EEE Web Service

Contents

Installation 5
WSDL 5
Upgrading to 1.48+ - Important Configuration Changes 5
EEEWebService Usage 6
Common Methods 6
EEEVersion 6
GetBranchList 6
Layby Methods 7
ItemsToPromoteToCustomer 7
RetrieveLayby 8
RecordLaybyPayment 9
Customer Registration Methods 10
GetCustomDemographics 10
Customer Registration 14
Customer Confirmation 16
Customer Confirmation With Details 16
Customer Validate Login 17
Customer Offline Login 18
Customer Details 19
CustomerDetailsByCustomerCode 21
Customer DetailsWithParameters 21
CustomerForgotPassword 22
CustomerEmailExists 22
CustomerDetailsExtended 23
CustomerDetailsExtendedWithParameters 23
CustomerConfirmationWithDetailsExtended 23
Loyalty Rewards Methods 24
Customer Loyalty Card Details 24
LoyaltyGiftCardStatus 25
CustomerLoyaltyStatement 25
LoyaltyPointsTransfer 26

Customer Promotion Coupon Methods 27
CouponRedemption 27
CouponRedemptionById 27
CouponValid 28
CouponsAvailableToCustomer 29
Transaction Methods 30
GetTransactionPrices 30
Email Statistics Methods 31
Email Statistics 31
Website Synchronisation Methods 32
Overview 32
Product 32
Product_Get 32
Product_AcknowledgeGet 32
ProductExtended_Get 33
ProductExtended_AcknowledgeGet 33
ProductPromotion_Get 33
ProductPromotion_AcknowledgeGet 33
ProductPromotionPrice_Get 34
ProductPromotionPrice_AcknowledgeGet 34
Inventory 34
Inventory _Get 34
Inventory_AcknowledgeGet 35
Inventory_LookUp 35
Price 35
Price_Get 35
Price _AcknowledgeGet 36
Price_LookUp 36
Cost 36
Cost_Get 36
Cost _AcknowledgeGet 37
Colour 37
Colour_Get 37
Colour_AcknowledgeGet 37
Size 38

Size_Get 38
Size_AcknowledgeGet 38
Style 38
Style_Get 38
Style_AcknowledgeGet 38
Analysis1 39
Analysis1_Get 39
Analysis1_AcknowledgeGet 39
Analysis2 39
Analysis2_Get 39
Analysis2_AcknowledgeGet 40
Analysis3 40
Analysis3_Get 40
Analysis3_AcknowledgeGet 40
Analysis4 40
Analysis4_Get 40
Analysis4_AcknowledgeGet 41
Product Images 41
ProductImage_Get 41
ProductImage _AcknowledgeGet 41
Master Customers 42
MasterCustomer_Get 42
MasterCustomer_AcknowledgeGet 42
Promotion Customers 42
PromotionCustomer_Get 42
PromotionCustomer_AcknowledgeGet 43
Price Book 43
PriceBook_Get 43
PriceBook_AcknowledgeGet 43
Sale 44
Sale_Insert 44
Sales_Insert (EEEWebserviceauth.asmx only) 46
Customer Order 47
CustomerOrder_Insert 47
CustomerOrderLineStatus_Get 50

CustomerOrderLineStatus_AcknowledgeGet 50
EEE / Enabler internal Messages 50
Message_Insert 50
Referral Cards 51
ReferralCardValidate 51
ReferralCardRedeem 51
Debtors 52
DebtorAuthorisation 52
Gift Cards 52
GiftCardBalance 52
GiftCardRedeem 53
GiftCardActivate 53
GiftCardVirtualReserve 54
GiftCardVirtualReserveIssue 54
Customer Communications 55
CustomerCommunication_GetAvailable 55
CustomerCommunication_Get 55
CustomerCommunication_AcknowledgeGet 56
Appendix A – Exceptions 57
Appendix B – EEE Configuration Options 59
Web service user permissions 59

Installation

Copy the following files and folders to the server plus the required asmx files

```
/bin/

/Web.config
```

Update the connection string to point to the appropriate database

Update SmtpAppender for Log4Net

1. Set threshold value to "ERROR"
2. Update smtpHost value
3. Update to value
4. Update subject and from value as necessary

Create application in IIS

```
e.g. http://localhost/EEEWebService
```

Make sure application pool is set to run with the correct identity. Also ensure that the identity used (e.g. EEEServices) has write access to the following directory:

'C:\Windows\Microsoft.NET\Framework\v2.0.50727\Temporary ASP.NET Files'

WSDL

Refer to EEEWebService.wsdl in /EEEProjects/EEEWebService

Upgrading to 1.48+ - Important Configuration Changes

Prior to 1.48 the settings for the webservice were stored in the web.config file.
From version 1.48.0 onwards some of these configuration settings appear in:
EEE -> Configuration Settings -> Comms -> Webservice Settings

When upgrading to 1.48.0 or later you must move setup these settings in EEE from the webservice web.config file. Below is the mapping of settings from the web.config to EEE setting:

Web.config Setting	EEE Websevice Setting	Example
Sale_Insert_TillCode	Till Code to use for Sales	01
Sale_Insert_WriteSaleToHostLogExportStagingFlag	Write to host log	False
Sale_Insert_CustomerRequired	Customer Required	False
CustomerOrder_Insert_DefaultInsertionUser	Insert Customer Orders using user	WEB
CustomerOrder_Insert_BinTransferSource	Warehouse Bin Transfer Source	MAINSTOCK
CustomerOrder_Insert_BinTransferDestination	Warehouse Bin Transfer Destination	WEBRESERVE
CustomerOrder_Insert_DefaultCustomerCategory	Default Customer Category	
CustomerOrder_Insert_Reference2	Reference2	WEB ORDER
CustomerOrder_Insert_TillCode	Till Code to use for Customer Orders	01

EEEWebService Usage

The uri for the webservice is the path to the host (e.g. http://localhost/eeewebservice) and

- With EEE web service user authorisation **ENABLED** – eeewebserviceauth.asmx

- With EEE web service user authorisation **DISABLED** – eeewebservice.asmx

All methods listed below are for the eeewebserviceauth.asmx. The eeewebservice.asmx uses the same methods however it does not implement the websiteUserCode and websitePassword parameters on any methods

Common Methods

EEEVersion

Method	EEEVersion
Parameters	Name / Type / Description websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	Version of EEE connected to
Sample Usage	`EEEWebService ews = new EEEWebService();` `string version = ews.EEEVersion();`

GetBranchList

Method	GetBranchList – gets a list of branches from EEE that are not flagged as Offline.
Parameters	Name / Type / Description websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	Datatable containing BranchCode, BranchDescription, RegionCode and RegionDescription
Sample Usage	`EEEWebService ews = new EEEWebService();` `DataTable branchlist = ews.GetBranchList();`

Layby Methods

ItemsToPromoteToCustomer

Method	ItemsToPromoteToCustomer		
Parameters	Name	Type	Description
	customerCode	string	Customer Code
	branchCode	string	Branch Code
	webSiteItemsOnly	int	Include items registered for display on websites only? 0 = No, 1 = Yes
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	DataSet with Data Table of structure Type (String) – this is the source of the recommendation ProductCode (String) – the product code of the product being recommended Code (String)		
Sample Usage	`DataSet ds = new DataSet();` `ds = ews.ItemsToPromoteToCustomer("07210128", "0721", 0);`		

If an empty or invalid parameter is supplied this method will throw an InvalidParameterException. Checking the detail of the SoapException should allow for better feedback to the user. An invalid customer or branch code is one that does not exist in the database.

RetrieveLayby

Method	RetrieveLayby
Parameters	Name / Type / Description customerSurname / string / The customers surname laybyNumber / string / The layby reference number websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	Data Table of structure LaybyNumber (string) CustomerCode (string) CreationDate (dateTime) – creation date of the layby DueDate (dateTime) – the date the layby is due to be finalised LastPaymentDate (dateTime) – the date of the last payment Total (decimal) – the layby total amount TotalPaymentsToDate (decimal) – the total of payments made to date Balance (decimal) – the balance remaining to be paid on the layby
Sample Usage	`DataTable dt = new DataTable();` `dt = ews.RetrieveLayby("graham","072101000027");`

If an empty or invalid parameter is supplied this method will throw an InvalidParameterException. Checking the detail of the SoapException should allow for better feedback to the user. An invalid layby number is one that does not exist in the database.

RecordLaybyPayment

Method	RecordLayByPayment		
Parameters	Name	Type	Description
	laybyNumber	string	The layby reference number
	transactionNumber	String	The external transaction number
	Line	Int	The line number of the payment
	WebSiteId	String	The Id of the website that is the source of the payment. This site must be registered in EEE.
	PaymentDate	String	The payment date in the format YYYYMMDD (E.g. 20090916)
	Amount	Double	The payment amount
	Narration	String	Any notes accompanying the payment
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	Boolean		
Sample Usage	`bool success = false;` `success = ews.RecordLaybyPayment("072101000027", "1",1,"MAGENTA","20090916",10.00,"Unit Test");`		

If an empty or invalid parameter is supplied this method will throw an InvalidParameterException. Checking the detail of the SoapException should allow for better feedback to the user. An invalid layby number is one that does not exist in the database.

Customer Registration Methods

GetCustomDemographics

Method	GetCustomDemographics		
Parameters	Name	Type	Description
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Sample Usage	`EEEWebService ews = new EEEWebService();` `XmlNode demographicsxml = ews.GetCustomDemographics();`		
Returns	Xml containing the details of the custom demographic questions and where applicable their available responses. The data sent conforms to GetCustomDemographics.xsd. Sample XML Below.		

```
<CustomDemographics>
  <DemographicQuestions QuestionId="9" QuestionText="Sports Played"
QuestionType="MLIST" ResponseGroup="SPORTSPLAYED">
    <DemographicResponses Answer="200" AnswerText="AFL" Sortkey="1.00000"/>
    <DemographicResponses Answer="201" AnswerText="Baseball" Inactive="1"
Sortkey="2.00000" />
    <DemographicResponses Answer="202" AnswerText="Basketball"
Sortkey="3.00000" />
    <DemographicResponses Answer="203" AnswerText="Boxing" Sortkey="4.00000"
/>
    <DemographicResponses Answer="204" AnswerText="Cricket"
Sortkey="5.00000"/>
    <DemographicResponses Answer="205" AnswerText="Cycling" />
    <DemographicResponses Answer="206" AnswerText="Golf" BranchCode="0765" />
    <DemographicResponses Answer="207" AnswerText="Hockey" />
    <DemographicResponses Answer="208" AnswerText="Netball" />
    <DemographicResponses Answer="209" AnswerText="Rugby League" />
    <DemographicResponses Answer="210" AnswerText="Rugby Union" />
    <DemographicResponses Answer="211" AnswerText="Soccor" />
    <DemographicResponses Answer="212" AnswerText="Softball" Inactive="1"
BranchCode="0765" />
    <DemographicResponses Answer="213" AnswerText="Swimming" />
    <DemographicResponses Answer="214" AnswerText="Other" />
  </DemographicQuestions>
</CustomDemographics>
```

Demographic Questions and Demographic Responses definition

Additional demographic questions can be recorded in EEE to collect specific information from a customer. Below details the different types of questions and how they behave.

Question Type TEXT

In a text question the user can respond with plain text.

Question format example:

```
<DemographicQuestions QuestionId="1" QuestionText="Sports" QuestionType="TEXT">
</DemographicQuestions>
```

Answer format example:

```
<DemographicQuestions QuestionId="1" Answer="This is a text answer"/>
```

Question Type DATE

In a text question the user can respond with a date. The answer is formatted YYYYMMDD.

Question format example:

```
<DemographicQuestions QuestionId="1" QuestionText="Date purchased" QuestionType="DATE">
</DemographicQuestions>
```

Answer format example:

```
<DemographicQuestions QuestionId="1" Answer="20100101"/>
```

Question Type BOOL

In a boolean question the user can respond yes or no.

Question format example:

```
<DemographicQuestions QuestionId="2" QuestionText="Has equipment" QuestionType="BOOL">
</DemographicQuestions>
```

Answer format example:

```
<DemographicQuestions QuestionId="2" Answer="true"/>
```

Question Type MLIST

In a MLIST (Multi select list) question the user can respond with one or more responses from a predefined list of responses. The list of responses is detailed in the DemographicResponses nodes within the DemographicQuestions node.

Question format example:

```
<DemographicQuestions QuestionId="9" QuestionText="Sports Played" QuestionType="MLIST"
ResponseGroup="SPORTSPLAYED">
        <DemographicResponses Answer="200" AnswerText="AFL" Inactive="1" />
        <DemographicResponses Answer="201" AnswerText="Baseball" BranchCode="007" />
        <DemographicResponses Answer="202" AnswerText="Foosball " Inctive="1" BranchCode="007" />
        <DemographicResponses Answer="203" AnswerText="Football" />
        <DemographicResponses Answer="210" AnswerText="Other" />
</DemographicQuestions>
```

Answer format example:

```
<DemographicQuestions QuestionId="9" Answer="200"/>
<DemographicQuestions QuestionId="9" Answer="201"/>
```

Question Type SLIST

In a SLIST (Single select list) question the user can respond by selecting a single response from a predefined list of responses. The list of responses is detailed in the DemographicResponses nodes within the DemographicQuestions node.

Question format example:

```
<DemographicQuestions QuestionId="50" QuestionText="Sports Played" QuestionType="SLIST"
ResponseGroup="SPORTSPLAYED">
        <DemographicResponses Answer="200" AnswerText="AFL" />
        <DemographicResponses Answer="201" AnswerText="Baseball" BranchCode="007" />
        <DemographicResponses Answer="202" AnswerText="Foosball " Inctive="1" BranchCode="007" />
        <DemographicResponses Answer="203" AnswerText="Football" />
</DemographicQuestions>
```

Answer format example:

```
<DemographicQuestions QuestionId="50" Answer="200"/>
```

Depends On Question types

How and when a question is displayed can be dependent on the answer the user provides to another question. This is where the DependsonQuestionId is used.

Question format example:

```
<DemographicQuestions QuestionId="10" QuestionText="Other" QuestionType="TEXT"
DependsOnQuestionId="9" DependsOnValue="Other" DependsOnAction="SHOW" ResponseGroup="" />
```

Answer format example:

```
<DemographicQuestions QuestionId="10" Answer="Fencing"/>
```

In the example above:

- Question 10 depends on the answer to Question 9 (DependsOnQuestionId=”9”)
- If the answer to Question 9 includes “Other” (DependsOnValue=”Other”) then Question 10 is shown (DependsOnAction=”SHOW”).

There are 2 depends on action types.

- SHOW – show the question if the dependencies are met.
- REPEAT – repeat the question for the number of times in the DependsOnValue attribute.

Repeating Demographic Questions and Associated Answers definition

For grouped questions new responses are denoted by a negative parent id. A positive parent id will update existing answers. The parent id of existing records is returned in the CustomerDetails function.

In the example question below the repeating group question relies on the answer to question 3. In the example answer for Question 3 the Answer is 2 so there will be 2 groups of responses for question 4 and its associated questions (denoted by the parentquestionid in the example question).

Question format example:

```
<DemographicQuestions QuestionId="3" QuestionText="Number of footballs" QuestionType="NUM"/>
<DemographicQuestions QuestionId="4" QuestionText="Number of footballs" QuestionType="GROUP"
DependsOnQuestionId="3" DependsOnAction="REPEAT" />
        <DemographicQuestions QuestionId="5" ParentQuestionId="4" QuestionText="Brand Name"
        QuestionType="TEXT" />
        <DemographicQuestions QuestionId="6" ParentQuestionId="4" QuestionText="Date of purchase"
        QuestionType="DATE" />
  </DemographicQuestions>
```

Answer format example:

```
<DemographicQuestions QuestionId="3" Answer="2">
<DemographicQuestions QuestionId="4" ParentId="-1" >
        <AssociatedAnswers QuestionId="5" ParentId="-1" Answer="Sherrin" />
        <AssociatedAnswers QuestionId="6" ParentId="-1" Answer="20100101" />
</DemographicQuestions>
<DemographicQuestions QuestionId="4" ParentId="-1" >
        <AssociatedAnswers QuestionId="5" ParentId="-1" Answer="Unknown" />
        <AssociatedAnswers QuestionId="6" ParentId="-1" Answer="20100102" />
</DemographicQuestions>
```

Customer Registration

Method	CustomerRegistration		
Parameters	Name	Type	Description
	customerXmlString	String	An xml string that contains information required to create or update a customer. See below for sample xml.
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	String - CustomerCode		
Sample .NET Usage	`string customerCode = "";` `customerCode = ews.CustomerRegistration("<xml string>");`		
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<string xmlns="http://www.magentaretail.com.au/">000001</string>`		

NOTE: If the CustomerCode is empty, a new customer record will be added. If the CustomerCode is not empty, the existing customer record will be updated. In the Customer node, the attributes are mapped into the same fields in a Customer record in EEE.

XML Input definition – see Customer.xsd for full definition.

Sample XML

```
<?xml version="1.0" encoding="utf-8"?>
<EEECustomer>
  <EEECustomerTable>
        <CustomerRecord        CustomerCode=""
                               CustomerCategoryCode="LOYALTY"
                               CustomerSubCategoryCode="SP"
                               Surname="Jones"
                               Firstname="Mary"
                               Salutation="Mrs"
                               Address1="50 George St"
                               Address2="" Address3=""
                               Address4="Sydney" Postcode="2000"
                               State="NSW"
                               Country="Australia" Email="mary.jones@nopond.com"
                               Phone="" Mobile="0000000000"
                               DOB="1985-01-01" Password="Password1"
                               BranchCode="99"/>
        <DemographicQuestions QuestionId="1" Answer="true"/>
        <DemographicQuestions QuestionId="2" Answer="20100501"/>
        <DemographicQuestions QuestionId="3" Answer="2"/>
        <DemographicQuestions QuestionId="4" ParentId="-1">
          <AssociatedAnswers QuestionId="5" Answer="GQ-1a" ParentId="-1"/>
          <AssociatedAnswers QuestionId="6" Answer="GQ-1b" ParentId="-1"/>
        </DemographicQuestions>
        <DemographicQuestions QuestionId="4" ParentId="-2">
          <AssociatedAnswers QuestionId="5" Answer="GQ-2a" ParentId="-2" />
          <AssociatedAnswers QuestionId="6" Answer="GQ-2b" ParentId="-2"/>
        </DemographicQuestions>
        <DemographicQuestions QuestionId="9" Answer="200"/>
        <DemographicQuestions QuestionId ="9" Answer="214"/>
        <DemographicQuestions QuestionId ="10" Answer="AFL"/>
        <DemographicQuestions QuestionId ="11" Answer="216"/>
        <DemographicQuestions QuestionId ="13" Answer="231"/>
        <DemographicQuestions QuestionId ="14" Answer="264"/>
        <DemographicQuestions QuestionId ="15" Answer="273"/>
  </EEECustomerTable>
```

```
</EEECustomer>
```

Customer Confirmation

Method	CustomerConfirmation
Parameters	Name / Type / Description emailId / String / A unique email Id that is in the email sent to the end user to confirm the email address. customerCode / String / The customer code included in the email sent websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	String - CustomerCode
Sample .NET Usage	`string customerCode = "";` `customerCode = ews.CustomerConfirmation("B54F85BC-DCDB-4897-8DD7-FE5E2A66A045","000001" );`
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<string xmlns="http://www.magentaretail.com.au/">000001</string>`

Customer Confirmation With Details

Method	CustomerConfirmationWithDetails
Parameters	Name / Type / Description emailId / String / A unique email Id that is in the email sent to the end user to confirm the email address. customerCode / String / The customer code included in the email sent websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	DataSet – The same format as the CustomerDetails method

Customer Validate Login

Method	CustomerValidateLogin
Parameters	Name / Type / Description LoyaltyCardNumber / String / The card number **OR** Emailaddress / String / The customers email address Password / String **AND** websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	String – CustomerCode if successful. Null if not successful.
Sample .NET Usage	`string login = ews.CustomerValidateLogin("test@test.com", "Password");`
	`string login = ews.CustomerValidateLogin("GC100000000000651313", "Password");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?> <string xmlns="http://www.magentaretail.com.au/">000001</string>`

Customer Offline Login

Method	CustomerValidateLoginOffline
Parameters	Name / Type / Description LoyaltyCardNumber / String / The card number OR CustomerCode / String / The customers customer code **Plus at least 2 of the following items** DOB / String / The customers date of birth in the format 'YYYYMMDD' Surname / String / The customers surname EmailAddress / String / The customers email address PostCode / String / The customers postcode Phone / String / The customers phone number Mobile / String / The customers mobile phone number **AND** websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	String – CustomerCode if successful. Null if not successful.
Sample .NET Usage	`string login = ews.CustomerValidateLoginOffline("GC100000000000651313", "Password", "19000101","Smith");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<string xmlns="http://www.magentaretail.com.au/">000001</string>`

Customer Details

Method	CustomerDetails
Parameters	Name / Type / Description Emailaddress / String / The customers email address Password / String / The customers password websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	DataSet - Table 0 is Customer details and Table 1 is Customer answers
Sample .NET Usage	`DataSet ds = new DataSet();` `ds = ews.CustomerDetails("test@test.com", "Password1");`

Return XML	`<diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">`
	`<NewDataSet xmlns="">`
	`<Table diffgr:id="Table1" msdata:rowOrder="0">`
	`<customercode>000001</customercode>`
	`<loyaltycardnumber>GC000000000000000012</loyaltycardnumber>`
	`<surname>Jones</surname>`
	`<firstname>Mary</firstname>`
	`<salutation>Mrs</salutation>`
	`<address1>50 George St</address1>`
	`<address2>Address21</address2>`
	`<address3>Address31</address3>`
	`<address4>Sydney</address4>`
	`<postcode>2000</postcode>`
	`<state>NSW</state>`
	`<country>Australia</country>`
	`<email>mary.jones@nopond.com</email>`
	`<mobile>0000000000</mobile>`
	`<dob>1985-01-01T00:00:00+10:00</dob>`
	`<gender>F</gender>`
	`<consentgiven>true</consentgiven>`
	`<consentdatetime></consentdatetime>`
	`<optoutemail />`
	`<optoutmobile />`
	`<donotcontact />`
	`<customercategorycode />`
	`<parentcustomercode />`
	`<customersubcategorycode />`
	`<externalid />`
	`<MasterCustomerCode />`
	`<MasterCustomerAssociationDate />`
	`<MasterCustomerAssociationExpiryDate />`
	`</Table>`
	`<Table1 diffgr:id="Table11" msdata:rowOrder="0">`
	`<Id>105</Id>`
	`<QuestionId>1</QuestionId>`
	`<Answer>true</Answer>`
	`</Table1>`
	`<Table1 diffgr:id="Table12" msdata:rowOrder="1">`
	`<Id>106</Id>`
	`<QuestionId>2</QuestionId>`
	`<Answer>20100501</Answer>`
	`</Table1>`
	`<Table1 diffgr:id="Table13" msdata:rowOrder="2">`
	`<Id>107</Id>`
	`<QuestionId>3</QuestionId>`
	`<Answer>2</Answer>`
	`</Table1>`
	`<Table1 diffgr:id="Table14" msdata:rowOrder="3">`
	`<Id>108</Id>`
	`<QuestionId>4</QuestionId>`

```
            <Answer />
</Table1>
<Table1 diffgr:id="Table15" msdata:rowOrder="4">
            <Id>112</Id>
            <QuestionId>4</QuestionId>
            <Answer />
</Table1>
<Table1 diffgr:id="Table16" msdata:rowOrder="5">
            <Id>116</Id>
            <QuestionId>4</QuestionId>
            <Answer />
</Table1>
<Table1 diffgr:id="Table17" msdata:rowOrder="6">
            <Id>120</Id>
            <QuestionId>4</QuestionId>
            <Answer />
</Table1>
<Table1 diffgr:id="Table18" msdata:rowOrder="7">
            <Id>109</Id>
            <ParentId>108</ParentId>
            <QuestionId>5</QuestionId>
            <Answer>Jimmy</Answer>
</Table1>
<Table1 diffgr:id="Table19" msdata:rowOrder="8">
            <Id>110</Id>
            <ParentId>108</ParentId>
            <QuestionId>6</QuestionId>
            <Answer>20090101</Answer>
</Table1>
<Table1 diffgr:id="Table110" msdata:rowOrder="9">
            <Id>111</Id>
            <ParentId>108</ParentId>
            <QuestionId>7</QuestionId>
            <Answer>M</Answer>
</Table1>
<Table1 diffgr:id="Table111" msdata:rowOrder="10">
            <Id>113</Id>
            <ParentId>112</ParentId>
            <QuestionId>5</QuestionId>
            <Answer>Jenny</Answer>
</Table1>
<Table1 diffgr:id="Table112" msdata:rowOrder="11">
            <Id>114</Id>
            <ParentId>112</ParentId>
            <QuestionId>6</QuestionId>
            <Answer>20080101</Answer>
</Table1>
<Table1 diffgr:id="Table113" msdata:rowOrder="12">
            <Id>115</Id>
            <ParentId>112</ParentId>
            <QuestionId>7</QuestionId>
            <Answer>M</Answer>
</Table1>
</NewDataSet>
</diffgr:diffgram>
</DataSet>
```


Notes	The following nodes are nullable and will not be present in the dataset if the Customer has a null value for these fields: • SubCategoryCode • MasterCustomerCode • MasterCustomerAssociationDate • MasterCustomerAssociationExpiryDate

See Demographic Questions and Associated Answers for information on the relationships of Answers

CustomerDetailsByCustomerCode

Method	CustomerDetailsByCustomerCode		
Parameters	Name	Type	Description
	CustomerCode	String	The customers EEE customer code
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	DataSet – The same format as the CustomerDetails method		

Customer DetailsWithParameters

Method	CustomerDetailsWithParameters		
Parameters	Name	Type	Description
	Emailaddress	String	The customers email address
	OR		
	LoyaltyCardNumber	String	The card number
	Plus at least 2 of the following items		
	DOB	String	The customers date of birth in the format 'YYYYMMDD'
	Surname	String	The customers surname
	EmailAddress	String	The customers email address
	PostCode	String	The customers postcode
	Phone	String	The customers phone number
	Mobile	String	The customers mobile phone number
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	DataSet – The same format as the CustomerDetails method		

CustomerForgotPassword

Method	CustomerForgotPassword
Parameters	Name / Type / Description LoyaltyCardNumber / String / The card number **OR** Emailaddress / String / The customers email address websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	Boolean – true if email sending was successful. False if not successful or a password has not been set.
Sample .NET Usage	`bool login = ews.CustomerForgotPassword("test@test.com");`
	`bool login = ews.CustomerForgotPassword("GC100000000000651313");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?> <boolean xmlns="http://www.magentaretail.com.au/">true</boolean>`

CustomerEmailExists

Method	CustomerEmailExists
Parameters	Name / Type / Description Emailaddress / String / The customers email address websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	Boolean – true if the email address exists in the customer database.
Sample .NET Usage	`bool emailexists = ews.CustomerEmailExists("test@test.com");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?> <boolean xmlns="http://www.magentaretail.com.au/">true</boolean>`

CustomerDetailsExtended

Description Returns the customers demographic detail and available coupons. The data conforms to CustomerDetailsExtended.xsd.

Parameters

Name	Type	Description
CustomerCode	String	The customers customer code
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns XML that conforms to CustomerDetailsExtended.xsd.

CustomerDetailsExtendedWithParameters

Description Locates the customer record using the supplied details and returns the customers demographic detail and available coupons. The data conforms to CustomerDetailsExtended.xsd.

Parameters

Name	Type	Description
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)
Emailaddress	String	The customers email address
	OR	
LoyaltyCardNumber	string	The card number
PLUS AS LEAST 2 OF THE FOLLOWING		
DOB	String	The customers date of birth in the format 'YYYYMMDD'
Surname	String	The customers surname
EmailAddress	String	The customers email address
PostCode	String	The customers postcode
Phone	String	The customers phone number
Mobile	String	The customers mobile phone number

Returns XML that conforms to CustomerDetailsExtended.xsd.

CustomerConfirmationWithDetailsExtended

Description Confirms a customers email address as valid and returns the customers demographic detail and available coupons. The data conforms to CustomerDetailsExtended.xsd.

Parameters

Name	Type	Description
emailId	String	A unique email Id that is in the email sent to the end user to confirm the email address.
CustomerCode	String	The customers customer code
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns XML that conforms to CustomerDetailsExtended.xsd.

Loyalty Rewards Methods

Customer Loyalty Card Details

Method	CustomerLoyaltyCardDetails
Parameters	Name / Type / Description LoyaltyCardNumber / String / The customers loyalty card number **OR** Emailaddress / String / The customers email address Password / String websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	DataSet Table 1 – Loyalty Card Details
Sample .NET Usage	`DataSet ds = new DataSet();` `ds = ews.CustomerLoyaltyCardDetails("GC100000000000651313","",”Password1”);`

Return XML	`<diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">`
	`<NewDataSet xmlns="">`
	`<Table diffgr:id="Table1" msdata:rowOrder="0">`
	`<CustomerCode>0000001</CustomerCode>`
	`<LoyaltyCardNumber>GC100000000000651313</LoyaltyCardNumber>`
	`<IssueDate>2010-02-12T10:56:39+10:00</IssueDate>`
	`<ExpiryDate>9999-12-31T00:00:00+10:00</ExpiryDate>`
	`<RemainingValue>0.0000</RemainingValue>`
	`<AccumulatedPoints>10</AccumulatedPoints>`
	`</Table>`
	`</NewDataSet>`
	`</diffgr:diffgram>`
	`</DataSet>`

LoyaltyGiftCardStatus

Method	LoyaltyGiftCardStatus
Parameters	Name / Type / Description LoyaltyCardNumber / String / The card number websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	0 = Card is valid and not issued and can be used 1 = Card is invalid 2 = Card is valid but is already issued to a customer
Sample .NET Usage	`int cardstatus = ews.LoyaltyGiftCardStatus("GC100000000000651313");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<int xmlns="http://www.magentaretail.com.au/">0</int>`

CustomerLoyaltyStatement

Method	CustomerLoyaltyStatement
Parameters	Name / Type / Description customerCode / String / The customers customer code dateFrom / String / Optional start date filter. Format YYYYMMDD dateTo / String / Optional end date filter. Format YYYYMMDD websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)

Return XML	`<?xml version="1.0" encoding="utf-8" ?>`
	`<LoyaltyStatement LoyaltyStatement="1">`
	`- <SaleHeader Reference1="07210100000484" RewardPoints="300.00" SaleDate="22/02/2010 14:38:35" BranchDescription="Store 1">`
	`<Saleline SequenceNumber="39138" ProductCode="103001" ProductDescription="Banana boat" Quantity="1.00" FCExtendedNetAmount="200.00" FCExtendedDiscountAmount="0.00" />`
	`<Saleline SequenceNumber="39139" ProductCode="35110000171-ASS-10" ProductDescription="Eco-d 2-ASS-10" Quantity="1.00" FCExtendedNetAmount="100.00" FCExtendedDiscountAmount="0.00" />`
	`</SaleHeader>`
	`- <SaleHeader Reference1="07210100000486" RewardPoints="90.00" SaleDate="22/02/2010 14:42:17" BranchDescription="Store 1">`
	`<Saleline SequenceNumber="39140" ProductCode="35110000145-BLK-OS" ProductDescription="Bath Sheet-BLK-OS" Quantity="1.00" FCExtendedNetAmount="105.00" FCExtendedDiscountAmount="15.00" />`
	`</SaleHeader>`
	`/LoyaltyStatement>`

NOTES	The XML returned is related in the following way: LoyaltyStatement is the root with 0 or more Sale headers SaleHeader is a single sale that can have 1 or more sale lines Saleline is a sale item in a sale.

LoyaltyPointsTransfer

Method	LoyaltyPointsTransfer
Parameters	Name / Type / Description CustomerCodeFrom / String / The customercode of the person transfer points to another loyalty customer LoyaltyCardNumberTo / String / The loyalty card number to transfer the points to PointstoTransfer / Decimal / The number of points to transfer websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	True = transfer complete False = transfer failed
Sample .NET Usage	`bool success = ews.LoyaltyPointsTransfer("000001", "GC500000000000000001", "20.00");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<boolean xmlns="http://www.magentaretail.com.au/">true</boolean>`

Customer Promotion Coupon Methods

CouponRedemption

Method	CouponRedemption		
Parameters	Name	Type	Description
	CouponCode	String	The coupon to redeem
	CustomerCode	String	The customers customer code
	RedeemTransactionNumber	String	The transaction number that the coupon was redeemed against
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	True = coupon redeemed False = coupon not redeemed		
Sample .NET Usage	`bool success = ews.CouponRedemption("0000C4", "000001");`		
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<boolean xmlns="http://www.magentaretail.com.au/">true</boolean>`		
NOTE:	The RedeemedDate of the coupon is set to the time that this method is executed		

CouponRedemptionById

Method	CouponRedemptionById		
Parameters	Name	Type	Description
	PromotionCouponId	Int	The id of the coupon to redeem
	CustomerCode	String	The customers customer code
	RedeemTransactionNumber	String	The transaction number that the coupon was redeemed against
	RedeemDate	DateTime	The date that the coupon was redeemed
	websiteUserCode	string	Usercode (see user permissions)
	websitePassword	string	Password (see user permissions)
Returns	True = coupon redeemed False = coupon not redeemed		
Sample .NET Usage	`bool success = ews.CouponRedemption(1, "000001", "TN1000","01/01/2010 10:00:00");`		
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<boolean xmlns="http://www.magentaretail.com.au/">true</boolean>`		

CouponValid

Method	CouponValid
Parameters	Name / Type / Description CouponCode / String / The coupon to redeem CustomerCode / String / The customers customer code websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	True = coupon able to be used False = coupon not able to be used
Sample .NET Usage	`bool success = ews.CouponValid("0000C4", "000001");`
Return XML	`<?xml version="1.0" encoding="utf-8" ?>` `<boolean xmlns="http://www.magentaretail.com.au/">true</boolean>`

CouponsAvailableToCustomer

Method	CouponsAvailableToCustomer
Parameters	Name / Type / Description CustomerCode / String / The customers customer code websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	Data Table containing coupons that can be used by the customer
Sample .NET Usage	`DataTable dt = ews.CouponsAvailableToCustomer("000001");`

Return XML	`<?xml version="1.0" encoding="utf-8" ?><DataTable xmlns="http://www.magentaretail.com.au/">`
	`<xs:schema id="NewDataSet" xmlns="" xmlns:xs="http://www.w3.org/2001/XMLSchema"`
	`xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">`
	`<xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:MainDataTable="Data"`
	`msdata:UseCurrentLocale="true"> <xs:complexType><xs:choice minOccurs="0"`
	`maxOccurs="unbounded"> <xs:element name="Data"><xs:complexType><xs:sequence>`
	`<xs:element name="CouponCode" type="xs:string" minOccurs="0" />`
	`<xs:element name="CouponNumber" type="xs:long" minOccurs="0" />`
	`<xs:element name="CouponImage" type="xs:base64Binary" minOccurs="0" />`
	`<xs:element name="PromotionCode" type="xs:string" minOccurs="0" />`
	`<xs:element name="StartDate" type="xs:dateTime" minOccurs="0" />`
	`<xs:element name="EndDate" type="xs:dateTime" minOccurs="0" />`
	`<xs:element name="CustomerCode" type="xs:string" minOccurs="0" />`
	`<xs:element name="IssueDate" type="xs:dateTime" minOccurs="0" />`
	`<xs:element name="ExpiryDate" type="xs:dateTime" minOccurs="0" />`
	`<xs:element name="RedeemDate" type="xs:dateTime" minOccurs="0" />`
	`<xs:element name="RedeemTransactionNumber" type="xs:string" minOccurs="0" />`
	`<xs:element name="PromotionType" type="xs:string" minOccurs="0" />`
	`<xs:element name="CouponPercentageValue" type="xs:decimal" minOccurs="0" />`
	`<xs:element name="CouponDollarValue" type="xs:decimal" minOccurs="0" />`
	`<xs:element name="PromotionDescription" type="xs:string" minOccurs="0" />`
	`<xs:element name="PromotionCouponId" type="xs:int" minOccurs="0" />`
	`</xs:sequence></xs:complexType></xs:element></xs:choice></xs:complexType></xs:element>`
	`</xs:schema> <diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"`
	`xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">`
	`<DocumentElement xmlns=""><Data diffgr:id="Data1" msdata:rowOrder="0">`
	`<CouponCode>0000C4</CouponCode>`
	`<CouponNumber>12</CouponNumber><CouponImage></CouponImage>`
	`<PromotionCode>FIRST</PromotionCode>`
	`<StartDate>2010-01-01T00:00:00+10:00</StartDate>`
	`<CustomerCode>000001</CustomerCode>`
	`<IssueDate>2010-06-02T10:14:12.527+10:00</IssueDate>`
	`<ExpiryDate>2010-06-02T23:59:59.997+10:00</ExpiryDate>`
	`<PromotionType>Acquisition Offer 1</PromotionType>`
	`<CouponPercentageValue>0.00</CouponPercentageValue>`
	`<CouponDollarValue>20.0000</CouponDollarValue>`
	`<PromotionDescription>AO2</PromotionDescription>`
	`<PromotionCouponId>5</PromotionCouponId>`
	`</Data></DocumentElement> </diffgr:diffgram> </DataTable>`

Transaction Methods

GetTransactionPrices

Method	GetTransactionPrices
Parameters	Name / Type / Description transactionXML / String / An xml string that meets the criteria as per the SalesImport.xsd (sample below) websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	An xml string that meets the criteria as per SalesImport.xsd that includes the prices as calculated by EEE.

Sample .NET Usage	`string xmlinput = "`
	`<Sales xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <Sale>`
	`<TranId>GF99999999</TranId>`
	`<TranDate>2009-09-11T00:00:00</TranDate>`
	`<CustomerCode>000001</CustomerCode>`
	`<SaleBranchCode>0721</SaleBranchCode>`
	`<StockBranchCode>0721</StockBranchCode>`
	`<MasterCustomerCode />`
	`<DINumber>11111</DINumber>`
	`<WebsiteCode>REBEL</WebsiteCode>`
	`<SaleLine xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">`
	`<SaleLineId>1</SaleLineId>`
	`<ProductCode>1</ProductCode>`
	`<Quantity>1.000000</Quantity>`
	`<UnitSellingPrice>149.9500</UnitSellingPrice>`
	`<ExtendedDiscount>10.0000</ExtendedDiscount>`
	`<DiscountReason>MD03</DiscountReason>`
	`<GiftCardNumber xsi:nil="true" />`
	`</SaleLine>`
	`</Sale>`
	`</Sales>`
	`string xmloutput = ews.GetTransactionPrices(xmlInput);`

Return XML	`<?xml version="1.0" encoding="utf-16" ?>`
	`<Sales xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">`
	`<Sale>`
	`<TranId>GF99999999</TranId>`
	`<TranDate>2009-09-11T00:00:00</TranDate>`
	`<CustomerCode>000001</CustomerCode>`
	`<SaleBranchCode>0721</SaleBranchCode>`
	`<StockBranchCode>0721</StockBranchCode>`
	`<MasterCustomerCode xsi:nil="true" />`
	`<DINumber>11111</DINumber>`
	`<WebsiteCode>REBEL</WebsiteCode>`
	`<SaleLine xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">`
	`<SaleLineId>1</SaleLineId>`
	`<ProductCode>1</ProductCode>`
	`<Quantity>1.000000</Quantity>`
	`<UnitSellingPrice>-5.10</UnitSellingPrice>`
	`<ExtendedDiscount>15.10</ExtendedDiscount>`
	`<DiscountReason>MD03</DiscountReason>`
	`<GiftCardNumber xsi:nil="true" />`
	`</SaleLine>`
	`</Sale>`
	`</Sales>`

Email Statistics Methods

Email Statistics

Method	Email Statistics
Parameters	Name / Type / Description emailStatsXmlString / String / An xml string that meets the criteria as per the EmailStats.xsd (sample below) websiteUserCode / string / Usercode (see user permissions) websitePassword / string / Password (see user permissions)
Returns	(Bool) true if the statistics were imported without errors
Sample .NET Usage	string xmlinput = “ <?xml version="1.0" ?> <stats xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <stat> <ccid>3008</ccid> <action>1</action> <linkid xsi:nil="true" /> <clicksourceid xsi:nil="true" /> <sourceid>1</sourceid> <utccreated>2010-06-15T06:17:21.993</utccreated> </stat> <stat> <ccid>3008</ccid> <action>2</action> <linkid>100</linkid> <clicksourceid xsi:nil="true" /> <sourceid>2</sourceid> <utccreated>2010-06-15T06:17:21.993</utccreated> </stat> <stat> <ccid>3008</ccid> <action>3</action> <linkid xsi:nil="true" /> <clicksourceid>2</clicksourceid> <sourceid>3</sourceid> <utccreated>2010-06-15T06:17:21.993</utccreated> </stat> </stats>” bool importResult = ews.EmailStatistics(xmlInput);
Return Values	True if the statistics were imported without errors False if there were errors Exception if there were errors

Website Synchronisation Methods

The website synchronisation methods are in place to enable customers to retrieve product and related data from EEE for use on their public facing web stores. There are also methods for the web stores to use to send sales and customer orders to the EEE database.

Overview

This document describes data extraction methods via web service from EEE to external systems.

As a general rule each extraction method (_Get) can be operated in either full load or incremental mode and each extraction supports an acknowledgement (_AcknowledgeGet) process to confirm successful receipt of the data. Acknowledging an extraction will let the system prepare and monitor changes for the next incremental extraction. This applies to both full and incremental mode.

Look Up methods (_LookUp) provide a real time lookup of EEE.

Insert (_Insert) methods commit data to EEE.

Product

Product_Get

Description Returns a gzipped UTF8 encoded byte array of Product xml data. In incremental mode (fullLoad = false), the set returned also identifies Products previously sent that are no longer available on the website.
The data conforms to ProductExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Products, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Product_AcknowledgeGet

Description Acknowledge a batch retrieved using Product_Get.

Parameters

Name	Type	Description
extractGuid	String	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

ProductExtended_Get

Description Returns a gzipped UTF8 encoded byte array of Product xml data including renamed Extra Info fields and Extended Attributes as defined in EEE. These fieldnames may change if they are changed in EEE. In incremental mode (fullLoad = false), the set returned also identifies Products previously sent that are no longer available on the website.

Parameters

Name	Type	Description
fullLoad	bool	True for all Products, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

ProductExtended_AcknowledgeGet

Description Acknowledge a batch retrieved using ProductExtended_Get.

Parameters

Name	Type	Description
extractGuid	String	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

ProductPromotion_Get

Description Returns a gzipped UTF8 encoded byte array of ProductPromotion xml data. In incremental mode (fullLoad = false), the set returned also identifies ProductPromotions previously sent that are no longer available on the website. The data conforms to ProductPromotionExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all ProductPromotions, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

ProductPromotion_AcknowledgeGet

Description Acknowledge a batch retrieved using ProductPromotion_Get.

Parameters

Name	Type	Description
extractGuid	String	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

ProductPromotionPrice_Get

Description Returns a gzipped UTF8 encoded byte array of ProductPromotionPrice xml data. In incremental mode (fullLoad = false), the set returned also identifies ProductPromotionPrices previously sent that are no longer available on the website. The data conforms to ProductPromotionPriceExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all ProductPromotionPrices, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

ProductPromotionPrice_AcknowledgeGet

Description Acknowledge a batch retrieved using ProductPromotionPrice_Get.

Parameters

Name	Type	Description
extractGuid	String	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Inventory

Inventory _Get

Description Returns a gzipped UTF8 encoded byte array of Inventory xml data. The data conforms to InventoryExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all web product Inventory, false for incremental.

websiteCode	string	The website code. Blank if none assigned. Max length 20.
branchCode	string	The branch code whose inventory is to be returned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Inventory_AcknowledgeGet

Description Acknowledge a batch retrieved using Inventory_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Inventory_LookUp

Description Real time Inventory look up for a single product.

Parameters

Name	Type	Description
productCode	string	Product code to look up. Max length 20.
branchCode	string	The branch code whose inventory is to be returned. Max length 20.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns decimal - Quantity on hand.

Price

Price_Get

Description Returns a gzipped UTF8 encoded byte array of Price xml data. The data conforms to PriceExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all web product Prices, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
branchCode	string	The branch code whose Prices are to be returned. Max length 20.

maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Price _AcknowledgeGet

Description Acknowledge a batch retrieved using Price_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Price_LookUp

Description Real time Price look up for a product. Considers Customer discounts if CustomerCode is specified.

Parameters

Name	Type	Description
productCode	string	Product code to look up.
quantity	decimal	The quantity.
branchCode	string	The branch code whose price is to be used. Max length 20.
customerCode	string	Blank if none. Customer making purchase for discount consideration. Max length 20.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns decimal - Extended Price for the specified Product and Quantity.

Cost

Cost_Get

Description Returns a gzipped UTF8 encoded byte array of Cost xml data. The data conforms to CostExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all web product Costs, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
branchCode	string	The branch code whose Costs are to be returned. Max length 20.

CostType	Int	0 = WAC Cost 1 = Buy price 1 2 = Both WAC and Buy price 1
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Cost _AcknowledgeGet

Description Acknowledge a batch retrieved using Cost_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Colour

Colour_Get

Description Returns a gzipped UTF8 encoded byte array of Colour xml data. The data conforms to ColourExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Colours, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Colour_AcknowledgeGet

Description Acknowledge a batch retrieved using Colour_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Size

Size_Get

Description Returns a gzipped UTF8 encoded byte array of Size xml data. The data conforms to SizeExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Size_AcknowledgeGet

Description Acknowledge a batch retrieved using Size_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Style

Style_Get

Description Returns a gzipped UTF8 encoded byte array of Style xml data. The data conforms to StyleExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Style_AcknowledgeGet

Description Acknowledge a batch retrieved using Style_Get.

Parameters

Name	Type	Description

extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Analysis1

Analysis1_Get

Description Returns a gzipped UTF8 encoded byte array of Analysis1 xml data. The data conforms to Analysis1Exchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Analysis1_AcknowledgeGet

Description Acknowledge a batch retrieved using Analysis1_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Analysis2

Analysis2_Get

Description Returns a gzipped UTF8 encoded byte array of Analysis2 xml data. The data conforms to Analysis2Exchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)

websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Analysis2_AcknowledgeGet

Description Acknowledge a batch retrieved using Analysis2_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Analysis3

Analysis3_Get

Description Returns a gzipped UTF8 encoded byte array of Analysis3 xml data. The data conforms to AnalysisExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Analysis3_AcknowledgeGet

Description Acknowledge a batch retrieved using Analysis3_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Analysis4

Analysis4_Get

Description Returns a gzipped UTF8 encoded byte array of Analysis4 xml data. The data conforms

to Analysis4Exchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Analysis4_AcknowledgeGet

Description Acknowledge a batch retrieved using Analysis4_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Product Images

ProductImage_Get

Description Returns a gzipped UTF8 encoded byte array of Product Image xml data. The data conforms to ProductImageExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all Sizes, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
sizeType	String	The size type to get. e.g.: THM for Thumbnails
Priority	Int	The image priority to get.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

ProductImage _AcknowledgeGet

Description Acknowledge a batch retrieved using ProductImage _Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585

websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Master Customers

MasterCustomer_Get

Description Returns a gzipped UTF8 encoded byte array of Master Customer xml data. The data conforms to MasterCustomerExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
customerCategoryCode	String	The customer category to get
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

MasterCustomer_AcknowledgeGet

Description Acknowledge a batch retrieved using MasterCustomer_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Promotion Customers

PromotionCustomer_Get

Description Returns a gzipped UTF8 encoded byte array of Promotion Customer xml data. The data conforms to PromotionCustomerExchange.xsd.
NOTE: This method only returns PromotionCustomer records where an explicit Customer is added to the promotion. It does not include PromotionCustomer records where a Customer Category or Customer Sub Category is added to the promotion.

Parameters

Name	Type	Description
fullLoad	bool	True for all, false for incremental.

websiteCode	string	The website code. Blank if none assigned. Max length 20.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

PromotionCustomer_AcknowledgeGet

Description Acknowledge a batch retrieved using PromotionCustomer_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Price Book

PriceBook_Get

Description Returns a gzipped UTF8 encoded byte array of Price Book Prices xml data. The data conforms to PriceBookExchange.xsd.

Parameters

Name	Type	Description
fullLoad	bool	True for all, false for incremental.
websiteCode	string	The website code. Blank if none assigned. Max length 20.
branchCode	string	The branch code. Blank for all branches for the websiteCode.
priceBookCode	string	The price book code. Blank for all price books that apply to the website/branch.
maxRows	int	The maximum number of rows to retrieve. 0 for all.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

PriceBook_AcknowledgeGet

Description Acknowledge a batch retrieved using PriceBook_Get.

Parameters

Name	Type	Description

extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

Sale

Sale_Insert

Description Commit one sale to EEE. This method accepts a gzipped UTF8 encoded byte array of Sale xml. The data must conform to SaleExchange.xsd and is rejected if non-conforming.

Parameters

Name	Type	Description
saleXml	byte[]	gzipped UTF8 encoded byte array
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, False otherwise.

Configuration The following settings need to be correctly configured in the EEE Configuration screen on the Comms tab.

Setting	Description
Sale_Insert_TillCode	The till to simulate when processing the sale e.g. 01. **Mandatory**.
Sale_Insert_WriteSaleToHostLogExportStagingFlag	True to write to Host Log. False otherwise. **Mandatory**.
Sale_Insert_CustomerRequired	True to ensure that a valid Customer Code is specified for the Sale. False otherwise. **Mandatory**.

Notes Nillable fields are required to have the attribute: xsi:nil="true" when no value is supplied. If this attribute is not set, Sale_Insert will throw business rule violation exceptions.

Emails alert to admin on exception

Discounts
The following examples illustrate how to use the UnitSellingPrice and ExtendedDiscount values to generate the correct Tender amount.

- UnitSellingPrice - price of individual item with discount applied
- ExtendedDiscount - total discount amount for sale line

Examples:

1. An item has a retail price of $100 and sells for $60 (discount of $40), only one item is sold.
   - Quantity = 1
   - UnitSellingPrice = $60
   - ExtendedDiscountPrice =$40
   - Tendered Amount = $60
2. An item is returned which had a retail price of $100 and sells for $60 (discount of $40), only one item is RETURNED.
   - Quantity = -1 -ve
   - UnitSellingPrice = $60 +ve
   - ExtendedDiscountPrice =-$40 -ve
   - Tendered Amount = $60 -ve
3. An item has a retail price of $100 and sells for $60 (discount of $40), TWO items sold.
   - Quantity = 2
   - UnitSellingPrice = $60
   - ExtendedDiscountPrice =$80
   - Tendered Amount = $120
4. An item is returned which had a retail price of $100 and sells for $60 (discount of $40), TWO items RETURNED.
   - Quantity = -2 -ve
   - UnitSellingPrice = $60 +ve
   - ExtendedDiscountPrice =-$80 -ve
   - Tendered Amount = -$120 -ve

Sales_Insert (EEEWebserviceauth.asmx only)

Description

Commits a batch of sales to EEE. This method accepts a gzipped UTF8 encoded byte array of Sales xml. The data must conform to SalesExchange.xsd and is rejected if non-conforming.

Parameters

Name	Type	Description
saleXml	byte[]	gzipped UTF8 encoded byte array
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns

Byte[] - gzipped UTF8 encoded byte array of xml data with a status for each sale insert as per the example below

```
<SaleInsertResults>
<Sale TranId="1121" Success="1" ReturnText="" />
<Sale TranId="1122" Success="0" ReturnText="A sale may not be imported if the sale
      date is older than 180 days." />
  </SaleInsertResults>
```

Configuration

Configuration and Notes are the same as those for the Sale_Insert method above

Customer Order

CustomerOrder_Insert

Description Commit one Customer Order to EEE. This method accepts a gzipped UTF8 encoded byte array of Customer Order xml. The data must conform to CustomerOrderExchange.xsd or CustomerOrderExchangeOptional.xsd (depending on configuration below) and is rejected if non-conforming.

Parameters

Name	Type	Description
customerOrderXml	byte[]	gzipped UTF8 encoded byte array
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, False otherwise.

Configuration The following settings need to be correctly configured in the EEE Configuration Screen on the Comms tab for this method:

Name	Configuration Setting Name	Description	Mandatory
Default Order Type	WebservicesCustomerOrder DefaultOrderType	The default order type to use for the order detail rows if the OrderType attribute is not specified in the customerOrderXml. If this is not passed in the xml and it is not configured here, it is defaulted to 'W'	N
Insert Customer orders using User	CustomerOrder_Insert _DefaultInsertionUser	The User Code to attribute the creation of the order to e.g. WEB. **Mandatory**.	**Y**
Warehouse Bin Transfer Source	CustomerOrder_Insert _BinTransferSource	If stock should be moved to a separate bin for the order, this is the transfer source e.g. MAINSTOCK. Note that both the Source and Destination settings are required to contain a value or both should be blank. **Optional** - leave blank/empty if stock should stay in the default bin MAINSTOCK.	N
Warehouse Bin Transfer Destination	CustomerOrder_Insert _BinTransferDestination	If stock should be moved to a separate bin for the order, this is the transfer destination e.g. WEBRESERVE. **Optional** - leave	N

		blank/empty if stock should stay in the default bin MAINSTOCK.	
Default Customer Category	CustomerOrder_Insert _DefaultCustomerCategory	If the Customer Code specified in the Order does not exist, EEE will create a new Customer assigned to this Customer Category e.g. WEB. **Optional** - leave blank/empty if no category is to be assigned.	N
Reference 2	CustomerOrder_Insert _Reference2	The Customer Order will have this note attached upon creation e.g. WEB ORDER. **Optional** - leave blank/empty for none.	N
Tillcode to use for Customer orders	CustomerOrder_Insert _TillCode	The till to simulate when processing the order e.g. 01. **Mandatory**.	**Y**
Customer record for customer orders must already exist	WebservicesCustomer OrderCustomerMustExist	If this option is on then the customer code specified in the customerOrderXml must already exist in EEE. The customer details do not need to be supplied in this instance. If this option is on then the xml is validated against CustomerOrderExchangeOptional.xsd	N
Customer order payment details are optional	WebservicesCustomer OrderPaymentOptional	If this option is on then the payment section of the customerOrderXml can be left out. If this option is on then the xml is validated against CustomerOrderExchangeOptional.xsd	N
Customer Orders are processed in Enabler dBase	WebservicesCustomer OrdersEnablerDbase	This option must be on if web customer orders are going to be processed in Enabler dBase. This limits customer order numbers to a maximum 12 characters	N
Get customer order status for orders marked as Web Orders only	WebservicesCustomer OrderStatusWebOrdersOnly	This option should be on if the status of web orders only are to be retrieved by the web service.	N

Notes

- Nillable fields are required to have the attribute: xsi:nil="true" when no value is supplied. If this attribute is not set, insert will throw business rule violation exceptions.
- Emails alert to admin on exception
- EEE will insert/update its copy of the customer record based on the existence in EEE of the customer code - which may not be blank.
- EEE will automatically assign customers imported/updated from the web site to a **configured customer category**.

- Also flag any new customer records as belonging to the **configured webstore branch**.
- Email alert on exception
- OrderTypes D, V, R, S, T, C are reserved order types and cannot be passed in using the web service

CustomerOrderLineStatus_Get

Description Obtain updated order status information for each order line. Returns a gzipped UTF8 encoded byte array of order line xml data. Conforms to OrderStatus.xsd.

Parameters

Name	Type	Description
fullLoad	boolean	True to get the current status for all order lines for all orders
branchCode	string	BranchCode of the website the orders were inserted against. If empty then orders for all branches are retreived.
maxRows	int	Maximum number of rows you want to recieve. Set to 0 to retreive all rows.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns byte[] - Returns a gzipped UTF8 encoded byte array.

Configuration

Name	Description	Default Setting
Get customer order status for orders marked as Web Orders only	This option should be on if the status of web orders only are to be retrieved by the web service.	N

CustomerOrderLineStatus_AcknowledgeGet

Description Acknowledge a batch retrieved using CustomerOrderLineStatus_Get.

Parameters

Name	Type	Description
extractGuid	string	The string representation of the extractGuid for the batch to acknowledge e.g. B0619126-F9D2-47E6-8555-59E493F31585
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success, false otherwise

EEE / Enabler internal Messages

Message_Insert

Description Inserts a new message into the EEE / Enabler internal notification system

Parameters

Name	Type	Description
BranchCode	string	The branch code of the branch to send the message to. This must be a valid branch code.
MessageSubject	String	The subject of the message

MessageText	String	The text of the message. NOTE: If this includes a URL the URL must be complete. E.g.: http://www.magentaretail.com.au
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success of message insert, false otherwise

Referral Cards

ReferralCardValidate

Description Validates a referral card and checks whether or not it can be redeemed.

Parameters

Name	Type	Description
CardNumber	string	The referral card number
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Xml. The Allowed element dictates if the card can be redeemed. If this is false then the Reason element will detail why the card is not able to be redeemed.

E.g.

```
<ReferralCardValidateResponse>
  <Allowed>FALSE</Allowed>
  <Reason>The referral card has already been redeemed</Reason>
</ReferralCardValidateResponse>
```

ReferralCardRedeem

Description Marks a referral card as redeemed.

Parameters

Name	Type	Description
CardNumber	string	The referral card number
RedeemTransactionNumber	String	The transaction id that the referral card is used on. If a sale then this is the Tran Id. If a customer order then this is the Order Number.
RedeemDate	DateTime	The date of the redemption
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success of redemption, false otherwise

Notes If the card is not marked as redeemed using this call, it will be marked as redeemed when the sale or customer order is inserted through the Sale_Insert or CustomerOrder_Insert methods.

Debtors

NOTE: A debtor is a customer with the ChargeAccount flag set on.

DebtorAuthorisation

Description Checks that the transaction amount can be charged to the debtors account

Parameters

Name	Type	Description
CustomerCode	string	The customer code of the debtor
Amount	double	The amount of the transaction to be authorised.
BranchCode	String	The branch code that the transaction will be applied against
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Xml. The Authorised element dictates if the authorisation was successful. If this is false then the Reason element will detail why the authorisation was refused.

E.g.

```
<DebtorAuthorisationResponse>
  <Authorised>FALSE</Authorised>
  <Reason>This customer does not have a charge account.</Reason>
</DebtorAuthorisationResponse>
```

Gift Cards

Please note that use of the previous GiftCardService.asmx web service is no longer supported.

GiftCardBalance

Description Checks the balance and status of a gift card (or Loyalty Gift Card)

Parameters

Name	Type	Description
voucherNumber	string	The voucher number to check
requirePin	Bool	True if the pin must be validated
pin	String	The pin assigned to the voucher
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Xml. Example:

```
<VoucherStatus>
  <VoucherNumber>11111</VoucherNumber>
  <CardValid>FALSE</CardValid>
  <PinValid>FALSE</PinValid>
  <Redeemable>FALSE</Redeemable>
  <Balance>0.0000</Balance>
  <ExpiryDate>2000-01-01T00:00:00</ExpiryDate>
  <Active>FALSE</Active>
  <LostStolen>FALSE</LostStolen>
  <LostStolenNotes />
  <Returned>FALSE</Returned>
  <PostedToExpenses>FALSE</PostedToExpenses>
  <PostedToRevenue>FALSE</PostedToRevenue>
  <StatusReason>Voucher 1111111111 is invalid</StatusReason>
</VoucherStatus>
```

GiftCardRedeem

Description

Checks that the redeemamount can be redeemed against this gift card and that the card is valid. If this is true the the redeemed amount is redeemed against the card. This method can also be used to redeem value from a loyalty gift card.

Parameters

Name	Type	Description
voucherNumber	string	The voucher number to check
requirePin	Bool	True if the pin must be validated
pin	String	The pin assigned to the voucher
redeemAmount	Double	The amount of the redemption
redeemBranchCode	String	The branch code that the transaction will be applied against
redeemTransactionNumber	String	The transaction id that the card is used on. If a sale then this is the Tran Id. If a customer order then this is order number.
redeemTransactionDate	DateTime	The transaction date of the transaction that the card is redeemed on. This must be the same value at the transaction date supplied to the sale or customer order methods.
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns

Xml. Note that the Balance in the xml is the balance AFTER the redemption has been processed OR the balance BEFORE the redemption if the redemption failed.

E.g.

```
<VoucherRedemptionStatus>
  <VoucherNumber>GC100000000000070135</VoucherNumber>
  <RedeemedSuccessfully>TRUE</RedeemedSuccessfully>
  <CardValid>TRUE</CardValid>
  <PinValid>TRUE</PinValid>
  <Redeemable>TRUE</Redeemable>
  <Balance>2.5000</Balance>
  <ExpiryDate>2012-07-14T00:00:00</ExpiryDate>
  <Active>TRUE</Active>
  <LostStolen>FALSE</LostStolen>
  <LostStolenNotes />
  <Returned>FALSE</Returned>
  <PostedToExpenses>FALSE</PostedToExpenses>
  <PostedToRevenue>FALSE</PostedToRevenue>
  <StatusReason />
</VoucherRedemptionStatus>
```

GiftCardActivate

Description

The Sale_Insert method xml now supports an activation code for physical gift cards. The activation code is generated by the web site and passed to EEE and stored against the vouchertransaction table for the gift card issue. This method activates those cards by passing in the transaction number and a valid activation code.

Parameters

Name	Type	Description
transactionNumber	string	The transaction number of the gift card sale
activationCode	String	The activation code for the cards (max 5 characters)
voucherNumber	string	The voucher number to activate (optional

		depending on configuration)
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns bool - True on success of card activation, false otherwise

GiftCardVirtualReserve

Description Reserves a number of virtual gift cards for a transaction so they can be activated through the Sale_Insert method. This method ensures that Gift Cards exist for purchase and provides the gift card numbers to be used in the sale lines of the purchase.
Note that you can reserve multiple gift cards of the same amount (eg: 2 x $20) in a single call, but you must use separate calls for different amounts (eg: 2 x $20 and 1 x $50)

Parameters

Name	Type	Description
transactionNumber	string	The transaction number of the gift card sale
transactionDate	datetime	The transaction date time of the gift card sale
issueAmount	Decimal	The $ value of each gift card requested
quantity	Int	The total number of gift cards in the transaction of the same value
websiteUserCode	String	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Byte[] - gzipped UTF8 encoded byte array of xml data containing the list of gift cards reserved for purchasing in this transaction. The xml document conforms to GiftCardVirtualReserveResult.xsd. Sample Xml:

```
<reservedVouchers>
  <voucher reference1="GF1" tranDate="2012-03-22T00:00:00"
voucherNumber="GC000000000012051959" pin="3734" amount="100.0000" />
  <voucher reference1="GF1" tranDate="2012-03-22T00:00:00"
voucherNumber="GC000000000012051967" pin="5096" amount="100.0000" />
  <voucher reference1="GF1" tranDate="2012-03-22T00:00:00"
voucherNumber="GC000000000012051975" pin="5123" amount="100.0000" />
</reservedVouchers>
```

GiftCardVirtualReserveIssue

Description The preferred method of activation for gift cards is via the Sale_Insert method. However if sales are imported on a delay and a virtual gift card needs to be issued immediately use this method. You should still import the gift card details through Sale_Insert afterwards.

Parameters

Name	Type	Description
transactionNumber	string	The transaction number of the gift card sale – must match the transactionNumber in the Sale_Insert
transactionDate	datetime	The transaction date time of the gift card sale – must match the transactionDate in the Sale_Insert
voucherNumber	string	The gift card number to issue

branchCode	String	The branchCode the sale will be imported against
tillNumber	String	The Till the sale will be imported against
websiteUserCode	String	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns A Byte[] - gzipped UTF8 encoded byte array of xml data containing the list of gift cards reserved in this transaction and their issued status. Sample Xml:

```
<issuedVouchers>
  <voucher voucherNumber="GC000000000012051959" pin="3734" amount="100.0000"
expiryDate="2014-03-22T00:00:00" status="Issued" />
  <voucher voucherNumber="GC000000000012051960" pin="3734" amount="100.0000"
expiryDate="2014-03-22T00:00:00" status="Issued but not active" />
  <voucher voucherNumber="GC000000000012051961" pin="3734" amount="100.0000"
expiryDate="2014-03-22T00:00:00" status="Reserved and not Issued" />
</issuedVouchers>
```

Customer Communications

CustomerCommunication_GetAvailable

Description Gets a list of available communications as configured in EEE by the user

Parameters

Name	Type	Description
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Byte[]. GZipped stream of Xml.

E.g.

```
<AvailableCommunication xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Communication xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Id="54"
Description="Test Web Service" />
  <Communication xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Id="55"
Description="web svc" />
</AvailableCommunication>
```

CustomerCommunication_Get

Description Gets the customer details of the customers attached to the communication

Parameters

Name	Type	Description
communicationBatchId	Int	The Id of the communication
communicationBatchDescription	String	The description of the communication (optional – only used if the batch id is not available)
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Byte[]. GZipped stream of Xml.

CustomerCommunication_AcknowledgeGet

Description Acknowledges that the customer records have been retreived for the specified communication batch

Parameters

Name	Type	Description
communicationBatchId	Int	The Id of the communication
websiteUserCode	string	Usercode (see user permissions)
websitePassword	string	Password (see user permissions)

Returns Boolean. True if acknowledgement successful

Appendix A – Exceptions

Web Method	Exception	Cause	Solution
All methods	NotLicencedException	The EEE installation is not licenced for WebSiteIntegration	Contact Magenta Support
CustomerRegistration	A customer record already exists with this email address	A customer record already exists with this email address	Email addresses must be unique in EEE.
	A customer code must be supplied when Auto-Generate Customer Numbers is not enabled in EEE	The Auto –generate customer numbers option is turned off	Turn the option on in EEE or supply data in the CustomerCode field
	The customer code cannot be blank	A customer code was not passed on a new customer and the Auto –generate customer numbers option is turned off	Turn the option on in EEE or supply data in the CustomerCode field
CustomerConfirmation **CustomerConfirmationWithDetails**	The supplied email id is invalid	The email id (which is a GUID) is not a valid guid.	It may be that the customer copied and pasted the confirmation link into their browser. They need to retry.
	The Email Id passed is not recognised. Confirmation was not successful	The email id was not located in the database	It may be that the customer copied and pasted the confirmation link into their browser. They need to retry.
CustomerDetails **CusotmerDetailsWithParameters**	An email address or loyalty card number must be supplied	An email address or loyalty card number must be supplied	Supply required details
	Either a password, date of birth, surname, postcode, phone or mobile number must be supplied for a details enquiry	Either a password, date of birth, surname, postcode, phone or mobile number must be supplied for a details enquiry	Supply required details
	Customer identity not verified. The loyalty card number or email address and at least 2 other items must be supplied correctly	Customer identity not verified. The loyalty card number or email address and at least 2 other items must be supplied correctly	Supply required details
CustomerValidateLogin	An email address or loyalty card number must be supplied		
	The email address or password supplied is invalid or the email address has not been confirmed		
	The email address supplied has not yet been confirmed		
CustomerValidateLoginOffline	A loyalty card number must be supplied		
	The loyalty card does not exist or has expired		
	Customer identity not verified. The loyalty card number and at least 2 other items must be supplied correctly		
CustomerForgotPassword	The email address or loyalty card number supplied is		

	invalid
	The email address has not been verified
CustomerEmailExists	An email address is required.
CouponRedemption	A customer code is required.
	A coupon code is required.
	A promotion coupon id or a coupon code must be specified
	A valid promotion coupon must be specified. The coupon entered does not exist or has expired
CouponValid	A customer code is required.
	A coupon code is required.
CouponsAvailableToCustomer	A customer code is required.

Appendix B – EEE Configuration Options

Web service user permissions

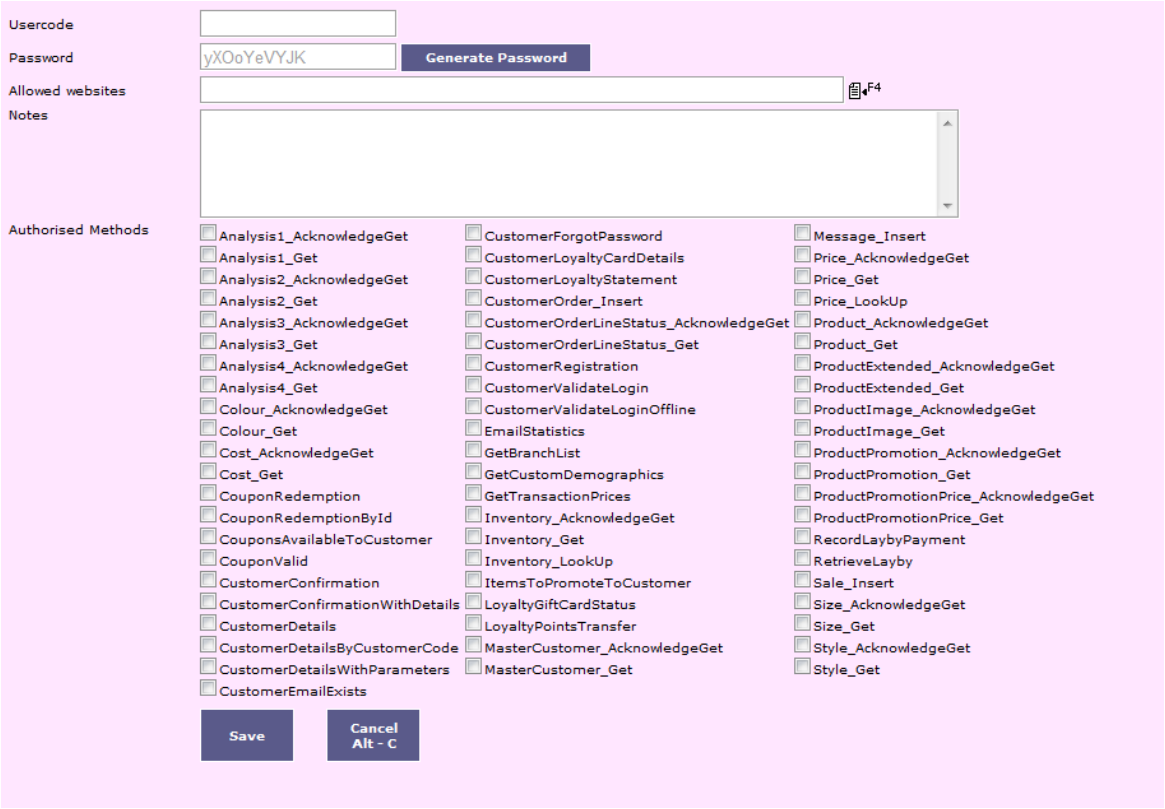
EEE clients can restrict access to the EEE web service by setting which methods a website can access.

Prior to enabling the configuration option "Use EEE web service user authorisation" any existing consumers of the EEE web service must have:

1. **been issued a User code and Password (see below)**
2. **been given access to the methods that they currently consume**
3. **been assigned the WEBSITE codes they can access (see allowed websites below)**
4. **implemented the use of the EEEWebServiceAuth.asmx web service and the use of the user code and password parameters on the web service methods.**

To configure EEE web service users, go to EEE->Utilities->Configuration->EEE Web Service Users

Click Add to see the screen below

Enter a user code, any applicable notes and select the methods that this user can access. If multiple website support is enabled then you must also specify which website codes this user can access.

Copy the password at this point as once it is saved it will not be re-displayed. If the password is lost by the consumer or by the EEE user then a new password must be generated using the Generate Password button