

# Business Message Standard (BMS)

for

## Align\_GDSN\_Price Synchronization

*(Process/Sub-process/Industry and/or Product/Geopolitical)*

### BRG: Align\_GDSN

**BMS Release: \_\_2.1\_\_**

**Document Version: \_\_0.0.3\_\_**

**Date: \_\_16.05.2007\_\_**

*(dd.mm.cyy example 27.10.2004)*



### Change Request Reference

<b>Refer to Change Request (CR) Number(s):</b>	05-000250
<b>CR Submitter(s):</b>	Tom Heist for RDD Team
<b>Date of CR Submission to GSMP:</b>	15 August 2005

### Business Requirements Document (BRAD) Reference

<b>BRAD Title: BRAD Price Synchronisation in the GDSN</b>
<b>BRAD Date: 30.04.2007</b>
<b>BRAD Version: 0.0.6</b>

### Document Summary

<b>Document Title:</b>	<b>BMS Align_GDSN_Price Synchronization</b>
<b>Document Version</b>	0.0.3
<b>Owner:</b>	GDSN
<b>Status:</b>	( <i>Check one box</i> ) <input type="checkbox"/> DRAFT <input checked="" type="checkbox"/> Approved
<b>BMS Template Version:</b>	0.4
<b>Targeted BMS Publication Version</b>	2.1

### Document Change History

*Note: During development include revisions in history. Upon Approval, eliminate revisions and include only delta from previous version.*

<b>Date of Change</b>	<b>Version</b>	<b>Changed By</b>	<b>Reason for Change</b>	<b>Summary of Change</b>	<b>Model Build #</b>
25.05.2006	0.0.1	Eric Kauz	Initial Draft		
05.04.2007	0.0.2	Eric Kauz	Pilot Feedback	Update to business rules based on pilot feedback. <ul style="list-style-type: none"> <li>o Updated UC-9 Related Rule 8 and UC-10 Related Rule 10 to read "Bracket Qualifiers for a Price Type can be sent providing that the brackets have not been sent as standard brackets".</li> </ul>	

30.04.2007	0.0.3	Eric Kauz	Comment Review	<ul style="list-style-type: none"><li>○ Changed Business Rule 18 in UC-9 and UC-10 to clarify the application of Allowances/Charges.</li><li>○ Update rule 9 in UC-9 and Rule 10 in UC-10 to clarify application of bracket qualifiers.</li></ul>	
------------	-------	-----------	----------------	---	--

# Business Message Standard

---

## Table of contents

---

Chapter	Page
<b>1 Business Solution .....</b>	<b>1</b>
1.1 Business Domain View.....	1
1.1.1 Problem Statement / Business Need.....	1
1.1.2 Objective.....	1
1.1.3 Audience.....	1
1.1.4 Artefacts .....	1
1.1.5 References .....	1
1.1.6 Acknowledgements .....	2
1.1.6.1 BRG Members .....	2
1.1.6.2 ITRG Members .....	2
1.1.6.3 Task/Project Group Participants ( <i>where applicable</i> ) .....	2
1.1.6.4 Design Team Members .....	2
1.2 Business Context .....	3
1.3 Additional Technical Requirements Analysis .....	3
1.3.1 Technical Requirements (optional).....	3
1.4 Business Transaction View .....	4
1.4.1 Use Case Diagram – Process Overview .....	4
1.4.2 Use Case Diagram – Manage Trading Relationship .....	5
1.4.3 Use Case Definitions – Add Trading Relationship.....	5
1.4.4 Use Case Diagram – Update Trading Relationship.....	7
1.4.5 Use Case Definitions – Update Trading Relationship .....	7
1.4.6 Use Case Definitions – Cancel Trading Relationship.....	9
1.4.7 Use Case Definitions – Discontinue Trading Relationship .....	11
1.4.8 Use Case Diagram – Synchronise Conditions .....	13
1.4.9 Use Case Definitions – Add Condition .....	13
1.4.10 Use Case Diagram – Modify Conditions.....	16
1.4.11 Use Case Definitions – Modify Condition .....	16
1.4.12 Use Case Definitions – Withdraw Condition.....	18
1.4.13 Use Case Definitions – Discontinue a Condition .....	20
1.4.14 Use Case Diagram – Synchronise Price Type .....	22
1.4.15 Use Case Definitions – Add Item Price Type .....	22
1.4.16 Use Case Diagram – Update Item Price Type .....	25

## Business Message Standard

---

### Table of contents

---

1.4.17	Use Case Definitions – Modify Item Price Type .....	25
1.4.18	Use Case Definitions – Withdraw Item Price Type .....	28
1.4.19	Use Case Definitions – Discontinue an Item Price Type .....	30
1.4.20	Business Transaction Activity Diagram(s) .....	31
1.4.21	Business Transaction Sequence Diagram(s) (optional) .....	31
1.5	Information Model (including GDD Report) .....	32
1.5.1	Data Description: .....	32
1.5.2	GDD Report : .....	32
1.5.3	Class Diagrams .....	55
1.5.3.1	Class Diagrams: Price Synchronisation Document .....	55
1.5.3.2	Class Diagrams: Price Synchronisation Confirmation .....	59
1.5.3.3	Class Diagrams: Common: To be removed from BMS Post Development. ....	60
1.5.4	Code Lists .....	61
1.6	Business Document Example .....	68
1.7	Implementation Considerations .....	68
1.8	Testing .....	68
1.8.1	Pass / Fail Criteria .....	68
1.8.2	Test Data .....	68
1.8.2.1	Price Synchronisation Document .....	68
1.8.2.2	Price Synchronisation Confirmation .....	70
1.9	Appendices .....	71
1.10	Summary of Changes .....	71
<b>2</b>	<b>2 Technical Solution Design .....</b>	<b>72</b>

## 1 Business Solution

### 1.1 Business Domain View

#### 1.1.1 Problem Statement / Business Need

Currently there is limited capability for electronically communicating accurate pricing information between trading partners using global standards that

*“accommodates all the different pricing business practices and facilitates an invoice amount equal to the expected payment amount equal to the actual payment”.*

Globally, pricing business practices range from simple pricing and transactional pricing to component based pricing. Component based pricing includes components such as promotions, allowances, charges and brackets.

#### 1.1.2 Objective

To supply the detail design of the (specific) business transaction needed to meet the requirements of the referenced in the BRAD for Price Synchronisation in the GDSN V 0.0.5

#### 1.1.3 Audience

The audience would be any participant in the global supply chain engaged in the GDSN. This would include the roles of suppliers (or sellers or data source), source data pools, recipient data pools, retailers (or buyers or data recipient ) and other third parties.

#### 1.1.4 Artefacts

Artefact name	State	Artefact / State description

#### 1.1.5 References

Reference #	Reference Name	Description
1	BRAD Price Synchronisation in the GDSN V 0.0.5	Requirements documentation for applying price synchronisation in the GDSN.
2	Align – BMS Trading Partner Profile	Approved global standard for price synchronization outside of the GDSN.
3	Align – BMS Condition Document and Monetary Documents	Approved global standard for price synchronization outside of the GDSN.

## Business Solution Design

---

---

### 1.1.6 Acknowledgements

#### 1.1.6.1 BRG Members

See Task/Project Group Participants

#### 1.1.6.2 ITRG Members

Not applicable

#### 1.1.6.3 Task/Project Group Participants (*where applicable*)

Function	Name	Company / organisation
Participant	Kraig Adams	Coca Cola
Participant	Gundeep Ahluwalia	Agentrics
Participant	Javier Arias	GS1 Spain
Participant	Neale Austen	GS1 Australia
Participant	Dan Beaudry	Procter & Gamble
Participant	Joe Bohning	Nestle Pet Care
Participant	Bryan Dunlap	Pillsbury
Participant	Mark Hann	GSX
Participant	John Durovec	1SYNC
Participant	Grant Kille	Agentrics
Participant	Carolyn Kroll	1SYNC
Participant	Ken Kubat	Tibco
Participant	Lynn Martinez	Cadbury
Participant	Randy Mercer	Lanza
Participant	Terry Mochar	Reckitt-Benkiser
Participant	Mrinalini Nayar	Pepsico
Participant	Nasir Qadeer	1SYNC
Participant	Nadine Radomski	Dean Foods
Participant	Steve Robba	Johnson & Johnson
Participant	Karen Spooner	Kraft
Participant	Diane Tetens	General Mills
Participant	Dave Wasielewski	Sterling Commerce
Participant	Greg Zwanziger	Supervalu

#### 1.1.6.4 Design Team Members

Function	Name	Organisation
Modeller	Eric Kauz	GS1
XML Technical Designer	Dipan Anarkat	GS1
EANCOM Technical Designer		
Peer Reviewer	Brian Bennett, John Ryu	GS1

### 1.2 Business Context

*(Note: The business context of the business)*

Context Category	Value(s)
Industry	All
Geopolitical	All
Product	All
Process	Align_GDSN_Price
System Capabilities	EAN.UCC
Official Constraints	None

### 1.3 Additional Technical Requirements Analysis

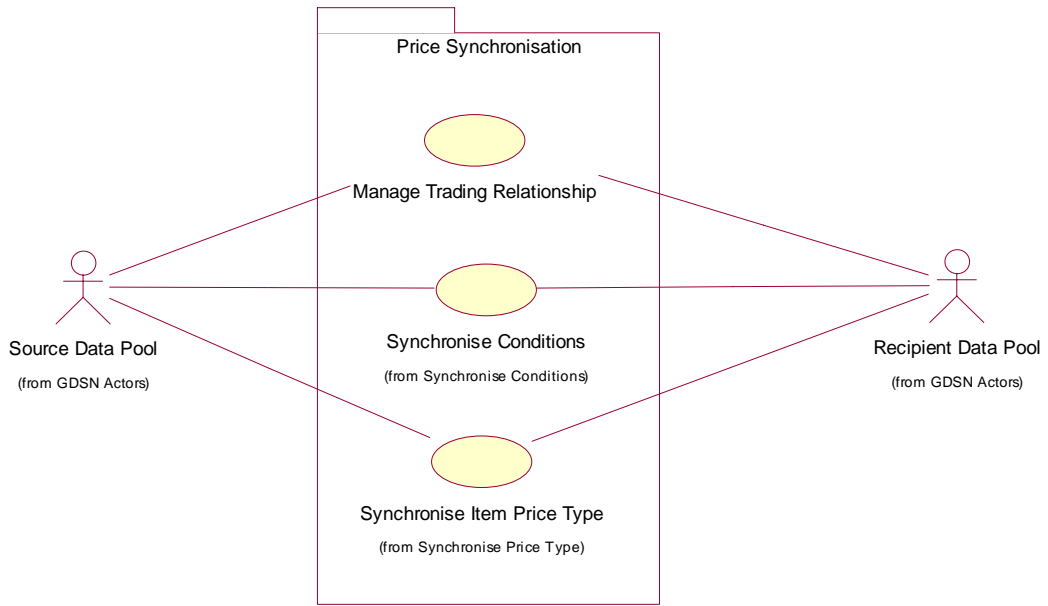
#### 1.3.1 Technical Requirements (optional)

Not applicable.



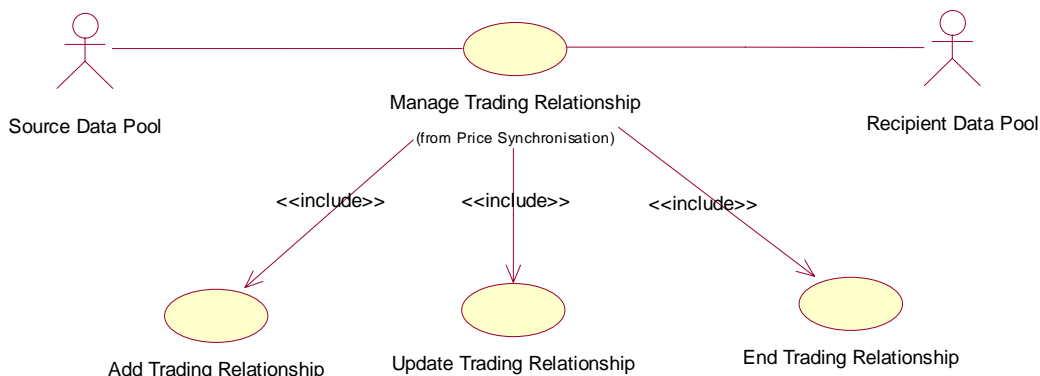
**1.4 Business Transaction View**

**1.4.1 Use Case Diagram – Process Overview**



**Figure 1 - Use Case Diagram: Price Synchronisation GDSN**

### 1.4.2 Use Case Diagram – Manage Trading Relationship



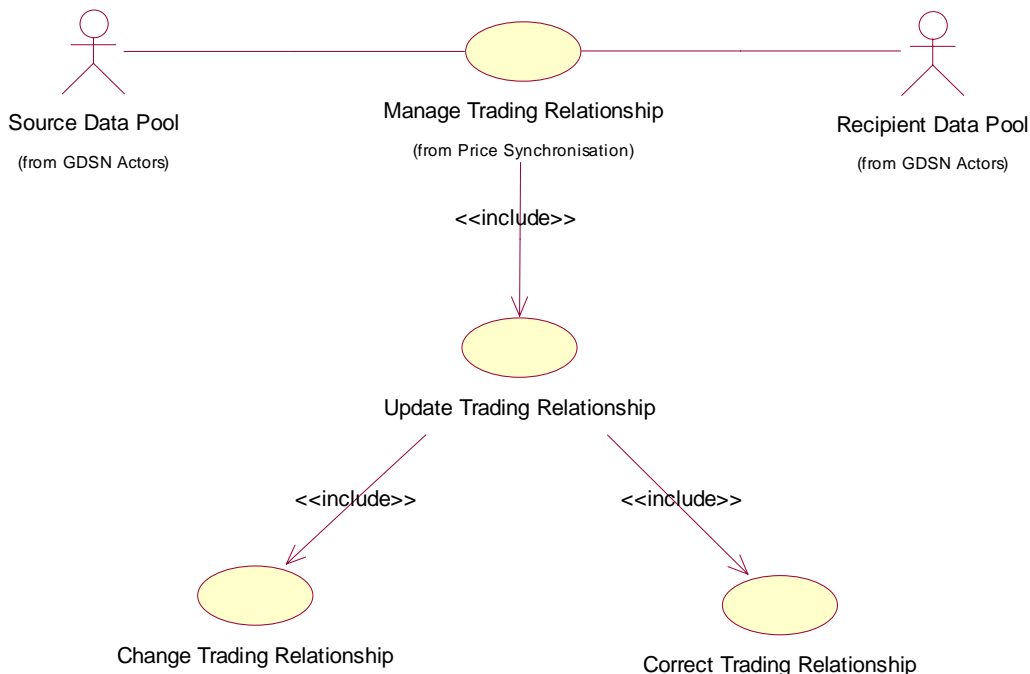
### 1.4.3 Use Case Definitions – Add Trading Relationship

<b>Use Case ID</b>	UC-1																						
<b>Use Case Name</b>	Add Trading Relationship																						
<b>Use Case Description</b>	This use case establishes a price synchronisation trading partner relationship.																						
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																						
<b>Performance Goals</b>	Initiate a price synchronisation relationship.																						
<b>Preconditions</b>	Trading partners have established a trading partner agreement including price synchronisation relationships, agreed-to pricing conditions; and are engaged in item synchronisation.																						
<b>Post conditions</b>	Price synchronisation relationship is active.																						
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of a new relationship and the SDP creates a price synchronisation list for the relationship. (Done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a relationship by sending a price synchronisation message with a document command of "add" with a relationship segment action code of "add" to the RDP.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Receives price synchronisation message and sends relationship information to data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends confirmation information to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b>data source receives the confirmation response and the pricing syn-</p>		Step #	Actor	Activity Step	1	SDP	Performs validations.	2	SDP	Creates a relationship by sending a price synchronisation message with a document command of "add" with a relationship segment action code of "add" to the RDP.	3	RDP	Receives price synchronisation message and sends relationship information to data recipient.	4	Data Recipient	Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends confirmation information to the data source.
Step #	Actor	Activity Step																					
1	SDP	Performs validations.																					
2	SDP	Creates a relationship by sending a price synchronisation message with a document command of "add" with a relationship segment action code of "add" to the RDP.																					
3	RDP	Receives price synchronisation message and sends relationship information to data recipient.																					
4	Data Recipient	Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.																					
5	RDP	Sends the confirmation response message to the SDP.																					
6	SDP	Updates the price synchronisation list and sends confirmation information to the data source.																					

## Business Solution Design

	chronisation is active.		
Alternative Scenario	<i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i>		
	<b>Step #</b>	<b>Actor</b>	<b>Activity Step</b>
		All	The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient's RDP. To reduce complexity the later is not shown in the activity steps in any scenario.
	3	Data Recipient	Data recipient responds with a confirmation status other than accept or no response is sent by the data recipient. See related rules below for status codes and their actions.
Related Requirements	1		
	2		
Related Rules	1.		Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship.
	2.		Action codes for the header segment other than initial load are: resend – used to indicate the message is to recover a lost or missing message; restart – used where a data recipient had rejected an item's pricing and wishes to resume synchronisation; and reload – used to "start over" by sending all current and future pricing.
	3.		A price synchronisation relationship can have only one active relationship segment at a time.
	4.		If the Document Header is "ADD", the Price Document ID must = "1"
	5.		When relationship action document header equals "add", there are no dependency checks.
	6.		The data recipient can override a previous confirmation status with another one through a confirmation response.
	7.		Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.

## 1.4.4 Use Case Diagram – Update Trading Relationship



## 1.4.5 Use Case Definitions – Update Trading Relationship

<b>Use Case ID</b>	UC-2									
<b>Use Case Name</b>	Update Trading Relationship									
<b>Use Case Description</b>	This use case maintains the price synchronisation trading partner relationship through either modifications or corrections to the relationship data.									
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient									
<b>Performance Goals</b>	To update the price synchronisation relationship.									
<b>Preconditions</b>	Trading partners have established a trading partner agreement including price synchronisation relationships, agreed-to pricing conditions; and are engaged in item synchronisation. Trading relationship data has been previously received and accepted by the data recipient.									
<b>Post conditions</b>	Price synchronisation relationship is updated.									
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of updates to a trading relationship (done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Validates trading relationship information.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Updates the relationship by sending a price synchronisation message</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	1	SDP	Validates trading relationship information.	2	SDP	Updates the relationship by sending a price synchronisation message
Step #	Actor	Activity Step								
1	SDP	Validates trading relationship information.								
2	SDP	Updates the relationship by sending a price synchronisation message								

## Business Solution Design

			with a document command of "Modify" with a relationship section action code of "Modify" (for a modification) or "Correct" (for a correct) to the RDP.
	3	RDP	Receives price synchronisation message and sends relationship information to data recipient.
	4	Data Recipient	Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.
	5	RDP	Sends the confirmation response message to the SDP.
	6	SDP	Sends confirmation information to the data source.
	<b>Ends when...</b> data source receives the confirmation response.		
<b>Alternative Scenario</b>	<i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i>		
	<b>Step #</b>	<b>Actor</b>	<b>Activity Step</b>
		All	The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipients RDP. To reduce complexity the later is not shown in the activity steps in any scenario.
	3	Data Recipient	Data Recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.
<b>Related Requirements</b>	1		
<b>Related Rules</b>	1	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the Data Recipient wishes to terminate the price synchronisation relationship.	
	2	A confirmation status of "rejected" results in a data recipient initiated termination of the trading relationship.	
	3	If the Document Header is "MODIFY", the Price Document ID must be greater than "1"	
	4	When Relationship action equals "modify", a positive response must be in the sync list for the Relationship segment before any adds/modifies/corrects/deletes to any other segment are sent. Note: a positive response is defined as any confirmation response other than "rejected" or "no response".	
	5	Relationship Start Effective Date can only be Corrected, not modified.	
	6	Relationship Start Effective Date can only be Corrected if the date is in the future.	
	7.	The data recipient can override a previous confirmation status with another one through a confirmation response.	
	8.	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.	

**1.4.6 Use Case Definitions – Cancel Trading Relationship**

<b>Use Case ID</b>	UC-3																						
<b>Use Case Name</b>	Cancel Trading Relationship																						
<b>Use Case Description</b>	This use case terminates a specific price synchronisation trading partner relationship that has not yet taken effect.																						
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																						
<b>Performance Goals</b>	To terminate a price synchronisation relationship.																						
<b>Preconditions</b>	Trading partners have established a trading partner agreement including price synchronisation relationships, agreed-to pricing conditions; and are engaged in item synchronisation. Trading relationship data has been previously received by the data recipient.																						
<b>Post conditions</b>	Price synchronisation relationship is terminated.																						
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of the need to terminate a trading relationship (done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs validations.</td> </tr> <tr> <td>1</td> <td>SDP</td> <td>Terminates the relationship by sending a price synchronisation message with a document command of "modify" with a relationship section action code of "Delete" to the RDP.</td> </tr> <tr> <td>2</td> <td>RDP</td> <td>Receives price synchronisation message and sends relationship information to data recipient.</td> </tr> <tr> <td>3</td> <td>Data Recipient</td> <td>Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>4</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>5</td> <td>SDP</td> <td>sends confirmation information to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b>data source receives the confirmation response.</p>		Step #	Actor	Activity Step	1	SDP	Performs validations.	1	SDP	Terminates the relationship by sending a price synchronisation message with a document command of "modify" with a relationship section action code of "Delete" to the RDP.	2	RDP	Receives price synchronisation message and sends relationship information to data recipient.	3	Data Recipient	Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.	4	RDP	Sends the confirmation response message to the SDP.	5	SDP	sends confirmation information to the data source.
Step #	Actor	Activity Step																					
1	SDP	Performs validations.																					
1	SDP	Terminates the relationship by sending a price synchronisation message with a document command of "modify" with a relationship section action code of "Delete" to the RDP.																					
2	RDP	Receives price synchronisation message and sends relationship information to data recipient.																					
3	Data Recipient	Receives the trading relationship information and confirms the relationship by responding with an "accept" response. The confirmation response message is sent to the RDP.																					
4	RDP	Sends the confirmation response message to the SDP.																					
5	SDP	sends confirmation information to the data source.																					
<b>Alternative Scenario</b>	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td></td> <td>All</td> <td>The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient's RDP. To reduce complexity the later is not shown in the activity steps in any scenario.</td> </tr> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>		Step #	Actor	Activity Step		All	The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient's RDP. To reduce complexity the later is not shown in the activity steps in any scenario.	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.												
Step #	Actor	Activity Step																					
	All	The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient's RDP. To reduce complexity the later is not shown in the activity steps in any scenario.																					
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																					
<b>Related Requirements</b>	<table border="1"> <tr> <td>1</td> <td></td> </tr> </table>		1																				
1																							
<b>Related Rules</b>																							

	1	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship.
	2	A confirmation status of "REJECTED" is not valid for the End Trading Relationship use case. A confirmation status of "REJECTED" implies that the data recipient initiated the termination of the trading relationship.
	3	If the Document Header is "MODIFY", the Price Document ID must be greater than "1"
	4	The "Delete" action code implies all data associated with the Relationship ID is no longer valid only for a relationship that has not taken effect. If the relationship is already in effect, the data source must send a Modify transaction and populate the End Effective Date.
	5	In order to end a relationship segment, all dependent condition and price type segments need to be deleted/end dated before a delete/end date can be sent for the Relationship Segment.
	6	Can only end at a Relationship Segment ID level (i.e. if you have 3 relationship IDs identified for a trading relationship, in order to end the ENTIRE relationship, all 3 relationship IDs must be deleted/end dated).

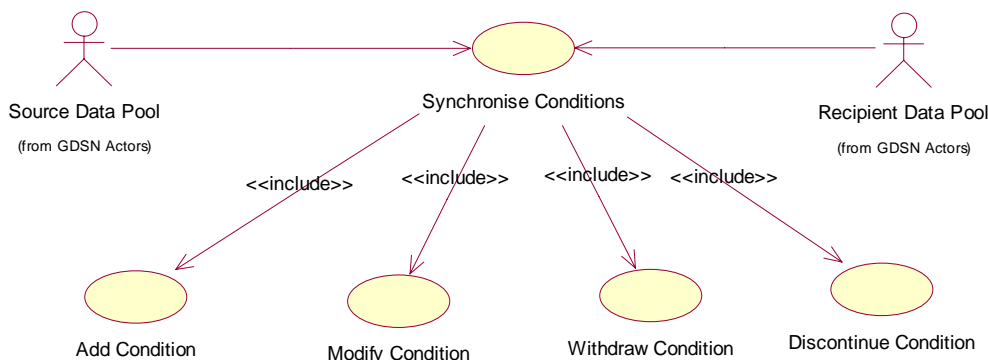
## 1.4.7 Use Case Definitions – Discontinue Trading Relationship

<b>Use Case ID</b>	UC-4																						
<b>Use Case Name</b>	Discontinue Trading Relationship																						
<b>Use Case Description</b>	This use case terminates a specific price synchronisation trading partner relationship that is currently in effect.																						
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																						
<b>Performance Goals</b>	To terminate a price synchronisation relationship.																						
<b>Preconditions</b>	Trading partners have established a trading partner agreement including price synchronisation relationships, agreed-to pricing conditions; and are engaged in item synchronisation. Trading relationship data has been previously received by the data recipient. The current date is greater than or equal to the effective date of the relationship.																						
<b>Post conditions</b>	Price synchronisation relationship is discontinued.																						
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of the need to discontinue a trading relationship (done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs validations.</td> </tr> <tr> <td>1</td> <td>SDP</td> <td>Terminates the relationship by sending a price synchronisation message with a document command of “modify” with a relationship section action code of “Modify” to the RDP and a populated relationship end effective date.</td> </tr> <tr> <td>2</td> <td>RDP</td> <td>Receives price synchronisation message and sends relationship information to data recipient.</td> </tr> <tr> <td>3</td> <td>Data Recipient</td> <td>Receives the trading relationship information and confirms the relationship change by responding with an “accept” response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>4</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>5</td> <td>SDP</td> <td>Sends confirmation information to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b>data source receives the confirmation response.</p>		Step #	Actor	Activity Step	1	SDP	Performs validations.	1	SDP	Terminates the relationship by sending a price synchronisation message with a document command of “modify” with a relationship section action code of “Modify” to the RDP and a populated relationship end effective date.	2	RDP	Receives price synchronisation message and sends relationship information to data recipient.	3	Data Recipient	Receives the trading relationship information and confirms the relationship change by responding with an “accept” response. The confirmation response message is sent to the RDP.	4	RDP	Sends the confirmation response message to the SDP.	5	SDP	Sends confirmation information to the data source.
Step #	Actor	Activity Step																					
1	SDP	Performs validations.																					
1	SDP	Terminates the relationship by sending a price synchronisation message with a document command of “modify” with a relationship section action code of “Modify” to the RDP and a populated relationship end effective date.																					
2	RDP	Receives price synchronisation message and sends relationship information to data recipient.																					
3	Data Recipient	Receives the trading relationship information and confirms the relationship change by responding with an “accept” response. The confirmation response message is sent to the RDP.																					
4	RDP	Sends the confirmation response message to the SDP.																					
5	SDP	Sends confirmation information to the data source.																					
<b>Alternative Scenario</b>	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td></td> <td>All</td> <td>The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient’s RDP. To reduce complexity the later is not shown in the activity steps in any scenario.</td> </tr> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>		Step #	Actor	Activity Step		All	The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient’s RDP. To reduce complexity the later is not shown in the activity steps in any scenario.	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.												
Step #	Actor	Activity Step																					
	All	The scenario shows the most anticipated choreography where the SDP sends to the RDP; but the SDP may send directly to the data recipient in situations where the SDP is also the data recipient’s RDP. To reduce complexity the later is not shown in the activity steps in any scenario.																					
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																					
<b>Related Requirements</b>	<table border="1"> <tr> <td>1</td> <td></td> </tr> </table>		1																				
1																							



Related Rules	
	<p><b>1</b> Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship.</p>
	<p><b>2</b> A confirmation status of "REJECTED" is not valid for the End Trading Relationship use case. A confirmation status of "REJECTED" implies that the data recipient initiated the termination of the trading relationship.</p>
	<p><b>3</b> If the Document Header is "MODIFY", the Price Document ID must be greater than "1"</p>
	<p><b>4</b> The "Delete" action code implies all data associated with the Relationship ID is no longer valid only for a relationship that has not taken effect. If the relationship is already in effect, the data source must send a Modify transaction and populate the End Effective Date.</p>
	<p><b>5</b> In order to end a relationship segment, all dependent condition and price type segments need to be deleted/end dated before a delete/end date can be sent for the Relationship Segment.</p>
	<p><b>6</b> Can only end at a Relationship Segment ID level (i.e. if you have 3 relationship IDs identified for a trading relationship, in order to end the ENTIRE relationship, all 3 relationship IDs must be deleted/end dated).</p>

### 1.4.8 Use Case Diagram – Synchronise Conditions



### 1.4.9 Use Case Definitions – Add Condition

<b>Use Case ID</b>	UC-5																					
<b>Use Case Name</b>	Add Condition																					
<b>Use Case Description</b>	This use case establishes non-line item conditions and summary conditions.																					
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																					
<b>Performance Goals</b>	Establish conditions for price synchronisation.																					
<b>Preconditions</b>	Price synchronisation relationship has been established and price synchronisation is active.																					
<b>Post conditions</b>	Conditions are synchronized.																					
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of price components to be added for a relationship (done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "modify" (if the trading relationship has already been established) and the condition segment with a segment action code of "add" to the RDP, indicates the condition types and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b> conditions and bracket qualifiers (as needed) have been synchro-</p>	Step #	Actor	Activity Step	1	SDP	Performs validations.	2	SDP	Creates a price synchronisation message using document command of "modify" (if the trading relationship has already been established) and the condition segment with a segment action code of "add" to the RDP, indicates the condition types and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.
Step #	Actor	Activity Step																				
1	SDP	Performs validations.																				
2	SDP	Creates a price synchronisation message using document command of "modify" (if the trading relationship has already been established) and the condition segment with a segment action code of "add" to the RDP, indicates the condition types and updates the price synchronisation list.																				
3	RDP	Sends the price message to the data recipient.																				
4	Data Recipient	Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.																				
5	RDP	Sends the confirmation response message to the SDP.																				
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																				

## Business Solution Design

	nized.																												
Alternative Scenario	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																						
	Step #	Actor	Activity Step																										
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																											
Related Requirements																													
Related Rules	<table border="1"> <tbody> <tr> <td>1</td> <td>When condition type equals bracket the bracket sub-section is used to identify the bracket qualifiers.</td> </tr> <tr> <td>2</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>3</td> <td>In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.</td> </tr> <tr> <td>4</td> <td>If there has been no response to relationship segment, the condition segment is still sent to the data recipient.</td> </tr> <tr> <td>5</td> <td>Cannot send condition if relationship has been rejected.</td> </tr> <tr> <td>6</td> <td>If the Condition Segment is sent in the first price message establishing the trading relationship, the Document Header Command = ADD.</td> </tr> <tr> <td>7</td> <td>If the Condition Segment is sent after establishing the trading relationship, the Document Header Command = MODIFY.</td> </tr> <tr> <td>8</td> <td>If Document Header Command = ADD, then Price Document ID must = "1".</td> </tr> <tr> <td>9</td> <td>If Document Header Command = MODIFY, then Price Document ID must be greater than "1".</td> </tr> <tr> <td>10</td> <td>Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.</td> </tr> <tr> <td>11</td> <td>Confirmation Statuses (assumes one response per condition) <ul style="list-style-type: none"> <li>• Accepted</li> <li>• Review</li> <li>• Synchronized</li> <li>• Rejected – no further price synchronisation may occur <ul style="list-style-type: none"> <li>○ For the specified condition ID</li> <li>○ Nor any price type referring to that condition ID</li> <li>○ Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.</li> </ul> </li> <li>• No Response – no further price synchronisation may occur <ul style="list-style-type: none"> <li>○ For the specified condition ID</li> <li>○ Nor any price type referring to that condition ID</li> </ul> </li> </ul> </td> </tr> <tr> <td>12</td> <td>To depict a line item allowance in the condition the Catalogue Items(s) or Global Product Classification "brick" code(s) must be specified in the condition segment.</td> </tr> <tr> <td>13</td> <td>The data recipient can override a previous confirmation status with another one through a confirmation response.</td> </tr> <tr> <td>14</td> <td>Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Ac-</td> </tr> </tbody> </table>	1	When condition type equals bracket the bracket sub-section is used to identify the bracket qualifiers.	2	Header segment is mandatory and must be sent with this message.	3	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.	4	If there has been no response to relationship segment, the condition segment is still sent to the data recipient.	5	Cannot send condition if relationship has been rejected.	6	If the Condition Segment is sent in the first price message establishing the trading relationship, the Document Header Command = ADD.	7	If the Condition Segment is sent after establishing the trading relationship, the Document Header Command = MODIFY.	8	If Document Header Command = ADD, then Price Document ID must = "1".	9	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".	10	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.	11	Confirmation Statuses (assumes one response per condition) <ul style="list-style-type: none"> <li>• Accepted</li> <li>• Review</li> <li>• Synchronized</li> <li>• Rejected – no further price synchronisation may occur <ul style="list-style-type: none"> <li>○ For the specified condition ID</li> <li>○ Nor any price type referring to that condition ID</li> <li>○ Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.</li> </ul> </li> <li>• No Response – no further price synchronisation may occur <ul style="list-style-type: none"> <li>○ For the specified condition ID</li> <li>○ Nor any price type referring to that condition ID</li> </ul> </li> </ul>	12	To depict a line item allowance in the condition the Catalogue Items(s) or Global Product Classification "brick" code(s) must be specified in the condition segment.	13	The data recipient can override a previous confirmation status with another one through a confirmation response.	14	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Ac-
	1	When condition type equals bracket the bracket sub-section is used to identify the bracket qualifiers.																											
	2	Header segment is mandatory and must be sent with this message.																											
	3	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.																											
	4	If there has been no response to relationship segment, the condition segment is still sent to the data recipient.																											
	5	Cannot send condition if relationship has been rejected.																											
	6	If the Condition Segment is sent in the first price message establishing the trading relationship, the Document Header Command = ADD.																											
	7	If the Condition Segment is sent after establishing the trading relationship, the Document Header Command = MODIFY.																											
	8	If Document Header Command = ADD, then Price Document ID must = "1".																											
	9	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".																											
	10	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.																											
	11	Confirmation Statuses (assumes one response per condition) <ul style="list-style-type: none"> <li>• Accepted</li> <li>• Review</li> <li>• Synchronized</li> <li>• Rejected – no further price synchronisation may occur <ul style="list-style-type: none"> <li>○ For the specified condition ID</li> <li>○ Nor any price type referring to that condition ID</li> <li>○ Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.</li> </ul> </li> <li>• No Response – no further price synchronisation may occur <ul style="list-style-type: none"> <li>○ For the specified condition ID</li> <li>○ Nor any price type referring to that condition ID</li> </ul> </li> </ul>																											
	12	To depict a line item allowance in the condition the Catalogue Items(s) or Global Product Classification "brick" code(s) must be specified in the condition segment.																											
	13	The data recipient can override a previous confirmation status with another one through a confirmation response.																											
14	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Ac-																												

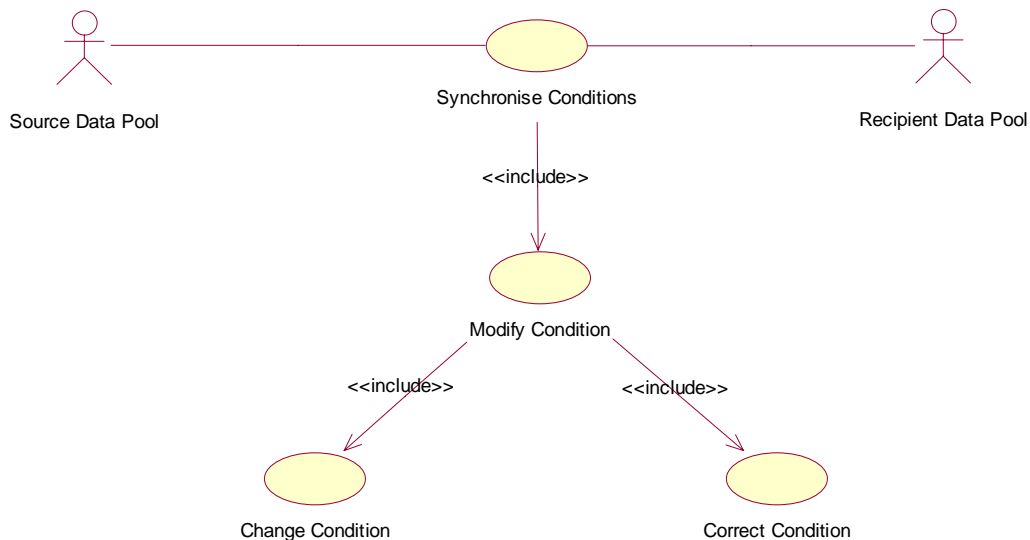
## Business Solution Design

---

---

		cepted' followed by 'Synchronised'.	
--	--	-------------------------------------	--

## 1.4.10 Use Case Diagram – Modify Conditions



## 1.4.11 Use Case Definitions – Modify Condition

<b>Use Case ID</b>	UC-6															
<b>Use Case Name</b>	Modify Condition															
<b>Use Case Description</b>	This use case modifies or corrects an existing non-line item conditions and summary conditions.															
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient															
<b>Performance Goals</b>	Modify or correct conditions for price synchronisation.															
<b>Preconditions</b>	Price synchronisation relationship exists and price component has been accepted by data source.															
<b>Post conditions</b>	Condition has been modified.															
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of modifications to item depictions and/or any related price types. (Done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "Modify" and the condition segment with a segment action code of "Modify" for a modification or "Correct" for a correct to the RDP, indicates the condition types and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the conditions by responding with an "Accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "Modify" and the condition segment with a segment action code of "Modify" for a modification or "Correct" for a correct to the RDP, indicates the condition types and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the conditions by responding with an "Accept" confirmation response. The confirmation response message is sent to the RDP.
Step #	Actor	Activity Step														
1	SDP	Performs necessary validations.														
2	SDP	Creates a price synchronisation message using document command of "Modify" and the condition segment with a segment action code of "Modify" for a modification or "Correct" for a correct to the RDP, indicates the condition types and updates the price synchronisation list.														
3	RDP	Sends the price message to the data recipient.														
4	Data Recipient	Receives the message and confirms the conditions by responding with an "Accept" confirmation response. The confirmation response message is sent to the RDP.														

## Business Solution Design

	5	RDP	Sends the confirmation response message to the SDP.																												
	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																												
	<b>Ends when...</b> conditions and bracket qualifiers (as needed) have been modified.																														
<b>Alternative Scenario</b>	<i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i>																														
	<b>Step #</b>	<b>Actor</b>	<b>Activity Step</b>																												
	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																												
<b>Related Requirements</b>																															
<b>Related Rules</b>	<table border="1"> <tr> <td>1</td> <td>When condition type equals bracket the bracket sub-section is used to identify the bracket qualifiers.</td> </tr> <tr> <td>2</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>3</td> <td>In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.</td> </tr> <tr> <td>4</td> <td>If there has been no response to relationship segment, the condition segment is still sent to the data recipient.</td> </tr> <tr> <td>5</td> <td>Cannot send condition if relationship has been rejected.</td> </tr> <tr> <td>6</td> <td>If Document Header Command = MODIFY, then Price Document ID must be greater than "1".</td> </tr> <tr> <td>7</td> <td>Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.  Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.</td> </tr> <tr> <td>8</td> <td>Condition Value Type cannot be modified.</td> </tr> <tr> <td>9</td> <td>Start Effective Date cannot be modified.</td> </tr> <tr> <td>10</td> <td>Start Effective Date can be corrected.</td> </tr> <tr> <td>11</td> <td>If a revised Start Effective Date is required for a condition that is not yet in effect, the condition must be deleted; indicating the condition or price has been withdrawn and will never apply to the trading relationship.</td> </tr> <tr> <td>12</td> <td>The data recipient can override a previous confirmation status with another one through a confirmation response.</td> </tr> <tr> <td>13</td> <td>Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.</td> </tr> <tr> <td>14</td> <td>Can't send an update for a segment if the previous add/modify/correct has had no response or has been rejected.</td> </tr> </table>			1	When condition type equals bracket the bracket sub-section is used to identify the bracket qualifiers.	2	Header segment is mandatory and must be sent with this message.	3	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.	4	If there has been no response to relationship segment, the condition segment is still sent to the data recipient.	5	Cannot send condition if relationship has been rejected.	6	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".	7	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.  Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.	8	Condition Value Type cannot be modified.	9	Start Effective Date cannot be modified.	10	Start Effective Date can be corrected.	11	If a revised Start Effective Date is required for a condition that is not yet in effect, the condition must be deleted; indicating the condition or price has been withdrawn and will never apply to the trading relationship.	12	The data recipient can override a previous confirmation status with another one through a confirmation response.	13	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.	14	Can't send an update for a segment if the previous add/modify/correct has had no response or has been rejected.
1	When condition type equals bracket the bracket sub-section is used to identify the bracket qualifiers.																														
2	Header segment is mandatory and must be sent with this message.																														
3	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.																														
4	If there has been no response to relationship segment, the condition segment is still sent to the data recipient.																														
5	Cannot send condition if relationship has been rejected.																														
6	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".																														
7	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.  Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.																														
8	Condition Value Type cannot be modified.																														
9	Start Effective Date cannot be modified.																														
10	Start Effective Date can be corrected.																														
11	If a revised Start Effective Date is required for a condition that is not yet in effect, the condition must be deleted; indicating the condition or price has been withdrawn and will never apply to the trading relationship.																														
12	The data recipient can override a previous confirmation status with another one through a confirmation response.																														
13	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.																														
14	Can't send an update for a segment if the previous add/modify/correct has had no response or has been rejected.																														

## 1.4.12 Use Case Definitions – Withdraw Condition

<b>Use Case ID</b>	UC-7																						
<b>Use Case Name</b>	Withdraw Condition																						
<b>Use Case Description</b>	This use case deletes an existing non-line item conditions and/or summary conditions prior to the effective start date of the condition.																						
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																						
<b>Performance Goals</b>	Withdraw a condition for price synchronisation.																						
<b>Preconditions</b>	Price synchronisation relationship exists and price component has been accepted by data source.																						
<b>Post conditions</b>	Condition has been withdrawn.																						
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of a need to withdraw a price component (done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "Modify" and the condition segment with a segment action code of "Delete" to the RDP, indicates the condition types and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b> conditions and bracket qualifiers (as needed) have been withdrawn.</p>		Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "Modify" and the condition segment with a segment action code of "Delete" to the RDP, indicates the condition types and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.
Step #	Actor	Activity Step																					
1	SDP	Performs necessary validations.																					
2	SDP	Creates a price synchronisation message using document command of "Modify" and the condition segment with a segment action code of "Delete" to the RDP, indicates the condition types and updates the price synchronisation list.																					
3	RDP	Sends the price message to the data recipient.																					
4	Data Recipient	Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.																					
5	RDP	Sends the confirmation response message to the SDP.																					
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																					
<b>Alternative Scenario</b>	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>		Step #	Actor	Activity Step	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.															
Step #	Actor	Activity Step																					
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																					
<b>Related Requirements</b>	<table border="1"> <tr> <td style="width: 50px;"> </td> <td> </td> </tr> </table>																						
<b>Related Rules</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>2</td> <td>In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respec-</td> </tr> </tbody> </table>		1	Header segment is mandatory and must be sent with this message.	2	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respec-																	
1	Header segment is mandatory and must be sent with this message.																						
2	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respec-																						

## Business Solution Design

---

---

		tive lists.
	3	If there has been no response to relationship segment, the condition segment is still sent to the data recipient.
	4	Cannot send condition if relationship has been rejected.
	5	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".
	6	A confirmation status of "REJECTED" is not valid for a withdraw.
	7	No response means that no further synchronisation can occur.
	8	The data recipient can override a previous confirmation status with another one through a confirmation response.
	9	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.



## 1.4.13 Use Case Definitions – Discontinue a Condition

<b>Use Case ID</b>	UC-8																						
<b>Use Case Name</b>	Discontinue Condition																						
<b>Use Case Description</b>	This use case discontinues an existing non-line item conditions and/or summary conditions that are already in effect.																						
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																						
<b>Performance Goals</b>	Discontinues a condition for price synchronisation.																						
<b>Preconditions</b>	Price synchronisation relationship exists and price component has been accepted by data source.																						
<b>Post conditions</b>	Condition has been discontinued.																						
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of a need to discontinue a price component. (Done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "Modify", the condition segment with a segment action code of "Modify" and an updated Condition End Effective Date to the RDP, indicates the condition types and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b> conditions and bracket qualifiers (as needed) have been discontinued.</p>		Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "Modify", the condition segment with a segment action code of "Modify" and an updated Condition End Effective Date to the RDP, indicates the condition types and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.
Step #	Actor	Activity Step																					
1	SDP	Performs necessary validations.																					
2	SDP	Creates a price synchronisation message using document command of "Modify", the condition segment with a segment action code of "Modify" and an updated Condition End Effective Date to the RDP, indicates the condition types and updates the price synchronisation list.																					
3	RDP	Sends the price message to the data recipient.																					
4	Data Recipient	Receives the message and confirms the conditions by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.																					
5	RDP	Sends the confirmation response message to the SDP.																					
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																					
<b>Alternative Scenario</b>	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>		Step #	Actor	Activity Step	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.															
Step #	Actor	Activity Step																					
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																					
<b>Related Requirements</b>																							
<b>Related Rules</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>2</td> <td>In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.</td> </tr> <tr> <td>3</td> <td>If there has been no response to relationship segment, the condition segment is</td> </tr> </tbody> </table>		1	Header segment is mandatory and must be sent with this message.	2	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.	3	If there has been no response to relationship segment, the condition segment is															
1	Header segment is mandatory and must be sent with this message.																						
2	In the condition segment, confirmations apply to the condition type and apply to all Catalogue Items or EANUCC Classification Category Codes in their respective lists.																						
3	If there has been no response to relationship segment, the condition segment is																						

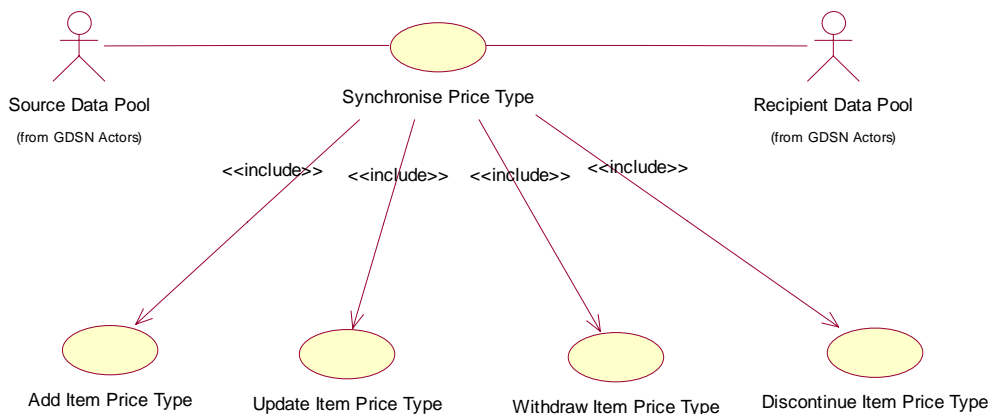
## Business Solution Design

---

---

		still sent to the data recipient.
	4	Cannot send condition if relationship has been rejected.
	5	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".
	6	No response means that no further synchronisation can occur.
	7	The data recipient can override a previous confirmation status with another one through a confirmation response.
	8	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.

### 1.4.14 Use Case Diagram – Synchronise Price Type



### 1.4.15 Use Case Definitions – Add Item Price Type

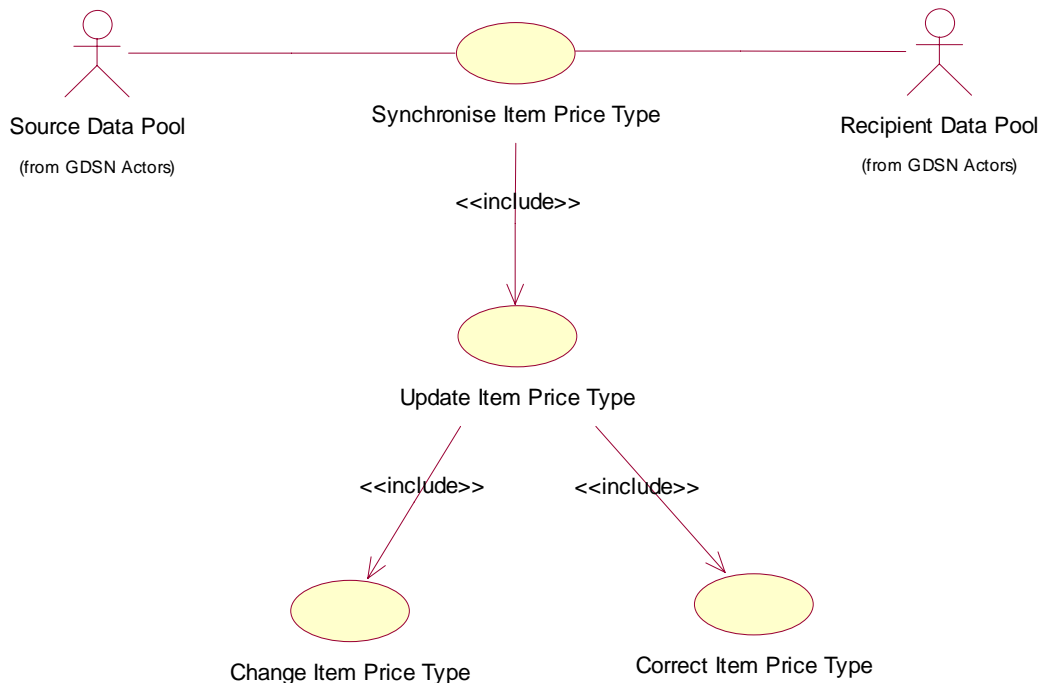
<b>Use Case ID</b>	UC-9																					
<b>Use Case Name</b>	Add Item Price Type																					
<b>Use Case Description</b>	This use case establishes an item depiction and any related price types.																					
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																					
<b>Performance Goals</b>	Establish an item depiction and all related price types for price synchronisation.																					
<b>Preconditions</b>	Price synchronisation relationship has been established and price synchronisation is active.																					
<b>Post conditions</b>	Item Depictions and related price types are synchronized.																					
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of new price types that they want to be synchronised with a trading partner. (Done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "modify" (if the trading relationship has already been established) and the item depiction and price type segments with a segment action code of "add" to the RDP and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "modify" (if the trading relationship has already been established) and the item depiction and price type segments with a segment action code of "add" to the RDP and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.
Step #	Actor	Activity Step																				
1	SDP	Performs necessary validations.																				
2	SDP	Creates a price synchronisation message using document command of "modify" (if the trading relationship has already been established) and the item depiction and price type segments with a segment action code of "add" to the RDP and updates the price synchronisation list.																				
3	RDP	Sends the price message to the data recipient.																				
4	Data Recipient	Receives the message and confirms the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.																				
5	RDP	Sends the confirmation response message to the SDP.																				
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																				

## Business Solution Design

	<b>Ends when...</b> item Depictions and related price types have been synchronized.		
<b>Alternative Scenario</b>	<i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i>		
	<b>Step #</b>	<b>Actor</b>	<b>Activity Step</b>
	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.
<b>Related Requirements</b>			
<b>Related Rules</b>	1	The Item Price Segment is mandatory if the Item Depiction qualifier has been populated.	
	2	Header segment is mandatory and must be sent with this message.	
	3	Relationship Segment and Condition Segment is not needed in a price synchronisation message to send Item Depiction and Price Type.	
	4	Cannot send Item Depiction and Price Type if the Trading Relationship has been rejected.	
	5	If the Target Condition ID in the Item Price Type segment is populated, the Confirmation Status for the targeted price type must be Accepted, Synchronized or Review.	
	6	If the target price type in the Item Price Type segment is populated, the Confirmation Status for the targeted price type must be Accepted, Synchronized or Review	
	7	Multiple Price Types may exist simultaneously for a Catalogue Item and each can have their own confirmation status	
	8	Bracket Qualifiers for a Price Type can be sent providing that the brackets have not been sent as standard brackets in the condition segment.	
	9	If Document Header Command = ADD, then Price Document ID must = "1".	
	10	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.	
	11	No Response stops further synchronisation of only the specific Price types for which the confirmation status is "No response"	
	12	A confirmation status of "Rejected" <ul style="list-style-type: none"> <li>•Can only be sent for a reason code of "item not accepted"</li> <li>•Stops further synchronisation of all Price Types for the associated trade item.</li> </ul>	
	13	If the Ship To or Ship From Business Location attributes are populated with more than one value (a list), the data recipient may only accept or reject all, not individually by Business Location	
	14	Condition segment is not required.	
	15	Relationship segment is not required.	
	16	At least one Item Price Segment is mandatory if the Item Depiction qualifier has been populated.	
	17	Can only populate Target Price type if "Price Type" is equal to "Allowance" or "Charge"	
	18	If the Target Price Type is not populated, the allowance/charge is tied to the	

		catalogue item itself, regardless of the base price it is using (i.e. will span all base price brackets).
	<b>19</b>	Data recipients cannot accept or reject individual brackets qualifiers.
	<b>20</b>	When an Item Price Type refers to a condition: <ul style="list-style-type: none"><li>○ The Condition Type must be of type "Bracket"</li><li>○ The Item Price segment must only be of Price type "Bracket".</li></ul>
	<b>21</b>	The data recipient can override a previous confirmation status with another one through a confirmation response.
	<b>22</b>	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.

## 1.4.16 Use Case Diagram – Update Item Price Type



## 1.4.17 Use Case Definitions – Modify Item Price Type

<b>Use Case ID</b>	UC-10									
<b>Use Case Name</b>	Modify Item Price Type									
<b>Use Case Description</b>	This use case modifies or corrects existing item depictions and/or any related price types.									
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient									
<b>Performance Goals</b>	Modify or correct item depictions and/or any related price types used for price synchronisation.									
<b>Preconditions</b>	Price synchronisation relationship exists and item depictions and/or any related price types have been accepted by data source.									
<b>Post conditions</b>	Item depictions and/or any related price types have been modified.									
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of modifications to price components (done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "Modify" and the item depiction and price type segments with a segment action code of "Modify" for modification and "Correct" for a correction to the RDP and updates the price synchronisation list.</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "Modify" and the item depiction and price type segments with a segment action code of "Modify" for modification and "Correct" for a correction to the RDP and updates the price synchronisation list.
Step #	Actor	Activity Step								
1	SDP	Performs necessary validations.								
2	SDP	Creates a price synchronisation message using document command of "Modify" and the item depiction and price type segments with a segment action code of "Modify" for modification and "Correct" for a correction to the RDP and updates the price synchronisation list.								

## Business Solution Design

	<table border="1"> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </table> <p><b>Ends when...</b> the item depiction and price type segments have been modified.</p>	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.												
3	RDP	Sends the price message to the data recipient.																							
4	Data Recipient	Receives the message and confirms the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.																							
5	RDP	Sends the confirmation response message to the SDP.																							
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																							
Alternative Scenario	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																		
Step #	Actor	Activity Step																							
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																							
Related Requirements																									
Related Rules	<table border="1"> <tr> <td>1</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>2</td> <td>If Document Header Command = MODIFY, then Price Document ID must be greater than "1".</td> </tr> <tr> <td>3</td> <td>Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.  Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.</td> </tr> <tr> <td>4</td> <td>Start Effective Date cannot be modified.</td> </tr> <tr> <td>5</td> <td>Start Effective Date can be corrected.</td> </tr> <tr> <td>6</td> <td>Cannot send Item Depiction and Price Type if the Trading Relationship has been rejected.</td> </tr> <tr> <td>7</td> <td>If the Target Condition ID in the Item Price Type segment is populated, the Confirmation Status for the Condition must be Accepted, Synchronized or Review</td> </tr> <tr> <td>8</td> <td>If the target price type in the Item Price Type segment is populated, the Confirmation Status for the Condition must be Accepted, Synchronized or Review.</td> </tr> <tr> <td>9</td> <td>Multiple Item price types may exist simultaneously for a Catalogue Item and each can have their own confirmation status</td> </tr> <tr> <td>10</td> <td>Bracket Qualifiers for a Price Type can be sent providing that the brackets have not been sent as standard brackets in the condition segment.</td> </tr> <tr> <td>11</td> <td>No Response stops further synchronisation of only the specific price types for which the confirmation status is "No response".</td> </tr> <tr> <td>12</td> <td>A confirmation status of "Rejected"  <ul style="list-style-type: none"> <li>•Can only be sent for a reason code of "item not accepted"</li> <li>•Stops further synchronisation of all price types for the associated</li> </ul> </td> </tr> </table>	1	Header segment is mandatory and must be sent with this message.	2	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".	3	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.  Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.	4	Start Effective Date cannot be modified.	5	Start Effective Date can be corrected.	6	Cannot send Item Depiction and Price Type if the Trading Relationship has been rejected.	7	If the Target Condition ID in the Item Price Type segment is populated, the Confirmation Status for the Condition must be Accepted, Synchronized or Review	8	If the target price type in the Item Price Type segment is populated, the Confirmation Status for the Condition must be Accepted, Synchronized or Review.	9	Multiple Item price types may exist simultaneously for a Catalogue Item and each can have their own confirmation status	10	Bracket Qualifiers for a Price Type can be sent providing that the brackets have not been sent as standard brackets in the condition segment.	11	No Response stops further synchronisation of only the specific price types for which the confirmation status is "No response".	12	A confirmation status of "Rejected" <ul style="list-style-type: none"> <li>•Can only be sent for a reason code of "item not accepted"</li> <li>•Stops further synchronisation of all price types for the associated</li> </ul>
1	Header segment is mandatory and must be sent with this message.																								
2	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".																								
3	Confirmation status codes other than accept are: review – message received and no action taken yet; synchronized – message received and implemented into the backend system; reject – message received and terms of a specific price message segment were rejected or the data recipient wishes to terminate the price synchronisation relationship. If no response is made the SDP will stop price synchronisation.  Can only send a reject if related to a specific catalogue item. This is used in the event that the data recipient does not wish to synchronise pricing for the indicated catalogue item.																								
4	Start Effective Date cannot be modified.																								
5	Start Effective Date can be corrected.																								
6	Cannot send Item Depiction and Price Type if the Trading Relationship has been rejected.																								
7	If the Target Condition ID in the Item Price Type segment is populated, the Confirmation Status for the Condition must be Accepted, Synchronized or Review																								
8	If the target price type in the Item Price Type segment is populated, the Confirmation Status for the Condition must be Accepted, Synchronized or Review.																								
9	Multiple Item price types may exist simultaneously for a Catalogue Item and each can have their own confirmation status																								
10	Bracket Qualifiers for a Price Type can be sent providing that the brackets have not been sent as standard brackets in the condition segment.																								
11	No Response stops further synchronisation of only the specific price types for which the confirmation status is "No response".																								
12	A confirmation status of "Rejected" <ul style="list-style-type: none"> <li>•Can only be sent for a reason code of "item not accepted"</li> <li>•Stops further synchronisation of all price types for the associated</li> </ul>																								

		trade item.
13		If the Ship To or Ship From Business Location attributes are populated with more than one value (a list), the data recipient may only accept or reject all, not individually by Business Location
14		Condition segment is not required.
15		Relationship segment is not required.
16		The Item Price Segment is mandatory if the Item Depiction qualifier has been populated.
17		Can only populate the target Item price type if the condition type is equal to "Allowance" or "Charge".
18		If the target item price type is not populated, the allowance/charge is tied to the catalogue item itself, regardless of the base price it is using (i.e. will span all base price brackets)
19		Data recipients cannot accept or reject individual brackets qualifiers.
20		When an Item Price Type refers to a condition: <ul style="list-style-type: none"> <li>o The Condition Type must be of type "Bracket"</li> <li>o The Item Price segment must only be of Price type "Bracket".</li> </ul>
21		The data recipient can override a previous confirmation status with another one through a confirmation response.
22		Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.
23		Can't send an update for a segment if the previous add/modify/correct has had no response or has been rejected.



## 1.4.18 Use Case Definitions – Withdraw Item Price Type

<b>Use Case ID</b>	UC-11																					
<b>Use Case Name</b>	Withdraw Item Price Type																					
<b>Use Case Description</b>	This use case deletes existing item depictions and/or any related price types prior to the effective start date of the price type.																					
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																					
<b>Performance Goals</b>	Withdraw existing item depictions and/or any related price types that are not in effect.																					
<b>Preconditions</b>	Price synchronisation relationship exists and item depictions and/or any related price types have been accepted by data source.																					
<b>Post conditions</b>	Item depictions and/or any related price types have been withdrawn.																					
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of a need to withdraw item depictions and/or any related price types. (Done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "Modify" and the item depiction and price type segments with a segment action code "Delete" and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the withdrawal of the item depiction and price type segments by responding with an "Accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b> item depictions and/or any related price types have been withdrawn.</p>	Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "Modify" and the item depiction and price type segments with a segment action code "Delete" and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the withdrawal of the item depiction and price type segments by responding with an "Accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.
Step #	Actor	Activity Step																				
1	SDP	Performs necessary validations.																				
2	SDP	Creates a price synchronisation message using document command of "Modify" and the item depiction and price type segments with a segment action code "Delete" and updates the price synchronisation list.																				
3	RDP	Sends the price message to the data recipient.																				
4	Data Recipient	Receives the message and confirms the withdrawal of the item depiction and price type segments by responding with an "Accept" confirmation response. The confirmation response message is sent to the RDP.																				
5	RDP	Sends the confirmation response message to the SDP.																				
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																				
<b>Alternative Scenario</b>	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>	Step #	Actor	Activity Step	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.															
Step #	Actor	Activity Step																				
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																				
<b>Related Requirements</b>																						
<b>Related Rules</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>2</td> <td>If Document Header Command = MODIFY, then Price Document ID must be greater than "1".</td> </tr> <tr> <td>3</td> <td>A confirmation status of "REJECTED" is not valid for a withdrawal of a Price</td> </tr> </tbody> </table>	1	Header segment is mandatory and must be sent with this message.	2	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".	3	A confirmation status of "REJECTED" is not valid for a withdrawal of a Price															
1	Header segment is mandatory and must be sent with this message.																					
2	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".																					
3	A confirmation status of "REJECTED" is not valid for a withdrawal of a Price																					

		Type.
	4	A confirmation status of "REVIEW" is not valid for a withdrawal or a Price Type.
	5	No response means that no further synchronisation can occur.
	6	Condition segment is not required.
	7	Relationship segment is not required.
	8	The Item Price Segment is mandatory if the Item Depiction qualifier has been populated.
	9	When the Item Price Type Segment Action Code = Delete <ul style="list-style-type: none"><li>•Must populate the End Effective Date for a parent and all children associated where the Price Type Start Effective Date is in the future.</li><li>•Must delete the parent and all children where the Price Type is still current</li><li>•End dating or deleting all price types associated with an item, not deleting the item</li></ul>
	10	The data recipient can override a previous confirmation status with another one through a confirmation response.
	11	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.

## 1.4.19 Use Case Definitions – Discontinue an Item Price Type

<b>Use Case ID</b>	UC-12																						
<b>Use Case Name</b>	Discontinue Item Price Type																						
<b>Use Case Description</b>	This use case deletes existing item depictions and/or any related price types that are already in effect.																						
<b>Actors (Goal)</b>	Data source, Source Data Pool, Recipient Data Pool, Data Recipient																						
<b>Performance Goals</b>	Discontinues existing item depictions and/or any related price types.																						
<b>Preconditions</b>	Price synchronisation relationship exists and existing item depictions and/or any related price types have been accepted by data source.																						
<b>Post conditions</b>	Item depictions and/or any related price types have been discontinued.																						
<b>Scenario</b>	<p><b>Begins when...</b>The data source notifies their SDP of a need to discontinue item depictions and/or any related price types. (Done outside of the network).</p> <p><b>Continues with...</b></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SDP</td> <td>Performs necessary validations.</td> </tr> <tr> <td>2</td> <td>SDP</td> <td>Creates a price synchronisation message using document command of "modify" and the item depiction and price type segments with a segment action code "Modify" and updates the price synchronisation list.</td> </tr> <tr> <td>3</td> <td>RDP</td> <td>Sends the price message to the data recipient.</td> </tr> <tr> <td>4</td> <td>Data Recipient</td> <td>Receives the message and confirms the discontinuation of the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.</td> </tr> <tr> <td>5</td> <td>RDP</td> <td>Sends the confirmation response message to the SDP.</td> </tr> <tr> <td>6</td> <td>SDP</td> <td>Updates the price synchronisation list and sends the confirmation response message to the data source.</td> </tr> </tbody> </table> <p><b>Ends when...</b> the item depiction and price type segments have been discontinued.</p>		Step #	Actor	Activity Step	1	SDP	Performs necessary validations.	2	SDP	Creates a price synchronisation message using document command of "modify" and the item depiction and price type segments with a segment action code "Modify" and updates the price synchronisation list.	3	RDP	Sends the price message to the data recipient.	4	Data Recipient	Receives the message and confirms the discontinuation of the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.	5	RDP	Sends the confirmation response message to the SDP.	6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.
Step #	Actor	Activity Step																					
1	SDP	Performs necessary validations.																					
2	SDP	Creates a price synchronisation message using document command of "modify" and the item depiction and price type segments with a segment action code "Modify" and updates the price synchronisation list.																					
3	RDP	Sends the price message to the data recipient.																					
4	Data Recipient	Receives the message and confirms the discontinuation of the item depiction and price type segments by responding with an "accept" confirmation response. The confirmation response message is sent to the RDP.																					
5	RDP	Sends the confirmation response message to the SDP.																					
6	SDP	Updates the price synchronisation list and sends the confirmation response message to the data source.																					
<b>Alternative Scenario</b>	<p><i>The step #s below are related to the step #s in the scenario and are alternatives to the scenario steps</i></p> <table border="1"> <thead> <tr> <th>Step #</th> <th>Actor</th> <th>Activity Step</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Data Recipient</td> <td>Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.</td> </tr> </tbody> </table>		Step #	Actor	Activity Step	3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.															
Step #	Actor	Activity Step																					
3	Data Recipient	Data recipient responds with a confirmation status other than accept. See related requirements below for status codes and their actions.																					
<b>Related Requirements</b>																							
<b>Related Rules</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Header segment is mandatory and must be sent with this message.</td> </tr> <tr> <td>2</td> <td>If Document Header Command = MODIFY, then Price Document ID must be greater than "1".</td> </tr> <tr> <td>3</td> <td>A confirmation status of "REJECTED" is not valid for a discontinue.</td> </tr> <tr> <td>4</td> <td>A confirmation status of "REVIEW" is not valid for a discontinue.</td> </tr> </tbody> </table>		1	Header segment is mandatory and must be sent with this message.	2	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".	3	A confirmation status of "REJECTED" is not valid for a discontinue.	4	A confirmation status of "REVIEW" is not valid for a discontinue.													
1	Header segment is mandatory and must be sent with this message.																						
2	If Document Header Command = MODIFY, then Price Document ID must be greater than "1".																						
3	A confirmation status of "REJECTED" is not valid for a discontinue.																						
4	A confirmation status of "REVIEW" is not valid for a discontinue.																						

	5	No response means that no further synchronisation can occur.
	6	Condition segment is not required.
	7	Relationship segment is not required.
	8	For a discontinue, the End Effective Date must be populated or updated.
	9	The data recipient can override a previous confirmation status with another one through a confirmation response.
	10	Multiple confirmations can be sent by data recipients for a single price message or message segment. For example, a data recipient can send a status of 'Accepted' followed by 'Synchronised'.

### 1.4.20 Business Transaction Activity Diagram(s)

### 1.4.21 Business Transaction Sequence Diagram(s) (optional)

## Business Solution Design

### 1.5 Information Model (including GDD Report)

#### 1.5.1 Data Description:

#### 1.5.2 GDD Report :

##### Price Synchronisation Document

Class (ABIE)	Attribute (BBIE)	Association (ASBIE)	Secondary Class	Official Dictionary Entry Name	Definition	Multiplicity	Related Requirements
BracketQualifier				Bracket Qualifier. Details	Identifiers conditions required to be met to qualify for a bracket.		
	bracketOperator			Bracket Qualifier. Bracket Operator. Code Value_ Text	A function to identify the logical relationship between multiple bracket qualifiers (And/Or).	0..1	{ref 1} 49
	bracketRangeQualifier-Code			Bracket Qualifier. Bracket Range Qualifier Code. Bracket Range Qualifier_Code	Specifies whether the bracket range is based upon an amount, a measurement or an other quantity	1..1	{ref 1} 48
	bracketTierMaximum			Bracket Qualifier. Bracket Tier Maximum. Total_Quantity	The upper limits for qualification for a bracket.	0..1	{ref 1} 47
	bracketTierMinimum			Bracket Qualifier. Bracket Tier Minimum. Total_Quantity	The lower limit for qualification for a bracket.	1..1	{ref 1} 46
ConditionTargetEntity				Condition Target Entity. Details	Provides the specific item or groups of items that the condition applies to.		
	classificationCategory-Code			Condition Target Entity. Classification Category Code. Text	The classification category associated with a specific condition.	1..*	
		None	CatalogueItemReference	Condition Target	Associates one or many	1..*	

## Business Solution Design

				Entity. Choice_Association. Catalogue Item Identification	catalogue items with a price type.		
ConditionValueInformation				Condition Value Information. Details	Provides the quantity or value associated with the condition for example 7 percent.		
	conditionValue			Condition Value Information.Condition Value.Total_Quantity	Provides a value or percent associated with a condition.	1..1	{ref 1} 37
	conditionValueType			Condition Value Information.Condition Value Type.Component Value Type_Code	A classification of the price component used to determine how to apply the amount for example value, rate or percent.	1..1	{ref 1} 36
	conditionValueCap			Condition Value Information. Condition Value Cap. Float_ Numeric	A quantity or measurement associated with the condition value qualifier to limit the calculation of rate to a specified maximum amount. This would be used where a trading partner sets a maximum value for an offer.	0..1	{ref 1} 39
IncotermInformation				Incoterm. Details	Incoterm is an abbreviation for International Commercial Terms. The International Chamber of Commerce created and manages the Incoterms and their definitions. There are 13 available for use in the buyer-seller contractual agreements.		
	incotermCode			Incoterm. Incoterm Code. Incoterm_Code	Incoterms is an abbreviation for International Commercial Terms. The	1..1	{ref 1} 25

## Business Solution Design

					International Chamber of Commerce created and manages the Incoterms and their definitions. There are 13 available for use in the buyer-seller contractual agreements. Incoterm references may be selected from an enumerated list.		
	incotermCodeLocation			Incoterm. Incoterm Code Location. Text	A description of the location required by an Incoterm.	1..1	{ref 1} 26
ItemDepictionQualifier				Item Depiction Qualifier. Details	A price synchronisation message segment used to show how the pricing information would be depicted on an invoice.		{ref 1} 50
		None	CatalogueItemReference	Item Depiction Qualifier. Association. Catalogue Item Identification	Associates a price type with a catalogue item.	1..1	{ref 1} 51
		None	ItemPriceType	Item Depiction Qualifier. Association. Item Price Type	Associates one or many item price types with an item depiction qualifier.	1..*	{ref 1} 57
ItemPriceType				Item Price Type. Details	Contains details of a price component associated with an item.		
	alternateLocationGrouping			Item Price Type. Alternate Location Grouping. Text	A string of characters used to describe a cluster of business locations mutually defined by the Information Provider and the Party Receiving Private Data.	0..1	{ref 1} 67
	distributionMethodCode			Item Price Type. Distribution Method Code. Code	The mode by which the Information Provider and the Party Receiving Private Data have agreed at what point(s)	1..1	{ref 1} 63

## Business Solution Design

					in the supply chain the Information Provider makes the goods available to the Party Receiving Private Data.		
	priceActionCode			Item Price Type. Price Action Code. Segment Action_ Code	A code assigned by the Information Provider to indicate to the Party Receiving Private Data, the reason for sending the price information contained within the specified segment within the Price Synchronization Message. The Party Receiving Private Data is able to use this code to determine the nature of the action associated with each price component within each price type segment. For example the addition of a new record, the modification of an existing record or the correction of an existing record.	1..1	{ref 1} 54
	priceActionReason			Item Price Type. Price Action Reason. Price Action Reason Code_ Code	A code to indicate the justification or explanation as to why the action associated with each price component has occurred. All actions may have an associated reason.	0..1	{ref 1} 55
	priceTargetMarketSub-division			Item Price Type. Price Target Market Subdivision. ISO3166_2_ Code	The code for country sub-division used to indicate the geo-political subdivision of the target market (=country ).	0..*	{ref 1} 64
	priceTypeApplicationSequence			Item Price Type. Price Type Application Sequence.	The order in which the value associated with a price type is applied in	1..1	{ref 1} 78



## Business Solution Design

				Integer_ Numeric	the process of calculating the net invoice price.		
	priceTypeCode			Item Price Type. Price Type Code. Text	A code assigned to identify the kind or class of a price component.	1..1	
	priceTypeDescription			Item Price Type. Price Type Description. Text	Text used to provide an additional description of the price component.	0..1	{ref 1} 58
	shipFrom			Item Price Type. Ship From. GLN_ Identifier	Identifies the origin location from where the goods will be shipped.	0..*	{ref 1} 66
	shipTo			Item Price Type. Ship To. GLN_ Identifier	The location destination to which goods will be shipped.	0..*	{ref 1} 65
	suggestedUnitRetailPrice			Item Price Type. Suggested Unit Retail Price. Amount	The retail (to consumer) price as suggested by the manufacturer. This is normally used to establish a proposed value for the trade item for marketing purposes. May or may not appear on the package.	0..1	{ref 1} 76
		None	BracketQualifier	Item Price Type. Association. Bracket Qualifier	Provides qualifiers required for eligibility for a price type of bracket.	0..*	
		parentCatalogueItem	CatalogueItemReference	Item Price Type. Parent Catalogue Item. Catalogue Item Identification	A reference to another trade item that is higher in the hierarchal configuration than the item referenced in the Item depiction. Used to vary the price of an item based on a higher level component in a hierarchal configuration.	0..1	{ref 1} 56
		itemPriceTypeSegmentIdentification	EntityIdentification	Item Price Type. Item Price Type Segment Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify a price component associated with an item.	1..1	{ref 1} 53

## Business Solution Design

		targetCondition	EntityIdentification	Item Price Type. Target Condition. Entity Identification	A reference to a previous Condition Identification that was used to define the Bracket Qualifiers - references back to the summary condition Identification.	0..1	{ref 1} 68
		targetPriceType	EntityIdentification	Item Price Type. Target Price Type. Entity Identification	A reference to a previous Price Type Identification that was used to define a component that this price is associated with.	0..1	{ref 1} 69
		None	PricePerformanceRequirementInformation	Item Price Type. Association. Price_Performance Requirement Information	Provides performance requirements for a price type.	0..*	
		None	PriceValueInformation	Item Price Type. Association. Price Value Information	Provides the numeric values associated with a price for example 5 percent.	1..1	
		None	ReferenceDocumentInformation	Item Price Type. Association. Reference Document Information	Provides reference information related to a given price for example a contract number.	0..1	
		priceTypeEffectiveEndDate	SegmentEffectiveEndDate	Item Price Type. Price Type Effective End Date. Segment Effective End Date	Provides end date details for a given price type.	0..*	{ref 1} 61
		priceTypeEffectiveStartDate	SegmentEffectiveStartDate	Item Price Type. Price Type Effective Start Date. Segment Effective Start Date	Provides details on start dates for a given price type.	1..*	{ref 1} 59
PricePerformanceRequirementInformation				Price_Performance Requirement Information. Details	A list of requirements which are types of price components to be met to receive a monetary value.		

## Business Solution Design

	performanceRequirementDescription			Price_ Performance Requirement Information. Performance Requirement Description. Text	A string of characters used to describe additional or more specific requirements to be met in order to receive a monetary value.	0..1	{ref 1} 83
		None	PricePerformanceRequirementOption	Price_ Performance Requirement Information. Association. Price Performance Requirement Option	Associates a performance requirement option and or dates with a performance requirement.	0..1	
PricePerformanceRequirementOption				Price Performance Requirement Option. Details	Standardized list of requirements which are types of price components to be met to receive a monetary value.		
	performanceRequirementEndDateTime			Price Performance Requirement Option. Performance Requirement End Date Time. Date Time	A date indicating the ending of a period during which the performance requirements should be met.	1..1	{ref 1} 86
	performanceRequirementOption			Price Performance Requirement Option. Performance Requirement Option. Text	Standardized list of requirements which are types of price components to be met to receive a monetary value.	0..1	{ref 1} 82
	performanceRequirementStartDateTime			Price Performance Requirement Option. Performance Requirement Start Date Time. Date Time	A date indicating the beginning of a period during which the performance requirements should be met.	1..1	{ref 1} 85
PriceSynchronisationCondition				Price Synchronisation Condition. Details	A price synchronisation message segment used to depict non- line-item conditions and summary conditions.		

## Business Solution Design

	conditionActionCode			Price Synchronisation Condition. Condition Action Code. Segment Action_ Code	A code assigned by the Information Provider to indicate to the Party Receiving Private Data, the reason for sending the price information contained within the specified segment within the Price Synchronisation Message. The Party Receiving Private Data is able to use this code to determine the nature of the action associated with each condition within each condition segment. For example the addition of a new record, the modification of an existing record or the correction of an existing record.	1..1	{ref 1} 29
	conditionApplicationSequence			Price Synchronisation Condition. Condition Application Sequence. Numeric	The order in which the value associated with a summary condition type of allowance or charge, is applied in the process of calculating the net invoice price.	0..1	{ref 1} 42
	conditionDescription			Price Synchronisation Condition. Condition Description. Text	Text used to provide an additional description of the condition.	1..1	{ref 1} 35
	conditionType			Price Synchronisation Condition. Condition Type. Text	Condition types are general classifications for a given condition. The treatment of the values in a price calculation are determined by the Condition Type.	1..1	{ref 1} 34
	conditionValueBasisQuantity			Price Synchronisation Condition. Condition Value	The base amount used for a condition in the case or a rate for exam-	0..1	{ref 1} 40

## Business Solution Design

				Basis. Measure	ple \$10 per '100' yards where 100 yards is the value basis.		
		None	BracketQualifier	Price Synchronisation Condition. Association. Bracket Qualifier	Provides conditions for being qualified for a given bracket.	0..*	
		None	ConditionTargetEntity	Price Synchronisation Condition. Association. Condition Target Entity	Provides an item of grouping of items associated with a price condition	0..1	{ref 1} 43
		None	ConditionValueInformation	Price Synchronisation Condition. Association. Condition Value Information	Provides a percentage or value associated with a given condition for example 15 percent.	0..1	
		priceSynchronisationConditionIdentification	EntityIdentification	Price Synchronisation Condition. Price Synchronisation Condition Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify a summary condition or an item condition of type bracket.	1..1	{ref 1} 28
		conditionEffectiveEndDate	SegmentEffectiveEndDate	Price Synchronisation Condition. Condition Effective End Date. Segment Effective End Date	Provides the effective end date and context for a price synchronisation condition	0..*	{ref 1} 32
		conditionEffectiveStartDate	SegmentEffectiveStartDate	Price Synchronisation Condition. Condition Effective Start Date. Segment Effective Start Date	Provides the effective start date and context for a price synchronisation condition.	1..*	{ref 1} 30
PriceSynchronisationDocument				Price Synchronisation Document. Details	An electronic document used to synchronise pricing information including pricing relationship, pricing elements and item price depiction between trading part-		

## Business Solution Design

					ners in order to facilitate an invoice amount equal to the expected payment amount equal to the actual payment.		
	informationProvider			Price Synchronisation Document. Information Provider. GLN_ Identifier	The party who owns the data.	1..1	{ref 1} 6
	partyReceivingPrivateData			Price Synchronisation Document. Party Receiving Private Data. GLN_ Identifier	Party, which is authorized to view, use, download the data provided by a Data Source.	1..1	{ref 1} 7
	priceDocumentType			Price Synchronisation Document. Price Document Type. Price Document Type_ Code	A code assigned by the Information Provider to indicate to the Party Receiving Private Data, the intended use or purpose of sending the Price Synchronisation Message. The Party Receiving Private Data is able to use this code to determine how to process the information contained within the message. For example, initial load of data, re-send of previously sent data or ongoing data synchronisation.	0..1	{ref 1} 8
		None	Document	Price Synchronisation Document. Association. Document	Not Applicable	1..1	
		priceSynchronisationDocumentIdentification	EntityIdentification	Price Synchronisation Document. Price Synchronisation Document Identification. Entity Identifica-	Within a given price synchronization relationship, a number assigned by the Source Data Pool to uniquely identify each instance of a Price Syn-	1..1	{ref 1} 4

## Business Solution Design

				tion	chronization Message sent from the Source Data Pool to the Party Receiving Private Data. The number is unique within each Price synchronization relationship.		
		priceSynchronisationRelationshipIdentification	EntityIdentification	Price Synchronisation Document. Price Synchronisation Relationship Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify each price synchronization relationship that exists between the Information Provider and the Party Receiving Private Data. Each Price Synchronisation Message can only contain price information related to a single price synchronization relationship.	1..1	{ref 1} 5
		None	ItemDepictionQualifier	Price Synchronisation Document. Association. Item Depiction Qualifier	Provides one or more item depictions for a price synchronisation document.	0..*	
		None	PriceSynchronisationCondition	Price Synchronisation Document. Association. Price Synchronisation Condition	Provides one or many price synchronisation conditions for a price synchronisation document.	0..*	
		None	PriceSynchronisationRelationship	Price Synchronisation Document. Association. Price Synchronisation Relationship	Provides the depiction of a price synchronisation relationship for a price synchronisation document.	0..1	
PriceSynchronisationRelationship				Price Synchronisation Relationship. Details	A message segment used to establish a price synchronisation relationship between trading partners.		
	priceSynchronisationRe-			Price Synchronisa-	The name assigned by	0..1	{ref 1}

## Business Solution Design

	relationshipName			Price Synchronisation Relationship. Price Synchronisation Relationship Name. Text	the buyer and seller to their price sync relationship.		20
	relationshipActionCode			Price Synchronisation Relationship. Relationship Action Code. Code	Indicates how the trading partner applies the information in the specified segment.	1..1	{ref 1} 11
	relationshipCurrency-Code			Price Synchronisation Relationship. Relationship Currency Code. ISO4217_ Code	A code used to indicate the system of money used within a particular country by the trading partners to conduct their commercial transactions.	1..1	{ref 1} 19
	relationshipEffectiveEndDateTime			Price Synchronisation Relationship. Relationship Effective End Date Time. Date Time	The day on which the price synchronization relationship ends.	0..1	{ref 1} 13
	relationshipEffectiveStartDateTime			Price Synchronisation Relationship. Relationship Effective Start Date Time. Date Time	The day on which the price synchronization relationship commences.	1..1	{ref 1} 12
	relationshipTradeChannel			Price Synchronisation Relationship. Relationship Trade Channel. Trade Channel_ Code	Used to specify how the trading partners within a price synchronization relation agree to define the distribution or marketing segmentation of products, customers and geographic areas into common groups that are supplied, serviced and measured in similar ways. The Trade Channel may be defined in the context of the Party Receiving Private Data.	1..1	{ref 1} 18
	targetMarketCountry-Code			Price Synchronisation Relationship. Target Market Country Code.	The target market code indicates the country level or higher geographical definition in	1..1	{ref 1} 17



## Business Solution Design

				ISO3166_1_ Code	which the price information is applicable.		
		priceSynchronisationRelationshipIdentification	EntityIdentification	Price Synchronisation Relationship. Price Synchronisation Relationship Identification. Entity Identification	Identifies a unique buyer-seller price sync relationship generated by the data source.	1..1	{ref 1} 10
		None	IncotermInformation	Price Synchronisation Relationship. Association. Incoterm	Provides incoterm details applicable to a trading partner relationship.	0..*	
		businessLocation	PartyIdentification	Price Synchronisation Relationship. Business Location. Party Identification	An entity that belongs to the Party Receiving Private Data, who is the intended recipient of the price information contained within the Price Synchronization Message.	1..1	{ref 1} 16
		informationProvider	PartyIdentification	Price Synchronisation Relationship. Information Provider. Party Identification	The party who owns the data.	1..1	{ref 1} 14
		partyReceivingPrivateData	PartyIdentification	Price Synchronisation Relationship. Party Receiving Private Data. Party Identification	Party, which is authorized to view, use, download the data provided by a Data Source.	1..1	{ref 1} 15
PriceValueInformation				Price Value Information. Details	None		
	priceBasisQuantity			Price Value Information. Price Basis Quantity. Measure	Price Basis Quantity qualifies Price with a 'Price Per' quantity. This must include a unit of measure to describe what the price and price quantity applies to, such as, a price of \$100 could apply to 1 case of prod-	1..1	{ref 1} 74

## Business Solution Design

					uct or to 25 Kilos. Price Basis Quantity includes a Unit of Measure.		
	priceValue			Price Value Information. Price Value.Total_Quantity	Associates a percent or integer value with a price value.	1..1	{ref 1} 70
	priceValueCap			Price Value Information. Price Value Cap. Float_ Numeric	A quantity or measurement associated with the price value qualifier to limit the calculation of rate to a specified maximum amount. This would be used where a trading partner sets a maximum value for an offer.	0..1	{ref 1} 71
	priceValueQualifier			Price Value Information. Price Value Qualifier. Code	A code assigned to identify the basis on which a specific price value is acted upon. For example, if the Price Value was 2%, the Price Value Qualifier would be 'percent'.	0..1	{ref 1} 73
	priceValueType			Price Value Information. Price Value Type. Price Value Type_ Code	A classification of the price component used to determine how to apply the amount for example value, rate or percent.	1..1	{ref 1} 72
ReferenceDocumentInformation				Reference Document Information. Details	This class enables the input of a reference document (e.g. contract ) for a specific condition.		
	referenceDocumentIdentifier			Reference Document Information. Reference Document_ Identifier. Identifier	Identifier that provides a link to further detail on the price condition, for example an associated contract between trading partners.	1..1	{ref 1} 79
	referenceDocumentDescription			Reference Document Information.	A free form text field used to describe a con-	0..1	{ref 1} 80

## Business Solution Design

				Reference Document Description. Text	tract or other document which contains more information about agreements made regarding a condition.		
SegmentEffectiveEndDateInformation				Segment Effective End Date. Details	The effective end date and associated context (e.g. last order date) for a condition.		
	effectiveEndDateTime			Segment Effective End Date. Effective End Date Time. Date Time	A date\time used to indicate when the component depicted in the segment is no longer available for use.	1..1	
	effectiveEndDateContextCode			Segment Effective End Date. Effective End Date Time Context. Effective End Date Context_Code	An associated event which gives significance to the effective start date for a segment for example first order date.	1..1	
SegmentEffectiveStartDateInformation				Segment Effective Start Date. Details	The start date and applicable context for the start date (first order date) for a condition type.		
	effectiveStartDateTime			Segment Effective Start Date. Effective Start Date Time. Date Time	The date on which the price synchronisation component begins.	1..1	
	effectiveStartDateContextCode			Segment Effective Start Date. Effective Start Date Time Context. Code	An associated event which gives significance to the effective start date for a segment for example first order date.	1..1	

## Business Solution Design

### Price Synchronisation Confirmation

Class (ABIE)	Attribute (BBIE)	Association (ASBIE)	Secondary Class	Official Dictionary Entry Name	Definition	Multiplicity	Related Requirements
PriceSynchronisationConfirmation				Price Synchronisation Confirmation. Details	The electronic communication from the Data Recipient to the Data Source indicating what action has been taken on the price synchronisation relationship, condition or price segment.		
	dataRecipient			Price Synchronisation Confirmation. Data Recipient. GLN_ Identifier	The party receiving the private data (for example, retailer). This information is taken from the price synchronization header of the price message to which this confirmation is responding.	1..1	{ref 1} 2
	dataSource			Price Synchro-	The informa-	1..1	{ref

## Business Solution Design

				nisation Confirmation. Data Source. GLN_Identifier	tion provider of the Price Synchronization message (for example, supplier). This is taken from the price synchronization header of the price message to which this confirmation is responding.		1} 1
		None	Document	Price Synchronisation Confirmation. Association. Document	Not Applicable	1..1	
		priceSynchronisationConfirmation-Identification	EntityIdentification	Price Synchronisation Confirmation. Price Synchronisation Confirmation Identification. Entity Identification	Uniquely identifies the Price Synchronisation Confirmation	1..1	
		priceSynchronisationDocument-Identification	EntityIdentification	Price Synchronisation Confirmation. Price Synchronisation Document Identification. Entity Identification	Within a given price synchronization relationship, a number assigned by the Source Data Pool to uniquely identify each instance of a Price Synchronization Message sent from the	1..1	{ref 1} 3

## Business Solution Design

					Source Data Pool to the Party Receiving Private Data. The number is unique within each Price synchronization relationship (from the price synchronization header of the price message to which this confirmation is responding).		
		priceSynchronisationRelationship-Identification	EntityIdentification	Price Synchronisation Confirmation. Price Synchronisation Relationship Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify each price synchronization relationship that exists between the Information Provider and the Party Receiving Private Data. Each Price Synchronisation Message can only contain price information related to a	1..1	{ref 1} 4

## Business Solution Design

					single price synchronization relationship (from the price synchronization header of the price message to which this confirmation is responding).		
		None	PriceSynchronisationSegmentConfirmation	Price Synchronisation Confirmation. Association. Price Synchronisation Segment Confirmation	Provides the confirmation status and the applicable price synchronization segment.	1..*	
PriceSynchronisationConfirmationStatusReason				Price Synchronisation Confirmation Status Reason. Details	Provides further details regarding the synchronization status for the price synchronization relationship, condition or price segment including the reason for the status, the action needed and any specific attribute.		
	actionNeeded			Price Synchronisation Confirmation Status Reason. Action Needed.	Identifies the type of action the data source needs to take in	1..1	{ref 1} 10

## Business Solution Design

				Text	order to resolve the data recipient's issue.		
	confirmationStatusReasonCode			Price Synchronisation Confirmation Status Reason. Confirmation Status Reason Code. Text	Identifies the type issue the data recipient has with the value communicated in the attribute name.	1..1	{ref 1} 9
	priceAttributeName			Price Synchronisation Confirmation Status Reason. Price Attribute Name. Text	Name of the attribute in the Price Synchronization message.	1..1	{ref 1} 7
	priceAttributeValue			Price Synchronisation Confirmation Status Reason. Price Attribute Value. Text	Value sent in the price synchronisation message that is associated with the attribute name.	1..1	{ref 1} 8
PriceSynchronisationSegment-Confirmation				Price Synchronisation Segment Confirmation. Details	The synchronisation status for the price synchronisation relationship, condition or price segment.		
	priceSynchronisationConfirmationStatus			Price Synchronisation Segment Confirmation. Price Synchronisation Confirmation Status.	Describes the data recipient's action taken on the information contained in a specific	1..1	{ref 1} 6



## Business Solution Design

				Synchronisation Confirmation Status_Code	segment of the price synchronization message.		
		itemPriceTypeSegmentIdentification	EntityIdentification	Price Synchronisation Segment Confirmation. Choice_Item Price Type Segment Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify a price component associated with an item (from the item price type segment of the price message to which this confirmation is responding).	1..1	{ref 1} 5
		priceSynchronisationConditionIdentification	EntityIdentification	Price Synchronisation Segment Confirmation. Choice_Price Synchronisation Condition Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify a summary condition or an item condition of type bracket (from the condition segment of the price message to which this confirmation is responding).	1..1	{ref 1} 5

## Business Solution Design

		priceSynchronisationRelationship- Identification	EntityIdentification	Price Synchronisation Segment Confirmation. Choice_ Price Synchronisation Relationship Identification. Entity Identification	A string of characters assigned by the Information Provider to uniquely identify each price synchronization relationship that exists between the Information Provider and the Party Receiving Private Data. Each Price Synchronisation Message can only contain price information related to a single price synchronization relationship (from the relationship segment of the price message to which this confirmation is responding).	1..1	{ref 1} 5
		None	PriceSynchronisationConfirmationStatusReason	Price Synchronisation Segment Confirmation. Association. Price Synchronisation Confirmation Status	Provides the reason, action required and relevant attributes and values connected with a syn-	0..*	

## Business Solution Design

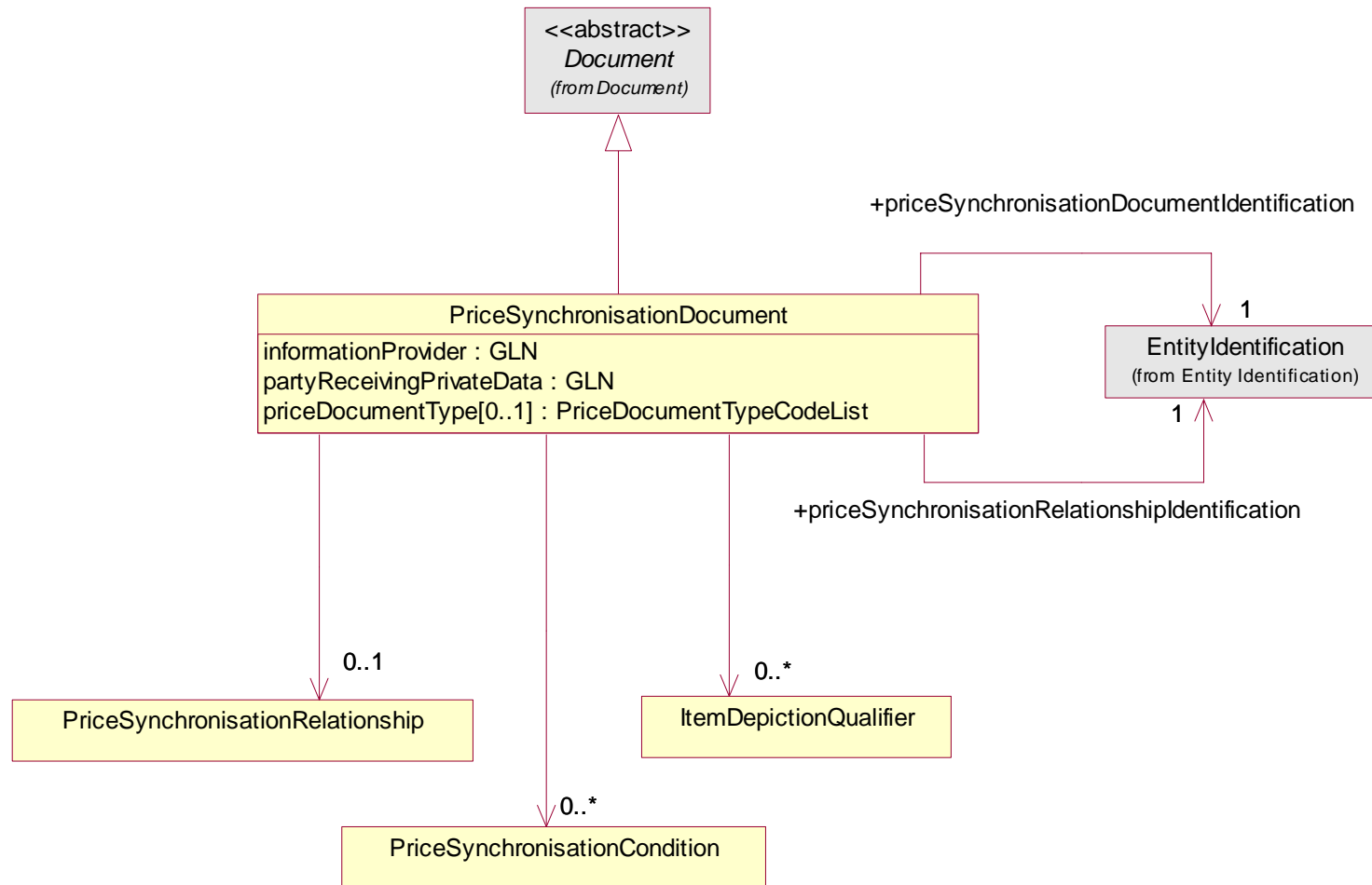
---

---

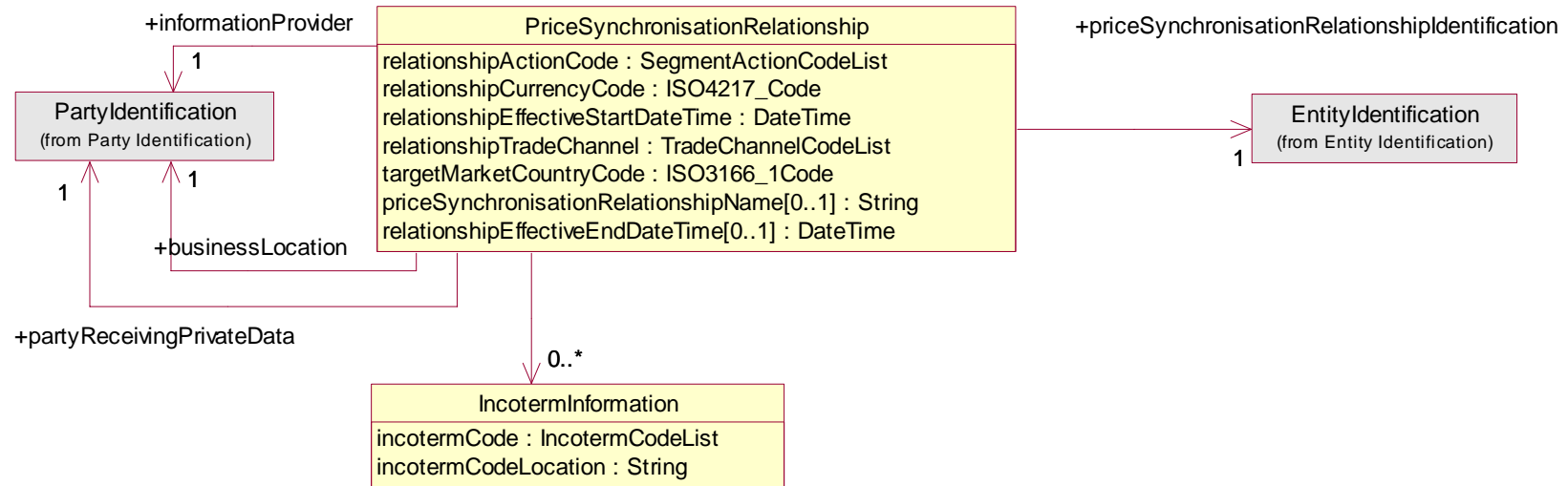
				Reason	chronisation status.		
--	--	--	--	--------	-------------------------	--	--

1.5.3 Class Diagrams

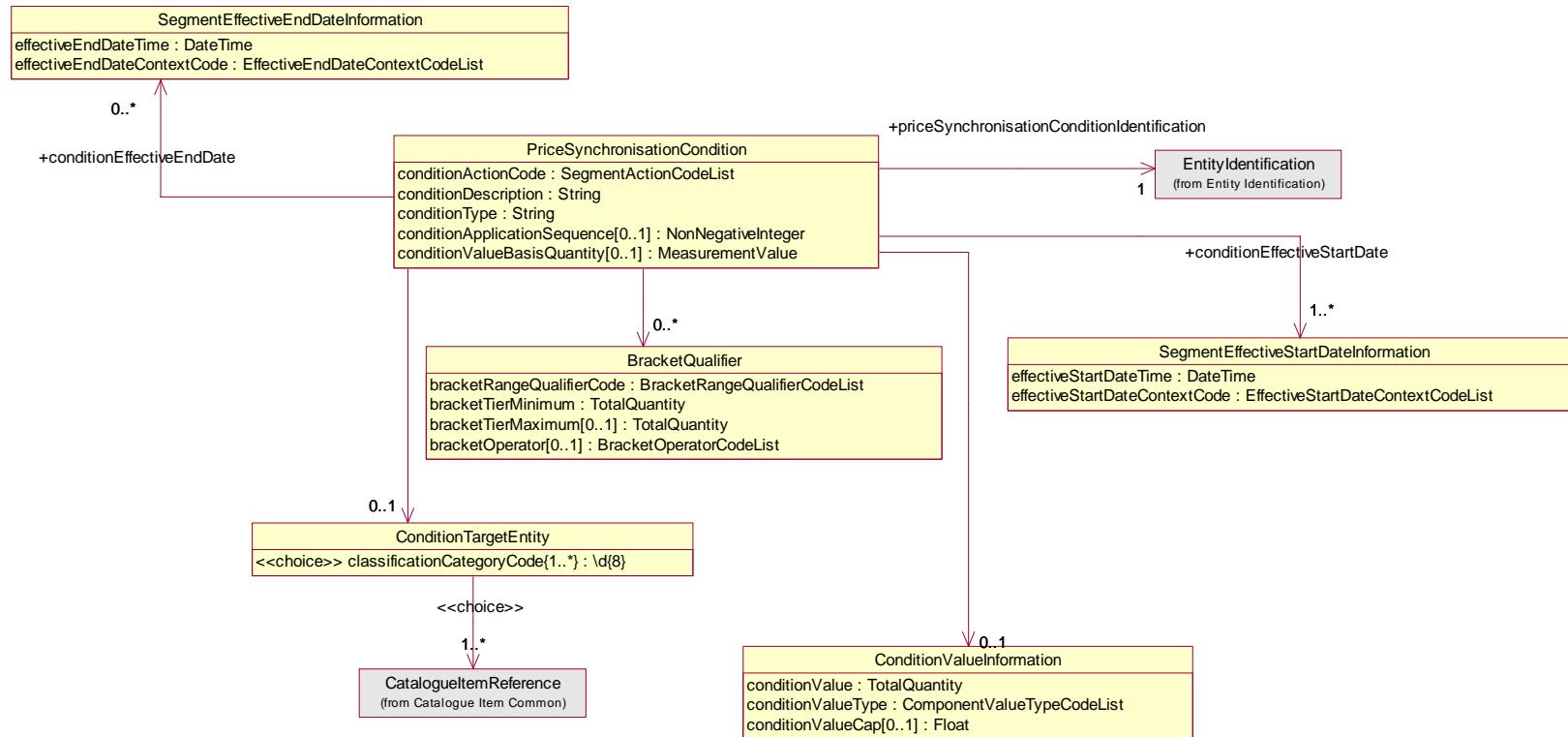
1.5.3.1 Class Diagrams: Price Synchronisation Document



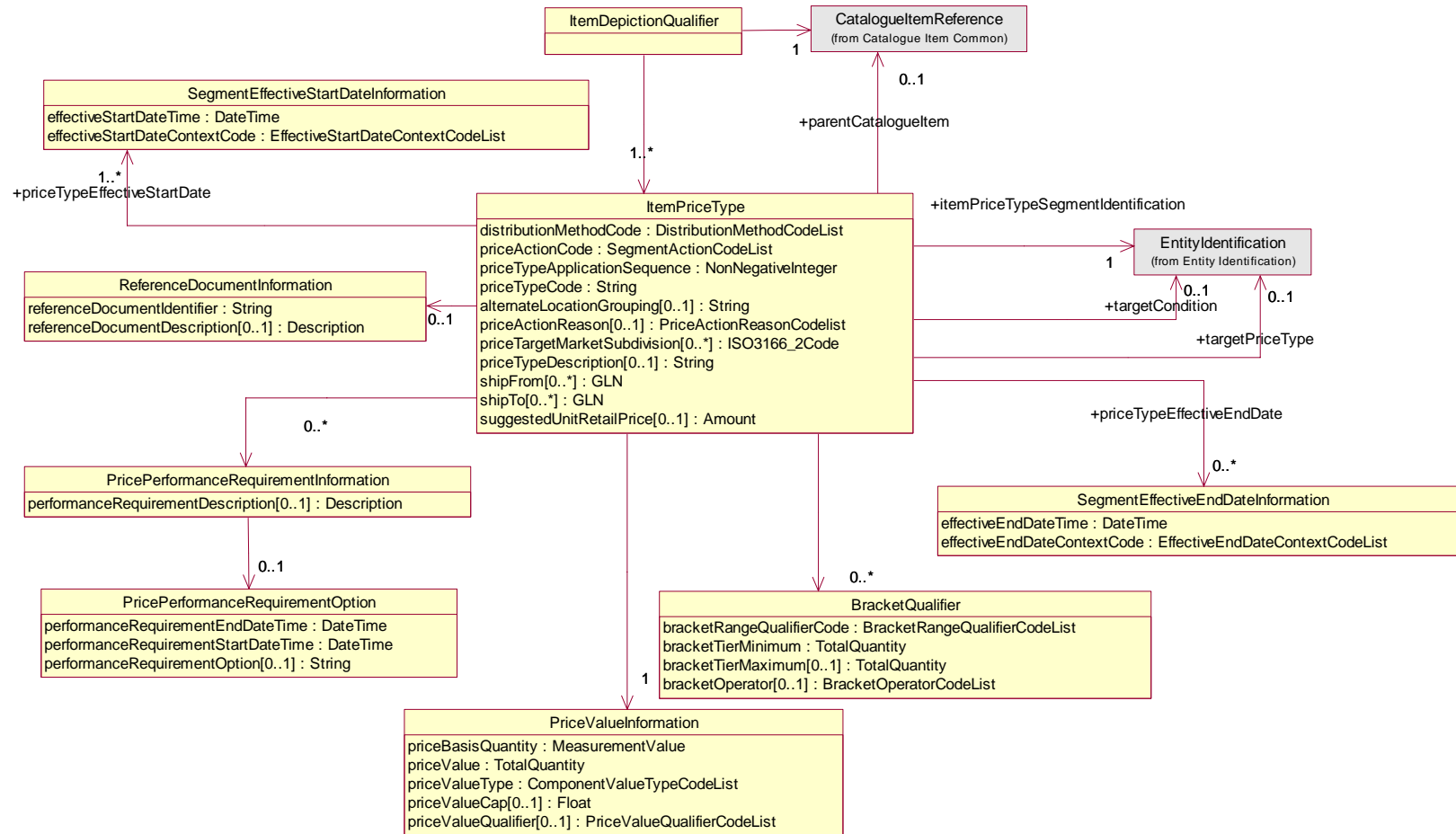
# Business Solution Design



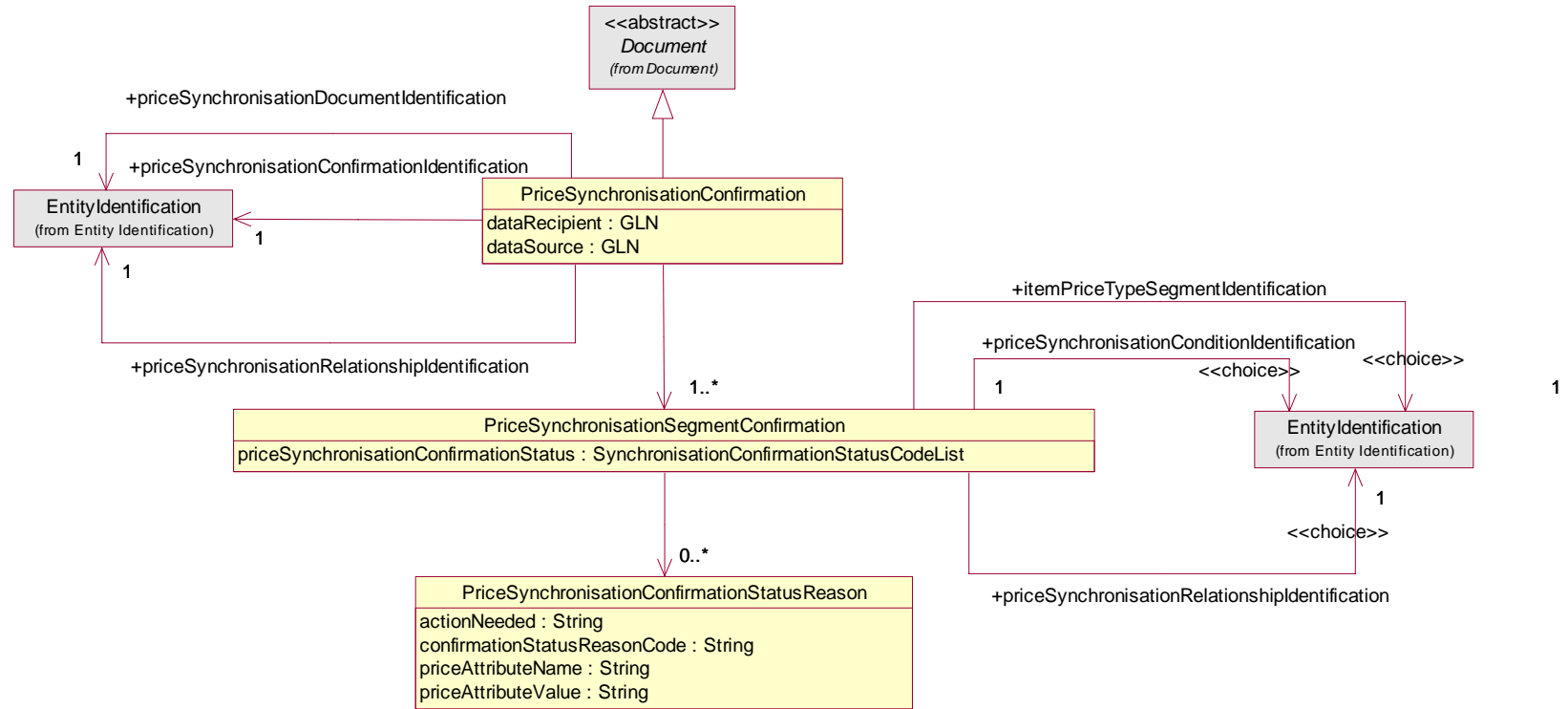
# Business Solution Design



# Business Solution Design

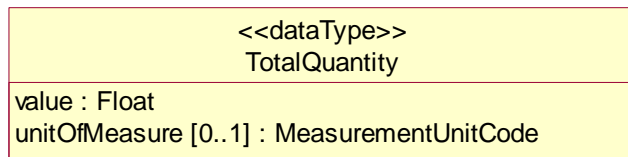


## 1.5.3.2 Class Diagrams: Price Synchronisation Confirmation





### 1.5.3.3 Class Diagrams: Common: To be removed from BMS Post Development.



## Business Solution Design

---

### 1.5.4 Code Lists

Code List Name	Code List Description
BracketOperatorCodeList	Determines the logical relationship between multiple bracket ranges.
Code Name	Code Description
AND	All previous qualifiers plus the current record.
OR	Any one qualifier but not more than one.

Code List Name	Code List Description
BracketRangeQualifierCode-List	Determines the type of range used to determine a bracket qualifier.
Code Name	Code Description
AMOUNT_RANGE	A range value with a currency.
MEASUREMENT_RANGE	A range value using a Unit Of Measure
RANGE	A numeric range.

Code List Name	Code List Description
ComponentValueTypeCode-List	A code which determines how the value associated with a price component is treated for a price calculation.
Code Name	Code Description
PERCENT	A part of a whole expressed in hundredths
RATE	A fixed ratio between two things.
VALUE	A numerical quantity that is assigned or is determined by calculation or measurement.

Code List Name	Code List Description
DistributionMethodCodeList	The means by which the supplier and retailer agree upon that at what point in the supply chain the supplier makes the goods available to the retailer.
Code Name	Code Description
CD	Cross Dock: The supplier picks and packs the goods per retail store location and delivers the goods to the retailers' distribution centre for transfer across the loading dock to local delivery vehicles. In this

## Business Solution Design

---

---

	case the goods are not stored in the distribution centre.
D2C	Direct to Consumer: The supplier delivers the goods direct to the end consumer's location.
DC	Distribution Centre: The supplier delivers the goods to the retailers' distribution centre. The goods are typically warehoused in the distribution centre prior to distribution.
DSD	Direct Store Delivery: The supplier delivers the goods direct to the retailers' retail store location.
FG	Factory Gate: The supplier makes the goods available to the retailer at the supplier's loading dock (factory gate). The retailer assumes responsibility for planning, scheduling, collection and transportation of the goods.
UNS	Unspecified

## Business Solution Design

---

---

Code List Name	Code List Description
EffectiveEndDateContextCodeList	A code to indicate the related circumstances associated with the effective end date.
Code Name	Code Description
AD_END_DATE	The end date for an advertisement for a given product.
LAST_DELIVERY_DATE	The day on which the last delivery is made.
LAST_ORDER_DATE	It indicates the latest date that an order can be placed for the trade item.
LAST_SHIP_DATE	It indicates the latest date that the trade item can be shipped. This is independent of any specific ship-from location.

Code List Name	Code List Description
EffectiveStartDateContextCodeList	A code to indicate the related circumstances associated with the effective start date.
Code Name	Code Description
AD_START_DATE	The start date for an advertisement for a given product.
FIRST_DELIVERY_DATE	The day on which the first delivery is made. Also know as First Arrival Date.
FIRST_ORDER_DATE	It indicates the earliest date that an order can be placed for the trade item.
FIRST_SHIP_DATE	It indicates the earliest date that the trade item can be shipped. This is independent of any specific ship-from location.

## Business Solution Design

---

---

<b>Code List Name</b>	<b>Code List Description</b>
IncotermCodeList	Incoterms is an abbreviation for International Commercial Terms. The International Chamber of Commerce created, owns and manages the Incoterms and their definitions.
<b>Code Name</b>	<b>Code Description</b>
CFR	Cost and Freight
CIF	Cost, Insurance and Freight
CIP	Carriage and Insurance Paid To
CPT	Carriage Paid To
DAF	Delivered at Frontier
DDP	Delivered Duty Paid
DDU	Delivered Duty Unpaid
DEQ	Delivered Ex Quay
DES	Delivered Ex Ship
EXW	Ex Works
FAS	Free Alongside Ship
FCA	Free Carrier
FOB	Free On Board

## Business Solution Design

---

Code List Name	Code List Description
PriceActionReasonCode	The reason as to why an action related to a Price Type has occurred.
Code Name	Code Description
NI	The introduction of a new item.
PD	Price decrease.
PI	A price increase.
RE	Range extension.
SC	Size change (Pack or Pallet).
TPR	Temporary price reduction.

Code List Name	Code List Description
PriceDocumentTypeCodeList	A code assigned by the information provider needed to indicate to the party receiving private data the intended use or purpose of sending the price synchronization message. The party receiving private data is able to use this code to determine how to process the information contained within the message.
Code Name	Code Description
INITIAL_LOAD	The sending of pricing information for the first time.
RELOAD	Sending all current and known future pricing. This is used to start over by replacing previously synchronised information.
RESEND	Indicates that the message is used to recover a lost or missing message.
RESTART	The status used when a data recipient had rejected an item's pricing but wishes to resume price synchronisation.

Code List Name	Code List Description
PriceValueQualifier	A code assigned to identify the basis on which a specific price value is acted upon. For example, if the Price Value was 2%, the Price Value Qualifier would be 'percent', if the Swell Allowance was 1 Euro, the Condition Value Qualifier would be 'monetary amount'
Code Name	Code Description
MONETARY_AMOUNT	An amount of or relating to money.
PERCENT	One part in a hundred.

## Business Solution Design

---

---

Code List Name	Code List Description
SegmentActionCodeList	
Code Name	Code Description
ADD	Used to signify that a Data Source is seeking to synchronise new data with the Data Recipient
CORRECT	Used to error correct or change the values of mandatory key attributes or an attribute where the change results in material financial impact.
DELETE	Used to remove one or many iterations of an existing segment.
MODIFY	Used to modify or change the values of any of the optional attributes within the segment.
NO_ACTION	No change or correction is being made to the segment.

Code List Name	Code List Description
SynchronisationConfirmationStateCodeList	The four states of synchronisation reflected by a Recipient Data Pool.
Code Name	Code Description
ACCEPTED	Data has been received by the Recipient, but no business decision has been made on the data.
REJECTED	The recipient requests that no further updates are desired. Data will no longer be synchronized or updates will no longer be provided.
REVIEW	A request to the data source to "review" their data because the data recipient has received discrepant data which they cannot synchronise.
SYNCHRONISED	Data is integrated, in sync and added to the synchronisation list.

## Business Solution Design

---



---

Code List Name	Code List Description
TradeChannelCodeList	A grouping of entities by common business model concentration to form a trade channel or industry sector.
Code Name	Code Description
CONVENIENCE	Small format retail store often outside or annexed to a gas/fuel station.
DRUG	Organisations or departments engaged in retailing prescription or non-prescription drugs and medicines.
FOOD_SERVICE	Trade channel that sells prepared food, for example restaurants, hotels, clubs.
GROCERY	Organisations or departments primarily engaged in retailing a general line of food products.
HARD_LINES	Organisations or departments primarily engaged in retailing a general line of hardware items, such as tools and builders' hardware.
HOME_GOODS	Not available
INDUSTRIAL	Not Available
INSTITUTIONAL	Not Available
MASS_MERCHANDISING	Organisations or departments primarily engaged in high volume, fast turnover variety of merchandise.
MILITARY	Sale of items to the military.
UNSPECIFIED	Trade Channel unknown or not relevant.
VENDING	The retailing merchandise through vending machines.



## Business Solution Design

---



---

### 1.6 Business Document Example

### 1.7 Implementation Considerations

### 1.8 Testing

#### 1.8.1 Pass / Fail Criteria

Number	Test Criteria	Related Requirement	Design Element	Pass Criteria	Fail Criteria
1					

#### 1.8.2 Test Data

##### 1.8.2.1 Price Synchronisation Document

Attribute	Value
PriceSynchronisationDocument	
<ul style="list-style-type: none"> <li>• informationProvider</li> </ul>	0012345000010
<ul style="list-style-type: none"> <li>• partyReceivingPrivateData</li> </ul>	0056345000022
<ul style="list-style-type: none"> <li>• priceDocumentType</li> </ul>	INITIAL_LOAD
<ul style="list-style-type: none"> <li>• priceSynchronisationDocumenta- tionIdentification <ul style="list-style-type: none"> <li>○ uniqueCreatorIdentification</li> <li>○ contentOwner</li> </ul> </li> </ul>	0012345000010
<ul style="list-style-type: none"> <li>○ contentOwner</li> </ul>	WG-000001
<ul style="list-style-type: none"> <li>• priceSynchronisationRelationship- Identification <ul style="list-style-type: none"> <li>○ uniqueCreatorIdentification</li> <li>○ contentOwner</li> </ul> </li> </ul>	0012345000010
<ul style="list-style-type: none"> <li>○ contentOwner</li> </ul>	WG-000002
PriceSynchronisationRelationship	
<ul style="list-style-type: none"> <li>• PriceSynchronisationRelationshipIden- tification <ul style="list-style-type: none"> <li>○ uniqueCreatorIdentification</li> <li>○ contentOwner</li> </ul> </li> </ul>	0012345000010
<ul style="list-style-type: none"> <li>○ contentOwner</li> </ul>	WG-000003
<ul style="list-style-type: none"> <li>• priceSynchronisationRelationship- Name</li> </ul>	Nana Corporation Food Service
<ul style="list-style-type: none"> <li>• relationshipActionCode</li> </ul>	ADD
<ul style="list-style-type: none"> <li>• relationshipCurrencyCode</li> </ul>	USD
<ul style="list-style-type: none"> <li>• relationshipEffectiveEndDateTime</li> </ul>	2007-01-10T12:00:01.000
<ul style="list-style-type: none"> <li>• relationshipEffectiveStartDateTime</li> </ul>	2006-01-10T12:00:01.000
<ul style="list-style-type: none"> <li>• relationshipTradeChannel</li> </ul>	FOOD_SERVICE
<ul style="list-style-type: none"> <li>• targetMarketCountryCode</li> </ul>	US
<ul style="list-style-type: none"> <li>• informationProvider</li> </ul>	0012345000010
<ul style="list-style-type: none"> <li>• businessLocation</li> </ul>	0012345000010
<ul style="list-style-type: none"> <li>• partyReceivingPrivateData</li> </ul>	0056345000022
IncotermInformation	
<ul style="list-style-type: none"> <li>• incotermCode</li> </ul>	CFR
<ul style="list-style-type: none"> <li>• incotermCodeLocation</li> </ul>	Port Charlotte

## Business Solution Design

PriceSynchronisationCondition	
• conditionActionCode	ADD
• conditionApplicationSequence	1
• conditionDescription	Extremely Large Order Bracket
• conditionType	BRACKET
• conditionValueBasisQuantity	10,000 YD
PriceSynchronisationConditionIdentification	
o uniqueCreatorIdentification	0012345000010
o contentOwner	WG-000007
conditionEffectiveStartDateInformation	
• effectiveStartDateTime	2006-01-10T12:00:01.000
• effectiveStartDateContextCode	FIRST_DELIVERY_DATE
conditionEffectiveEndDateInformation	
• effectiveEndDateTime	2007-01-10T12:00:01.000
• effectiveEndDateContextCode	LAST_DELIVERY_DATE
BracketQualifier	
• bracketRangeQualifierCode	MEASUREMENT_RANGE
• bracketTierMaximum	500,000 YD
• bracketTierMinimum	100,000 YD
ConditionValueInformation	
• conditionValue	2,000.00
• conditionValueType	RATE
• conditionValueCap	10,000
ConditionTargetEntity	
• CatalogueItemReference	
o gtin	06110123456784
o dataSource	0012345000010
o targetMarketCountryCode	US
ItemPriceType	
• alternateLocationGrouping	72436437
• distributionMethodCode	CD
• priceActionCode	ADD
• priceActionReason	NI
• priceTargetMarketSubdivision	US-CA
• priceTypeApplicationSequence	1
• priceTypeCode	INTRODUCTORY_PRICE
• shipFrom	0012345000011
• shipTo	0056345000025
• suggestedUnitRetailPrice	30.00 USD
priceTypeEffectiveStartDateInformation	
• effectiveStartDateTime	2006-01-10T12:00:01.000
• effectiveStartDateContextCode	FIRST_DELIVERY_DATE
ReferenceDocumentationInformation	
• referenceDocumentIdentifier	123232334334
• referenceDocumentDescription	Contract dated 2006-07-01
PricePerformanceRequirementInformation	
• performanceRequirementEnd-Date DateTime	2007-01-10T12:00:01.000
• performanceRequirementOption	INSERT
• performanceRequirementStart-Date DateTime	2006-01-10T12:00:01.000
PriceValueInformation	
• priceBasisQuantity	100 YD

## Business Solution Design

• priceValue	10.00
• priceValueQualifier	MONETARY_AMOUNT
• priceValueType	VALUE
targetPriceType	
o uniqueCreatorIdentification	0012345000010
o contentOwner	WG-000005
itemPriceTypeSegmentIdentification	
o uniqueCreatorIdentification	0012345000010
o contentOwner	WG-000006

### 1.8.2.2 Price Synchronisation Confirmation

Attribute	Value
PriceSynchronisationConfirmation	
• dataRecipient	0012345000010
• dataSource	0056345000022
• priceSynchronisationConfirmation-Identification	
o uniqueCreatorIdentification	0056345000022
o contentOwner	WG-000013
• priceSynchronisationRelationship-Identification	
o uniqueCreatorIdentification	0012345000010
o contentOwner	WG-000002
• priceSynchronisationDocumentationIdentification	
o uniqueCreatorIdentification	0012345000010
o contentOwner	WG-000001
PriceSynchronisationSegmentConfirmation	
• PriceSynchronisationConditionIdentification	
o uniqueCreatorIdentification	0012345000010
o contentOwner	WG-000007
• priceSynchronisationConfirmation-Status	REVIEW
PriceSynchronisationConfirmationStatus	
• actionNeeded	Correct Value
• confirmationStatusReasonCode	Details not as agreed
• priceAttributeName	conditionValueBasisQuantity
• priceAttributeValueName	100,000 YD

### 1.9 Appendices

#### 1.10 Summary of Changes

*(Details changes to BMS for each version by BMS Section)*

<b>Change</b>	<b>BMS Ver- sion</b>	<b>Associated CR Number</b>
<ul style="list-style-type: none"><li>• Updated UC-9 Related Rule 8 and UC-10 Related Rule 10 to read “Bracket Qualifiers for a Price Type can be sent providing that the brackets have not been sent as standard brackets”.</li><li>• Changed Business Rule 18 in UC-9 and UC-10 to clarify the application of Allowances/Charges.</li><li>• Updated rule 9 in UC-9 and Rule 10 in UC-10 to clarify application of bracket qualifiers.</li></ul>	0.0.3	07-000149

### 2 2 Technical Solution Design

This section provides the specifications for the standards content ITRG approves. It is called the Technical Solution Design (TSD).

The Technical Solution Design contains:

- TSD Zip file Table of Contents
- Business Message Standard Section Technical Level GDD Report
- XSD (XML Schema Documents)
- XML Instance File and HTML Form View (XML and HTML files containing sample data specified in Section 1.6)

In the process of approving the Technical Solution Design, the ITRG will be provided the following artifacts:

- Any relevant Business Requirements Analysis Document (BRAD)
- Any relevant Business Requirements Document (BRD)
- Section 1 of Business Message Standard (Business Solution Design)
- Comment Resolution Template from Technical Public Review
- XML Test Report
- Change Request
- Other informative or reference documents