



Write Back Functional Specification

Part B: The Claim Event Notify Service

Version: 2.0

Issue Date: 26 November 2015

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.

SECTION 4: CONTENTS

| | | |
|------------|--|-----------|
| 1 | Claim Event Notify Message Interactions..... | 4 |
| 1.1 | Service Overview | 4 |
| 1.2 | Message Data Definition..... | 5 |
| 1.2.1 | Request Message | 7 |
| 1.2.2 | Response message | 7 |
| 1.3 | Operational Behaviour and Messages | 8 |
| 1.3.1 | Behaviour 1 - Request Successful | 9 |
| 1.3.2 | Behaviour 2- Request Failed - Invalid Request Message Format | 10 |
| 1.3.3 | Behaviour 3 - Request Failed - Duplicate Message | 10 |
| 1.3.4 | Behaviour 4 - Request Failed - Message Receiver Not the Intended Recipient | 11 |
| 1.4 | Claim Event Notify Exception Handling..... | 11 |
| 1.4.1 | Technical Errors | 11 |
| 1.4.2 | Timeout & Retry | 11 |
| 1.4.3 | Message Persistence..... | 11 |
| 1.5 | Claim Event Notify Non-Functional Characteristics | 12 |
| 1.5.1 | Integration Security | 12 |
| 1.5.2 | Availability | 12 |
| 1.5.3 | Service Response Times | 12 |
| 1.5.4 | Performance and Maximum Load | 13 |
| 1.5.5 | Invoking the Service | 13 |
| 1.6 | Conventions and Considerations | 14 |
| 1.6.1 | Identification of Parties | 14 |
| 1.6.2 | Claim Event Notify and LIRMA Message Sequencing | 14 |
| 1.6.3 | Fixed-Width Repeatable Data Elements | 14 |
| 1.6.4 | Warning Messages..... | 15 |
| 1.6.5 | Market Information..... | 15 |
| 1.6.6 | Claim Event Notify when Claim Response Fails | 15 |

| | | |
|----------|--|-----------|
| 1.6.7 | Claim Event Notify when a Claim Is Deleted | 17 |
| 2 | Document Control | 18 |
| 2.1 | Document Information | 18 |
| 2.2 | Revision History | 18 |
| 2.3 | PARCI | 18 |

APPENDICES

| | |
|--|-------------------------------------|
| Appendix 4A: Claim Event Notify Triggers | 22 |
| Claim Notification Event Triggers | 22 |
| CWT Action Details on a VCS Claim | 24 |
| Appendix 4B: Code Sets..... | 29 |
| WB-CODE-001: Warning Messages | 29 |
| WB-CODE-002: Response Codes | 29 |
| WB-CODE-003: Outstanding Amount Qualifier | 29 |
| WB-CODE-004: Business Class | 29 |
| WB-CODE-005: Transaction Status Value | 29 |
| WB-CODE-006: Claim Category..... | 30 |
| WB-CODE-007: Claim Sub Category..... | 30 |
| WB-CODE-008: Date of Loss Qualifier | 30 |
| Appendix 4C: Claim Event Notify XML Message Example | Error! Bookmark not defined. |
| Request Message | Error! Bookmark not defined. |
| Response Message..... | Error! Bookmark not defined. |
| Appendix 4D: Clarifications and SOAPS | 31 |
| Delete Event Notify Message - SOAP..... | 31 |

1 CLAIM EVENT NOTIFY MESSAGE INTERACTIONS

This section describes the data sent to carriers and the interface which will be offered by carriers for a Claim Event Notify service to be processed. This covers the business data and document information sent from central systems that is required to be processed by Carrier systems, together with the data returned in the response message from the Carrier.

1.1 Service Overview

Carrier systems will expose a web service that will allow the central system to notify a claim event to them. This notification will consist of the following information:

- The information that would normally be provided with the current CWT
- The current claim information that would normally be available to the carrier from CAS
- A current list of policy and claim documents for a claim at the point at which the service is invoked that would normally be available to the carrier from the IMR. This will be in a similar format to that obtained from an ACORD DRI Search.

The information will be provided in a structured XML format that resembles ACORD as far as practicable. This service will also contain appropriate failure, retry and exception handling to mitigate any risk of non-delivery.

The response message will contain an acknowledgment of successful receipt of the message or a rejection with the reason.

Please note that the technical content of the message e.g. SOAP or REST protocols, unique message identifiers etc. will be defined in detailed interface design which will likely involve further collaboration with carriers and their software providers.

Carrier Organisations using Outbound DRI

For Carrier organisations who already have an Outbound DRI implementation, Write Back should operate as a complementary service. Carrier systems will be able to compare the list of documents provided in this web service with documentation already notified via the Outbound DRI channel using the unique identifier for the document (either Document ID or a combination of Document Reference and Document Version). If any documents are notified which are not currently held on the Carrier's own repository - for example, where the Carrier role on a claim has changed and previously loaded documents have not been notified via Outbound DRI or where documents have been loaded to the IMR but no CWT event has been triggered - the Carrier system can invoke the Document Download service in order to obtain copies of the documents. This will ensure that Carriers have a full set of claim documents when adjusting a claim.

1.2 Message Data Definition

Conventions

The following conventions are valid for the Claim Event Notify request message.

| Area | Statement |
|-----------------|--|
| Message | A Claim Event Notify Request will be at the same level as existing CWT notifications. |
| Message | A Claim Event Notify Request always has only one sender and one recipient. However, the Request message will contain information about other markets on the risk. |
| Data Completion | Usage for each data element is provided using: <ul style="list-style-type: none"> • (M)andatory - The data element will always be supplied in the message • (C)onditional - The provision of the data element is dependent on the provision and content of other data elements in the message. • (O)ptional - The data element will be supplied if the related data field in the source system is populated. • (N)ot used - The data element will not be supplied. |
| Code Lists | Code lists are used as indicated in the Code Lists section. <ul style="list-style-type: none"> – See the ACORD code list (www.ACORD.org) for further information where ACORD codes are used – ISO 4217 currency codes will be applied to all currencies – See Appendix 4B: for specific London Market code lists. |

The following conventions are valid for the Claim Event Notify response message

| Area | Statement |
|---------------|--|
| Orchestration | One Claim Event Notify response message must be sent per successfully received Claim Event Notify request message. |
| Orchestration | Once a Claim Event Notify request has been responded to there should be no subsequent acknowledgment responses for that message. |
| Code Lists | <p>Code lists are used as indicated in the Code Lists section.</p> <ul style="list-style-type: none"> – See the ACORD code list (www.ACORD.org) for further information where ACORD codes are used – ISO 4217 currency codes will be applied to all currencies <p>See Appendix 5A: for specific London Market code lists.</p> |

Data Dictionary Structure

The table below provides details on each of the column headers used in the data dictionary.

| Column | Description |
|-----------------|--|
| Field Name | The name assigned to the data element |
| Xpath | The xpath for the data elements |
| Definition | Business description of the field |
| Comments | Additional commentary regarding usage |
| Data Type | The type of data |
| Business Format | Indicates the format of the field |
| Code set | Provides, where applicable, the relevant code list |
| Code set values | Valid values |

| Column | Description |
|----------------|--|
| Multiplicity | Indicates field repeatability |
| ILU | Message completion rules where the receiving /responding carrier is participating as an ILU company carrier on the claim. |
| LIRMA Lead | Message completion rules where the receiving /responding carrier is participating as the LIRMA lead company on the claim. |
| LIRMA Follower | Message completion rules where the receiving /responding carrier is participating as the LIRMA following company on the claim. |
| Lloyd's | Message completion rules where the receiving /responding carrier is participating as a Lloyd's syndicate on the claim. |

1.2.1 Request Message

Please refer to the Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for the Request Message structure, multiplicity and business usage.

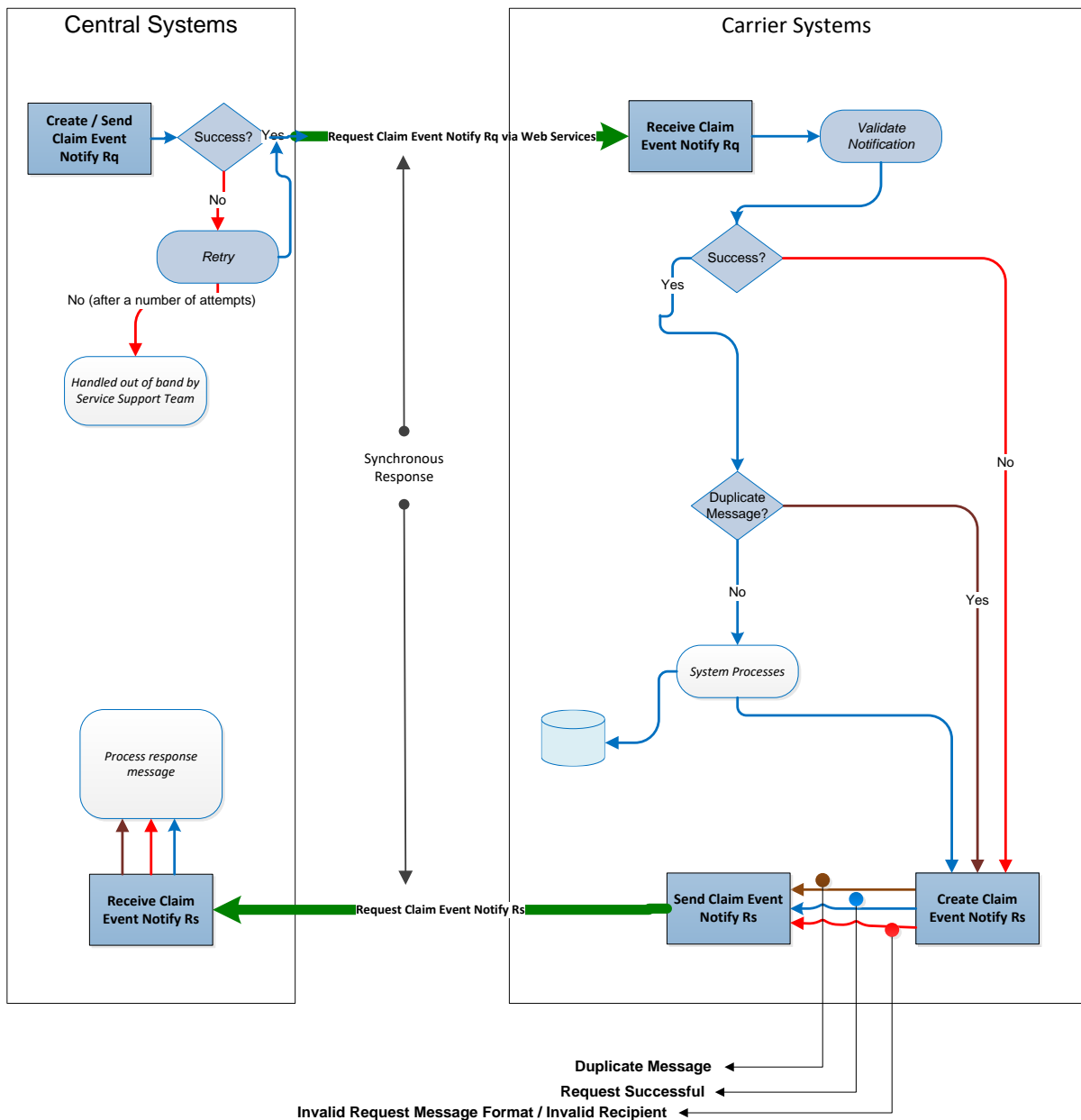
1.2.2 Response message

Please refer to the Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for the Response Message structure, multiplicity and business usage.

1.3 Operational Behaviour and Messages

The Claim Event Notify response is a synchronous response and is used to confirm if the request is acknowledged by the carriers. This does not include the confirmation if the notify service request message is successfully processed by the carriers.

The following diagram illustrates the service operational behaviours:



The response message contains three elements which together convey the response status of the message.

| Field Name | Definition | Usage |
|-------------------------------|---|-----------|
| AcknowledgementLevelIndicator | Code which indicates the level of acknowledgement provided in a response. Valid values are: translation_validation application_validation | Mandatory |
| AcknowledgementStatus | Code which indicates the status of the acknowledgment given within a response. Valid values are: acknowledged rejected | Mandatory |
| ResponseDescription | Narrative to support the acknowledgement status | Optional |

The service will exhibit the following operational behaviors:

- Behaviour 1 - Request Successful
- Behaviour 2 - Request Failed - Invalid Request Message Format
- Behaviour 3 - Request Failed - Duplicate Message
- Behaviour 4 - Request Failed - Invalid Recipient

1.3.1 Behaviour 1 - Request Successful

If the message is successful then a service response 'Success' will be returned.

| Behaviour 1 - Request Successful | |
|----------------------------------|--|
| Pre-condition | A valid request message is sent to the service and the request has been successfully received |
| Post-condition | <ul style="list-style-type: none"> • <i><AcknowledgementLevelIndicator></i> will be set to 'application_validation'. • <i><AcknowledgementLevelStatus></i> will be set to "acknowledged" |

1.3.2 Behaviour 2- Request Failed - Invalid Request Message Format

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

| Behaviour 2 - Invalid Request Message Format | |
|--|---|
| Pre-condition | An invalid request message is sent to the service and the request has failed schema validation within the carrier's receiving system |
| Post-condition | <ul style="list-style-type: none"> • <i><AcknowledgementLevelIndicator></i> will be set to 'translation_validation'. • <i><AcknowledgementLevelStatus></i> will be set to rRejected" • <i><ResponseDescription></i> will be set to "Fail - Request Invalid" |

1.3.3 Behaviour 3 - Request Failed - Duplicate Message

If a duplicate message is received by the carrier i.e. bearing the same unique message identifier as a previous successfully received message, then a service response 'Fail' will be returned.

An example of how this might happen is where the central system fails to receive the Carrier system's successful response message. This would have been indistinguishable from a timeout condition and the (duplicate) message would be retransmitted accordingly.

| Behaviour 3 - Duplicate Message | |
|---------------------------------|---|
| Pre-condition | A message that has previously been received and responded as successful is received again |
| Post-condition | <ul style="list-style-type: none"> • <i><AcknowledgementLevelIndicator></i> will be set to 'translation_validation'. • <i><AcknowledgementLevelStatus></i> will be set to "rejected" • <i><ResponseDescription></i> will be set to "Fail - Duplicate Message" |

1.3.4 Behaviour 4 - Request Failed - Message Receiver Not the Intended Recipient

If a message is sent to a party who is not registered for this service or is not the intended recipient for the message then a service response 'Fail - Invalid Recipient' will be returned.

| Behaviour 4 - Invalid Recipient | |
|---------------------------------|---|
| Pre-condition | The recipient of the message is not the intended recipient |
| Post-condition | <ul style="list-style-type: none"> • <i><AcknowledgementLevelIndicator></i> will be set to 'translation_validation'. • <i><AcknowledgementLevelStatus></i> will be set to "rejected" • <i><ResponseDescription></i> will be set to "Fail - Invalid Recipient" |

1.4 Claim Event Notify Exception Handling

1.4.1 Technical Errors

Technical errors e.g. SOAP fault handling will be defined in detailed design which will likely involve further collaboration with carriers and their software providers.

1.4.2 Timeout & Retry

Central systems will wait for a response message for an agreed wait period. When this threshold has been reached the Claim Event Notify message will be retransmitted.

If a response message has not been received after an agreed number of attempts then central systems will abort transmission and initiate exception handling.

The wait period and the number of attempts will be variable parameters (Please refer to the ECF - WriteBack - ClaimNotify - Interface Specification for more details).

1.4.3 Message Persistence

Sent Claim Event Notify messages will be persisted within central systems and retained for a period to be defined in a separate Non Functional Requirement document.

1.5 Claim Event Notify Non-Functional Characteristics

The Claim Event Notify service will consider the following non-functional characteristics:

1.5.1 Integration Security

Technical integration security, identification and authentication are to be defined in a separate Non Functional Requirement document.

1.5.2 Availability

Claim Event Notify messages will be triggered by ECF during core service availability hours which are 7am-7pm UK time, Monday to Friday excluding public and bank holidays (existing standard but to be confirmed in full Non Functional Requirement document). Carrier systems are expected to be available for at least these ECF core service availability hours to minimise failure, retry and exception handling overheads within central systems.

1.5.3 Service Response Times

1.5.3.1 Message Transmission from ECF

Claim Event Notify messages will be transmitted within a period (to be defined in a separate Non Functional Requirement document) of the claim transaction event being processed within central systems, provided this transmission time does not breach the ECF core service availability hours which are 7am-7pm UK time, Monday to Friday excluding public and bank holidays.

If the response time for any given message goes beyond the cessation of the ECF service, then the Claim Event Notify message will be transmitted within the core service availability hours following the resumption of the ECF service, it will not be queued.

1.5.3.2 Message Response from Carrier's System

Response messages will be transmitted from the carrier's system within an agreed period (to be defined in a separate Non Functional Requirement document) of the receipt of a Claim Event Notify message.

Failure to respond within this time will result in the Claim Event Notify message being re-transmitted and/or aborted as described in timeout/retry processing.

1.5.4 Performance and Maximum Load

1.5.4.1 Message Size

The average and maximum message sizes are to be defined in a separate Non Functional Requirement document.

1.5.4.2 Volumes

The average volume of CWT triggers expected per day is 16,500 events. Note: This figure relates to the number of recorded activities on CLASS per day and should not be taken as the number of CWT messages issued per day. The number of CWT messages issued may be considerably higher, for example, a single CLASS event on a claim with three markets registered for CWT notifications will result in three CWT messages being generated and issued.

Anticipated Volumes

Assuming 3.5 carriers per claim and a 20% retry figure, the anticipated volume of Claim Event Notify messages issued by central systems per day is anticipated to be in the vicinity of 69,300 messages (16,500 events x 3.5 carriers x 20% retry)

The anticipated volumes provided above relate to the estimate volume of messages central systems will generate across all carriers. Carriers who are registered for CWT today should base their estimates on the number of CWT notifications processed by their systems today.

1.5.5 Invoking the Service

Upon registration for this service, Carriers will provide a Production URL to receive Claim Event Notify messages applicable for each line/stamp which will be held by Central systems.

From time to time separate URLs will be provided by the carrier for lower environments e.g. MAT to carry out testing.

1.6 Conventions and Considerations

1.6.1 Identification of Parties

Wherever London Market parties are expressed in the message the following convention will be used:

- [Party/ID/@Agency](#) set to Lloyds, ILU or LIRMA (whichever is appropriate)
- [Party/ID](#) set to Broker, Syndicate or Company Number
- [Party/Name](#) set to Broker, Syndicate or Company Name

1.6.2 Claim Event Notify and LIRMA Message Sequencing

On creation of a transaction for the LIRMA market, a Claim Event Notify will be sent to all LIRMA participants. However the LIRMA Lead is the only one who can review and respond on the transaction, whereas the following market can only review the transaction.

Once the LIRMA Lead has responded with a CIR response, this will trigger another Claim Event Notify message which the following market can then review and respond to.

1.6.3 Fixed-Width Repeatable Data Elements

The Claim Event Notify request message will include a number of data elements which must be presented to the Carrier exactly as provided in the messages.

Carrier system providers should take care to:

- Use fixed-width (non-proportional) font when displaying the information
- Present the information in line item order

This relates to the following data elements;

| Data Block | Description |
|--------------------|--|
| Broker Comments | Consists of a repeatable block (up to 60). Each block contains a line number and a description. |
| Slip Lead Comments | Consists of a repeatable block (up to 15). Each block contains a line number and a description. |
| Public Comments | Block repeats per reinsurer on risk. For each reinsurer, a repeatable block (up to 15) is allowed. Each block contains a line number and a description. |

| Data Block | Description |
|-------------------|---|
| Private Comments | <p>Block repeats per agreement party in the receiving carriers' bureau.</p> <p>For each agreement party a response field is provided plus a repeatable block (up to 15 instances). Each block contains a line number and a description.</p> |
| Aggregate Details | <p>Block repeats up to 495 times.</p> <p>Each instance consists of a line number and a comment (up to 70 characters).</p> |

1.6.4 Warning Messages

Warnings generated at the time the broker creates the claim is conveyed in:

wb:Extension/wb:Claim/wb:Indicators/wb:SoftWarnings/SoftWarningDescription

- The element will not be conveyed where no warnings have been recorded.
- The element will repeat per warning.

1.6.5 Market Information

The Claim Event Notify Request message has only one sender and one recipient. However, the Request message will contain information about other markets on the risk.

The information relating to other markets on the risk is contained within the <wb:ContractMarket> tag (see sample XML Request message in Appendix 4C) and will repeat for each market on the claim.

1.6.6 Claim Event Notify when Claim Response Fails

If central systems are unavailable e.g. where the Claim Response request has been received outside of ECF core service availability hours, the request will be queued and a synchronous service response 'Request has been queued' will be returned as Claim Response Service message. The sending of the response closes the Claim Response messaging pair.

Central systems will attempt to update ECF when it is next available. If the update request is rejected, a *special case* of the Claim Event Notify request message i.e. the Claim Error Notify will be issued back to the originating Carrier. Unlike a "regular" Claim Event Notify request message, the Claim Error Notify:

- Will not be triggered by a CLASS event;
- Will not be initiated by the CWT service; and
- Will only be issued to the originating Carrier.

The following data items are included in the message where the event is in relation to this special case.

| Field Name | Definition | Usage |
|-------------------------------|---|----------|
| AcknowledgementLevelIndicator | Code which indicates the level of acknowledgement provided in a response. Valid values are: translation_validation application_validation For the Claim Event Notify this will always be set to application_validation | Optional |
| AcknowledgementStatus | Code which indicates the status of the acknowledgment given within a response. Valid values are: acknowledged rejected For the Claim Event Notify this will always be set to rejected | Optional |
| ResponseDescription | Narrative to support the acknowledgement status. This will be error messages where validation has failed | Optional |

1.6.6.1 Error Message

The following table shows the error message structure when the queued claim response message has not been successfully processed in the central system (i.e. ECF):

| Claim Event Notify Message | Field Name | Usage |
|----------------------------|------------|----------|
| <ErrorsandWarnings> | | Optional |
| <Result> | | Optional |

| | | |
|------------------------------|-------------------------|----------|
| <Level>-</Level> | ErrorWarningLevel | Optional |
| <ErrorCode>-</ErrorCode> | ErrorCode | Optional |
| <Description>-</Description> | ErrorWarningDescription | Optional |
| </Result> | | Optional |
| </ErrorsandWarnings> | | Optional |

Note: The above error notification service message design will be confirmed during detailed interface design.

1.6.7 Claim Event Notify when a Claim Is Deleted

The Claim Notify Request Message will contain only the UCR, TR, Explanation and MessageSequenceNumber within AdditonallInformation as Key Value pairs. Please refer to Delete Event Notify Message - SOAP in the appendices ([Appendix 4C](#)) for detailed information. The following table contains an example of the message structure

| Claim Event Notify Message | Field Name | Usage |
|---|---|------------|
| <wb:AdditionalInformation> | | Optional |
| <wb:ExtendedFields> | | Repeatable |
| <wb:FieldName></wb:FieldName> | Name of the data field being passed in the message e.g. UCR, TR, Explanation, MessageSequenceNumber | Mandatory |
| <wb:FieldValue></wb:FieldValue> | Value of the data field being passed in the message e.g. B0001UCR001, B0001TR001 | Mandatory |
| <wb:FieldDescription></wb:FieldDescription> | Description of the data field being passed in the message | Mandatory |
| </wb:ExtendedFields> | | |
| </wb:AdditionalInformation> | | |

2 DOCUMENT CONTROL

2.1 Document Information

| | |
|---------------------------|---------------------|
| Prepared by: | Clarissa Montecillo |
| Project Manager: | Patrick Bergin |
| Programme Manager: | John Philips |

2.2 Revision History

| Version | Date | Author | Description |
|---------|------------|------------------------------|--|
| 0.3 | 02/10/2014 | Clarissa Montecillo | Initial Draft issued to Paul T. |
| 0.4 | 15/10/14 | Parminder Kaur/Kajal Bhardwa | Updated to incorporate market review comments and other updates include Xchanging's internal reviews i.e. Inclusion of Section 1.6.6 and data definition table is further refined to add more context. |
| 0.4.1 | 04/11/14 | Kajal Bhardwa | Updated based on final comments received from working group |
| 1.0 | 14/11/14 | Kajal Bhardwa | Final signed off version |
| 1.1 | 15/09/15 | Sonal Quadros | Updated to add references to the Notify Data Dictionary and included section 1.6.7 |
| 2.0 | 26/11/2015 | Sonal Quadros | Incorporated review comments received from the market team. |

2.3 PARCI

The following roles relate to the most recent version of this document as listed in the Revision History above.

| PARCI | Project Role | Name | Signature | Date |
|----------|------------------|----------------|-----------|------|
| Producer | Business Analyst | Parminder Kaur | | |

| PARCI | Project Role | Name | Signature | Date |
|-----------|---------------------------|--|-----------|------|
| Approver | Business Requestor | Write Back Working Group (London Market Carrier community) | | |
| Reviewer | Project Manager | Pat Bergin | | |
| | Solution Architect | Vikas Acharya | | |
| | Technical Consultant | Nitin Jain | | |
| | Sponsoring Architect | Chris Hendry | | |
| | Principal Architect | Rob Jillings, John Ticehurst | | |
| | System Manager | David Burnett | | |
| | Test Manager | Simon Taylor | | |
| | Offshore Tech Project Mgr | Sonia Thakur | | |
| | Technical Architect | Mark Fillier | | |
| Consulted | Business Architect | Victoria Jandrell | | |
| Informed | Configuration Manager | Robin Winfield | | |

| PARCI | Project Role | Name | Signature | Date |
|-------|-------------------------------------|-----------------|-----------|------|
| | Enterprise Architecture Mgr | Kiwi Wilkinson | | |
| | Design Team Manager | Stuart Plummer | | |
| | Enterprise Apps Architect | Praveen Nagpal | | |
| | Enterprise Info Architect | David Lee | | |
| | Enterprise Infrastructure Architect | Aaron Goodship | | |
| | Application Lead | Ross Daines | | |
| | Technical Project Manager | Tarun Narang | | |
| | PMO | Rubina Chaudhry | | |

Section 4: Appendices

Appendix 4A: Claim Event Notify Triggers

Claim Notification Event Triggers

The following table describes the CLASS events which will trigger the generation of Claim Event Notify Triggers.

| Action | Action Code | Response Code | Notification | Recipients |
|--|-------------------------------------|---------------|--|---|
| Broker creation of new claims transaction | A | Blank | Yes - new entry | All registered carriers in the bureau participating in that claim transaction NB: please note that participating carriers include non-agreement parties on a claim, as well as agreement parties. |
| Broker cancels a transaction and creates a corrected claim transaction and re-submission | E (optional, online deletion), D, A | Blank | Yes - deletion of original and new/add entry | All registered carriers in the bureau participating in that claim transaction NB: if market is changed, the E and D action items go to original market, An action item goes to new market |
| Broker updates a transaction | E (optional, online update), U | Blank | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |
| Carrier Delegates Lead responsibilities to XCS (or a reversal of this) | U | Blank | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |

| Action | Action Code | Response Code | Notification | Recipients |
|---|--|---------------|---------------------------|---|
| On-line response by agreement party (including lead response of Circulate by LIRMA lead) | R | Response code | Yes - response entry | All registered carriers in the bureau participating in that claim transaction |
| Changes to agreement parties and changes to data that an agreement party is permitted to change | R | Response code | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |
| LIRMA user responds with > 1 company line on the claim | R | Response code | Yes - response entry | All registered carriers in the bureau participating in that claim transaction, with each line included separately in file |
| Deletion of the transaction by the broker or lead | E (optional, online deletion), D | Blank | Yes - deletion entry | All registered carriers in the bureau participating in that claim transaction |
| Completion of the transaction within CLASS | U | Blank | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |
| Market check on claim loaded by broker | Not reported | N/A | No | Managed outside of CWT; manual notification |
| Release of claim after market check | A | Blank | Yes - new/add transaction | All registered carriers in the bureau participating in that claim transaction |
| Market check and claim not released to market | Not reported | N/A | No | No recipients |

| Action | Action Code | Response Code | Notification | Recipients |
|--|--|---------------|---|--|
| Carrier declares individual/organisation conflict of interest (without change of lead) | Not reported | N/A | No | No recipients |
| Carrier declares individual/organisation conflict of interest (change of lead) | E (optional, online deletion), D (original) A (resubmit) | Blank | Yes - deletion of original and new entry for resubmission | All registered carriers in the bureau participating in that claim transaction |
| Broker action taken resulting in transaction status < 10 (in error) | E | Blank | Yes - update entry. No further advice until status becomes > 10 | All registered carriers participating on the claim. |
| Purge of claim | E | Blank | Yes | All participants on a claim. NB: the action participant type = X and the action participant = System |
| Lead un-purges a claim | U | Blank | Yes | All participants on a claim. NB: the action participant type = 'C' and the action participant = Lead Underwriter Stamp Code |

CWT Action Details on a VCS Claim

The following table provides the CWT action details on a VCS claim (Lloyd's only):

| Action | Action Code | Response Code | Notification | Recipients |
|--|-------------|---|--------------------|---|
| Broker creation of new claim i.e. 1 st transaction. | A | Blank | Yes - new entry | All registered carriers in the bureau participating in that claim transaction NB: please note that participating carriers include non-agreement parties on a claim, as well as agreement parties. |
| Leader Triages a claim as VCS | U | Blank | Yes - update entry | Registered Carrier (Leader) |
| | | VCS Entry | | VCS Service Provider |
| VCS Service Provider removes a claim from VCS | U | Blank | Yes - update entry | Registered Carrier (Leader) |
| | | VCS EXIT XCS Or VCS EXIT C-FORD | | VCS Service Provider |
| Broker creates a new transaction on an existing VCS claim | A | Blank | Yes - new entry | All registered carriers in the bureau participating in that claim transaction NB: please note that participating carriers include non-agreement parties on a claim, as well as agreement parties. |
| | | | Yes - new entry | VCS Service Provider |

| Action | Action Code | Response Code | Notification | Recipients |
|---|---|---------------|--|--|
| Broker cancels a VCS transaction and creates a corrected claim transaction and re-submission | E (optional, online deletion), D, A | Blank | Yes - deletion of original and new/add entry | All registered carriers in the bureau participating in that claim transaction NB: if market is changed, the E and D action items go to original market, An action item goes to new market. |
| | | | | VCS Service Provider |
| Broker updates a VCS transaction | E (optional, online update), U | Blank | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |
| | | | | VCS Service Provider |
| Changes to agreement parties and changes to data that an agreement party is permitted to change | R | Response code | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |
| | | | | VCS Service Provider |
| Deletion of the transaction by the broker or lead | E (optional, online deletion), D | Blank | Yes - deletion entry | All registered carriers in the bureau participating in that claim transaction |
| | | | | VCS Service Provider |

| Action | Action Code | Response Code | Notification | Recipients |
|--|--|---------------|---|--|
| Completion of the transaction within CLASS | U | Blank | Yes - update entry | All registered carriers in the bureau participating in that claim transaction |
| Carrier declares individual/organisation conflict of interest (without change of lead) | Not reported | N/A | No | No recipients |
| Carrier declares individual/organisation conflict of interest (change of lead) | E (optional, online deletion), D (original) A (resubmit) | Blank | Yes - deletion of original and new entry for resubmission | All registered carriers in the bureau participating in that claim transaction |
| | | | | VCS Service Provider |
| Broker action taken resulting in transaction status < 10 (in error) | E | Blank | Yes - update entry. No further advice until status becomes > 10 | All registered carriers participating on the claim. |
| | | | | VCS Service Provider |
| Purge of claim | E | Blank | Yes | All participants on a claim. NB: the action participant type = X and the action participant = System |
| | | | | VCS Service Provider |

| Action | Action Code | Response Code | Notification | Recipients |
|------------------------|-------------|---------------|--------------|---|
| Lead un-purges a claim | U | Blank | Yes | All participants on a claim. NB: the action participant type = 'C' and the action participant = Lead Underwriter Stamp Code |
| | | | | VCS Service Provider |

Appendix 4B: Code Sets

WB-CODE-001: Warning Messages

Warning messages are stored on broker input and displayed when an agreement party opens a transaction that they can respond to, or update their response or on request.

Please refer to Lloyd's Warnings_Errors, LIRMA Lead Warnings_Errors, LIRMA Co Warnings_Errors, ILU Warnings_Errors tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for a list of all warnings; some are no longer generated, but it is possible that they may be displayed on older claims.

WB-CODE-002: Response Codes

Please refer to the Response Code tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid response codes per bureau

WB-CODE-003: Outstanding Amount Qualifier

Please refer to the Outstanding Qualifier Code tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid outstanding amount qualifiers

WB-CODE-004: Business Class

Please refer to the Business Class Code tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid outstanding amount qualifiers

WB-CODE-005: Transaction Status Value

Please refer to the Transaction Status Values tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid outstanding amount qualifiers

WB-CODE-006: Claim Category

Please refer to the Claim Category Code tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid outstanding amount qualifiers

WB-CODE-007: Claim Sub Category

Please refer to the Claim Sub Category Code tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid outstanding amount qualifiers

WB-CODE-008: Date of Loss Qualifier

Please refer to the Loss Date Qualifier tab in the latest Claim Event Notify Data Dictionary embedded in the ECF - WriteBack - ClaimNotify - Interface Specification for all valid outstanding amount qualifiers

Appendix 4C: Clarifications and SOAPS

Delete Event Notify Message - SOAP

Issue Summary: Currently when a Broker deletes a claim transaction, the CLASS mainframe system deletes it from the database (Hard Delete) and sends a CWT trigger with skinny data to the subscribed carriers. We are currently trying to replicate this for the write back notify message by extracting the information from CWT into the Notify message however we are facing the following problems

- a. The data contained in the CWT message is not sufficient to populate all the data elements and Business mandatory fields in the Claim Notify message
- b. The enrichment process fails at Orchestration layer because there is no data available in the CLASS system due to hard delete.

In order to resolve this, we are proposing the following options

Solution Summary:

1. Option-1: Populate the fields given below which are mandatory at schema level and for the business but not available in CWT/CAS with some static data on the premise that these fields will not be relevant for a delete event and their data can be ignored.

- BureauLeadingInsurerOrReinsurer/Party/Name
- LossDetails
- ClaimStatus
- AddedCirculatedOrAuthorisedDate

Pros:

- No change to existing schema
- No need to undo some of the build that is already done, just need to add functionality on top to populate static data

Cons:

- Change to the DD to update the comments section of the given fields

2. Option-2[**Preferred**]: Do not populate the Claim Movement aggregate as it is optional and pass only the UCR, TR, Explanation and MessageSequenceNumber within the AdditionalInformation tag of the Claim Notify message as Key Value pairs. For example:

<wb:AdditionalInformation>

```
<wb:ExtendedFields>  
<wb:FieldName>MessageSequenceNumber</wb:FieldName>  
<wb:FieldValue>100</wb:FieldValue>  
<wb:FieldDescription>Message Sequence Number</wb:FieldDescription>  
<wb:ExtendedFields>
```

```
<wb:ExtendedFields>  
<wb:FieldName>UMR</wb:FieldName>  
<wb:FieldValue>B0001UMR1</wb:FieldValue>  
<wb:FieldDescription>Broker Reference</wb:FieldDescription>  
<wb:ExtendedFields>
```

```
<wb:ExtendedFields>  
<wb:FieldName>UCR</wb:FieldName>  
<wb:FieldValue>B0001UCR1</wb:FieldValue>  
<wb:FieldDescription>Claim Reference</wb:FieldDescription>  
<wb:ExtendedFields>
```

```
<wb:ExtendedFields>  
<wb:FieldName>TR</wb:FieldName>  
<wb:FieldValue>B0001TR1</wb:FieldValue>  
<wb:FieldDescription>Transaction Reference</wb:FieldDescription>  
<wb:ExtendedFields>
```

```
<wb:ExtendedFields>  
<wb:FieldName>Explanation</wb:FieldName>  
<wb:FieldValue>D</wb:FieldValue>  
<wb:FieldDescription>Transaction Deleted</wb:FieldDescription>  
<wb:ExtendedFields>
```

```
</wb:AdditionalInformation>
```

Pros:

- Keeps the message clean with only information required
- No change to the XSD
- Minor change to build to implement this approach

Cons

- Update to the interface spec to include this implementation for the delete scenario