



# ECF Enhancements Programme

## Write Back – Web Services

### Interface Specification – Claim Respond

Version: 3.6.3~~2~~

Issue Date: ~~13/02~~03/072018

#### **Copyright Information**

© Velonetic™ 2023

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical photocopying, recording, or otherwise, without the prior permission of Velonetic.

Note only signed hard copies and electronic masters of documents will be controlled. Any other copy may not be current.

#### **Trademark Information**

Company, product, or brand names mentioned in this document, may be the trademarks of their owners.

## Contents

1	Introduction	3
1.1	Background.....	3
1.2	Document Scope.....	3
1.3	Service Overview.....	3
1.4	Operations.....	4
2	Operation - Claim Respond	5
2.1	Request/Input message.....	5
2.2	Response/Output message.....	9
2.3	Operation behaviours.....	10
3	Non-functional Characteristics	16
3.1	Security.....	16
3.2	Web Service Availability.....	18
3.3	Service versioning.....	18
3.4	Performance and Maximum Load.....	18
3.5	Service Support and Maintenance.....	19
3.6	Invoking the Service.....	19
4	Acronyms	20
5	Document Control	21
6	Appendix A: Sample messages	24
7	Appendix B: Claim Respond Data dictionary	25
8	Appendix C: Service XML Resources	26
8.1	Claim Respond XSDs & Sample messages.....	26
9	Appendix D: Crossing the post scenarios	27
9.1	Scenario A: Carriers must ensure they respond to the latest transaction.....	27
9.2	Scenario B: Carrier response for 1 <sup>st</sup> notification while Broker updates transaction	27
9.3	Scenario C: Carrier response for 1 <sup>st</sup> notification while Broker updates transaction	27
10	Appendix E: Multiple Carrier Lines	31
11	Appendix F: VCS Triage Category Clarification	32

# 1 INTRODUCTION

## 1.1 Background

The ECF Enhancements Programme delivers a Write Back solution that will enable Carriers to manage the claim within their internal systems without the need to duplicate effort via CAS. This is achieved by the provision of services that will allow a Carrier's system to receive timely notifications of claim transaction events, claim information and documents and to similarly respond by message.

This document defines the interface that allows a Carrier's system to respond to a claim transaction from their internal systems.

## 1.2 Document Scope

This document covers the information adhered to by this service along with describing logical functional and exceptional behaviours before describing how the service can be called.

The document covers the data involved in calling this service's operations and the operation behaviours.

Any changes to the interface design will be advised to the market and software vendors through a robust change control process so that the findings and potential changes can be considered.

## 1.3 Service Overview

*Claim Respond* service provides a mechanism for carrier systems to respond to a claim from within their own system and to sync the response with the central services system.

The service request must be associated with a UCR, TR and Carrier Code, apart from other correlation keys mentioned in the data dictionary.

The service performs these validations on the receipt of a request

- ❖ XML Schema conformity
- ❖ Verifying authenticity of the caller - authentication
- ❖ Verifying if the carrier is registered for the service - authorisation

If the request is well formed, adhering to a message schema and if the carrier is registered to invoke the service, a synchronous acknowledgement will be sent to the carriers about the successful receipt of the request.

Xchanging will process the request asynchronously - if the processing was successful, a successful claim notify event will be generated to notify all the carriers (participants in the claim) about the response. If the processing has resulted in business validation errors, an error notify will be sent only to that carrier (as part of Claim Notify message).

Business validation rules will vary depending on:

- ❖ The Bureau the responding organisation belongs to
- ❖ The role of the organisation on the claim
- ❖ The specific user issuing the response
- ❖ The status of the claim

Refer to the functional specification (*ANA-FS-Write Back Project Part F*) to understand business validations of all the functional scenarios.

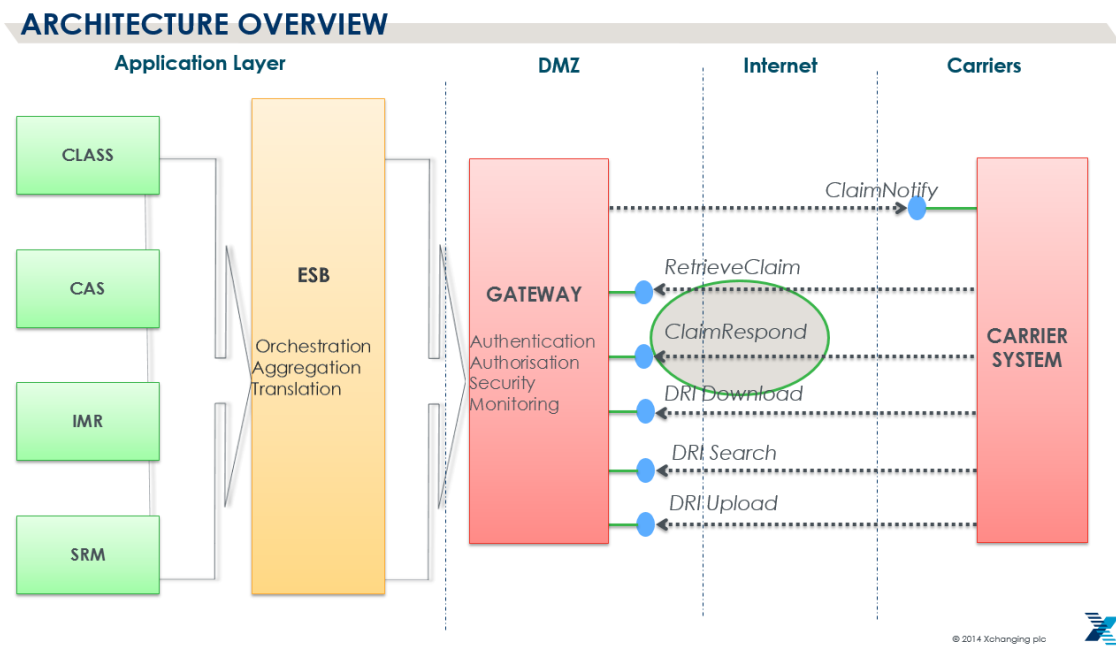


Figure 1: Operational Behaviour of Claim Respond Service

## 1.4 Operations

There is only one operation 'Claim Respond available within this service.

## 2 OPERATION - CLAIM RESPOND

This section describes the interface for the *Claim Respond* operation offered by Xchanging and its message details required to call the operation along with the data returned in the response message. It covers both the message headers and the message body.

This operation allows the carriers to respond to a claim from their internal systems.

### 2.1 Request/Input message

The request is contained in a SOAP envelope split between headers and the body payload data. The request message in this context is the message that contains information while the carrier responds to or updates a claim transaction.

#### 2.1.1 Message Header

Apart from standard SOAP headers as per WS\* standard, following additional information will be sent in the SOAP header of the message.

- Sender and Receiver party information will be passed in the request header.

```
<Sender>
  <Party><Id> {Carrier's party ID - Stamp code}</Id></Party>
</Sender>

<Receiver>
  <Party><Id> {Xchanging's party ID - DUNS code} </Id></Party>
</Receiver>
```

The Party IDs of the sender & receiver will be expected to in this format

#	Format
LIRMA	urn:lirma:XNNNN
Lloyds	urn:lloyds:NNNN
Xchanging	urn:duns:236196817
ILU	urn:ilu:NNNNNN

- <MsgTypeCd> will be added to the header to represent the type of the message. Value of the field in this service will be '**ClaimRespond**'
- <Timestamp> will have message creation date and time

- RequestUUID will be added to the header for auditing purposes at gateway level. The value of RequestUUID tag would be the same as the value of UUID tag in the message body
- <ResponseUUID> is an optional tag - this will be populated by Xchanging as part of synchronous response to the request. The value should be the same as the value of ReferredMessage/UUID tag in the response message body
- Carrier will be sending encrypted hash value of the signature in <wsse:security> tag to allow Xchanging to identify and authenticate the user based on the Carrier's public key. Please see the Security Section of this document to get more information on Service authentication.

Refer to the [SOAP header XSD](#) for structure of the header fields.

### **2.1.2 Message Body**

Apart from the mandatory fields that are required to be passed in the request message, it should contain information that has been updated on the claim.

#### **Only delta information needs to be sent in the request message.**

The request message should also contain the actual user (carrier system user) who is responding to the claim. This user needs to be on-boarded and registered in the central services system with appropriate privileges. Carrier should also be registered for *Claim Notify* service in order to receive notification of the claim responses that result in claim updates and or error messages when business validation fails. The CLASS User ID is passed in the following element

*ClaimResponseRq/Insurer/Contact/ClassUserId*

Account Code is also passed in the request message to ensure proper validation is triggered within the central services system.

Only <Insurer> fields should be populated in the request and <Reinsurer> field should be ignored in all scenarios.

**Referred UUID** field is added to interface as part of CR11. **(NOTE - CR11 is part of November release, only DD changes are included. Sample message and XSD do not include this field for September release)**

Referred UUID is the UUID of the latest Claim Event Notify or Retrieve Claim Data message sent to carrier system.

The Referred UUID passed in Response request message will be validated against the UUID present in XSB-ECF Database.

It is to ensure the Claim Response is being made against a specific Claim Event Notify or Retrieve Claim Data message.

Note - **Referred UUID** field is optional during the transitional phase, defined as 3 months from go-live, after which it will become mandatory

**AsOfDate (Time Stamp)** field is added to interface as part of CR11. **(NOTE - CR11 is part of November release, only DD changes are included. Sample message and XSD do not include this field for September release)**

AsOfDate is the latest Class record update date and time to the millisecond.

AsOfDate field should be populated in Response request message with the latest AsOfDate passed to carrier system from latest Notify or Retrieve response message.

This will ensure proper validation is triggered within central services system and only Response request on latest information available in carrier systems is successfully applied.

Note - **AsOfDate** field is optional during the transitional phase, defined as 3 months from go-live, after which it will become mandatory

If a specific field is updated as part of the response, only those fields need to be sent (as per the data structure defined in the message schema). The service internally will fetch the information from central services system and ensures that the given user (passed in the request message) has permissions to update certain fields. If user has no permission to edit certain fields that the Claim Respond request message contains, appropriate error is returned back to the carrier as a notify message.

If a specific field is updated to blank field/empty, carrier system should pass in 'null' (literal string, case sensitive) as part of the field value.

### ***Respond Comments***

If any new comments are added as part of the response, only those comments should be passed in the request message. There is no provision to update or delete already submitted comments. Previous responded comments should not be sent as part of a new claim respond request message.

Central services system supports 15 lines of comments, each line supporting upto 70 characters. As part of Claim Notify message, line number and comment details are sent to the carriers. When carriers are responding to a claim, they should take appropriate action (as part of their business process) when the number of comments are nearing 15. , Central services system will truncate the comment to 70 characters, should it be more than 70 characters.

### ***Agreement Parties***

- ❖ <AgreementPartyModifiedIndicator> within each insurer tag will define if an agreement party is added or removed or modified. Refer to the data dictionary for enumerated values of the indicator.
  - If 'add' is selected, the role of the carrier will change from Follower to Agreement Party. The carrier can optionally amend the contact details of an Agreement Party they are adding by populating the <Contact> aggregate at the same time.
  - If 'mod' is selected then only editable fields should be passed in the request (apart from other mandatory fields for correlation). If contact details are passed in the message, contact details of the specified carrier will be updated.
  - If 'rem' is selected the role of the specified carrier will change from Agreement Party to follower.
- ❖ If the response has no modification to the agreement parties, agreement party tags will not be populated

Please refer to the data dictionary to understand which fields are applicable for response for specific Bureau.

### ***Claim Entry Financial Amounts***

- ❖ <ClaimEntryAmounts> structure repeats for each currency. Central services system supports maximum of three currencies to settle for a given transaction.

A new <ClaimEntryAmounts> can be added in a new currency - in which case, carrier system should pass all the necessary fields that are required to attach the record to the given claim transaction. If the record with that currency does not exist, the system will create a new < ClaimEntryAmounts> with the new currency. If the record with that currency already exists, system will update the existing <ClaimEntryAmounts> record with the



editable values (e.g. VAT, imported services amount, imported services narrative etc.) passed in the request message. Note that, entries of <ClaimEntryAmounts> should be passed along with all mandatory correlation keys that are required for central services system to identify the record within the claim transaction.

To elaborate, A LIRMA Lead can supply a Lead Reserve in a currency not supplied by the broker up to the limit of three currencies.

If the LIRMA lead is adding a new reserve, they should specify the new currency and populate the <wb:LeadReserveAmount>. Other amount related fields could also be supplied as long as they are editable.

It won't be possible to remove already existing < ClaimEntryAmounts> record of a given currency.

### ***Bulk Transactions***

- ❖ <BulkList> - It will not be possible to add/remove from the bulk transactions, but for component claims within that bulk list carriers can amend certain fields (VAT, Imported Services Amount, Imported Services Narrative etc.). Apart from correlation keys (Component Claim Ref/TR and currency), only the amended fields, within the respective component claim, needs to be passed in the request.

Refer data dictionary defined in [Appendix-B](#).

The data dictionary that defines

- the set of fields and their metadata
- business description of the fields along with certain validation rules

Refer XML Schema defined in [Appendix-C](#) to get all the data definition of the request and response message structure of Claim Respond service operation.

## **2.2 Response/Output message**

If the request message is validated successfully and carrier is registered for the service, a synchronous acknowledgement will be sent to the caller/carrier system.

Subsequently, the request message is processed asynchronously and claim responses are applied to CLASS.

If any business validation errors occur, service will return all (maximum 10) errors in a *Claim Notify* message to the carrier system with *MsgTypeCd* SOAP header as '**RespondErrorNotify**' along with the correlation key (<wb:ClaimRespondUUIId>) of the original claim respond request. Please refer to Claim Notify data dictionary for the metadata of this field.

On a success scenario where *Claim Respond* request is successfully processed, there will not be any correlation between the subsequent claim movement notify message and the original *Claim Respond* request.

### 2.2.1 Message headers

Xchanging system will respond with the UUID of the synchronous response along with that of the request to allow any correlation between request and response at both Xchanging's and carrier systems' gateway.

Refer to the [SOAP header XSD](#) for structure of the header fields.

### 2.2.2 Message body

The synchronous response payload should also contain acknowledgement indicator and statuses.

```
<Response>
  <AcknowledgementLevelIndicator>-</AcknowledgementLevelIndicator>
  <AcknowledgementStatus>-</AcknowledgementStatus>
  <ResponseDescription>-</ResponseDescription>
</Response>
```

Acknowledgement Indicator values are = ***translation\_validation*** or ***application\_validation***.

Acknowledgement Status values are ***acknowledged*** or ***rejected***.

`<ReferredMessage><UUIId></UUIId></ReferredMessage>` defines the Unique Universal Identifier (UUID) created for the request message being responded to.

The web service response is a synchronous reply and hence does not need to be signed.

## 2.3 Operation behaviours

These operational behaviours are explained in the following sequence diagram

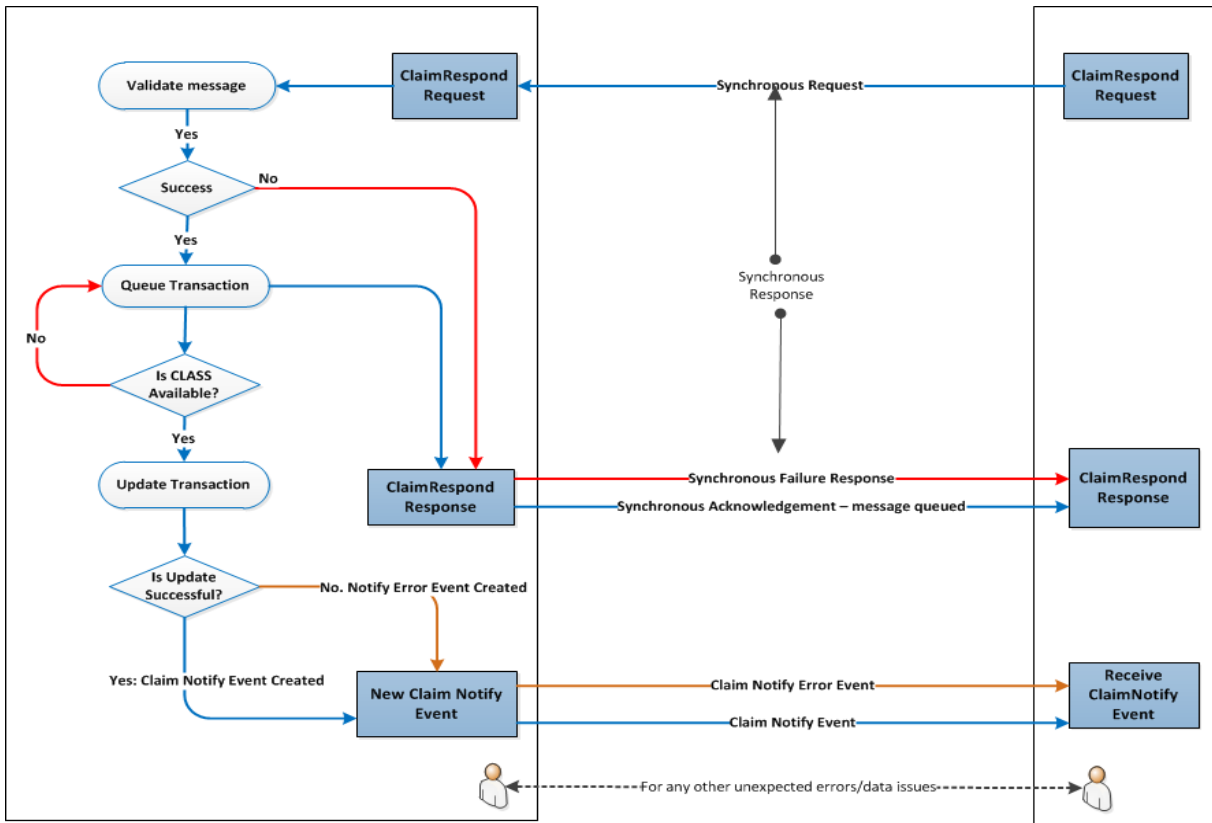


Figure 2: Operational Behaviour of ClaimRespondService

The Claim Respond operation will exhibit the following functional behaviours.

### 2.3.1 WB Claim Respond - carrier not registered for the service

WB Claim Respond - carrier not registered for the service	
Pre-condition	User is not registered for invoking Claim Respond service
Post-condition	<ul style="list-style-type: none"> <li>• <i>&lt;AcknowledgementLevelIndicator&gt;</i> will be set to <b>'application_validation'</b></li> <li>• <i>&lt;AcknowledgementLevelStatus&gt;</i> will be set to <b>"rejected"</b></li> <li>• <i>&lt;ResponseDescription&gt;</i> will be set to <b>"Carrier not registered to invoke the service"</b></li> </ul>

### 2.3.2 Invalid Request Message Format

If the message is in an invalid format then a service response 'Fail' will be returned. The originator is responsible for correcting and re-sending the message if and as appropriate.

Behaviour 4 - Invalid Request Message Format	
Pre-condition	An invalid request message is sent to the service and the request has failed validation within central systems
Post-condition	<ul style="list-style-type: none"> <li>• <i>&lt;AcknowledgementLevelIndicator&gt;</i> will be set to <b>'translation_validation'</b></li> <li>• <i>&lt;AcknowledgementLevelStatus&gt;</i> will be set to <b>"rejected"</b></li> <li>• <i>&lt;ResponseDescription&gt;</i> will be set to <b>"Fail - Request Invalid"</b></li> <li>• The response message will not include claim business or claim document information.</li> </ul>

### 2.3.3 WB Claim Respond - successfully received and processed

Pre-condition	A valid request message containing the all mandatory fields and the carrier's response/comments are sent to the service either within ECF core hours or outside of ECF core hours.
Post-condition	<ul style="list-style-type: none"> <li>• Synchronous acknowledgement is sent with               <ul style="list-style-type: none"> <li>○ <i>&lt;AcknowledgementLevelIndicator&gt;</i> will be set to <b>'translation_validation'</b></li> <li>○ <i>&lt;AcknowledgementLevelStatus&gt;</i> will be set to <b>"acknowledged"</b></li> <li>○ <i>&lt;ResponseDescription&gt;</i> will be set to <b>"Request has been queued"</b></li> </ul> </li> <li>• Asynchronously, central services system will be updated with the carrier's responses.</li> <li>• <i>Claim Notify</i> Event will be generated to inform all the participants of that Bureau about the response.</li> </ul>

### 2.3.4 WB Claim Respond - Successfully received but Process failure

WB Claim Respond - Successfully received but Process failure	
Pre-condition	A valid request message containing the all mandatory fields and the carrier's response/comments are sent to the service either within ECF core hours or outside of ECF core hours.
Post-condition	<ul style="list-style-type: none"> <li>• Synchronous acknowledgement is sent with               <ul style="list-style-type: none"> <li>○ <i>&lt;AcknowledgementLevelIndicator&gt;</i> will be set to <b>'translation_validation'</b>.</li> <li>○ <i>&lt;AcknowledgementLevelStatus&gt;</i> will be set to <b>"acknowledged"</b></li> <li>○ <i>&lt;ResponseDescription&gt;</i> will be set to <b>"Request has been queued"</b></li> </ul> </li> <li>• If asynchronous processing of the response is not successful because of business validation failures, error will be sent <u>only to the carrier who responded</u> (not all participants) as <i>Claim Notify</i> via <i>&lt;wb:ErrorsAndWarnings&gt;</i> tag. SOAP header of the message will have <i>MsgTypeCd</i> value as <i>'RespondErrorNotify'</i> <ul style="list-style-type: none"> <li>○ Refer to the <i>Claim Notify</i> Interface Spec for further details on data elements of the service.</li> </ul> </li> <li>• Any warnings generated by central services system while updating the claim transaction will be ignored by central services system and not communicated in the claim notification.</li> </ul>

### 2.3.5 WB respond requests from carrier while another claim update is being applied

WB respond requests from carrier while another claim update is being applied	
Pre-condition	A valid respond request is submitted by carrier for the same UMR, UCR and TR while the other carrier or broker update is being applied on the same UCR and TR
Post-condition	<ul style="list-style-type: none"> <li>• Synchronous acknowledgement is sent with               <ul style="list-style-type: none"> <li>○ <i>&lt;AcknowledgementLevelIndicator&gt;</i> will be set to <b>'translation_validation'</b></li> <li>○ <i>&lt;AcknowledgementLevelStatus&gt;</i> will be set to</li> </ul> </li> </ul>

	<p style="text-align: center;"><b>"acknowledged"</b></p> <ul style="list-style-type: none"> <li>○ <i>&lt;ResponseDescription&gt;</i> will be set to <b>"Request has been queued"</b></li> <li>• Asynchronously, system will try to update the claim transaction into CLASS system</li> <li>• If update on the Claim fails the validation because of the record being locked by another update on the same TR, the service retries configurable number of times to process the request.</li> <li>• If the processing of the response is successful, Claim Notify event will be generated.</li> <li>• If processing the response is not successful even after retry attempts, error will be sent to the carrier as a <i>Claim Notify</i> message via <i>&lt;wb:ErrorsAndWarnings&gt;</i> tag.</li> </ul>
--	--

### 2.3.6 Authentication failure

Pre-condition	Carrier sends a request with invalid certificate details
Post-condition	SOAP fault message with authentication error or certificate related issue.

### 2.3.7 Other unexpected runtime errors

In this scenario, Xchanging system will send the response to the Claim Respond request and carrier systems encounter any runtime exceptions or errors; this will result in SOAP fault that will propagate to the client.

### 2.3.8 Crossing the post scenarios

If a broker re-releases a transaction then the carrier should not respond to the previous version, as it is now out-of-date. However if the carrier has already responded but that response is pending in the central systems, then that response could subsequently be applied to the new version of the transaction, which is incorrect.

To address this there is some validation when a Claim Response is processed. If there have been any subsequent Notify events generated since the time the response was sent by the carrier then the response is rejected using a Respond Error Notify message. Refer to the Claim Notify Interface Spec for further details on data elements of the message used while notifying errors during processing of a claim response.

The validation is currently based on timestamps, so relies on the server clocks at carriers being aligned with those at central services. This is a potential source of error although in practice should not occur if all parties are mindful of this dependency. There are also a few unlikely scenarios where the current approach will not detect an out-of-date response and process it incorrectly, although this should subsequently be detected and remedial action can be taken. The timestamps will be in UTC format and should be passed along with timezone offsets.

The medium-term solution is to replace the synchronised timestamp approach with another more robust mechanism. This is likely to be based on correlation identifiers and central services timestamps. The intention is to include this in the scope of a subsequent project phase. The same issues are likely to arise with ACORD messaging in Phase 2, so a common solution will be sought.

### **2.3.9 Multiple lines for a carrier on the contract market**

When there are multiple lines for a carrier on the contract market, only the first line gets activated for selection as agreement party on CAS. The lead is able to select one or more active carrier lines while choosing an agreement party on a response. For example if a carrier has a consortium line (line 2) and a non-consortium line (line 3), the line that appears first on the market i.e. the consortium line in this case, will be activated for selection on CAS.

For write back, this active/inactive line information is not relayed back to the carriers on the notification and hence they may not be able to make an informed decision while selecting the agreement party.

Please refer to [Multiple Carrier Lines](#) appendix for the options and preferred solution.

### **2.3.10 VCS Triage Category Clarification**

Refer to the clarification paper available under [Appendix F](#) for detailed examples on data populated in the Triage Category field for various scenarios.

## 3 NON-FUNCTIONAL CHARACTERISTICS

The *Claim Respond* service should comply with the following non-functional characteristics.

### 3.1 Security

The service will be exposed on a secure end point (transport level security). There is no message level security enforced.

Central services system recommends the following authentication and security strategy for the ECF2 write back services.

This is consistent with the ACORD security profiles, please refer to [ACORD Security Profiles V1.1.0](#) link (section 5) for more details.

#### Security Techniques

- SSL/TLS Server Authentication
- SSL/TLS Integrity
- SSL/TLS Encryption
- WSS Signature SOAP Envelope

#### 3.1.1 Identification

When carrier systems invoke the service, Xchanging will receive the sender and receiver details in the SOAP header (along with other headers specified [above](#)).

Xchanging's gateway will identify the caller by reading the SOAP headers.

#### 3.1.2 Authentication

The service authentication of all ECF write back services will be handled via Public Key Infrastructure (PKI) standards.<sup>1</sup>

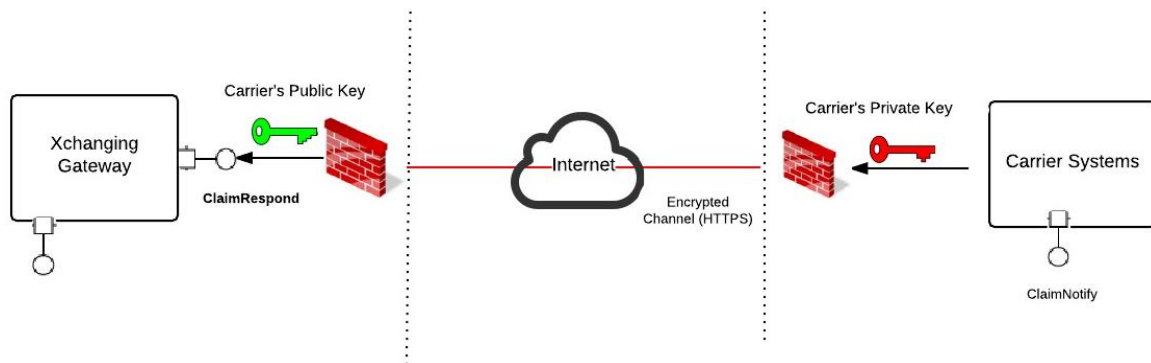
Carrier systems will send a respond request by signing the message with its own private key. The hash-value of the message is encrypted and passed in the SOAP header to the Xchanging system. Xchanging gateway, on the other end, will identify the sender (as defined above), decrypt the hash based on carrier's public key and verify the signature.

The exchange of certificate keys will be done as part of registration process.

---

<sup>1</sup> PKI approach is considered over other options such as OAuth, SAML, API Keys to ensure write back services are DRI consistent and aligns with Write back phase-2 strategy





One of the key advantages of this approach is that carrier system can reuse the same approach (and same set of keys) across other services including DRI services.

### 3.1.3 Authorisation

All carriers will be registered in Xchanging's 'Service Registration Management (SRM)' application as part of the service registration process.

If carriers are not registered for the Claim Respond service in SRM, the web service will send a failure response to the carrier (as part of synchronous acknowledgement) with suitable error code. See this [section](#) for further clarification.

There is another layer of user level authorisation enforced by central services when the transaction response updates are being processed.

- Party aggregate in the message is populated with the CLASS user ID of the user making the response

*XPATH: Party/Contact/ClassUserId*

- User is validated against CLASS system to verify the authorisation rules
- Carrier systems should also be sending CLASS account code in the request message
- Carrier systems should ensure necessary authorisation and access rights are set up as per their individual security policies to cover respective governance, audit and compliance procedures
  - E.g. read only versus respond access rights, managing authorisation limits individual conflict of interest etc.

The user level authorisation errors (or any business validation errors) and issues will be recognised while applying the transaction updates in the central services system. Any user level authorisation errors will be notified as part of asynchronous *Claim Notify* error response.

### 3.1.4 Encryption

No message level encryption is enforced<sup>2</sup>. But, transport level (SSL) encryption is enabled.

## 3.2 Web Service Availability

The web service will be 'technically' available beyond ECF core hours (7am - 7pm), but supported only during ECF core hours.

Any Claim Respond requests sent outside of core hours will be processed (and notification will be sent) when CLASS system is up and running.

## 3.3 Service versioning

Major version = 1

Minor version = 0

For an incompatible service change, it is assumed that the changes is major and change the major version number as a result. Old version: N.m => New version: (N+1).0

For backward-compatible service changes, we'll assume the change is minor and change only the minor version number as a result. Old version: N.m => New version: N.(m+1)

## 3.4 Performance and Maximum Load

### 3.4.1 Anticipated Volumes

#	Metrics
Maximum number of Claim Respond requests (as per current volumes)	20000 daily
Peak hour volumes (% age of daily volume)	15%
Maximum time for carriers to receive a synchronous acknowledgement	10 seconds
Number of out of core hours Claim Respond requests (assumption)	10% of daily volume
In normal scenarios <sup>3</sup> , time it takes for business response (Claim Notify) to be received when a claim response is processed successfully	3 minutes <sup>4</sup>

<sup>2</sup> This is in accordance with ACORD messaging on DRI services.

<sup>3</sup> Scenarios on average load that excludes cases such as respond lock conditions, retries, CLASS being unavailable etc.

<sup>4</sup> This should not be considered as the SLA. In technical terms, this is the average time it takes to send business response to a Claim Respond request under normal conditions.

---

Average size of the request message	80 KB
Average size of the acknowledgment message	5 KB

---

Refer to *Claim Notify Interface specification* to understand the non-functional and operational behaviours of the *Claim Notify* service

### **3.5 Service Support and Maintenance**

Xchanging's write back service support and incident management process will be published to allow carrier support teams to collaborate with Xchanging's service desk to handle any issues.

### **3.6 Invoking the Service**

The test, MAT and production URLs of the Claim Respond Web service end point will be shared with the carriers at the point of registration.

## 4 ACRONYMS

Acronym	Definition
CLASS	Claims Loss Advice and Settlement System
HTTP	Hyper Text Transfer Protocol
HTTPS	HTTP over SSL
IP	Internet Protocol
SOAP	Simple Object Access Protocol - an industry standard protocol for invoking remote web services, usually used with HTTP
SSL	Secure Sockets Layer - a protocol for secure interaction over IP
WB	Write Back
WSDL	Web Services Definition Language

## 5 DOCUMENT CONTROL

Version	Date	Author	Brief description
0.1	20/01/15	Vikas Acharya, Nitin Jain	Data dictionary, service behaviour, NFRs, SOAP Headers, XSD resources.
0.2	22/01/15	Vikas Acharya	Internally reviewed
0.3	23/01/15	Vikas Acharya	Clarifying Document Scope section 1.2
1.0	03/02/2015	Vikas Acharya	<p>Comments and clarifications from market workshop on 28/1.</p> <p>More clarity on interface security, data definitions and XSDs</p> <p>Internally reviewed comments.</p> <p>Added sample messages. Re-versioned to 1.0.</p> <p>Issued for sign-off.</p>
1.0	13/02/2015	Vikas Acharya, Nitin Jain	<p>Few data definition optimization based on feedback from the market. Updated XML resources.</p> <p>Re-issued for Sign-off.</p>
1.0	20/02/2015	Vikas Acharya	Signed-off.
1.1	25/03/2015	Vikas Acharya	<p>QueryReason is added for AgreementParty Responses apart from Lloyds lead responses</p> <p>Updated comments for ParticipantFunction to describe significance of coded values</p> <p>Split out YES and CES responses on the response code tab</p> <p>Added new error code D001 in ILU_Warnings_Errors tab, LIRMA Co</p>

			Warnings_Errors, Lloyd's Warnings_Errors, LIRMA Lead Warnings_Errors tabs
			Removed the security tag from the Xchanging's header as it will be inserted into the SOAP header in the same level.
1.2	22/04/2015	Vikas Acharya, Nitin Jain	Clarifications on "Crossing the post" scenarios. Updated DD and XSD.
1.3	27/05/2015	Vikas Acharya	Few clarifications in DD. No impacts or changes to XSD.
2.0	18/11/2015	Nitin Jain	Updated DD,  Attached the latest version of published clarification papers
2.1	04/02/2016	Sue Brett	Updated DD, Appendix B  Added 2 new error codes under the ILU, LIRMA Lead, LIRMA Co's and Lloyd's Warnings_Errors tabs for conflict of interest.
2.2	13/04/2016	Adnan Nasar-Ullah	Changes to Claim Scheme Values in DD  Line 55 - '2006 and Prior' changed to '2006 and prior' i.e. small p
2.3	19/04/2016	Adnan Nasar-Ullah	Changes to DD in appendix B to include CR11 changes
2.4	20/03/2016	Sudhir Kumar	Updated for CR11. Two new fields are added to response request message.  Reffered UUID  AsOfDate
2.5	25/04/2016	Adnan Nasar-Ullah	Changed AsOfDate to milliseconds in DD
2.6	27/04/2016	Sudhir Kumar	Attached latest XSD and Sample message.

2.7	20/06/2016	Sudhir Kumar	Updated comments for fields added for CR11 as that is part of November 2016 release.  DD include changes for both CR11 and CR19.  XSD and Sample message conforms to CR19 changes.
2.8	27/06/2016	Sudhir Kumar	Attached latest XSD and Sample message.
2.9	05/07/2016	Paul Sterrett	Updated DD
3.0	14/07/2016	Paul Sterrett	Updated DD and Sample Message (Respond Request)
3.1	28/07/2016	Paul Sterrett	Updated DD
3.2	15/09/2016	Paul Sterrett	Updated DD and Sample Messages
3.3	22/09/2016	Paul Sterrett	Refreshed version to coincide with Retrieve and Notify updates.
3.4	07/11/2016	Paul Sterrett	Updated as part of Documentation corrections from Onboarding
3.5	20/04/2017	Sudhir Kumar	Updated the version history to keep all three-interface spec at same version. No Change in Document
3.6	24/08/2017	George Cruickshanks	Updated the version history to keep all three-interface spec at same version. No Change in Document
3.6.1	26/09/2017	George Cruickshanks	Updated the version history to keep all three-interface spec at same version. No Change in Document
3.6.2	13/02/2018	Dave Smith	Updated the version history to keep all three-interface spec at same version. No Change in Document
<a href="#">3.6.3</a>	<a href="#">03/07/2018</a>	<a href="#">Dave Smith</a>	<a href="#">Updated DD</a>

## 6 APPENDIX A: SAMPLE MESSAGES

Please see the [following section](#) for the sample messages for both request and response. Note that these sample messages does not contain a real claim response data, but it should give enough information around the structure of the message.



## 7 APPENDIX B: CLAIM RESPOND DATA DICTIONARY

Refer to Respond Data Dictionary circulated with this document.

Please note that the "Editable" column has marked some fields as 'MR', indicating that they are "Mandatory and Read-only". These fields form the core set of correlation keys for Xchanging systems to fetch the latest claim transaction information before processing the carrier response.



ClaimRespond Data  
Dictionary v3 6 1.xls) **[Deleted]**



ClaimRespond Data  
Dictionary v3 6 3.xls)

## 8 APPENDIX C: SERVICE XML RESOURCES

### 8.1 Claim Respond XSDs & Sample messages

Refer to XSDs and sample messages circulated with this document.



Sample XMLs (2).zip



Respond XSDs V3.6.2.zip

[Deleted]

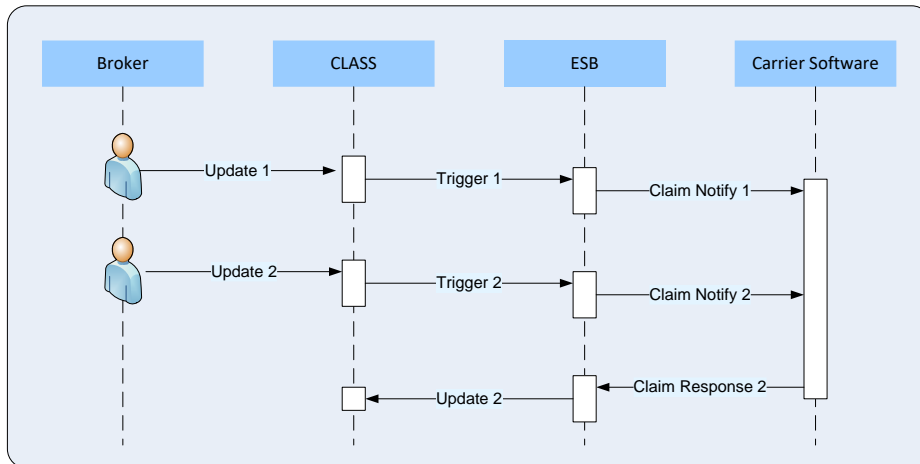


Respond XSDs V3.6.3.zip

## 9 APPENDIX D: CROSSING THE POST SCENARIOS

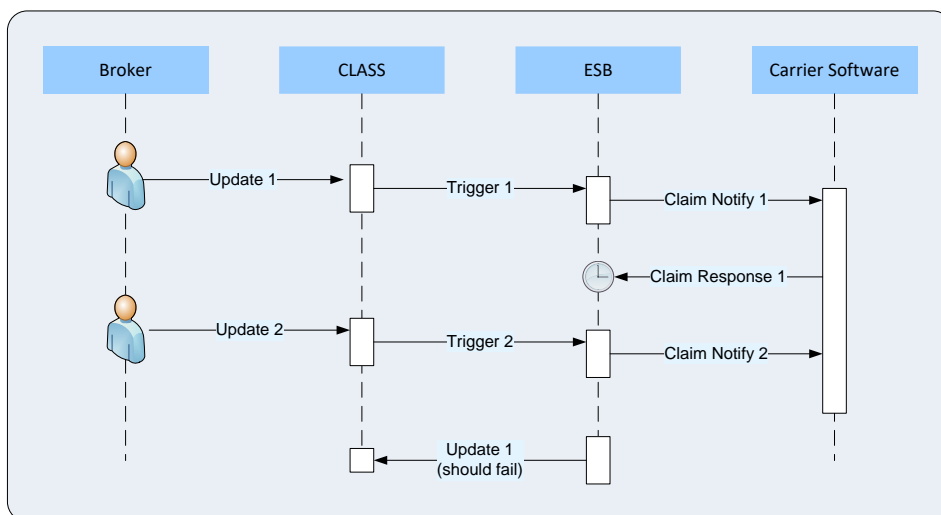
### 9.1 Scenario A: Carriers must ensure they respond to the latest transaction

Carriers have been communicated to ensure they respond to the latest version of the notification.



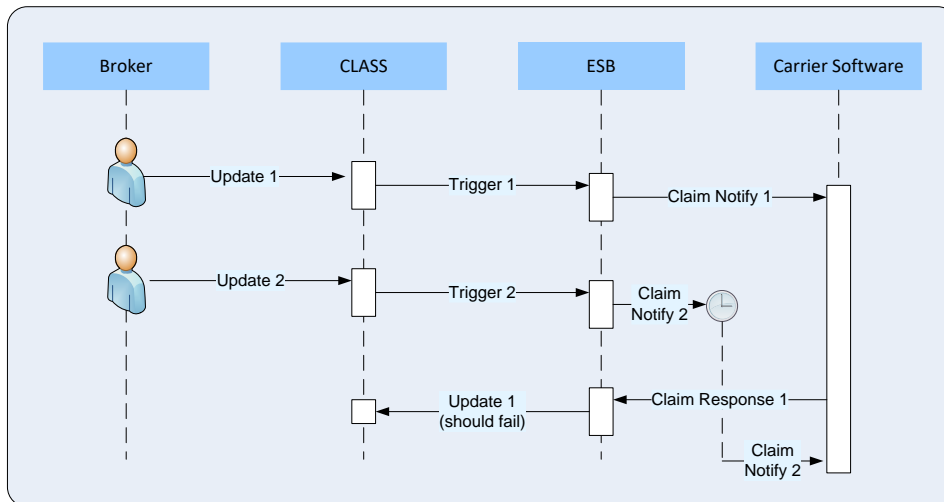
### 9.2 Scenario B: Carrier response for 1<sup>st</sup> notification while Broker updates transaction

The broker makes an update which is notified to the carrier; carrier responds and this is queued for processing; broker makes a 2<sup>nd</sup> update which is notified; the first carrier response should subsequently fail



### 9.3 Scenario C: Carrier response for 1<sup>st</sup> notification while Broker updates transaction

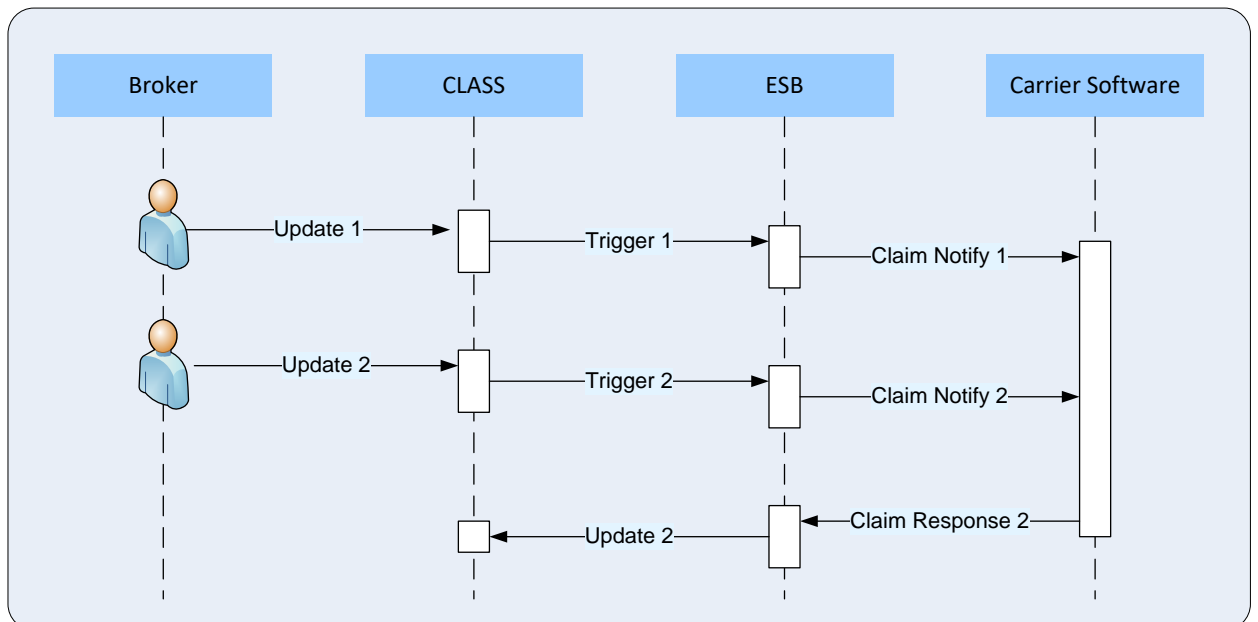
The broker makes an update which is notified to the carrier; broker makes a 2<sup>nd</sup> update and the notification is queued in the central services or carrier gateway; the carrier responds to the 1<sup>st</sup> notification (which should fail)



## 9.4 CR11: Extension to Crossing the post to handle Scenarios which are not handled in Phase-1

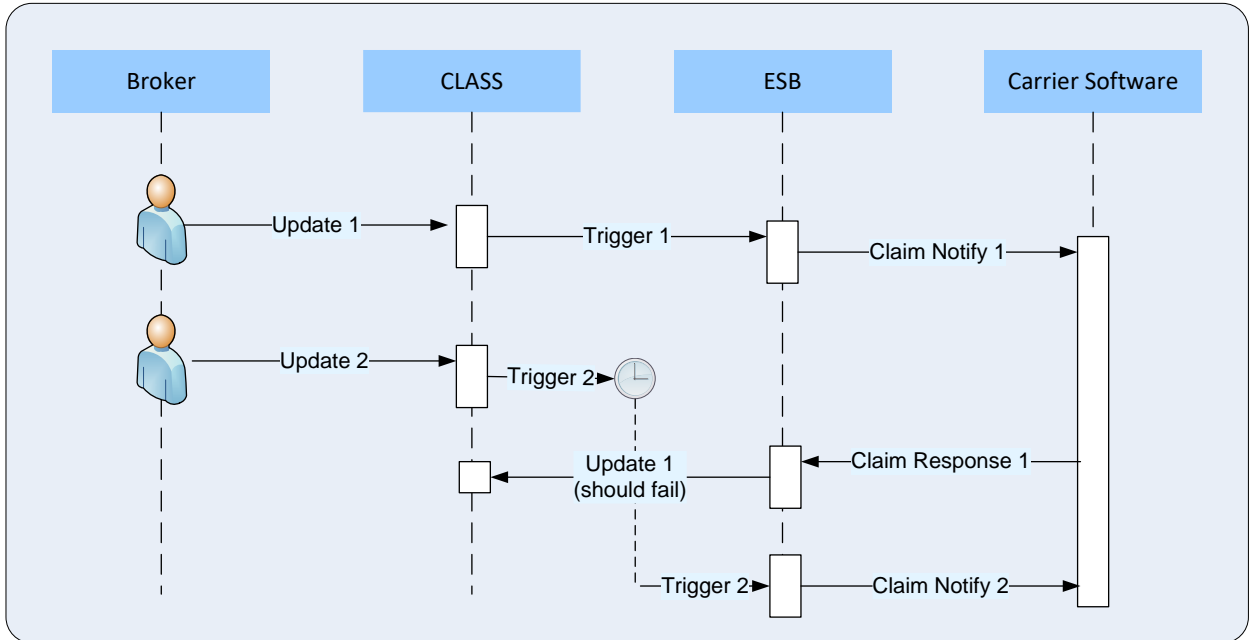
### 9.4.1 Two Updates Then One Response

Broker makes an update which is notified to the Carrier. The Broker then makes a 2nd update which is notified, the Carrier responds. It is the Carrier's responsibility to ensure that they respond only to the latest notification that they have received.



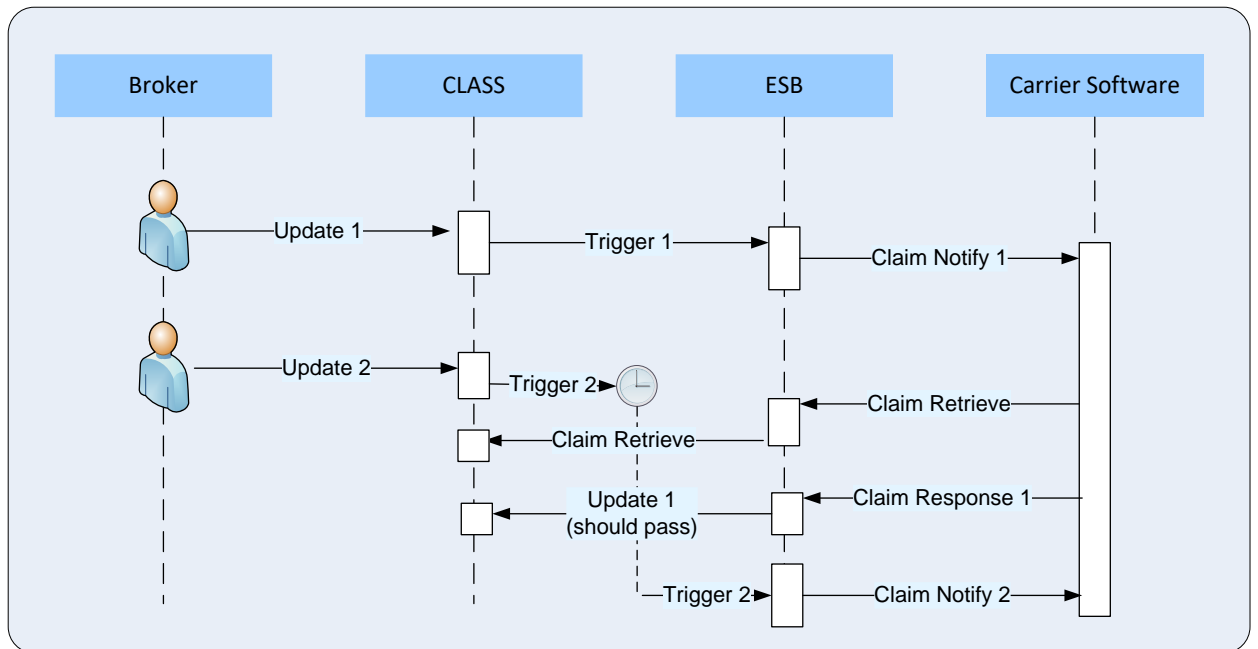
### 9.4.2 Queued Trigger

Broker makes an update which is notified to the carrier, Broker makes a 2nd update and the notification is queued before reaching the ESB. Carrier responds to the 1st notification (which should fail)



### 9.4.3 Queued Trigger but Retrieved Successfully

Broker makes an update which is notified to the carrier, Broker makes a 2nd update and the notification is queued before reaching the ESB. Carrier retrieves the claim transaction using the ClaimRetrieve service to get most up-to-date claim transaction details. Carrier then responds to the claim transaction using ClaimRespond service, this should pass.



#### 9.4.4 Solution overview

**Assumption:** Carriers will be using either Claim Notify or Claim Retrieve service before responding to a claim transaction using the Claim Respond service

The notification message UUID should be sent in the Respond messages as a referred UUID. This would enable us to verify that the Carrier is responding to the latest Notify message successfully sent to them.

Compare AsOfDate available in the mainframe with the AsOfdate available in ClaimRespond request message. If these two dates don't match then reject the transaction.

#### **During Transition Phase - For the interim period (i.e. 3 months):**

- If AsOfDate is provided, then it must be correctly populated and validated
- If AsOfDate is not provided, then:
  - o If the field is provided in the message but the data is not available in the source system, a dummy date should be entered (i.e. 1 Jan 1900, 00:00:00:0001 hours). The AsOfDate validation will then be ignored
  - o If the field is not provided in the message then the AsOfDate validation will be ignored

After the interim period the AsOfDate and Referred UUID fields will be mandatory and the AsOfDate validation will be applied

## 10 APPENDIX E: MULTIPLE CARRIER LINES



Multiple carrier  
lines.docx



## **11 APPENDIX F: VCS TRIAGE CATEGORY CLARIFICATION**



VCS Triage  
Category Clarificatic