list with at least 2 different timestamps (it is not the fault of the pillar if this fails, but the test needs this to be satisfied to make sense).	
2	Request file ids with MaxTimeStamp set to the timestamp of the newest file id	All file ids should be returned.	
3	Request file ids with MaxTimeStamp set to the timestamp of the oldest file id	Only file id with the timestamp equal to MaxTimeStamp are returned.	
4	Request file ids with MaxTimeStamp set to the timestamp of the oldest file id - 10 ms	No file ids are returned.	

Test 4: minTimeStampTest

Purpose
Test the pillar support for only retrieving file ids newer that a given time. Note that this test assumes there is at least 2 file ids with different timestamps.

Step	Stimuli	Expected result	Result
1	Request default file ids for all files on the pillar	A list with at least 2 different timestamps (it is not the fault of the pillar if this fails, but the test needs this to be satisfied to make sense).	
2	Request file ids with MinTimeStamp set to the timestamp of the oldest file id	All file ids should be returned.	
3	Request file ids with MinTimeStamp set to the timestamp of the newest file id	Only file id with the timestamp equal to MinTimeStamp are returned.	
4	Request file ids with MinTimeStamp set to the timestamp of the newest file id + 10 ms	No file ids are returned.	