Web Service Application Programming Interface Documentation

20.05.2024
Version 4.1.3.14
London & Zurich Ltd

Table of Contents

Overview

Security

The web services are accessed over HTTPS using SSL. TLS 1.1 and 1.2 are the supported transport layer protocols. TLS 1.0 and SSL3 and below are not available for use.

Each web service user will be issued with a security code, up to 32 characters in length which, in association with a group number, can be used for web service authentication.

All web service requests referred to in this document must include a Security data structure named Credentials, which should contain a group number and its corresponding security code.

Test Environment

There is a separate environment that can be used for testing web service methods. This can be used to test requests and responses, however it does not similulate processes that occur in the production system, such as:

- Generating collections from schedules
- Direct Debit Submissions
- Generating BACS Reports/Failed Collections/Indemnity Claims
- Reviewing or accepting Pending Customer Account changes

Messages that would be forwarded outside the system (e.g. to BACS for processing) are not created by the test environment.

Time-Constrained Data

Incoming data is processed in real time. BACS collection requests, BACS Reports processing and client payments are performed on a daily basis. When performing searches on data expected for the current day, it is recommended that queries be made after 9:30am.

Method Request Sizes

There is a limit to the size of a request, 9048576 bytes. If you find you are creating requests that exceed that size, please split the requests up. The header response will contain Request Entity Too Large

Method Response Sizes

Depending upon the exact contents of a request, a web service method can return a large amount of data. A caller should be prepared to handle a large response size under the relevant circumstances.

Code Examples

The XML examples in this document have been created using an open-source program called SoapUI (https://www.soapui.org/), to illustrate the XML structure for web service requests and responses. Optional parameters can either be populated or omitted from the call.

Web Service Locations

The Live web services can be accessed from this web address:

https://webservices.landz.co.uk/wcf/4.0/managed.svc
with the wsdl accessible at:
https://webservices.landz.co.uk/wcf/4.0/managed.svc?wsdl

The Test web services can be accessed from this web address:

https://test.landz.co.uk/wcf/4.0/managed.svc

With the wsdl accessible at:

https://test.landz.co.uk/wcf/4.0/managed.svc?wsdl

Base Class Properties

All web service methods expect a request parameter which is a data structure that extends from the RequestBase data structure. Some fields of RequestBase are mandatory and some are optional.

All web service methods respond to requests with responses which are data structures that extend the ResponseBase data structure.

RequestBase

Credentials Property

The Credentials data structure is a mandatory part of the RequestBase data structure, so every web service method request must include Credentials. Each web service method called will use this information to check that the caller has access to the requested method and is authorised to access data for the Group.

Property	Length	Required
SecurityCode	29, 30 or 32 characters	Yes
Group	4 Numerical Characters	Yes

Example

```
<Credentials>
 <Group>4000</Group>
 <SecurityCode>TEST1234TEST1234TEST1234TEST1234</SecurityCode>
</Credentials>
```

Scope

This document outlines details and instructions for use of the web services available to users of London & Zurich's Direct Debit system.

The services are written using Microsoft .NET technologies, based on XML messages (eXtensible Mark-up Language) and the SOAP (Simple Object Access Protocol) communications interface. The service methods can be used with non-Microsoft technologies[1].

RemoveErrorDescription

The RemoveErrorDescription field can help to reduce the network traffic in a production environment. This is an optional field, taking a Boolean value ("true" or "false"). If a request contains a RemoveErrorDescription field with a value of "true" then the response to the request will not include

verbose descriptions with any error codes it contains. If the RemoveErrorDescription field's value is set to "false", or if the field is omitted, then verbose descriptions will accompany any errors returned in the response.

In order to remove verbose error strings in a production environment include the following in request data structures:

```
<RemoveErrorDescription>true</RemoveErrorDescription>
```

Examples

An example of a web service method response to a request that had RemoveErrorDescription set to "true":

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetCustomerStatusResponse xmlns="https://webservices.landz.co.uk">
   <GetCustomerStatusResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
     <b:string>003</b:string>
     <b:string>004</b:string>
    </a:Errors>
    <a:Message>Authentication Failed, contact London and Zurich for assistance</a:Message>
    <a:ResponseCode>1</a:ResponseCode>
    <a:CustomerStatuses i:nil="true"/>
   </GetCustomerStatusResult>
  </GetCustomerStatusResponse>
 </s:Body>
</s:Envelope>
```

An example of a web service method response to a request that had RemoveErrorDescription set to "false" or had the RemoveErrorDescription field omitted:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetCustomerStatusResponse xmlns="https://webservices.landz.co.uk">
   <GetCustomerStatusResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
     <b:string>003: Security code is the wrong length</b:string>
     <b:string>004: Group Number is invalid</b:string>
    </a:Errors>
    <a:Message>Authentication Failed, contact London and Zurich for assistance</a:Message>
    <a:ResponseCode>1</a:ResponseCode>
    <a:CustomerStatuses i:nil="true"/>
   </GetCustomerStatusResult>
  </GetCustomerStatusResponse>
 </s:Body>
</s:Envelope>
```

ResponseBase

ResponseCode

This is an integer code which indicates the execution status of a call.

If the ResponseCode is 0 then the requested web service method was executed. If the ResponseCode is anything other than 0 then the web service method was not executed.

Value	Meaning	Remark
0	Success	The web service method was successfully called
1	Authentication Failed	The combination of the supplied group number and security code did not pass authentication
2	System Failure	An unexpected error occurred – should this happen, please contact London & Zurich as soon as possible
3	Invalid Parameters	The values submitted in the web service call are not valid, e.g. malformed date-time values
4	Invalid Data	An action was requested that could not be performed due to a data-related issue, e.g. trying to suspend a customer's account when the customer is already suspended

Message

A string containing general information about the response.

Errors

If you receive a ResponseCode that is not 0 then the Errors field will be populated with data indicating reasons why the ResponseCode is non-zero. If the RemoveErrorDescription field in the request was set to "true" then only error numbers will be included; for example "002: No security code supplied" would be returned as "002". These Errors are typically related to the validation of the data in the request.

Common Error Codes

Value	Meaning	Remark
001	No Request received	This implies that the SOAP message did not contain the main body of the request
002	No security code supplied	The SecurityCode field was empty
003	Security code is the wrong length	The SecurityCode should be between 29 and 32 characters in length
004	Group Number is invalid	A group number should be 4 digits
005	Authentication failed, please contact London and Zurich	The combination of Group and SecurityCode does not pass the system's authentication checks

099	System Error	Check the Errors data structure returned in the response for more information

Date Handling

If a date or date range is supplied, each date will be checked and could elicit warnings from the following table. Additional date checks might be carried out depending on the purpose of the particular parameter – these will be noted in the relevant method's documentation.

Dates should be supplied following the ISO 8601 standard. This standard expresses an individual day using a YYYY-MM-DD format. A date that is supplied as an argument in any web service method call should not contain a time element – time elements may typically be ignored, but behaviour is not defined in such cases.

Examples of date-handling error codes, and their accompanying error messages, include the following:

Value	Meaning
102	Start Date is too far in the past (6 years or more)
103	Start Date is too far in the future (1 month or more)
104	End Date is too far in the past (6 years or more)
105	End Date is too far in the future (1 month or more)
106	Start Date is after End Date

Customer Reference Handling Codes

Many of the web service calls require a Customer Reference. A Customer Reference is the concatenation of a group number, a colon, and a DDI Reference. For example, if a group number is "1234" and the DDI Reference supplied is "ABA5678" then the corresponding CustomerRef value is "1234:ABA5678".

Examples of customer reference-handling error codes, and their accompanying error messages, include the following:

Value	Meaning
120	Customer Reference is missing
121	Customer Reference is too long
122	Customer Reference is too short
123	Customer Reference does NOT match Group
124	Customer does not exist
125	The Customer account is suspended – no actions regarding payments will take place when the customer is suspended (also known as end dated)

Examples

Example Request

This is an example of the information in the request data structure being invalid, prompting the Errors field to be populated in the response data.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
   <web:GetScheduledActivePayments>
     <web:request>
       <web1:Credentials>
         <web:Group>4000</web:Group>
         <web:SecurityCode>TEST1234TEST1234TEST1234TEST1234</web:SecurityCode>
       </web1:Credentials>
       <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
       <web1:CustomerRef>6000:test</web1:CustomerRef>
       <web1:EndDate>2015-01-01</web1:EndDate>
       <web1:ShowFuturePayments>false</web1:ShowFuturePayments>
       <web1:StartDate>2015-01-01</web1:StartDate>
     </web:request>
   </web:GetScheduledActivePayments>
 </soapenv:Body>
</soapenv:Envelope>
```

Example Response

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
   <GetScheduledActivePaymentsResponse xmlns="https://webservices.landz.co.uk">
     <GetScheduledActivePaymentsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
       <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
         <b:string>122: Customer Reference is too short</b:string>
         <b:string>123: Customer Reference does NOT match Group</b:string>
         <b:string>124: Customer does not exist</b:string>
       </a:Errors>
       <a:Message>Please check the errors to identify the cause</a:Message>
       <a:ResponseCode>3</a:ResponseCode>
       <a:AllPayments i:nil="true"/>
       <a:PaymentDefinitions i:nil="true"/>
     </GetScheduledActivePaymentsResult>
   </GetScheduledActivePaymentsResponse>
 </s:Body>
</s:Envelope>
```

Web Services

IsServiceAvailable

The IsServiceAvailable method provides a way to check whether the web service is available.

Request

A call to the IsServiceAvailable web service method is unique in that the caller does not need to supply Credentials.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk">
 <soapenv:Header/>
 <soapenv:Body>
   <web:IsServiceAvailable/>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

If this method returns then the service is available.

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
   <IsServiceAvailableResponse xmlns="https://webservices.landz.co.uk">
     <IsServiceAvailableResult>true</IsServiceAvailableResult>
   </IsServiceAvailableResponse>
 </s:Body>
</s:Envelope>
```

BankModulusAndSortcodeDDEnabledCheck

Call this method to perform a modulus check on a UK bank account number and sort code. This method will also display whether a sort code is direct debit enabled and/or electronic direct debit enabled.

Request

A BankModulusAndSortcodeDDEnabledCheck data structure contains all the fields of the RequestBase data structure in addition to the following fields:

Property	Type	Remark
AccountNo	String	The UK Bank Account number to be checked. Numerical characters only.
Sortcode	String	The UK Sort Code for the Bank Account. Numerical characters only.

Response

Property	Type	Remark
IsValid	Boolean	A value indicating whether the combination of account number and sort code passes modulus checking. Value can be “true” or “false”
ResponseCode	BankValidationResponseCode	An value indicating the response code - see table below
IsDirectDebitEnabled	Boolean	Value can be “true” or “false” This indicates whether the branch indicated by the supplied sort code is able to process Direct Debits. If this cannot be determined, this will also be “false”. This is advisory, based on the current Extended Industry Sort Code Directory data.
IsElectronicDirectDebitEnabled	Boolean	Value can be “true” or “false” This indicates whether the branch indicated by the supplied sort code is able to process Electronic Direct Debits. If this cannot be determined, this will also be “false” . This is advisory, based on the current Extended Industry Sort Code Directory data.

BankModulusAndSortCodeDDEnabledCheckResponse.ResponseCode enumeration

Value	Remark
SystemFailure	An unexpected error occurred
UnableToCheckAsForeignCurrencyAccount	The supplied account is not a sterling based bank account and therefore cannot be checked
AccountNumberOrSortCodeBlank	Either the bank account number or sort code was not supplied

ModulusCheckFail	The combination of the supplied bank account number and sort code does not pass the modulus check
AccountNumberNotNumeric	The supplied account number contains invalid characters (should only be numerical character [0-9])
AccountNumberLengthInvalid	The length of the supplied bank account number is invalid
SortCodeNotNumeric	The supplied sort code contains invalid characters (should only be numerical character [0-9])
SortCodeLengthInvalid	The length of the supplied sort code is invalid
DetailsOk	The supplied details successfully passed the requested checks (modulus at minimum, and the direct debit and electronic direct debit checks as well, if the checks were requested)
SortCodeCannotBeMatched	The supplied sort code cannot be matched to the EISCD database of sort codes
SortCodeNotWithinKnownRange	The supplied sort code cannot be found within the known ranges of sort codes for the purpose of modulus checking

If the input data is invalid for more than one reason, then only one reason will be returned.

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
   <BankModulusAndSortcodeDDEnabledCheckResponse xmlns="https://webservices.landz.co.uk">
     <BankModulusAndSortcodeDDEnabledCheckResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
       <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
       <a:Message>Success</a:Message>
       <a:ResponseCode>0</a:ResponseCode>
       <a:BankModulusAndSortCodeDDEnabledCheckResponse
xmlns:b="http://schemas.datacontract.org/2004/07/Solar.BankValidation.Responses">
         <b:IsValid>false</b:IsValid>
         <b:ResponseCode>ModulusCheckFail</b:ResponseCode>
         <b:IsDirectDebitEnabled >false</b:IsDirectDebitEnabled >
         <b:IsElectronicDirectDebitEnabled >false</b:IsElectronicDirectDebitEnabled >
       </a:BankModulusAndSortCodeDDEnabledCheckResponse>
     </BankModulusAndSortcodeDDEnabledCheckResult>
   </BankModulusAndSortcodeDDEnabledCheckResponse>
 </s:Body>
</s:Envelope>
```

AddCustomer

Call the AddCustomer method to ask that a customer be added to the system.

The AddCustomer method takes an AddCustomerRequest data structure and returns an AddCustomerResponse data structure.

AddCustomerRequest extends the RequestBase data structure. AddCustomerResponse extends the ResponseBase data structure.

Request

An AddCustomerRequest data structure contains all the fields of the RequestBase data structure in addition to containing a Customer data structure.

All supplied string values will be trimmed of any leading and/or trailing spaces before processing.

Customer Entity

Property	Required	Remark
DDIReference[1]	Yes	The reference of the DDI that will be lodged with Bacs. Must be between 6 and 10 characters long, inclusive. May only contain alphanumeric characters. Must be unique across all customers in a single Group and/or using a single SUN[3]. NB: Bacs only allows upper case characters, so any alphabetic characters supplied in this string will be converted to upper case before processing.
CustomerName	Yes	The customer's name. Must be at least 1 character long. Any characters after the 200th will be ignored.
Address1	Yes	When customers call us, we need information to verify their identity. This data would also be used for addressing post, should that be required. Must be between 1 and 50 characters long, inclusive.
Address2	No	If this field is present its value must not be more than 50 characters long.
Address3	No	If this field is present its value must not be more than 50 characters long.
Address4	No	If this field is present its value must not be more than 50 characters long.
Postcode	Yes	Must be between 1 and 8 characters long, inclusive. NB: Any alphabetic characters supplied in this string will be converted to upper-case before processing.
ContactName	Yes	Must be at least 1 character long. Any characters after the 40th will be ignored.
Email	No	If this field is present its value must not be more than 200 characters long.
Telephone	Yes	Must be between 1 and 14 characters long, inclusive.
Mobile	No	If this field is present its value must not be more than 14 characters long.
DebitName	Yes	The name of the paying bank account holder. Must be between 1 and 18 characters long, inclusive. Can contain only alphanumeric characters and/or the characters " ", "-", ".", "&" and "/". NB: Any alphabetic characters supplied in this string will be converted to upper case upon reception.
Sortcode[2]	Yes	The sort code of the paying bank. May be 6 numeric characters or 3 pairs of 2 numeric characters separated by a minus sign.
AccountNo[2]	Yes	The paying bank account number. Must be 8 numeric characters.

CompanyNumber	No	If this field is present its value must not be more than 10 characters long.
Comment	No	Any comment, at the caller's discretion. If this field is present its value must not be more than 500 characters long.

1 It is not possible to change the DDIReference once the account has been created. If a mistake has been made, London & Zurich *may* be able to delete the customer before any charges are incurred. The customer will need to be added again.

2 The supplied bank sort code and account number will be checked for; sort code validity, account number modulus, direct debit compatibility and the existence of a bank with the given sort code. If any of the checks fail then the response will contain the error details.

3 The check for the uniqueness of DDIReferences across a single SUN is enabled by default but can be turned off on request.

Example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddCustomer>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:Customer>
     <web1:AccountNo>12345678</web1:AccountNo>
     <web1:Address1>1st line of address</web1:Address1>
     <web1:Address2>2nd line</web1:Address2>
     <web1:Address3></web1:Address3>
     <web1:CompanyNumber></web1:CompanyNumber>
     <web1:ContactName>Fred Bloggs</web1:ContactName>
     <web1:CustomerName>Widgets 4 U</web1:CustomerName>
     <web1:DDIReference>98989ABC</web1:DDIReference>
     <web1:DebitName>F Bloggs</web1:DebitName>
     <web1:Email>anyone@widget.com</web1:Email>
     <web1:Postcode>B1 2RX</web1:Postcode>
     <web1:Sortcode>1111</web1:Sortcode>
     <web1:Telephone>0121 234 79999</web1:Telephone>
    </web1:Customer>
   </web:request>
  </web:AddCustomer>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

An AddCustomerResponse data structure contains the fields of the RequestBase data structure in addition to the following;

Property	Type	Remark

CustomerRef	String	The Customer Reference of the newly-added customer. A Customer Reference comprises a Group and DDIReference separated by a colon as described above.

Example:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <AddCustomerResponse xmlns="https://webservices.landz.co.uk">
   <AddCustomerResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <a:Message>Success</a:Message>
    <a:ResponseCode>0</a:ResponseCode>
    <a:CustomerRef>4000:98989ABC</a:CustomerRef>
   </AddCustomerResult>
  </AddCustomerResponse>
 </s:Body>
</s:Envelope>
```

Note: The “Errors” array has no content in this example because the “Message” is “Success” and the “ResponseCode” is 0.

Additional Errors

Value	Meaning
130	No customer information supplied
131	DDIReference (Bacs Account Reference) is required
132	Customer name is required
133	Contact name is required
134	First line of address (Address1) is required
135	Address postcode is required
136	Telephone number is required
137	Bank Sortcode is required
138	Bank account number is required
139	Bank account holder's name (DebitName) is required
141	DDIReference is too short
142	DDIReference is too long
143	Address1 is too long
144	Address2 is too long
145	Address3 is too long
146	Address4 is too long
147	Postcode is too long
148	Bank account holder's name (DebitName) is too long
149	Email Address is too long

150	Telephone number is too long
151	Mobile number is too long
152	Comment is too long
153	Company Number is too long
154	DDIReference must contain only letters and numbers
155	Bank account holder's name (DebitName)
156	Bank account number should be numbers only and 8 characters long
157	Bank sortcode is not valid
158	Bank account number and sortcode combination does not pass modulus check
159	Bank sortcode not found
160	You cannot use test account details on live accounts
162	DDIReference already exists (<group:reference>)
163[1]	The DDIReference already exists within the same Service User Number

1 It is possible, if using a shared Service User Number (SUN), that there could be a duplication of the same DDIReference registered with BACS. This error (and check) is enabled by default but can be excluded on request.

UpdateCustomer

Call the UpdateCustomer method to ask that a customer record that is already on the system be updated to reflect the supplied data. The customer is identified by a combination of the value in the DDIReference field and the Group number that was supplied in the call's Credentials data structure.

The UpdateCustomer method takes an UpdateCustomerRequest data structure and returns an UpdateCustomerResponse data structure.

UpdateCustomerRequest extends the RequestBase data structure.

UpdateCustomerResponse extends the ResponseBase data structure.

UpdateCustomer uses similar data structures to AddCustomer but with an additional option to reinstate a customer. It is not possible to change a customer's DDIReference.

Request

An UpdateCustomerRequest data structure contains all the fields of the RequestBase data structure in addition to containing a ReinstateCustomer field, a ReinstateReason field and a Customer data structure.

Property	Required	Remark
ReinstateCustomer	No	Optional. If the account is suspended, set this to "true" to remove the suspended date and reinstate the customer.
ReinstateReason	Required only if ReinstateCustomer is set to "true".	If ReinstateCustomer is "true" then a comment must also be supplied. The supplied comment must not consist solely of whitespace.

Customer

Property	Required	Remark
DIReference	Yes	A customer is identified by a reference, which is formed from a concatenation of the group number, a colon and their DDIReference (the value returned when the customer was added via AddCustomer). The value supplied in this field will be used as part of that concatenation (along with the Group number supplied in the Credentials data structure) in order to identify the customer whose record is to be updated. Must be between 6 and 10 characters long, inclusive, after trimming. May only contain alphanumeric characters. NB: Alphabetic characters in this string will be converted to upper case before processing.
CustomerName	Yes	The customer's name. Must be at least 1 character long after trimming. Any characters after the 200th will be ignored.
Address1	Yes	When customers call us, we need information to verify their identity. This data would also be used for addressing post, should that be required. Must be between 1 and 50 characters long, inclusive, after trimming.
Address2	No	If this field is present its value must not be more than 50 characters long after trimming.
Address3	No	If this field is present its value must not be more than 50 characters long after trimming.
Address4	No	If this field is present its value must not be more than 50 characters long after trimming.
Postcode	Yes	Must be between 1 and 8 characters long, inclusive, after trimming. NB: Any alphabetic characters in this string will be converted to upper case before processing.
ContactName	Yes	Must be at least 1 character long after trimming. Any characters after the 40th, after trimming, will be ignored.
Email	No	If this field is present its value must not be more than 200 characters long after trimming. NB: Any alphabetic characters in this string will be converted to lower case before processing.
Telephone	Yes	Must be between 1 and 14 characters long, inclusive, after trimming.
Mobile	No	If this field is present its value must not be more than 14 characters long after trimming.
DebitName	Yes	The name of the paying bank account holder. Must be between 1 and 18 characters long, inclusive. Can contain only alphanumeric characters and/or the characters " ", "-", ".", "&" and "/". NB: Any alphabetic characters in this string will be converted to upper case, and the string trimmed, upon reception.
Sortcode[1]	Yes	The sort code of the paying bank. May be 6 numeric characters or 3 pairs of 2 numeric characters separated by a minus sign.
AccountNo[1]	Yes	The paying bank account number. Must be 8 numeric characters.

CompanyNumber	No	If this field is present its value must not be more than 10 characters long after trimming.
Comment	No	General comment. If this field is present its value must not be more than 500 characters long after trimming.

1 The supplied bank sort code and account number will be checked for; sort code validity, account number modulus, direct debit compatibility and the existence of a bank with the given sort code. If any of the checks fail then the response will contain the error details.

Response

The UpdateCustomerResponse data structure contains no further fields beyond those of the ResponseBase.

Additional Errors

Typical errors generated by calls to UpdateCustomer are the same as those for AddCustomer.

Notes

If a call to UpdateCustomer results in changes to bank details (DebitName, Sortcode or AccountNo) then the changes may take some time to propagate through the system. During this time, the Customer is said to have “pending” changes. The “GetCustomers” method offers access to a list of customers that have “pending” changes. A Customer may not have their details updated while the Customer has “pending” changes.

GetCustomerStatus

The GetCustomerStatus method allows a caller to obtain the *current* status and associated details of a list of customers. A customer's current status relates to whether or not the customer is currently considered as "suspended" on the London & Zurich system, a date on which the customer became suspended, the creation date of the customer's account and the most recent date that a Direct Debit Instruction was submitted on behalf of the customer.

The GetCustomerStatus method takes a GetCustomerStatusRequest data structure and returns a GetCustomerStatusRepsonse data structure.

GetCustomerStatusRequest extends the RequestBase data structure. GetCustomerStatusResponse extends the RepsonseBase data structure.

Request

A GetCustomerStatusRequest data structure contains all the fields of the RequestBase data structure in addition to containing a CustomerReference – an array of string values, each string being a customer reference for a customer:

Property	Required	Remark
CustomerReferences	Yes	An array of strings, each string being the Customer Reference of a customer

Example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:arr="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
 <soapenv:Header/>
 <soapenv:Body>
   <web:GetCustomerStatus>
     <web:request>
       <web1:Credentials>
         <web:Group>4000</web:Group>
         <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
       </web1:Credentials>
       <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
       <web1:CustomerReferences>
         <arr:string>4000:98989ABC</arr:string>
         <arr:string>4000:3646MA001</arr:string>
         <arr:string>4000:364ABD001</arr:string>
<arr:string>4000:40004000</arr:string>
       </web1:CustomerReferences>
     </web:request>
   </web:GetCustomerStatus>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

A GetCustomerStatusResponse data structure contains all the fields of the ResponseBase data structure in addition to containing a CustomerStatuses object, which contains an array of CustomerStatus objects

Property	Type	Remark
CustomerStatuses	Array of CustomerStatus objects	An array of CustomerStatus objects – a CustomerStatus object will exist for every supplied valid customer reference

CustomerStatus

Property	Type	Remark
CustomerCreationDate	DateTime	The datetime at which the customer's account was entered into the London & Zurich system
CustomerName	String	The name currently associated with the supplied Customer Reference.
CustomerReference	String	The customer reference to which these status details relate.
CustomerSuspendedDate	DateTime	The datetime that the Customer was suspended, if the Customer is suspended, otherwise NULL.
DDIReference	String	The reference to the direct debit instruction that is to be (or has been) lodged with BACS
DDIStatus	String	The current status of the Customer: 'Active' if this customer has a NULL CustomerSuspendedDate, or 'Inactive' otherwise.
MostRecentDDISubmissionDate	DateTime	The datetime that the most recent DDI lodgement request was sent to BACS, or NULL if no such request has been sent.

Example:

In the below example, four customer references were submitted. In the response object, one of the customer references produced an error (the customer reference did not exist) and the other three were matched successfully. This is reported in the ResponseCode as a failure, but any successfully matched customer references still have their status details returned.

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetCustomerStatusResponse xmlns="https://webservices.landz.co.uk">
   <GetCustomerStatusResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
     <b:string>124: Customer 4000:40004000 does not exist</b:string>
    </a:Errors>
    <a:Message>Please check the errors to identify the cause</a:Message>
    <a:ResponseCode>3</a:ResponseCode>
    <a:CustomerStatuses>
     <a:CustomerStatus>
```

```
        <a:CustomerCreationDate>2016-07-18T14:30:34.957</a:CustomerCreationDate>
        <a:CustomerName>Widgets 4 U</a:CustomerName>
        <a:CustomerReference>4000:98989ABC</a:CustomerReference>
        <a:CustomerSuspendedDate i:nil="true"/>
        <a:DDIReference>98989ABC</a:DDIReference>
        <a:DDIStatus>Active</a:DDIStatus>
        <a:MostRecentDdiSubmissionDate i:nil="true"/>
      </a:CustomerStatus>
      <a:CustomerStatus>
        <a:CustomerCreationDate>2010-11-01T00:00:00</a:CustomerCreationDate>
        <a:CustomerName>Mrs Vernell Vashon</a:CustomerName>
        <a:CustomerReference>4000:3646MA001</a:CustomerReference>
        <a:CustomerSuspendedDate>2013-10-27T06:31:30.983</a:CustomerSuspendedDate>
        <a:DDIReference>3646MA001</a:DDIReference>
        <a:DDIStatus>Inactive</a:DDIStatus>
        <a:MostRecentDdiSubmissionDate>2011-08-27T10:00:07.763</a:MostRecentDdiSubmissionDate>
      </a:CustomerStatus>
      <a:CustomerStatus>
        <a:CustomerCreationDate>2009-05-21T00:00:00</a:CustomerCreationDate>
        <a:CustomerName>Mrs Tennille Toussaint</a:CustomerName>
        <a:CustomerReference>4000:364ABD001</a:CustomerReference>
        <a:CustomerSuspendedDate i:nil="true"/>
        <a:DDIReference>364ABD001</a:DDIReference>
        <a:DDIStatus>Active</a:DDIStatus>
        <a:MostRecentDdiSubmissionDate>2011-08-27T10:00:08.077</a:MostRecentDdiSubmissionDate>
      </a:CustomerStatus>
    </a:CustomerStatuses>
  </GetCustomerStatusResult>
 </GetCustomerStatusResponse>
</s:Body>
</s:Envelope>
```


Notes

If a customer is “suspended” then no Direct Debit collections will be requested from that customer. A suspended customer must be reinstated in order for collections to recommence. See “ReinstateCustomer” for more details.

Remarks

In order to calculate whether a Customer’s DDI is lodged at Bacs it is possible to examine the CustomerStatus. If the MostRecentDDISubmissionDate is at least three working days ago and the DDIStatus is ‘Active’ then the Customer’s DDI is lodged and collections may be requested.

If a collection is requested (via a call to AddScheduledPayment) before the Customer’s DDI is successfully lodged then an error will be returned. The error will include an indication of the expected lodgement date.

GetEarliestCollectionDate

This method will return the earliest available collection date for a given CustomerRef

GetEarliestCollectionDateRequest extends the RequestBase data structure.

GetEarliestCollectionDateResponse extends the ResponseBase data structure.

Request

A GetEarliestCollectionDateRequest data structure contains all the fields of the RequestBase data structure in addition to containing a CustomerRef:

Property	Required	Remark
CustomerRef	No	The Customer Reference you wish to find the earliest collection date for, e.g. 4000:ABC123 If not supplied the call will assume it's a new account.

Response

A GetEarliestCollectionDateResponse data structure contains all the fields of the ResponseBase data structure in addition to containing an EarliestDate value

Property	Type	Remark
EarliestDate	Datetime	The earliest date of a collection that can be made, relating to the CustomerRef supplied in the Request object

Example:

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk"
xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
   <soapenv:Header/>
   <soapenv:Body>
      <web:GetEarliestCollectionDate>
         <web:request>
            <web1:Credentials>
               <web:Group>4000</web:Group>
               <web:SecurityCode>----</web:SecurityCode>
            </web1:Credentials>
            <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
            <web1:CustomerRef>4000:ABC123</web1:CustomerRef>
         </web:request>
      </web:GetEarliestCollectionDate>
   </soapenv:Body>
</soapenv:Envelope>
```

Response

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
   <s:Body>
      <GetEarliestCollectionDateResponse xmlns="https://webservices.landz.co.uk">
```

```
        <GetEarliestCollectionDateResult
xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
            <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
            <a:Message>Success</a:Message>
            <a:ResponseCode>0</a:ResponseCode>
            <a:EarliestDate>2021-11-24T00:00:00</a:EarliestDate>
        </GetEarliestCollectionDateResult>
      </GetEarliestCollectionDateResponse>
   </s:Body>
</s:Envelope>
```

GetLatestSubmissionDate

Similar to GetEarliestCollectionDate, GetLatestSubmissionDate will return the latest date a Client can add a Collection to the system, for a given CustomerRef, for it to successfully collect on a given date.

GetLatestSubmissionDateRequest extends the RequestBase data structure.

GetLatestSubmissionDateResponse extends the ResponseBase data structure.

Request

A GetLatestSubmissionDateRequest data structure contains all the fields of the RequestBase data structure in addition to containing a CustomerRef:

Property	Required	Remark
CustomerRef	No	The Customer Reference you wish to find the earliest collection date for, e.g. 4000:ABC123
CollectionDate	Datetime	The intended date you with the collection to occur on

Response

A GetLatestSubmissionDateResponse data structure contains all the fields of the ResponseBase data structure in addition to containing an EarliestDate value

Property	Type	Remark
LatestSubmissionDate	Datetime	The latest date a collection request can be added into the system, for a given CustomerRef, so that it collects on the Collection Date supplied in the Request object

Example:

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk"
xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
   <soapenv:Header/>
   <soapenv:Body>
      <web:GetLatestSubmissionDate>
         <web:request>
            <web1:Credentials>
               <web:Group>4000</web:Group>
               <web:SecurityCode>----</web:SecurityCode>
            </web1:Credentials>
            <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
            <web1:CollectionDate>2021-11-24</web1:CollectionDate>
            <web1:CustomerRef>4000:ABC123</web1:CustomerRef>
         </web:request>
      </web:GetLatestSubmissionDate>
   </soapenv:Body>
</soapenv:Envelope>
```

Response

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
   <s:Body>
      <GetLatestSubmissionDateResponse xmlns="https://webservices.landz.co.uk">
         <GetLatestSubmissionDateResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
            <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
            <a:Message>Success</a:Message>
            <a:ResponseCode>0</a:ResponseCode>
            <a:LatestSubmissionDate>2021-11-19T00:00:00</a:LatestSubmissionDate>
         </GetLatestSubmissionDateResult>
      </GetLatestSubmissionDateResponse>
   </s:Body>
</s:Envelope>
```

AddScheduledPayment

Call the AddScheduledPayment method to ask that a new scheduled payment (also known as a "Continuous Authority" or "CA") be added to the system for a customer.

The AddScheduledPayment method takes an AddScheduledPaymentRequest data structure and returns an AddScheduledPaymentResponse data structure.

AddScheduledPaymentRequest extends the RequestBase data structure.

AddScheduledPaymentResponse extends the ResponseBase data structure.

A schedule can be set up to request:

- A one-off, once-only payment
- A fixed number of payments, the first payment different to the rest
- Continuous payments, the first payment different to the rest
- A fixed number of payments, all the same value
- Continuous payments, all the same value

Each payment can have a different payment frequency, i.e. a continuous authority can be applied to collect every one month, one week, six months etc. The parameter for this is RegularPaymentFrequency.

The available options for the RegularPaymentFrequency field are:

RegularPaymentFrequency field value	Description
1	Monthly
3	Quarterly
6	Semi-annually (every 6 months)
12	Annually
101	Weekly
102	Every 2 weeks
104	Every 4 weeks

NumberOfRegularPayments is used to show how many same-valued payments are to be made for a continuous authority (the first payment may be for a different amount). Any number above 0 indicates the number of same-valued payments that will be made. A value of 0 represents a continuous schedule without a defined end-point, so payments will be collected at the specified interval until the continuous authority is cancelled.

Request

- A schedule that describes a single payment should be specified by including the fields; CustomerRef, FirstPaymentDate, FirstPaymentAmount

- A schedule that describes a series comprising 2 or more same-valued payments should be specified by including the following fields; CustomerRef, RegularPaymentStartDate, RegularPaymentAmount, RegularPaymentFrequency, NumberOfRegularPayments
- A schedule that describes a series comprising a first collection of one value, followed by a series of one or more same-valued collections[IS1] , should be specified by including the following fields; CustomerRef, FirstPaymentDate, FirstPaymentAmount, RegularPaymentStartDate, RegularPaymentAmount, RegularPaymentFrequency, NumberOfRegularPayments

Property	Required	Remark
CustomerRef	Yes	The unique identifier that is the Customer Reference. A customer is identified by a reference which is formed from a concatenation of the group number, a colon and the DDIReference that was supplied when AddCustomer was called to create the customer. (This is also the value returned when the customer was added via AddCustomer.)
FirstPaymentDate	Should be present when describing a series of collections where the first collection is for a different amount to subsequent collections, or if describing a single collection that is not part of a series.	Use for a single payment or an initial payment with recurring different payment value to the first. ISO 8601 format (YYYY-MM-DD).
FirstPaymentAmount	Should be present when describing a series of collections where the first collection is for a different amount to subsequent collections, or if describing a single collection that is not part of a series.	The amount that is to be collected on the FirstPaymentDate. The value of this field is treated as an amount in pounds sterling (GBP). The value of this field will be parsed as a decimal, expecting to receive some digits, followed by a ".", followed by 2 digits.
RegularPaymentStartDate	Should be present when describing a series of collections. Should not be present when describing a single collection.	The date that the first collection in the series of same-valued collections is due to take place. ISO 8601 format (YYYY-MM-DD). The day-part of the date must be in the range 01-28, inclusive. If every collection in the series of collections is for the same value then this date should be the date that the first collection is due. If the collection schedule has a first collection that has a different value to subsequent collections, then this date should represent the date of the first of the recurring collections

RegularPayment Amount	Should be present when describing a series of collections. Should not be present when describing a single collection.	The amount that is to be collected in the series of same-valued collections. The value of this field is treated as an amount in pounds sterling (GBP). The value of this field will be parsed as a decimal, expecting to receive some digits, followed by a ".", followed by 2 digits. If every collection in the series of collections is for the same value then this field should hold that value. If the series of collections has a first collection that has a different value to the subsequent collections then this field should hold the value requested during the subsequent collections. The value of the first collection is specified in the FirstPaymentAmount field (see below).
RegularPayment Frequency	Should be present when describing a series of collections. Should not be present when describing a single collection.	Refer to the table earlier in this section.
NumberOfRegularPayments	Should be present when describing a series of collections. Should not be present when describing a single collection.	Must be an integer and should not be negative. 0 represents an on-going series without a defined, finite number of collections (in which case, collections will continue until the CancelScheduledPayment method is called). A number greater than 0 represents the number of same-valued collections that will be made. [e.g. If a series is one collection of £20 followed by 3 collections of £10, the NumberOfRegularPayments is 3. If a series is 5 collections of £15, the NumberOfRegularPayments is 5. If a series is one collection of £5, followed by a recurring collection of £4 with no defined final payment then the NumberOfRegularPayments is 0. If a series is a single collection of £5 then the NumberOfRegularPayments element should not be present.]

Example: A one-off, once-only payment

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddScheduledPayment>
   <web:request>
    <web1:Credentials>
```

```
      <web:Group>4000</web:Group>
      <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
     </web1:Credentials>
     <web1:CustomerRef>4000:98989ABC</web1:CustomerRef>
     <web1:FirstPaymentAmount>12.50</web1:FirstPaymentAmount>
     <web1:FirstPaymentDate>2015-05-01</web1:FirstPaymentDate>
    </web:request>
   </web:AddScheduledPayment>
 </soapenv:Body>
</soapenv:Envelope>
```

Example: fixed number of payments per month, the first payment different to the rest

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddScheduledPayment>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:CustomerRef>4000:98989ABC</web1:CustomerRef>
    <web1:FirstPaymentAmount>13.50</web1:FirstPaymentAmount>
    <web1:FirstPaymentDate>2015-05-01</web1:FirstPaymentDate>
    <web1:NumberOfRegularPayments>5</web1:NumberOfRegularPayments>
    <web1:RegularPaymentAmount>10.00</web1:RegularPaymentAmount>
    <web1:RegularPaymentFrequency>1</web1:RegularPaymentFrequency>
    <web1:RegularPaymentStartDate>2015-06-15</web1:RegularPaymentStartDate>
   </web:request>
  </web:AddScheduledPayment>
 </soapenv:Body>
</soapenv:Envelope>
```

Example: Continuous payments quarterly, the first payment different to the rest

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddScheduledPayment>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:CustomerRef>4000:98989ABC</web1:CustomerRef>
    <web1:FirstPaymentAmount>16.50</web1:FirstPaymentAmount>
    <web1:FirstPaymentDate>2015-07-01</web1:FirstPaymentDate>
    <web1:NumberOfRegularPayments>0</web1:NumberOfRegularPayments>
    <web1:RegularPaymentAmount>12.00</web1:RegularPaymentAmount>
    <web1:RegularPaymentFrequency>3</web1:RegularPaymentFrequency>
```

```
    <web1:RegularPaymentStartDate>2015-08-15</web1:RegularPaymentStartDate>
   </web:request>
  </web:AddScheduledPayment>
 </soapenv:Body>
</soapenv:Envelope>
```

Example: A fixed number of payments, all the same value

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddScheduledPayment>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:CustomerRef>4000:98989ABC</web1:CustomerRef>
    <web1:NumberOfRegularPayments>10</web1:NumberOfRegularPayments>
    <web1:RegularPaymentAmount>15.00</web1:RegularPaymentAmount>
    <web1:RegularPaymentFrequency>3</web1:RegularPaymentFrequency>
    <web1:RegularPaymentStartDate>2015-12-11</web1:RegularPaymentStartDate>
   </web:request>
  </web:AddScheduledPayment>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

An AddScheduledPaymentResponse data structure contains the fields of the RequestBase data structure in addition to the following;

Property	Type	Remark
ID	Integer	The unique identifier of the scheduled payment that has been created. if there was an error in creating the scheduled payment (which would be indicated by a ResponseCode other than 0) then this value is not defined.

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
        <AddScheduledPaymentResponse xmlns="https://webservices.landz.co.uk">
        <AddScheduledPaymentResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:ID>3545331</a:ID>
        </AddScheduledPaymentResult>
        </AddScheduledPaymentResponse>
```

```
  </s:Body>
</s:Envelope>
```

Additional Errors

Examples of AddScheduledPayment-specific error codes, and their accompanying error messages, include the following:

Value	Meaning
130	Regular Payment Frequency value is invalid
131	First Payment Date lead time is too short: earliest collection date would be <DD-MM-YYYY>
132	First Payment Date supplied without First Payment Amount
133	First Payment Amount is less or equal to zero
134	First Payment Date is too far into the future (over a year)
135	Regular Payment Start Date lead time is too short: earliest collection date would be <DD-MM-YYYY>
136	Regular Payment Start Date supplied without Regular Payment Amount
137	Regular Payment Amount is less or equal to zero
138	Regular Payment Date is too far into the future (over a year)
139	Frequency is set as a one-off collection, but a value has been supplied for a regular number of payments
140	The Frequency is set as a one-off collection, but the first payment amount or first payment date is invalid
141	A regular payment frequency has been supplied, but the regular payment date or amount are invalid
142	A fixed number of Regular Payments has been specified, but the regular payment date or regular payment amount are invalid
143	The Payment Frequency is specified as a regular payment, but the regular payment date is blank
144	Regular Payment Start Date day is after the 28th
145	It is not possible to have the first payment and regular payment on the same day
146	First Payment Date cannot be after the Regular Payment Date
149	Number of regular payments specified as a one off payment but with a incorrect payment frequency value
150	Payment will clash with the payment due on the <date>
151	Payment exceeds the limit of payment amount. Maximum value is <amount>
152	Payments for the month exceeds the limit of payment amount. Maximum value is <amount>
153	Maximum Payments for the month has been exceeded. Maximum value is <amount> collections per month

AddCollectionSchedule

Call the AddCollectionSchedule method to ask that a new collection schedule (also known as a "Continuous Authority" or "CA" or "scheduled payment") be added to the system.

The AddCollectionSchedule method takes an AddCollectionScheduleRequest data structure and returns an AddCollectionScheduleResponse data structure.

AddCollectionScheduleRequest extends the RequestBase data structure.

AddCollectionScheduleResponse extends the ResponseBase data structure.

A schedule can be set up to request:

- A one-off, once-only collection
- A fixed number of collections, the first collection amount being different to the rest
- Continuous collections, the first collection amount being different to the rest
- A fixed number of collections, all for the same value
- Continuous collections, all for the same value

Each schedule can have a different collection frequency, i.e. a schedule can be created which will attempt to collect every one month, one week, six months etc. The parameter for this is RegularPaymentFrequency.

The available options for the RegularPaymentFrequency field are:

RegularPaymentFrequency field value	Description
1	Monthly
3	Quarterly
6	Semi-annually (every 6 months)
12	Annually
101	Weekly
102	Every 2 weeks
104	Every 4 weeks

NumberOfRegularPayments is used to show how many same-valued collections are to be made via a schedule (the first collection amount may be for a different amount to the rest). Any number above 0 indicates the number of same-valued collections that will be made. A value of 0 represents a continuous schedule without a defined end-point, so collections will be attempted at the specified interval until the schedule is cancelled.

Request

- A schedule that describes a single collection should be specified by including the fields; CustomerReference, FirstPaymentDate, FirstPaymentAmount

- A schedule that describes a series comprising 2 or more same-valued collections should be specified by including the following fields; CustomerReference, RegularPaymentStartDate, RegularPaymentAmount, RegularPaymentFrequency, NumberOfRegularPayments
- A schedule that describes a series comprising a first collection of one value, followed by a series of one or more same-valued collections, should be specified by including the following fields; CustomerReference, FirstPaymentDate, FirstPaymentAmount, RegularPaymentStartDate, RegularPaymentAmount, RegularPaymentFrequency, NumberOfRegularPayments

Property	Required	Remarks
CustomerReference	Yes	The unique identifier that is the Customer Reference. A customer is identified by a reference which is formed from a concatenation of the group number, a colon and the DDIReference that was supplied when AddCustomer was called to create the customer. (This is also the value returned when the customer was added via AddCustomer.)
FirstPaymentDate	Should be present when describing a series of collections where the first collection is for a different amount to subsequent collections, or if describing a single collection that is not part of a series.	Use for a single collection, or a schedule consisting of an initial collection with recurring collection values different to the first. ISO 8601 format (YYYY-MM-DD).
FirstPaymentAmount	Should be present when describing a series of collections where the first collection is for a different amount to subsequent collections, or if describing a single collection that is not part of a series.	The amount that is to be collected on the FirstPaymentDate. The value of this field is treated as an amount in pounds sterling (GBP). The value of this field will be parsed as a decimal, expecting to receive some digits, followed by a ".", followed by 2 digits.

RegularPaymentStartDate	Should be present when describing a series of collections. Should not be present when describing a single collection.	The date that the first collection in the series of same-valued collections is due to take place. ISO 8601 format (YYYY-MM-DD). The day-part of the date must be in the range **01-28**, inclusive. If every collection in the series of collections is for the same value then this date should be the date that the first collection is due. If the collection schedule has a first collection that has a different value to subsequent collections, then this date should represent the date of the first of the recurring collections
RegularPaymentAmount	Should be present when describing a series of collections. Should not be present when describing a single collection.	The amount that is to be collected in the series of same-valued collections. The value of this field is treated as an amount in pounds sterling (GBP). The value of this field will be parsed as a decimal, expecting to receive some digits, followed by a ".", followed by 2 digits. If every collection in the series of collections is for the same value then this field should hold that value. If the series of collections has a first collection that has a different value to the subsequent collections then this field should hold the value requested during the subsequent collections. The value of the first collection is specified in the FirstPaymentAmount field (see below).
RegularPaymentFrequency	Should be present when describing a series of collections. Should not be present when describing a single collection.	Refer to the table earlier in this section for possible values and their meanings.

NumberOfRegularPayments	Should be present when describing a series of collections. Should not be present when describing a single collection.	Must be an integer and should not be negative. 0 represents an on-going series without a defined, finite number of collections (in which case, collections will continue until the CancelScheduledPayment method is called). A number greater than 0 represents the number of same-valued collections that will be made. [e.g. If a series is one collection of £20 followed by 3 collections of £10, the NumberOfRegularPayments is 3. If a series is 5 collections of £15, the NumberOfRegularPayments is 5. If a series is one collection of £5, followed by a recurring collection of £4 with no defined final collection then the NumberOfRegularPayments is 0. If a series is a single collection of £5 then the NumberOfRegularPayments element should not be present.]
ExternalReference	No	This is an optional field which may, if present, contain any series of up to 40 characters at the caller's discretion. If present the string must be unique across all the caller's schedules. This reference is named "External" since it is not part of the London & Zurich system. Its only use is as a parameter in the webservice call to GetSchedulesByExternalReference(). Feel free to use this field as you see fit. *Note that the value will be trimmed of any leading or trailing whitespace characters before being saved. The 40 character limit applies after this trimming has taken place.*

Example: A one-off, once-only collection

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                  xmlns:web=https://webservices.landz.co.uk
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddCollectionSchedule>
   <web:request>
```

```
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:CustomerReference>4000:98989ABC</web1:CustomerReference>
    <web1:FirstPaymentAmount>12.50</web1:FirstPaymentAmount>
    <web1:FirstPaymentDate>2015-05-01</web1:FirstPaymentDate>
    <web1:ExternalReference>Our Ref = abc123xyz</web1:ExternalReference>
   </web:request>
  </web:AddCollectionSchedule>
 </soapenv:Body>
</soapenv:Envelope>
```

Example: A fixed number of monthly collections (five, at £10 a time), following a first collection which is for a different amount (£13.50)

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                  xmlns:web=https://webservices.landz.co.uk
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddCollectionSchedule>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:CustomerReference>4000:98989ABC</web1:CustomerReference>
    <web1:FirstPaymentAmount>13.50</web1:FirstPaymentAmount>
    <web1:FirstPaymentDate>2015-05-01</web1:FirstPaymentDate>
    <web1:NumberOfRegularPayments>5</web1:NumberOfRegularPayments>
    <web1:RegularPaymentAmount>10.00</web1:RegularPaymentAmount>
    <web1:RegularPaymentFrequency>1</web1:RegularPaymentFrequency>
    <web1:RegularPaymentStartDate>2015-06-15</web1:RegularPaymentStartDate>
    <web1:ExternalReference>some text</web1:ExternalReference>
   </web:request>
  </web:AddCollectionSchedule>
 </soapenv:Body>
</soapenv:Envelope>
```

Example: Continuous quarterly collections, the first collection amount differing from the rest (one for £16.50, followed by £12 each quarter)

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                  xmlns:web=https://webservices.landz.co.uk
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web: AddCollectionSchedule>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
```

```
      <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:CustomerRef>4000:98989ABC</web1:CustomerRef>
    <web1:FirstPaymentAmount>16.50</web1:FirstPaymentAmount>
    <web1:FirstPaymentDate>2015-07-01</web1:FirstPaymentDate>
    <web1:NumberOfRegularPayments>0</web1:NumberOfRegularPayments>
    <web1:RegularPaymentAmount>12.00</web1:RegularPaymentAmount>
    <web1:RegularPaymentFrequency>3</web1:RegularPaymentFrequency>
    <web1:RegularPaymentStartDate>2015-08-15</web1:RegularPaymentStartDate>
   </web:request>
  </web: AddCollectionSchedule>
 </soapenv:Body>
</soapenv:Envelope>
```

Example: A fixed number of quarterly collections (ten), all for the same amount (£15)

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                  xmlns:web=https://webservices.landz.co.uk
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web: AddCollectionSchedule>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:CustomerRef>4000:98989ABC</web1:CustomerRef>
    <web1:NumberOfRegularPayments>10</web1:NumberOfRegularPayments>
    <web1:RegularPaymentAmount>15.00</web1:RegularPaymentAmount>
    <web1:RegularPaymentFrequency>3</web1:RegularPaymentFrequency>
    <web1:RegularPaymentStartDate>2015-12-11</web1:RegularPaymentStartDate>
    <web1:ExternalReference>His birthday is on Wednesday the 23rd</web1:ExternalReference>
   </web:request>
  </web: AddCollectionSchedule>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

An AddCollectionScheduleResponse data structure contains the fields of the RequestBase data structure in addition to the following;

Property	Type	Remark
ID	Integer	The unique identifier of the collection schedule that has been created. if there was an error in creating the schedule - which would be indicated by a ResponseCode other than 0 - then this value is not defined.

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
   <AddCollectionScheduleResponse xmlns="https://webservices.landz.co.uk">
      <AddCollectionScheduleResult xmlns:a=http://schemas.datacontract.org/2004/07/WebApi
                                   xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
         <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
         <a:Message>Success</a:Message>
         <a:ResponseCode>0</a:ResponseCode>
         <a:ID>3545331</a:ID>
      </AddCollectionScheduleResult>
   </AddCollectionScheduleResponse>
  </s:Body>
</s:Envelope>
```

Additional Errors

Examples of AddCollectionSchedule-specific error codes, and their accompanying error messages, include the following:

Value	Meaning
130	Regular Payment Frequency value is invalid
131	First Payment Date lead time is too short: earliest collection date would be <DD-MM-YYYY>
132	First Payment Date supplied without First Payment Amount
133	First Payment Amount is less or equal to zero
134	First Payment Date is too far into the future (over a year)
135	Regular Payment Start Date lead time is too short: earliest collection date would be <DD-MM-YYYY>
136	Regular Payment Start Date supplied without Regular Payment Amount
137	Regular Payment Amount is less or equal to zero
138	Regular Payment Date is too far into the future (over a year)
139	Frequency is set as a one-off collection, but a value has been supplied for a regular number of payments
140	The Frequency is set as a one-off collection, but the first payment amount or first payment date is invalid
141	A regular payment frequency has been supplied, but the regular payment date or amount are invalid
142	A fixed number of Regular Payments has been specified, but the regular payment date or regular payment amount are invalid
143	The Payment Frequency is specified as a regular payment, but the regular payment date is blank
144	Regular Payment Start Date day is after the 28th
145	It is not possible to have the first payment and regular payment on the same day
146	First Payment Date cannot be after the Regular Payment Date

149	Number of regular payments specified as a one off payment but with a incorrect payment frequency value
150	Payment will clash with the payment due on the <date>
151	Payment exceeds the limit of payment amount. Maximum value is <amount>
152	Payments for the month exceeds the limit of payment amount. Maximum value is <amount>
153	Maximum Payments for the month has been exceeded. Maximum value is <amount> collections per month
154	The value of ExternalReference may not be more than 40 characters long
155	ExternalReference is not unique

GetCollectionsByDate

Call the GetCollectionsByDate method to retrieve a list of CollectionEntry data structures, each of which represents a Direct Debit collection.

The GetCollectionsByDate method takes a GetCollectionsByDateRequest data structure and returns a GetCollectionsByDateResponse data structure.

GetCollectionsByDateRequest extends the RequestBase data structure. GetCollectionsByDateResponse extends the ResponseBase data structure.

This method accepts an optional start date, an optional end date and an optional customer reference, using default values where necessary. This method returns CollectionEntry data structures containing details of all collection requests that were/will be due to be made against the given customer's (or customers') DDI(s) where the collection due date fell/falls within the supplied date range. The returned CollectionEntry objects represent both successful and failed collections, including collections that were successful prior to indemnity claims being raised.

Request

A GetCollectionsByDateRequest data structure contains the fields of the RequestBase data structure in addition to the following;

Property	Required	Remark
StartDate	No	If the StartDate field is supplied then the returned data will contain only those collections with a due date on or after the given date. If the StartDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.
EndDate	No	If the EndDate field is supplied then the returned data will contain only those collections with a due date on or before the given date. If the EndDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.

CustomerRef	No	If the CustomerRef field is supplied then the returned data will contain only those collections made against the customer with the given reference. If the CustomerRef field is not supplied then all collections for the entire group within the relevant time period will be returned. A customer is identified by a reference which is formed from a concatenation of the group number, a colon and the DDIReference that was supplied when AddCustomer was called to create the customer. (This is also the value returned when the customer was added via AddCustomer.)

Examples

Request - a specific customer of a client:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:GetCollectionsByDate>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:CustomerRef>4000:TEST123</web1:CustomerRef>
    <web1:EndDate>2015-04-30</web1:EndDate>
    <web1:StartDate>2015-01-01</web1:StartDate>
   </web:request>
  </web:GetCollectionsByDate>
 </soapenv:Body>
</soapenv:Envelope>
```

Request - all customers of a client:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
<soapenv:Header/>
<soapenv:Body>
 <web:GetCollectionsByDate>
  <web:request>
   <web1:Credentials>
    <web:Group>4000</web:Group>
    <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
   </web1:Credentials>
   <web1:EndDate>2011-12-19</web1:EndDate>
   <web1:StartDate>2011-12-19</web1:StartDate>
  </web:request>
 </web:GetCollectionsByDate>
</soapenv:Body>
</soapenv:Envelope>
```

Response

A GetCollectionsByDateResponse data structure contains the fields of the RequestBase data structure in addition to containing "Detail", which is an array of 0 or more CollectionEntry data structures.

Property	Type	Remark
Detail	Array of CollectionEntry	Each individual collection and its status.

CollectionEntry

Property	Type	Remark
CollectionID	Integer	The unique reference on the system for the collection.
CustomerRef	String	The customer reference.
CustomerName	String	Name of the customer.
CollectionDate	Date	The due date of the collection.
Amount	Decimal	The amount of the collection.
ProcessingStatus	String	See below.
StatusDescription	String	Description of the ProcessingStatus.
FailureCode	String	See below.
FailureDescription	String	The reason for the failure.
FailureReference	Integer	The key used to link back to the failure record if the collection failed.
MajorResult	String	deprecated
MajorDescription	String	deprecated
MinorResult	String	deprecated
MinorDescription	String	deprecated
ReconcileReference	Integer	An integer value that links the collection to a client payment
ReconcileDate	Date	The date on which the net payment to the client that relates to this collection was ready to be paid
ScheduledPaymentID	Integer	The key for the continuous authority that created this payment.

ProcessingStatus values and descriptions

The ProcessingStatus of a CollectionEntry indicates whether a collection request has been sent to Bacs and, if so, whether the request resulted in a successful collection or a failed collection attempt.

ProcessingStatus	StatusDescription	Interpretation
n	Unprocessed	No collection request was/will be sent to Bacs.
j	Processing	A collection request has been sent to Bacs.
H	Successful Transaction	This collection has succeeded and will be settled.
F	Failed Transaction	This collection failed. See FailureCode for details of the reasons for failure.

FailureCode values and descriptions

If the ProcessingStatus of a CollectionEntry is 'F' ("Failed Transaction") then FailureCode is an indication of the reason that the collection attempt failed. If the ProcessingStatus of a CollectionEntry if not 'F' then FailureCode is undefined.

FailureCode	FailureDescription
0	Refer To Payer
1	Instruction Cancelled
2	Payer Deceased
3	A/C Transferred to another Bank
4	Advance Notice Disputed
5	No Account
6	No Instruction
7	Amount Differs
8	Amount Not Yet Paid
9	Presentation Overdue
A	Originator Differs
B	Account Closed
I	Indemnity Claim
G	Group Request
X	CMS Request

MajorResult is a legacy indicator of whether or not London & Zurich sent a request to Bacs to perform this collection.

MinorResult is a legacy indicator of the reason that a collection request failed.

Response - a specific customer of a client

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetCollectionsByDateResponse xmlns="https://webservices.landz.co.uk">
   <GetCollectionsByDateResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <a:Message>Success</a:Message>
    <a:ResponseCode>0</a:ResponseCode>
    <a:Detail>
     <a:CollectionEntry>
      <a:Amount>113.4000</a:Amount>
      <a:CollectionDate>2011-08-08T00:00:00</a:CollectionDate>
      <a:CollectionID>11030376</a:CollectionID>
      <a:CustomerName>Mrs Holli Halberg</a:CustomerName>
      <a:CustomerRef>4000:364ACC001</a:CustomerRef>
      <a:FailureCode i:nil="true"/>
      <a:FailureDescription i:nil="true"/>
      <a:FailureReference i:nil="true"/>
      <a:MajorDescription>Processed Line</a:MajorDescription>
      <a:MajorResult>m</a:MajorResult>
      <a:MinorDescription>Successful Transaction</a:MinorDescription>
      <a:MinorResult>H</a:MinorResult>
```

```
    <a:ProcessingStatus>H</a:ProcessingStatus>
    <a:ReconcileDate>2011-08-11T00:00:00</a:ReconcileDate>
    <a:ReconcileReference>503</a:ReconcileReference>
    <a:ScheduledPaymentID>447811</a:ScheduledPaymentID>
    <a:StatusDescription>Successful Transaction</a:StatusDescription>
  </a:CollectionEntry>
  <a:CollectionEntry>
    <a:Amount>113.4000</a:Amount>
    <a:CollectionDate>2011-09-08T00:00:00</a:CollectionDate>
    <a:CollectionID>11030657</a:CollectionID>
    <a:CustomerName>Mrs Holli Halberg</a:CustomerName>
    <a:CustomerRef>4000:364ACC001</a:CustomerRef>
    <a:FailureCode i:nil="true"/>
    <a:FailureDescription i:nil="true"/>
    <a:FailureReference i:nil="true"/>
    <a:MajorDescription>Processed Line</a:MajorDescription>
    <a:MajorResult>m</a:MajorResult>
    <a:MinorDescription>Successful Transaction</a:MinorDescription>
    <a:MinorResult>H</a:MinorResult>
    <a:ProcessingStatus>H</a:ProcessingStatus>
    <a:ReconcileDate>2011-09-10T00:00:00</a:ReconcileDate>
    <a:ReconcileReference>400</a:ReconcileReference>
    <a:ScheduledPaymentID>447811</a:ScheduledPaymentID>
    <a:StatusDescription>Successful Transaction</a:StatusDescription>
  </a:CollectionEntry>
  <a:CollectionEntry>
    <a:Amount>40.0000</a:Amount>
    <a:CollectionDate>2011-11-10T00:00:00</a:CollectionDate>
    <a:CollectionID>11031233</a:CollectionID>
    <a:CustomerName>Mrs Holli Halberg</a:CustomerName>
    <a:CustomerRef>4000:364ACC001</a:CustomerRef>
    <a:FailureCode i:nil="true"/>
    <a:FailureDescription i:nil="true"/>
    <a:FailureReference i:nil="true"/>
    <a:MajorDescription>Processed Line</a:MajorDescription>
    <a:MajorResult>m</a:MajorResult>
    <a:MinorDescription>Successful Transaction</a:MinorDescription>
    <a:MinorResult>H</a:MinorResult>
    <a:ProcessingStatus>H</a:ProcessingStatus>
    <a:ReconcileDate>2011-11-12T00:00:00</a:ReconcileDate>
    <a:ReconcileReference>391</a:ReconcileReference>
    <a:ScheduledPaymentID i:nil="true"/>
    <a:StatusDescription>Successful Transaction</a:StatusDescription>
  </a:CollectionEntry>
  <a:CollectionEntry>
    <a:Amount>40.0000</a:Amount>
    <a:CollectionDate>2012-11-08T00:00:00</a:CollectionDate>
    <a:CollectionID>11034874</a:CollectionID>
    <a:CustomerName>Mrs Holli Halberg</a:CustomerName>
    <a:CustomerRef>4000:364ACC001</a:CustomerRef>
    <a:FailureCode i:nil="true"/>
    <a:FailureDescription i:nil="true"/>
    <a:FailureReference i:nil="true"/>
    <a:MajorDescription>Processed Line</a:MajorDescription>
    <a:MajorResult>m</a:MajorResult>
    <a:MinorDescription>Successful Transaction</a:MinorDescription>
```

```
            <a:MinorResult>H</a:MinorResult>
            <a:ProcessingStatus>H</a:ProcessingStatus>
            <a:ReconcileDate>2012-11-10T00:00:00</a:ReconcileDate>
            <a:ReconcileReference>311</a:ReconcileReference>
            <a:ScheduledPaymentID>767910</a:ScheduledPaymentID>
            <a:StatusDescription>Successful Transaction</a:StatusDescription>
          </a:CollectionEntry>
        </a:Detail>
      </GetCollectionsByDateResult>
    </GetCollectionsByDateResponse>
  </s:Body>
</s:Envelope>
```


Response - all customers of a client:


```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetCollectionsByDateResponse xmlns="https://webservices.landz.co.uk">
      <GetCollectionsByDateResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Detail>
          <a:CollectionEntry>
            <a:Amount>150.0000</a:Amount>
            <a:CollectionDate>2011-12-19T00:00:00</a:CollectionDate>
            <a:CollectionID>11031793</a:CollectionID>
            <a:CustomerName>Mrs Margurite Meachum</a:CustomerName>
            <a:CustomerRef>4000:364BUC001</a:CustomerRef>
            <a:FailureCode i:nil="true"/>
            <a:FailureDescription i:nil="true"/>
            <a:FailureReference i:nil="true"/>
            <a:MajorDescription>Processed Line</a:MajorDescription>
            <a:MajorResult>m</a:MajorResult>
            <a:MinorDescription>Successful Transaction</a:MinorDescription>
            <a:MinorResult>H</a:MinorResult>
            <a:ProcessingStatus>H</a:ProcessingStatus>
            <a:ReconcileDate>2011-12-23T00:00:00</a:ReconcileDate>
            <a:ReconcileReference>398</a:ReconcileReference>
            <a:ScheduledPaymentID>539353</a:ScheduledPaymentID>
            <a:StatusDescription>Successful Transaction</a:StatusDescription>
          </a:CollectionEntry>
          <a:CollectionEntry>
            <a:Amount>35.0000</a:Amount>
            <a:CollectionDate>2011-12-19T00:00:00</a:CollectionDate>
            <a:CollectionID>11031794</a:CollectionID>
            <a:CustomerName>Mrs Myriam Mcbee</a:CustomerName>
            <a:CustomerRef>4000:364CSE001</a:CustomerRef>
            <a:FailureCode i:nil="true"/>
            <a:FailureDescription i:nil="true"/>
            <a:FailureReference i:nil="true"/>
            <a:MajorDescription>Processed Line</a:MajorDescription>
            <a:MajorResult>m</a:MajorResult>
            <a:MinorDescription>Successful Transaction</a:MinorDescription>
            <a:MinorResult>H</a:MinorResult>
            <a:ProcessingStatus>H</a:ProcessingStatus>
```

```
        <a:ReconcileDate>2011-12-23T00:00:00</a:ReconcileDate>
        <a:ReconcileReference>398</a:ReconcileReference>
        <a:ScheduledPaymentID>539303</a:ScheduledPaymentID>
        <a:StatusDescription>Successful Transaction</a:StatusDescription>
      </a:CollectionEntry>
      <a:CollectionEntry>
        <a:Amount>25.0000</a:Amount>
        <a:CollectionDate>2011-12-19T00:00:00</a:CollectionDate>
        <a:CollectionID>11031804</a:CollectionID>
        <a:CustomerName>Mrs Ione Israel</a:CustomerName>
        <a:CustomerRef>4000:364DED001</a:CustomerRef>
        <a:FailureCode i:nil="true"/>
        <a:FailureDescription i:nil="true"/>
        <a:FailureReference i:nil="true"/>
        <a:MajorDescription>Unprocessed</a:MajorDescription>
        <a:MajorResult>n</a:MajorResult>
        <a:MinorDescription>Account is Ended</a:MinorDescription>
        <a:MinorResult>p</a:MinorResult>
        <a:ProcessingStatus>n</a:ProcessingStatus>
        <a:ReconcileDate>2011-12-23T00:00:00</a:ReconcileDate>
        <a:ReconcileReference>398</a:ReconcileReference>
        <a:ScheduledPaymentID>494172</a:ScheduledPaymentID>
        <a:StatusDescription>Unprocessed</a:StatusDescription>
      </a:CollectionEntry>
      <a:CollectionEntry>
        <a:Amount>43.7600</a:Amount>
        <a:CollectionDate>2011-12-19T00:00:00</a:CollectionDate>
        <a:CollectionID>11031795</a:CollectionID>
        <a:CustomerName>Mrs Natashia Ney</a:CustomerName>
        <a:CustomerRef>4000:364EDE001</a:CustomerRef>
        <a:FailureCode i:nil="true"/>
        <a:FailureDescription i:nil="true"/>
        <a:FailureReference i:nil="true"/>
        <a:MajorDescription>Processed Line</a:MajorDescription>
        <a:MajorResult>m</a:MajorResult>
        <a:MinorDescription>Successful Transaction</a:MinorDescription>
        <a:MinorResult>H</a:MinorResult>
        <a:ProcessingStatus>H</a:ProcessingStatus>
        <a:ReconcileDate>2011-12-23T00:00:00</a:ReconcileDate>
        <a:ReconcileReference>398</a:ReconcileReference>
        <a:ScheduledPaymentID>497420</a:ScheduledPaymentID>
        <a:StatusDescription>Successful Transaction</a:StatusDescription>
      </a:CollectionEntry>
    </a:Detail>
  </GetCollectionsByDateResult>
 </GetCollectionsByDateResponse>
</s:Body>
</s:Envelope>
```

GetBacsCustomerUpdates

Call the GetBacsCustomerUpdates method to retrieve a list of updates that Bacs have issued instructing London & Zurich to change one or more customers' details. Customer records are updated every working day prior to 9.30 am.

The GetCustomerDetails method offers a way to examine the current state of a Customer via its Customer Reference.

The GetBacsCustomerUpdates method takes a GetBacsCustomerUpdatesRequest data structure and returns a GetBacsCustomerUpdatesResponse data structure.

GetBacsCustomerUpdatesRequest extends the RequestBase data structure.
GetBacsCustomerUpdatesResponse extends the ResponseBase data structure.

Request

A GetBacsCustomerUpdatesRequest data structure contains the fields of the RequestBase data structure in addition to the following;

Property	Required	Remark
StartDate	No	If the StartDate field is supplied then the returned data will contain only those customer updates that were applied to our system on or after the given date. If the StartDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within the six years prior to today.
EndDate	No	If the EndDate field is supplied then the returned data will contain only those customer updates that were applied to our system on or before the given date. If the EndDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within the six years prior to today.

Response

A GetBacsCustomerUpdatesResponse data structure contains the fields of the RequestBase data structure in addition to containing "CustomerUpdates", which is an array of 0 or more BacsCustomerUpdate data structures.

BacsCustomerUpdate

Property	Type	Remark
CustomerRef	String	The Customer Reference of the customer whose record has been updated.

CurrentAccountNumber	String	The bank account number associated with the customer's DDI at the time of the update request.
CurrentSortCode	String	The sort code that was associated with the customer's DDI at the time of the update request.
DueDate	Date	When this customer information change should be carried out. (In most cases this date implies that the change is immediate.)
CustomerName	String	The name of the customer.
NewAccountNumber	String	If not blank then the current bank account number needs be changed to this.
NewDebitName	String	If not blank then this field holds the new debit name to be used.
NewSortCode	String	If not blank then the bank account needs to use this new sort code.
ReasonCode	String	The reason the Bacs update has been raised.
ReasonDescription	String	A description of the reason for the update.
ReportGenerationDate	Date	The date when Bacs created the update report.

Reason Codes

Code	Description
0	Instruction cancelled – refer to payer
1	Instruction cancelled by payer
2	Payer deceased
3	Instruction transferred to another bank/building society
5	No account
6	No instruction
7	DDI amount not zero
B	Account closed
C	Account/Instruction transferred to a different branch of a bank/building society
D	Advance notice disputed
E	Instruction amended
F	Invalid account type
G	Bank will not accept Direct Debits on account
H	Instruction expired
I	Payer reference is not unique
K	Instruction cancelled by bank
L	Incorrect payer's account details
M	Transaction code/User status incompatible
N	Transaction disallowed at payer's branch
O	Invalid reference
P	Payer's name not present
Q	Originator's name blank
R	Instruction re-instated

Example


```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetBacsCustomerUpdatesResponse xmlns="https://webservices.landz.co.uk">
   <GetBacsCustomerUpdatesResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <a:Message>Success</a:Message>
    <a:ResponseCode>0</a:ResponseCode>
    <a:CustomerUpdates>
     <a:BacsCustomerUpdate>
      <a:CurrentAccountNumber>06774744</a:CurrentAccountNumber>
      <a:CurrentSortCode>086090</a:CurrentSortCode>
      <a:CustomerName>Prof Dara Desiderio</a:CustomerName>
      <a:CustomerRef>4000:364CHI002</a:CustomerRef>
      <a:DueDate>2015-03-06T00:00:00</a:DueDate>
      <a:NewAccountNumber>06774744</a:NewAccountNumber>
      <a:NewDebitName>NSMSGHGCC AFLHEE</a:NewDebitName>
      <a:NewSortCode>086090</a:NewSortCode>
      <a:ReasonCode>3</a:ReasonCode>
      <a:ReasonDescription>Instruction transferred to another bank/building society</a:ReasonDescription>
      <a:ReportGenerationDate>2015-03-06T00:00:00</a:ReportGenerationDate>
     </a:BacsCustomerUpdate>
     <a:BacsCustomerUpdate>
      <a:CurrentAccountNumber>46238510</a:CurrentAccountNumber>
      <a:CurrentSortCode>871427</a:CurrentSortCode>
      <a:CustomerName>Mrs Verda Speight</a:CustomerName>
      <a:CustomerRef>4000:364OUT001</a:CustomerRef>
      <a:DueDate>2015-03-07T00:00:00</a:DueDate>
      <a:NewAccountNumber/>
      <a:NewDebitName>IQVRMQQG OFNKKVMI</a:NewDebitName>
      <a:NewSortCode>871427</a:NewSortCode>
      <a:ReasonCode>1</a:ReasonCode>
      <a:ReasonDescription>Instruction cancelled by payer</a:ReasonDescription>
      <a:ReportGenerationDate>2015-03-07T00:00:00</a:ReportGenerationDate>
     </a:BacsCustomerUpdate>
     <a:BacsCustomerUpdate>
      <a:CurrentAccountNumber>02355688</a:CurrentAccountNumber>
      <a:CurrentSortCode>309070</a:CurrentSortCode>
      <a:CustomerName>Miss Yu Yadon</a:CustomerName>
      <a:CustomerRef>4000:364PRO005</a:CustomerRef>
      <a:DueDate>2015-03-07T00:00:00</a:DueDate>
      <a:NewAccountNumber/>
      <a:NewDebitName>VVQRLLAJ VLIGLWG</a:NewDebitName>
      <a:NewSortCode>309070</a:NewSortCode>
      <a:ReasonCode>1</a:ReasonCode>
      <a:ReasonDescription>Instruction cancelled by payer</a:ReasonDescription>
      <a:ReportGenerationDate>2015-03-07T00:00:00</a:ReportGenerationDate>
     </a:BacsCustomerUpdate>
     <a:BacsCustomerUpdate>
      <a:CurrentAccountNumber>66374958</a:CurrentAccountNumber>
      <a:CurrentSortCode>089999</a:CurrentSortCode>
      <a:CustomerName>Mrs Malissa Maring</a:CustomerName>
      <a:CustomerRef>4000:364BEN001</a:CustomerRef>
      <a:DueDate>2015-03-08T00:00:00</a:DueDate>
```

```
            <a:NewAccountNumber/>
            <a:NewDebitName>MKRIYK PPMERZ</a:NewDebitName>
            <a:NewSortCode>089999</a:NewSortCode>
            <a:ReasonCode>1</a:ReasonCode>
            <a:ReasonDescription>Instruction cancelled by payer</a:ReasonDescription>
            <a:ReportGenerationDate>2015-03-08T00:00:00</a:ReportGenerationDate>
          </a:BacsCustomerUpdate>
          <a:BacsCustomerUpdate>
            <a:CurrentAccountNumber>12345112</a:CurrentAccountNumber>
            <a:CurrentSortCode>074456</a:CurrentSortCode>
            <a:CustomerName>Mrs Ofelia Oboyle</a:CustomerName>
            <a:CustomerRef>4000:364ARI001</a:CustomerRef>
            <a:DueDate>2015-11-03T00:00:00</a:DueDate>
            <a:NewAccountNumber/>
            <a:NewDebitName/>
            <a:NewSortCode>074456</a:NewSortCode>
            <a:ReasonCode>B</a:ReasonCode>
            <a:ReasonDescription>Account closed</a:ReasonDescription>
            <a:ReportGenerationDate>2015-11-27T00:00:00</a:ReportGenerationDate>
          </a:BacsCustomerUpdate>
        </a:CustomerUpdates>
      </GetBacsCustomerUpdatesResult>
    </GetBacsCustomerUpdatesResponse>
  </s:Body>
</s:Envelope>
```

GetCollectionByCollectionID

The GetCollectionByCollectionID method allows the caller to obtain details about a Collection by supplying its Collection ID.

Request

Property	Required	Remarks
CollectionID	Yes	The identifier of a collection.
GetSchedule	No	An optional Boolean value (True or False). Defaults to False. If supplied as True then the response contains a populated Schedule object. A response with a populated Schedule object will be about twice the size of one without.

Response

The response, over and above those in the base Response class, returns a Collection object and a Schedule object.

Property	Type	Remark
Collection	Data Structure	See the CollectionEntry object described under GetFailedCollectionsByDate. A FailureDate is additionally supplied.
Schedule	Data Structure	See the CollectionSchedule object described under GetSchedulesByCustomer. If the call contained no GetSchedule property, or had a GetSchedule value of False then this Schedule object will not be populated. A response with a populated Schedule object will be about twice the size of one without.

Additional Errors

Value	Meaning
165	Collection not found

Examples

An example of a call to the GetCollectionByCollectionID web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
```

```
  <soapenv:Body>
    <web:GetCollectionByCollectionId>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
          <web:SecurityCode>wxyzCDUH7VxyzTxGW876543219uvwxyz</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:CollectionId>49728000</web1:CollectionId>
        <web1:GetSchedule>true</web1:GetSchedule>
      </web:request>
    </web:GetCollectionByCollectionId>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetCollectionByCollectionID web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetCollectionByCollectionIdResponse xmlns="https://webservices.landz.co.uk">
      <GetCollectionByCollectionIdResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Collection>
          <a:Amount>26.7500</a:Amount>
          <a:CollectionDate>2024-04-05T00:00:00</a:CollectionDate>
          <a:CollectionID>49728000</a:CollectionID>
          <a:CustomerName>MRS A BEE </a:CustomerName>
          <a:CustomerRef>1234:113RXQLWO0</a:CustomerRef>
          <a:FailureCode i:nil="true"/>
          <a:FailureDescription i:nil="true"/>
          <a:FailureReference i:nil="true"/>
          <a:MajorDescription>Processed Line</a:MajorDescription>
          <a:MajorResult>m</a:MajorResult>
          <a:MinorDescription>Successful Transaction</a:MinorDescription>
          <a:MinorResult>H</a:MinorResult>
          <a:ProcessingStatus>H</a:ProcessingStatus>
          <a:ReconcileDate>2024-04-08T00:00:00</a:ReconcileDate>
          <a:ReconcileReference>1019087401</a:ReconcileReference>
          <a:ScheduledPaymentID>28316450</a:ScheduledPaymentID>
          <a:StatusDescription>Successful Transaction</a:StatusDescription>
          <a: FailureDate i:nil="true"/>
        </a:Collection>
        <a:Schedule>
          <a:AdvanceNoticeSentDate i:nil="true"/>
          <a:ClientId>1234</a:ClientId>
          <a:CollectionScheduleId>28316450</a:CollectionScheduleId>
          <a:CreatedDatetime>2024-04-01T04:24:23.94</a:CreatedDatetime>
          <a:CustomerEndDate i:nil="true"/>
          <a:CustomerId>1234:113RXQLWO0</a:CustomerId>
          <a:CustomerName>Mrs Ava Bee</a:CustomerName>
          <a:FirstPaymentAmount>26.75</a:FirstPaymentAmount>
          <a:FirstPaymentDate>2024-04-05T00:00:00</a:FirstPaymentDate>
          <a:LastPaymentAmount>26.75</a:LastPaymentAmount>
          <a:LastPaymentDate>2024-04-05T00:00:00</a:LastPaymentDate>
```

```
            <a:NextPaymentAmount>0</a:NextPaymentAmount>
            <a:NextPaymentDate>2024-04-05T00:00:00</a:NextPaymentDate>
            <a:PaymentType>One-Off</a:PaymentType>
            <a:RegularPaymentAmount>0</a:RegularPaymentAmount>
            <a:RegularPaymentDate i:nil="true"/>
            <a:RegularPaymentDay>0</a:RegularPaymentDay>
            <a:RegularPaymentFrequency>0</a:RegularPaymentFrequency>
            <a:RegularPaymentsExpectedCount>-1</a:RegularPaymentsExpectedCount>
            <a:RegularPaymentsReceivedCount>1</a:RegularPaymentsReceivedCount>
            <a:StatusId>2</a:StatusId>
            <a:StatusText>Completed</a:StatusText>
          </a:Schedule>
        </GetCollectionByCollectionIdResult>
      </GetCollectionByCollectionIdResponse>
    </s:Body>
</s:Envelope>
```

GetFailedCollectionsByDate

Call the GetFailedCollectionsByDate method to retrieve a list of collections that were requested but failed, where the due date of the collection fell within a specified date range.

The GetFailedCollectionsByDate method takes a GetFailedCollectionsByDateRequest data structure and returns a GetFailedCollectionsByDateResponse data structure.

GetFailedCollectionsByDateRequest extends the RequestBase data structure.
GetFailedCollectionsByDateResponse extends the ResponseBase data structure.

Request

A GetFailedCollectionsByDateRequest data structure contains the fields of the RequestBase data structure in addition to the following:

Property	Required	Remark
CustomerRef	No	If the CustomerRef field is supplied then the returned data will contain only those failed collection requests that were due to be made against the customer with the given reference. If the CustomerRef field is not supplied then all failed collections for the entire group within the relevant time period will be returned. A customer is identified by a reference which is formed from a concatenation of the group number, a colon and the DDIReference that was supplied when AddCustomer was called to create the customer. (This is also the value returned when the customer was added via AddCustomer.)
StartDate	No	If the StartDate field is supplied then the returned data will contain only those failed collections that had a due date on or after the given date. If the StartDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.
EndDate	No	If the EndDate field is supplied then the returned data will contain only those failed collections that had a due date on or before the given date. If the EndDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
<soapenv:Header/>
```

```
<soapenv:Body>
  <web:GetFailedCollectionsByDate>
    <web:request>
      <web1:Credentials>
        <web:Group>4000</web:Group>
        <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
      </web1:Credentials>
      <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
      <web1:EndDate>2016-05-01</web1:EndDate>
      <web1:StartDate>2016-01-01</web1:StartDate>
    </web:request>
  </web:GetFailedCollectionsByDate>
</soapenv:Body>
</soapenv:Envelope>
```

Response

A GetFailedCollectionsByDateResponse data structure contains the fields of the RequestBase data structure in addition to containing “Detail”, which is an array of 0 or more CollectionEntry data structures.

Property	Type
Detail	Array of CollectionEntry

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetFailedCollectionsByDateResponse xmlns="https://webservices.landz.co.uk">
      <GetFailedCollectionsByDateResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Detail>
          <a:CollectionEntry>
            <a:Amount>300.0000</a:Amount>
            <a:CollectionDate>2016-01-08T00:00:00</a:CollectionDate>
            <a:CollectionID>11051287</a:CollectionID>
            <a:CustomerName>Miss Minna Ange</a:CustomerName>
            <a:CustomerRef>4000:CON020</a:CustomerRef>
            <a:FailureCode>0</a:FailureCode>
            <a:FailureDescription>Refer To Payer</a:FailureDescription>
            <a:FailureReference>494270</a:FailureReference>
            <a:MajorDescription>Processed Line</a:MajorDescription>
            <a:MajorResult>m</a:MajorResult>
            <a:MinorDescription>Refer To Payer</a:MinorDescription>
            <a:MinorResult>0</a:MinorResult>
            <a:ProcessingStatus>F</a:ProcessingStatus>
            <a:ReconcileDate>2016-01-10T00:00:00</a:ReconcileDate>
            <a:ReconcileReference>424</a:ReconcileReference>
            <a:ScheduledPaymentID>7650198</a:ScheduledPaymentID>
            <a:StatusDescription>Failed Transaction</a:StatusDescription>
```

```
        </a:CollectionEntry>
        <a:CollectionEntry>
          <a:Amount>39.9800</a:Amount>
          <a:CollectionDate>2016-01-08T00:00:00</a:CollectionDate>
          <a:CollectionID>11051295</a:CollectionID>
          <a:CustomerName>Miss Terica Reidhead</a:CustomerName>
          <a:CustomerRef>4000:CRY001</a:CustomerRef>
          <a:FailureCode>1</a:FailureCode>
          <a:FailureDescription>Instruction Cancelled</a:FailureDescription>
          <a:FailureReference>494274</a:FailureReference>
          <a:MajorDescription>Processed Line</a:MajorDescription>
          <a:MajorResult>m</a:MajorResult>
          <a:MinorDescription>Instruction Cancelled</a:MinorDescription>
          <a:MinorResult>1</a:MinorResult>
          <a:ProcessingStatus>F</a:ProcessingStatus>
          <a:ReconcileDate>2016-01-10T00:00:00</a:ReconcileDate>
          <a:ReconcileReference>424</a:ReconcileReference>
          <a:ScheduledPaymentID>7650440</a:ScheduledPaymentID>
          <a:StatusDescription>Failed Transaction</a:StatusDescription>
        </a:CollectionEntry>
      </a:Detail>
    </GetFailedCollectionsByDateResult>
  </GetFailedCollectionsByDateResponse>
 </s:Body>
</s:Envelope>
```


CollectionEntry

Property	Type	Remark
CollectionID	Integer	The system's unique reference for the collection.
CustomerRef	String	The customer reference.
CustomerName	String	Name of the person.
CollectionDate	Date	The due date of the collection.
Amount	Decimal	The amount of the collection.
ProcessingStatus	String	'F' - see below.
StatusDescription	String	"Failed Transaction" – see below.
MinorResult	String	See below.
MinorDescription	String	deprecated
FailureCode	String	The code for reason for failure.
FailureDescription	String	The reason of the failure.
FailureReference	Integer	The key used to link back to the failure record if the collection failed.
MajorResult	String	deprecated
MajorDescription	String	deprecated
ReconcileReference	Integer	An integer value that links the collection to a client payment
ReconcileDate	Date	The date when London & Zurich reconciled the indemnity claim.
ScheduledPaymentID	Integer	The key for the continuous authority that created this payment.

ProcessingStatus values and descriptions

The ProcessingStatus of a CollectionEntry indicates whether a collection request has been sent to Bacs and, if so, whether the request resulted in a successful collection or a failed collection attempt. GetFailedCollectionsByDate will return only those collection attempts which failed, so the ProcessingStatus of the returned CollectionEntry objects will always be 'F' and its StatusDescription "Failed Transaction".

MinorResult values and descriptions

The MinorResult of CollectionEntry objects returned by GetFailedCollectionsByDate is an indicator of whether an indemnity claim was made in respect of the collection. If the MinorResult is an 'I' then an indemnity claim has been made in respect of the collection, otherwise no such indemnity claim has been made.

Consult the FailureCode to examine the reason for the failed collection. The FailureCode will represent the reason that the collection failed, when MinorResult is not 'I', or the reason for the indemnity claim, when MinorResult is 'I'.

FailureCode values and descriptions

If the ProcessingStatus of a CollectionEntry is 'F' ("Failed Transaction") then FailureCode is an indication of the reason that the collection attempt failed.

The range of possible values that may be returned in the FailureCode and FailureDescription fields depends on the value returned in the MinorResult field. When a CollectionEntry has the value "I" as its MinorResult then that CollectionEntry represents a collection that resulted in an indemnity claim. When a CollectionEntry does not have the value "I" as its MinorResult then that CollectionEntry does not represent a collection that resulted in an indemnity claim.

FailureCode values and descriptions when MinorResult is NOT "I"

FailureCode when MinorResult is NOT "I"	FailureDescription
0	Refer To Payer
1	Instruction Cancelled
2	Payer Deceased
3	A/C Transferred to another Bank
4	Advance Notice Disputed
5	No Account
6	No Instruction
7	Amount Differs
8	Amount Not Yet Paid
9	Presentation Overdue
A	Originator Differs
B	Account Closed

G	Group Request
X	CMS Request

FailureCode values and descriptions when MinorResult is "I"

FailureCode when MinorResult is "I"	FailureDescription
1	Amount and/or date of Direct Debit differs from advance notice
2	No advance notice was received by the payer
3	DDI cancelled by paying bank
4	Payer has cancelled DDI direct with service user
5	No instruction held. Payer disputes having given authority
6	Signature on DDI is fraudulent or not in accordance with account authorised signature(s).
7	Claim raised at service user's request after Direct Debit applied to payer's account.
8	Service user name disputed. Payer does not recognise service user collecting Direct Debit.

MajorResult is a legacy indicator of whether or not London & Zurich sent a request to Bacs to perform this collection.

GetPublicHolidays

Call the GetPublicHolidays method to obtain a list of forthcoming public UK holidays. No collections can take place on these days. Any collections due to be generated as a result of payment schedules will be collected on the next possible banking day. For example, if a schedule would generate an attempt to collect on a Saturday, the collection will actually be requested on the next banking day (probably the Monday).

The GetPublicHolidays method takes a GetPublicHolidaysRequest data structure and returns a GetPublicHolidaysResponse data structure.

GetPublicHolidaysRequest extends the RequestBase data structure. GetPublicHolidaysResponse extends the ResponseBase data structure.

Request

The GetPublicHolidaysRequest data structure contains no further fields beyond those of the RequestBase.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:GetPublicHolidays>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
   </web:request>
  </web:GetPublicHolidays>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

Property	Type
Holidays	Array of dateTime values representing public holiday days. (The time part of the returned data should be ignored.)

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
```

```
   <GetPublicHolidaysResponse xmlns="https://webservices.landz.co.uk">
     <GetPublicHolidaysResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
       <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
       <a:Message>Success</a:Message>
       <a:ResponseCode>0</a:ResponseCode>
       <a:Holidays xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
         <b:dateTime>2016-08-29T00:00:00</b:dateTime>
         <b:dateTime>2016-12-26T00:00:00</b:dateTime>
         <b:dateTime>2016-12-27T00:00:00</b:dateTime>
         <b:dateTime>2017-01-02T00:00:00</b:dateTime>
         <b:dateTime>2017-04-14T00:00:00</b:dateTime>
         <b:dateTime>2017-04-17T00:00:00</b:dateTime>
         <b:dateTime>2017-05-01T00:00:00</b:dateTime>
         <b:dateTime>2017-05-29T00:00:00</b:dateTime>
         <b:dateTime>2017-08-28T00:00:00</b:dateTime>
         <b:dateTime>2017-12-25T00:00:00</b:dateTime>
         <b:dateTime>2017-12-26T00:00:00</b:dateTime>
       </a:Holidays>
     </GetPublicHolidaysResult>
   </GetPublicHolidaysResponse>
 </s:Body>
</s:Envelope>
```

GetIndemnitiesByDate

Call this method to obtain a list of indemnity claims which were raised between a supplied start date and end date. An indemnity claim is where the customer claims back money from a collection after the collection operation has been completed.

Request

Property	Type	Required	Remark
StartDate	Date	No	The date on or after which the indemnity was raised (not the date of collection). ISO 8601 format (YYYY-MM-DD) The date must be a date that falls within a time period beginning six years prior to today and ending one month from now. If the StartDate field is not supplied then it defaults to today's date.
EndDate	Date	No	The date on or before which the indemnity was raised (not the date of collection). ISO 8601 format (YYYY-MM-DD) The date must be a date that falls within a time period beginning six years prior to today and ending one month from now. If the EndDate field is not supplied then it defaults to today's date.

Response

Property	Type
Detail	Array of IndemnityEntry

IndemnitiyEntry

Property	Type	Remark
CustomerRef	String	The Customer Reference
CustomerName	String	The name of the customer
CollectionDate	Date	The date the collection occurred on
Amount	Decimal	Amount to be reclaimed
FailureReference	Integer	A link back to the original failure
ReportRaised	Date	The date when London & Zurich was notified of the claim
PayingBankReference	String	A reference to query the claim with the bank
ReconcileDate	Date	The date on which London & Zurich reconciled the indemnity claim.

ReconcileReference	Integer	The payment reference.
ReasonCode	String	The DDICA-related reason code
Reason	String	Description of the ReasonCode.

Indemnity Reason Codes

Code	Description
1	Amount and / or date of Direct Debit differ from Advance Notice.
2	No Advance Notice received by Payer/or the amount quoted is disputed.
3	DDI cancelled by paying bank.
4	Payer has cancelled DDI direct with service user.
5	No Instruction held. Payer disputes having given authority.
6	Signature on DDI is fraudulent or not in accordance with account authorised signature(s).
7	Claim raised at service users request after Direct Debit applied to payers account.
8	Service user name disputed. Payer does not recognise service user collecting Direct Debit.

GetAllIndemnitiesByCreatedDate

Call this method to obtain a list of indemnity claims which were raised between a supplied start date and end date. An indemnity claim is where the customer claims back money from a collection after the collection operation has been completed.

Note : This call returns same Indemnity Claims as returned by GetIndemnitiesByDate but Indeminity Claim Entity returned by this call differs than earlier and has more details.

Request

Property	Type	Required	Remark
StartDate	Date	No	The date on or after which the indemnity was raised (not the date of collection). ISO 8601 format (YYYY-MM-DD) The date must be a date that falls within a time period beginning six years prior to today and ending one month from now. If the StartDate field is not supplied then it defaults to today's date.
EndDate	Date	No	The date on or before which the indemnity was raised (not the date of collection). ISO 8601 format (YYYY-MM-DD) The date must be a date that falls within a time period beginning six years prior to today and ending one month from now. If the EndDate field is not supplied then it defaults to today's date.

Response

Property	Type
Detail	Array of IndemnityEntry

IndemnitiyEntry

Property	Type	Remark
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
CustomerId	String	The Customer Reference
CustomerName	String	The name of the customer
CollectionDate	Date	The date the collection occurred on
CollectionId	Integer	Unique Id of Collection related to the Indemnity Claim.
Amount	Decimal	Amount to be reclaimed
IndeminityClaimId		

IndemnityClaimReason	Date	The date when London & Zurich was notified of the claim
IndemnityClaimReasonCode	String	A reference to query the claim with the bank
ReconcileDate	Date	The date on which London & Zurich reconciled the indemnity claim.
ReconcileReference	Integer	The payment reference.
ReasonCode	String	The DDICA-related reason code
Reason	String	Description of the ReasonCode.

GetScheduledActivePayments

This method returns a list of active collection schedules, with the option of also retrieving a list of future collections that are due to occur within a supplied date range.

The <CustomerRef> field is an optional field which can be used to restrict the result set to a specific customer. If omitted, the results will include details for all customers.

The field <ShowFuturePayments> is used to determine whether the response should include collections that are scheduled to be requested in the future, and takes a Boolean value of true or false.

If a forecast of future collection requests is required, the service will return up to 2 years' worth of data. If the collection schedule is fixed term and starts less than 2 years in the future, the returned value will be based on the fixed term length.

Request

Property	Type	Required	Remark
CustomerRef	String	No	If supplied then the results are limited to the specified customer, otherwise the results include details for all customers.
ShowFuturePayments	Boolean	No	Supply the value "true" to request when collections will be made in the future based on the schedule's definition. If not supplied, the method will assume the value 'false'.
StartDate	Date	No	If ShowFuturePayments is set to 'true' then the returned forecast of future collection requests will cover the date range given by StartDate and EndDate. Use ISO 8601 format (YYYY-MM-DD). If StartDate is not supplied then it will take the default value of the current date. If EndDate is not supplied, and ShowFuturePayments is set to 'true', then all planned collection requests on or after the StartDate will be returned. (StartDate and EndDate do not influence the active schedules that the method returns.)
EndDate	Date	No	

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
```

```
  <web:GetScheduledActivePayments>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:EndDate>2016-06-01</web1:EndDate>
    <web1:ShowFuturePayments>true</web1:ShowFuturePayments>
    <web1:StartDate>2016-01-01</web1:StartDate>
   </web:request>
  </web:GetScheduledActivePayments>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

Property	Type
PaymentDefinitions	Array of ScheduledPaymentDefinition
AllPayments	Array of ScheduledPayment

ScheduledPaymentDefinition

Property	Type	Remark
ScheduledPaymentID	Integer	The identifier of the schedule
CustomerRef	String	The Customer Reference
CustomerName	String	The name of the customer
FirstPaymentAmount	Decimal	As described in AddScheduledPayment
FirstPaymentDate	Date (nullable)	As described in AddScheduledPayment
RegularPaymentAmount	Decimal	As described in AddScheduledPayment
RegularPaymentStartDate	Date (nullable)	As described in AddScheduledPayment
RegularPaymentFrequency	Integer	As described in AddScheduledPayment
NumberOfRegularPayments	Integer	As described in AddScheduledPayment
NextPaymentDate	Date	The date the next payment is due
NextAmount	Decimal	The amount due on the next payment
LastPaymentDate	Date (nullable)	The date that the last payment was processed by London & Zurich
LastAmount	Decimal	The amount that was requested the last time this schedule generated a collection request.
NumberOfPaymentsRecieved	Integer	How many payments have been processed so far on this schedule
ScheduleSetupDate	Date	Date when the schedule was set up.

ScheduledPayment

Property	Type	Remark

CustomerRef	String	The Customer Reference
CollectionDate	Date	The date the collection is due
Amount	Decimal	The amount to be collected

GetScheduledFuturePayments

Call this method to obtain a list of planned collection requests from one or more customers.

Request

Property	Type	Required	Remark
CustomerRef	String	No	If supplied then the results are limited to the specified customer, otherwise the results include details for all customers.
StartDate	Date	Yes	ISO 8601 format (YYYY-MM-DD)
EndDate	Date	Yes	

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:GetScheduledFuturePayments>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:EndDate>2016-12-31</web1:EndDate>
    <web1:StartDate>2016-08-01</web1:StartDate>
   </web:request>
  </web:GetScheduledFuturePayments>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

Property	Type
Payments	Array of ScheduledPayment

ScheduledPayment

Property	Type	Remark
CustomerRef	String	The Customer Reference
CollectionDate	Date	The date the collection is due
Amount	Decimal	The amount to be collected

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
```

```
 <s:Body>
   <GetScheduledFuturePaymentsResponse xmlns="https://webservices.landz.co.uk">
     <GetScheduledFuturePaymentsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
       <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
       <a:Message>Success</a:Message>
       <a:ResponseCode>0</a:ResponseCode>
       <a:Payments>
         <a:ScheduledPayment>
           <a:Amount>35.00</a:Amount>
           <a:CollectionDate>2016-08-01T00:00:00</a:CollectionDate>
           <a:CustomerRef>4000:HSSSMI001</a:CustomerRef>
         </a:ScheduledPayment>
         <a:ScheduledPayment>
           <a:Amount>1200.00</a:Amount>
           <a:CollectionDate>2016-08-08T00:00:00</a:CollectionDate>
           <a:CustomerRef>4000:HSSSMI001</a:CustomerRef>
         </a:ScheduledPayment>
         <a:ScheduledPayment>
           <a:Amount>20.00</a:Amount>
           <a:CollectionDate>2016-08-01T00:00:00</a:CollectionDate>
           <a:CustomerRef>4000:MLSUTTON</a:CustomerRef>
         </a:ScheduledPayment>
         <a:ScheduledPayment>
           <a:Amount>100.00</a:Amount>
           <a:CollectionDate>2016-08-01T00:00:00</a:CollectionDate>
           <a:CustomerRef>4000:MRS11011</a:CustomerRef>
         </a:ScheduledPayment>
       </a:Payments>
     </GetScheduledFuturePaymentsResult>
   </GetScheduledFuturePaymentsResponse>
 </s:Body>
</s:Envelope>
```

GetSchedulesByCustomer

The GetSchedulesByCustomer method allows the caller to obtain a list of schedules that belong to a supplied Customer.

Request

Property	Required	Remarks
CustomerRef	Yes	The Customer Reference

Response

If there are any Schedules that belong to the supplied Customer, they will be listed I the response.

Property	Type
Schedules	Array of CollectionSchedules

CollectionSchedule

Property	Type	Remark
AdvanceNoticeSentDate	Datetime	Contains the datetime the advance notice was sent for this schedule. Null if not sent
ClientId	Int	The Client ID that the schedule belongs to
CollectionScheduleId	Int	The identifier of the collection schedule
CreatedDatetime	Datetime	The datetime that the schedule was added to the system
CustomerEndDate	Datetime	The datetime that the Customer account was suspended. Null if the Customer is active
CustomerId	String	The Customer Reference that the schedule belongs to (also referred to as Account Ref), e.g. 4000:ABC123m
CustomerName	String	The Customer Account name
FirstPaymentAmount	Decimal	As described in AddScheduledPayment
LastPaymentAmount		As described in GetScheduledActivePayments
LastPaymentDate		As described in GetScheduledActivePayments
NextPaymentAmount		The next amount due to be Collected via this Schedule
NextPaymentDate		When the next Collection from this Schedule is due to be made

PaymentType	string	Can be either “One-Off” when there is only one Collection to be made from this Schedule, “Continuous” if the Schedule will collect until cancelled, or a number representing the number of Collections to be made by this Schedule.
RegularPaymentAmount	Decimal	As described in AddScheduledPayment
RegularPaymentDate	Date	As described in AddScheduledPayment under RegularPaymentStartDate
RegularPaymentDay	Int	This the day number in the month that regular Collections started on e.g. for a RegularPaymentDate of 21/02/2024 would be 21
RegularPaymentFrequency	String	As described under RegularPaymentFrequency in AddScheduledPayment
RegularPaymentsExpectedCount	Int	The number of Collections expected to be made from this Schedule: -1 if One Off, 0 if continuous, positive integer if a fixed number of Collections
RegularPaymentsReceivedCount	Int	The number of Collections that have been made by this Schedule
StatusId	Int	The integer status of the Schedule, see Status table below
StatusText	String	The text status of the Schedule, see Status table below

The possible Status integers are as follows.

Item2 field (Status) value	Description
1	Active
2	Completed
3	Cancelled
0	Unknown

Additional Errors

Additional errors will be the same as those described in “Customer Reference Handling” section

Examples

An example of a call to the GetSchedulesByCustomer web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
```

```
  <soapenv:Body>
    <web:GetSchedulesByCustomer>
      <web:request>
        <web1:Credentials>
          <web:Group>4000</web:Group>
          <web:SecurityCode>xxxxxxxxxxxxxxxxxx</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:CustomerRef>4000:ABC123</web1:CustomerRef>
      </web:request>
    </web:GetSchedulesByCustomer>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetSchedulesByCustomer web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetSchedulesByCustomerResponse xmlns="https://webservices.landz.co.uk">
      <GetSchedulesByCustomerResult xmlns:a=http://schemas.datacontract.org/2004/07/WebApi
                                    xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Schedules>
          <a:CollectionSchedule>
            <a:AdvanceNoticeSentDate i:nil="true"/>
            <a:ClientId>4000</b:ClientId>
            <a:CollectionScheduleId>1231231</a:CollectionScheduleId>
            <a:CreatedDatetime>2021-12-25T14:56:15</a:CreatedDatetime>
            <a:CustomerEndDate i:nil="true"/>
            <a:CustomerId>4000:ABC1235</a:CustomerId>
            <a:CustomerName>Mrs Mary Madre</a:CustomerName>
            <a:FirstPaymentAmount>350</a:FirstPaymentAmount>
            <a:FirstPaymentDate>2022-01-08T00:00:00</a:FirstPaymentDate>
            <a:LastPaymentAmount>350</a:LastPaymentAmount>
            <a:LastPaymentDate>2022-01-08T00:00:00</a:LastPaymentDate>
            <a:NextPaymentAmount>0</a:NextPaymentAmount>
            <a:NextPaymentDate>2022-01-08T00:00:00</a:NextPaymentDate>
            <a:PaymentType>One-Off</a:PaymentType>
            <a:RegularPaymentAmount>0</a:RegularPaymentAmount>
            <a:RegularPaymentDate i:nil="true"/>
            <a:RegularPaymentDay>0</a:RegularPaymentDay>
            <a:RegularPaymentFrequency>0</a:RegularPaymentFrequency>
            <a:RegularPaymentsExpectedCount>-1</a:RegularPaymentsExpectedCount>
            <a:RegularPaymentsReceivedCount>1</a:RegularPaymentsReceivedCount>
            <a:StatusId>2</a:StatusId>
            <a:StatusText>Completed</a:StatusText>
          </a:CollectionSchedule>
          <a:CollectionSchedule>
            <a:AdvanceNoticeSentDate i:nil="true"/>
            <a:ClientId>4000</a:ClientId>
            <a:CollectionScheduleId>987987</a:CollectionScheduleId>
            <a:CreatedDatetime>2020-12-27T15:57:35</a:CreatedDatetime>
            <a:CustomerEndDate i:nil="true"/>
            <a:CustomerId>4000:ABC123</a:CustomerId>
```

```
            <a:CustomerName>Mrs Mary Madre</a:CustomerName>
            <a:FirstPaymentAmount>0</a:FirstPaymentAmount>
            <a:FirstPaymentDate i:nil="true"/>
            <a:LastPaymentAmount>49.992</a:LastPaymentAmount>
            <a:LastPaymentDate>2023-04-10T00:00:00</a:LastPaymentDate>
            <a:NextPaymentAmount>49.992</a:NextPaymentAmount>
            <a:NextPaymentDate>2023-05-11T00:00:00</a:NextPaymentDate>
            <a:PaymentType>Continuous</a:PaymentType>
            <a:RegularPaymentAmount>49.992</a:RegularPaymentAmount>
            <a:RegularPaymentDate>2021-02-08T00:00:00</a:RegularPaymentDate>
            <a:RegularPaymentDay>15</a:RegularPaymentDay>
            <a:RegularPaymentFrequency>1</a:RegularPaymentFrequency>
            <a:RegularPaymentsExpectedCount>0</a:RegularPaymentsExpectedCount>
            <a:RegularPaymentsReceivedCount>27</a:RegularPaymentsReceivedCount>
            <a:StatusId>1</a:StatusId>
            <a:StatusText>Active</a:StatusText>
          </a:CollectionSchedule>
        </a:Schedules>
      </GetSchedulesByCustomerResult>
    </GetSchedulesByCustomerResponse>
  </s:Body>
</s:Envelope>
```

GetScheduleByScheduleID

The GetScheduleByScheduleID method allows the caller to obtain details about a Collection Schedule by supplying its Schedule id.

Request

Property	Required	Remarks
ScheduleID	Yes	The identifier of a schedule.

Response

The response, over and above those in the base Response class, returns a Schedule object.

Property	Type	Remark
Schedule	Data Structure	See the CollectionSchedule object described under GetSchedulesByCustomer

Additional Errors

Value	Meaning
164	Invalid Schedule ID

Examples

An example of a call to the GetScheduleByScheduleID web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetScheduleByScheduleId>
      <web:request>
        <web1:Credentials>
          <web:Group>4000</web:Group>
          <web:SecurityCode>xxxxxxxxxxxxxxx</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:ScheduleID>45646464</web1:ScheduleID>
      </web:request>
    </web:GetScheduleByScheduleId>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetScheduleByScheduleID web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetScheduleByScheduleIdResponse xmlns="https://webservices.landz.co.uk">
      <GetScheduleByScheduleIdResult xmlns:a=http://schemas.datacontract.org/2004/07/WebApi
                                     xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Schedule>
          <a:AdvanceNoticeSentDate i:nil="true"/>
          <a:ClientId>4000</a:ClientId>
          <a:CollectionScheduleId>45646464</a:CollectionScheduleId>
          <a:CreatedDatetime>2024-02-23T15:59:58.33</a:CreatedDatetime>
          <a:CustomerEndDate i:nil="true"/>
          <a:CustomerId>4000:FGH123</a:CustomerId>
          <a:CustomerName>Ms Jo Bloggs</a:CustomerName>
          <a:FirstPaymentAmount>0</a:FirstPaymentAmount>
          <a:FirstPaymentDate i:nil="true"/>
          <a:LastPaymentAmount>0</a:LastPaymentAmount>
          <a:LastPaymentDate i:nil="true"/>
          <a:NextPaymentAmount>134</a:NextPaymentAmount>
          <a:NextPaymentDate>2024-03-05T00:00:00</a:NextPaymentDate>
          <a:PaymentType>Continuous</a:PaymentType>
          <a:RegularPaymentAmount>134</a:RegularPaymentAmount>
          <a:RegularPaymentDate>2024-03-05T00:00:00</a:RegularPaymentDate>
          <a:RegularPaymentDay>15</a:RegularPaymentDay>
          <a:RegularPaymentFrequency>1</a:RegularPaymentFrequency>
          <a:RegularPaymentsExpectedCount>0</a:RegularPaymentsExpectedCount>
          <a:RegularPaymentsReceivedCount>0</a:RegularPaymentsReceivedCount>
          <a:StatusId>1</a:StatusId>
          <a:StatusText>Active</a:StatusText>
        </a:Schedule>
      </GetScheduleByScheduleIdResult>
    </GetScheduleByScheduleIdResponse>
  </s:Body>
</s:Envelope>
```

GetSchedulesByExternalReference

This method returns a list of collection schedules which have an ExternalReference value which includes the caller's supplied parameter value. (ExternalReference values may be supplied, at the caller's discretion, when creating collection schedules via calls to the AddCollectionSchedule method. See above.)

The CustomerReference field is an optional field which can be used to restrict the result set to a specific customer. If omitted, the results will include schedules for all customers.

Note that if ExternalReference is not supplied then the results will not be limited by a schedule's external reference value. Similarly, if CustomerReference is not supplied then the results will not be limited by a schedule's external reference value.

Although both CustomerReference and ExternalReference are optional fields both may not be null at the same time – at least one of the fields must have a non-null value.

Request

Property	Type	Required	Remark
ExternalReference	String	No	If supplied then the results are limited to collection schedules with an ExternalReference which includes this string. NB: The external reference "123abc xyz" includes • "123" • "xyz" • "3a" • "bc xy" • "123abc xyz" • "23abc xy" but does not include • "1 2 3" • "xyz!" • " abc" • "bcxy" • "bc xy" • "x&y" • "123 abc"
CustomerReference	String	No	If supplied then the results are limited to the specified customer, otherwise the results include schedules from all customers.

Example

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                  xmlns:web=https://webservices.landz.co.uk
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:GetSchedulesByExternalReference>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:ExternalReference>friday</web1:ExternalReference>
   </web:request>
  </web: GetSchedulesByExternalReference>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

Property	Type
Schedules	Array of CollectionScheduleExt objects

CollectionScheduleExt

Property	Type	Remark
CollectionScheduleID	Integer	The identifier of the schedule
CreatedDatetime	Date	The timestamp of the moment that the schedule was created.
StatusId	Integer	As described in GetScheduleStatuses
StatusText	String	As described in GetScheduleStatuses
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
CustomerId	String	The unique identifier of this customer, formed by Group and DDIReference, separated by a colon
CustomerName	String	The name of the customer
CustomerEndDate	Date (nullable)	If CustomerEndDate has a value, and that value is in the past, then the account is suspended. No collections will be attempted whilst the account is suspended.
PaymentType	String	Possible values are “One-Off”, “Continuous” or a string representing the number of regular payments comprising this schedule
FirstPaymentAmount	Decimal	As described in AddCollectionSchedule
FirstPaymentDate	Date (nullable)	As described in AddCollectionSchedule

NextPaymentAmount	Decimal	As described in AddCollectionSchedule
NextPaymentDate	Date (nullable)	As described in AddCollectionSchedule
LastPaymentAmount	Decimal	The amount that was collected on the most LastPaymentDate
LastPaymentDate	Date (nullable)	The date that the most recent payment was processed by London & Zurich
RegularPaymentAmount	Decimal	As described in AddCollectionSchedule
RegularPaymentDate	Date (nullable)	RegularPaymentStartDate as described in AddCollectionSchedule ???
RegularPaymentDay	Integer	The day of the month on which the collectins are scheduled, or 0 if the schedule does not describe regular payments
RegularPaymentFrequency	Integer	As described in AddCollectionSchedule, or 0 if the schedule does not describe regular payments
RegularPaymentsExpectedCount	Integer	A value of -1 implies that the schedule does not describe regular payments
RegularPaymentsReceivedCount	Integer	How many payments have been processed so far on this schedule
AdvanceNoticeSentDate	Date (nullable)	
ExternalReference	String (nullable)	The external reference that is associated with the schedule, as described in AddCollectionSchedule

Example


```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/" >
  <s:Body>
    <GetSchedulesByExternalReferenceResponse xmlns="https://webservices.landz.co.uk">
      <GetSchedulesByExternalReferenceResult xmlns:a=http://schemas.datacontract.org/2004/07/WebApi
                                             xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Schedules>
          <a:CollectionScheduleExt>
            <a:AdvanceNoticeSentDate i:nil="true"/>
            <a:ClientId>4000</a:ClientId>
            <a:CollectionScheduleId>7774648</a:CollectionScheduleId>
            <a:CreatedDatetime>2024-02-26T09:41:01.693</a:CreatedDatetime>
            <a:CustomerEndDate i:nil="true"/>
            <a:CustomerId>4000:00004771</a:CustomerId>
            <a:CustomerName>Mrs Sue Dandoo</a:CustomerName>
            <a:ExternalReference>extref 1</a:ExternalReference>
            <a:FirstPaymentAmount>5</a:FirstPaymentAmount>
            <a:FirstPaymentDate>2024-04-01T00:00:00</a:FirstPaymentDate>
            <a:LastPaymentAmount>0</a:LastPaymentAmount>
```

```
        <a:LastPaymentDate i:nil="true"/>
        <a:NextPaymentAmount>5</a:NextPaymentAmount>
        <a:NextPaymentDate>2024-04-01T00:00:00</a:NextPaymentDate>
        <a:PaymentType>One-Off</a:PaymentType>
        <a:RegularPaymentAmount>0</a:RegularPaymentAmount>
        <a:RegularPaymentDate i:nil="true"/>
        <a:RegularPaymentDay>0</a:RegularPaymentDay>
        <a:RegularPaymentFrequency>0</a:RegularPaymentFrequency>
        <a:RegularPaymentsExpectedCount>-1</a:RegularPaymentsExpectedCount>
        <a:RegularPaymentsReceivedCount>0</a:RegularPaymentsReceivedCount>
        <a:StatusId>1</a:StatusId>
        <a:StatusText>Active</a:StatusText>
      </a:CollectionScheduleExt>
      <a:CollectionScheduleExt>
        <a:AdvanceNoticeSentDate i:nil="true"/>
        <a:ClientId>4000</a:ClientId>
        <a:CollectionScheduleId>7774649</a:CollectionScheduleId>
        <a:CreatedDatetime>2024-02-26T09:41:38.79</a:CreatedDatetime>
        <a:CustomerEndDate i:nil="true"/>
        <a:CustomerId>4000:00004771</a:CustomerId>
        <a:CustomerName>Mrs Sue Dandoo</a:CustomerName>
        <a:ExternalReference>extref 2</a:ExternalReference>
        <a:FirstPaymentAmount>5</a:FirstPaymentAmount>
        <a:FirstPaymentDate>2024-05-01T00:00:00</a:FirstPaymentDate>
        <a:LastPaymentAmount>0</a:LastPaymentAmount>
        <a:LastPaymentDate i:nil="true"/>
        <a:NextPaymentAmount>5</a:NextPaymentAmount>
        <a:NextPaymentDate>2024-05-01T00:00:00</a:NextPaymentDate>
        <a:PaymentType>One-Off</a:PaymentType>
        <a:RegularPaymentAmount>0</a:RegularPaymentAmount>
        <a:RegularPaymentDate i:nil="true"/>
        <a:RegularPaymentDay>0</a:RegularPaymentDay>
        <a:RegularPaymentFrequency>0</a:RegularPaymentFrequency>
        <a:RegularPaymentsExpectedCount>-1</a:RegularPaymentsExpectedCount>
        <a:RegularPaymentsReceivedCount>0</a:RegularPaymentsReceivedCount>
        <a:StatusId>1</a:StatusId>
        <a:StatusText>Active</a:StatusText>
      </a:CollectionScheduleExt>
    </a:Schedules>
  </GetSchedulesByExternalReferenceResult>
 </GetSchedulesByExternalReferenceResponse>
</s:Body>
</s:Envelope>
```


Additional Errors

Examples of GetSchedulesByExternalReference-specific error codes, and their accompanying error messages, include the following:

Value	Meaning
176	Either ExternalReference or CustomerReference may be null, or contain nothing but whitespace, but not both

GetNonProcessingDates

The GetNonProcessingDates method retrieves a list of future dates where collections cannot be processed or where it is recommended that a collection is not requested to occur for a given customer. Although a payment will be scheduled to be collected at the next earliest working day following the collection date, this method allows you to be clear when the money is actually to be collected.

The reasons codes for this are:

Code	Remark
WE	This date falls on a weekend
PH	This date fails on a public holiday
CA	The customer has a payment schedule that will generate a collection request that falls on this date. It is possible to set up another collection for the same date and value, but we advise against it – banks sometimes reject collections when the date and value are the same.
AC	The date does not fall within the possible time period for a Direct Debit. (See accompanying documentation.)

Request

GetNonProcessingDatesRequest

Property	Type	Required	Remark
CustomerRef	String	Yes	The Customer Reference
StartDate	Date	Yes	The start of the range of dates that you'd like to be returned – ISO 8601 format (YYYY-MM-DD). This value should not represent a date earlier than today.
EndDate	Date	Yes	The end of the range of dates that you'd like to be returned – ISO 8601 format (YYYY-MM-DD). This value should not represent a date that is more than 365 days after StartDate.

Response

GetNonProcessingDatesResponse

Property	Type
NonProcessingDates	Array of NonProcessingDate

NonProcessingDate

Property	Type	Remark
NonProcDate	Date	The date on which a Direct Debit collection cannot be made.
Description	String	The reason why a collection cannot be requested for the given date.

Additional Errors

Value	Meaning
130	Date range must span one year or less
131	You can only select dates on or after today

CancelScheduledPayment

Cancels a collection schedule, using a payment schedule (continuous authority) ID.

Note that this method attempts to cancel a collection schedule, not the collecitons themselves. Cancelling a schedule will stop any Collections being created from the schedule, but will not affect a collection that has already processed. In order to stop a Collection from occurring, you would need to send the cancellation request 3 working days before the due date of the Collection.

Request

Property	Type	Max length	Required	Remark
CustomerRef	String	15	No	The Customer Reference for which the scheduled payment or payments should be cancelled. If not supplied then no schedules will be cancelled.
ScheduledPayment ID	Integer	N/A	No	If not supplied then all schedules associated with the Customer Reference will be cancelled. If supplied, any schedule that has the given ID that is associated with the Customer Reference will be cancelled.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:CancelScheduledPayment>
   <web:request>
    <web1:Credentials>
     <web:Group>4000</web:Group>
     <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
    </web1:Credentials>
    <web1:CustomerRef>4000:140378ST2</web1:CustomerRef>
    <web1:ScheduledPaymentID>4733395</web1:ScheduledPaymentID>
    </web:request>
  </web:CancelScheduledPayment>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

The CancelScheduledPaymentResponse contains a Message and ResponseCode, along with the ScheduledPaymentID of the cancelled schedule.

Example

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
   <CancelScheduledPaymentResponse xmlns="https://webservices.landz.co.uk">
     <CancelScheduledPaymentResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
       <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
       <a:Message>Success</a:Message>
       <a:ResponseCode>0</a:ResponseCode>
       <a:Payments xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
         <b:int>4733395</b:int>
       </a:Payments>
     </CancelScheduledPaymentResult>
   </CancelScheduledPaymentResponse>
 </s:Body>
</s:Envelope>
```

Additional Errors

Value	Meaning
130	The PaymentID must be a positive integer value Or No scheduled payments have been cancelled

In the case where a non-postive (zero or lower) number has been supplied, the error message will be “130: The PaymentID must be a positive integer value”.

In the case where an attempt to cancel a collection schedule, where the collection request has already been submitted, the error message will be “130: No scheduled payments have been cancelled”.

GetCustomers

Call this method to obtain a list of some or all customers which are active and/or suspended, depending on call parameters.

Request

Property	Type	Remark
GetSuspendedOnly	Boolean (Optional)	If false, bring back all customers that are not suspended. If true, bring back only those customers that are suspended. If omitted, all customers will be returned whether or not they are suspended.
GetPendingOnly	Boolean (Optional)	If true, return all customers with pending detail changes. If false, return all customers with no pending detail changes. If omitted, all customers will be returned whether or not there are pending changes.

Response

Property	Type
Summary	Array of CustomerDetail

CustomerDetail

Property	Type	Remark
CustomerRef	String	The unique identifier of this customer, formed by Group and DDIReference, separated by a colon
DDIReference	String	As described in AddCustomer
CustomerName	String	As described in AddCustomer
DebitName	String	As described in AddCustomer
Address1	String	As described in AddCustomer
Address2	String	As described in AddCustomer
Address3	String	As described in AddCustomer
Address4	String	As described in AddCustomer
Postcode	String	As described in AddCustomer
ContactName	String	As described in AddCustomer
Email	String	As described in AddCustomer
Telephone	String	As described in AddCustomer
Mobile	String	As described in AddCustomer
CompanyNumber	String	As described in AddCustomer
Sortcode	String	As described in AddCustomer
AccountNo	String	As described in AddCustomer

Status	String	Possible values: A = Accepted: The customer's DDI has either not been submitted to BACS or has been submitted to BACS and accepted D = Declined: The customer's DDI has been submitted to and rejected by BACS
SuspendedDate	Date (Nullable)	If SuspendedDate has a value and that value is in the past then the account is suspended. No collections will be attempted whilst the account is suspended.
LastUpdated	Date	The most recent date on which changes to this customer's record occurred.
IsPending	Boolean	This signifies whether there are changes pending on a customer's account that have not yet been approved. If there are pending changes then no further updates can take place until the customer has moved from pending

SuspendCustomer

This is used to place a customer's account on hold from further processing of any collections.

Request

Property	Type	Required	Remark
CustomerRef	String	Yes	The reference of the customer that is to be suspended
CancelAllScheduled Payments	Boolean	Yes	If true, all scheduled payments set up against the supplied customer reference will be cancelled. If false then schedules will be unaltered, although no collections will be processed while the customer is suspended.
Reason	String	Yes	A comment stating the reason for suspending the customer. May not consist solely of whitespace characters. Characters after the 300th will be ignored.

Response

No extra data beyond the ResponseBase data structure.

Additional Errors

Value	Meaning
130	No reason given
131	System Error in suspending customer
132	Customer was not found
133	Customer was already suspended

ReinstateCustomer

The ReinstateCustomer method offers a way to undo the suspension of a customer's account. Processing of collections from the customer will restart.

Request

Property	Type	Required	Remark
CustomerRef	String	Yes	The reference of the customer to be reinstated
Reason	String	Yes	A comment stating the reason for reinstating the customer. May not consist solely of whitespace characters. Characters after the 300th will be ignored.

Response

No extra data beyond the ResponseBase data structure.

Additional Errors

Value	Meaning
131	System Error in reinstating customer
132	Customer was not found
133	Customer was not suspended
201	Customer already in pending process

GetCustomerDetails

Use this method to list the customer details, comments, successful and failed collections and future payments. The details that are required to be present in the response can be flagged on and off by the use of the four Boolean parameters.

Request

Property	Required	Remark
CustomerRef	Yes	The Customer Reference
GetComments	Yes	If true, this will retrieve all comments made on a customer's account, otherwise no such comments will be returned
GetSuccessfulCollections	Yes	If true, this will retrieve all successful collections made on a customer's account that have been paid on to you
GetFailedCollections	Yes	If true, this will retrieve all Collections that have not been paid onto you – this will include unprocessed, failed collections, collections that have been made but not yet paid on, and collections that have had an indemnity claim raised against them.
GetScheduledPayments	Yes	If true, this will return predicted collections based off schedules that have been set up against a customer's account, otherwise no such predicted collections will be returned

Response

CustomerDetail

Property	Type	Remark
CustomerRef	String	The unique identifier of this customer, formed by Group and DDIReference, separated by a colon
DDIReference	String	As described in AddCustomer
CustomerName	String	As described in AddCustomer
DebitName	String	As described in AddCustomer
Address1	String	As described in AddCustomer
Address2	String	As described in AddCustomer
Address3	String	As described in AddCustomer
Address4	String	As described in AddCustomer
Postcode	String	As described in AddCustomer
ContactName	String	As described in AddCustomer
Email	String	As described in AddCustomer
Telephone	String	As described in AddCustomer

Mobile	String	As described in AddCustomer
Sortcode	String	As described in AddCustomer
AccountNo	String	As described in AddCustomer
CompanyNumber	String	As described in AddCustomer
Status	String	Possible values: A = Accepted: The customer's DDI has either not been submitted to BACS or has been submitted to BACS and accepted D = Declined: The customer's DDI has been submitted to and rejected by BACS
SuspendedDate	Date	If SuspendedDate has a value, and that value is in the past, then the account is suspended. No collections will be attempted whilst the account is suspended.
LastUpdated	Date	The most recent date on which changes to this customer's record occurred.
IsPending	Boolean	This value indicates whether there are pending changes to the details of a customer's account. If the customer has detail changes pending then no further updates can take place until the customer has moved from pending.

Comment

Property	Remark
Comment	0 or more instances of the Comment data structure

Comments

Property	Type	Remark
CreatedDate	Date	When the comment was added
Comment	String	The comment

Collections

Property	Remark
Collections	0 or more instances of the CollectionEntry data structure

Failures

Property	Remark
Failures	0 or more instances of the CollectionEntry data structure

CollectionEntry

Property	Type	Remark
CollectionID	Integer	A unique identifier for the collection
CustomerRef	String	The Customer Reference
CustomerName	String	The name given to the customer associated with the Customer Reference
CollectionDate	Date	The due date of the collection

Amount	Decimal	The value of the collection
FailureCode	String	A code describing failure
FailureDescription	String	The reason for the failure
ProcessingStatus	String	Overall status
StatusDescription	String	A description of the ProcessingStatus
MinorResult	String	See below
MinorDescription	String	deprecated
MajorResult	String	deprecated
MajorDescription	String	deprecated
FailureReference	Integer (Nullable)	The key used to link back to the failure record if the collection failed
ReconcileReference	Integer (Nullable)	An integer value that links the collection to a client payment
ReconcileDate	Date (Nullable)	The date when London & Zurich reconciled the indemnity claim.
ScheduledPaymentID	Integer (Nullable)	The continuous authority ID that created this collection

ProcessingStatus values and descriptions

The ProcessingStatus of a CollectionEntry indicates whether a collection request has been sent to Bacs and, if so, whether the request resulted in a successful collection or a failed collection attempt.

ProcessingStatus	StatusDescription	Interpretation
n	Unprocessed	No collection request was/will be sent to Bacs.
j	Processing	A collection request has been sent to Bacs.
H	Successful Transaction	This collection has succeeded and will be settled.
F	Failed Transaction	This collection failed. See FailureCode for details of the reasons for failure.

MinorResult values and descriptions

The MinorResult of a CollectionEntry is only defined when the ProcessingStatus of the CollectionEntry is 'F'.

The MinorResult of CollectionEntry objects which have ProcessingStatus 'F' is an indicator of whether an indemnity claim was made in respect of the collection. If the MinorResult is an 'I' then an indemnity claim has been made in respect of the collection, otherwise no indemnity claim has been made.

Consult the FailureCode to examine the reason for the failed collection. The FailureCode will represent the reason that the collection failed, when MinorResult is not 'I', or the reason for the indemnity claim, when MinorResult is 'I'.

FailureCode values and descriptions

If the ProcessingStatus of a CollectionEntry is 'F' ("Failed Transaction") then FailureCode is an indication of the reason that the collection attempt failed. If the ProcessingStatus of a CollectionEntry is not 'F' then FailureCode is not defined.

The range of possible values that may be returned in the FailureCode and FailureDescription fields depends on the value returned in the MinorResult column. When a CollectionEntry has the ProcessingStatus 'F' and the value "I" as its MinorResult then that CollectionEntry represents a collection that resulted in an indemnity claim. A CollectionEntry that has the ProcessingStatus 'F' and a value other than "I" as its MinorResult represents a collection that failed for other reasons.

FailureCode values and descriptions when ProcessingStatus is 'F' and MinorResult is NOT "I"

FailureCode when MinorResult is NOT "I"	FailureDescription
0	Refer To Payer
1	Instruction Cancelled
2	Payer Deceased
3	A/C Transferred to another Bank
4	Advance Notice Disputed
5	No Account
6	No Instruction
7	Amount Differs
8	Amount Not Yet Paid
9	Presentation Overdue
A	Originator Differs
B	Account Closed
G	Group Request
X	CMS Request

FailureCode values and descriptions when ProcessingStatus is 'F' and MinorResult is "I"

FailureCode when MinorResult is "I"	FailureDescription
1	Amount and/or date of Direct Debit differs from advance notice
2	No advance notice was received by the payer
3	DDI cancelled by paying bank
4	Payer has cancelled DDI direct with service user
5	No instruction held. Payer disputes having given authority
6	Signature on DDI is fraudulent or not in accordance with account authorised signature(s).
7	Claim raised at service user's request after Direct Debit applied to payer's account.

8	Service user name disputed. Payer does not recognise service user collecting Direct Debit.

MajorResult is a legacy indicator of whether or not London & Zurich sent a request to Bacs to perform this collection.

GetClientPayments

The GetClientPayments method is used to obtain what payments were, or are due to be, made by London & Zurich to the Client within a supplied date range. Updated payment information is available after 12 pm every working day.

Request

Property	Required	Remark
StartDate	No	The earliest payment date you want the query to report on – ISO 8601 format (YYYY-MM-DD) If this value is not supplied then today's date is used. The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.
EndDate	No	The latest payment date you want the query to report on – ISO 8601 format (YYYY-MM-DD). If this value is not supplied then today's date is used. The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
   <web:GetClientPayments>
     <web:request>
       <web1:Credentials>
         <web:Group>4000</web:Group>
         <web:SecurityCode>XXXXXXXXXXXXXXXXXXXXXXXX</web:SecurityCode>
       </web1:Credentials>
       <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
       <web1:EndDate>2016-08-17</web1:EndDate>
       <web1:StartDate>2010-08-19</web1:StartDate>
     </web:request>
   </web:GetClientPayments>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

Property	Type	Remark
Payments	Data Structure	Contains 0 or more ClientPayment data structures

ClientPayment

Property	Remark

ReconcileReference	An integer reference relating to the payment
CollectedDate	The date on which the collections relating to this payment were made
ScheduledCollectionTotal	A decimal value representing the expected total prior to any deductions being made
FailedCollectionTotal	A decimal value representing the total deductions from the payment
PaymentTotal	A decimal value representing the actual payment amount
ReconcileDate	The date when the payment was made.

Example


```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetClientPaymentsResponse xmlns="https://webservices.landz.co.uk">
   <GetClientPaymentsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <a:Message>Success</a:Message>
    <a:ResponseCode>0</a:ResponseCode>
    <a:Payments>
     <a:ClientPayment>
      <a:CollectedDate>2011-11-11T00:00:00</a:CollectedDate>
      <a:FailedCollectionTotal>48.0000</a:FailedCollectionTotal>
      <a:PaymentTotal>27344.7200</a:PaymentTotal>
      <a:ReconcileDate>2011-11-12T00:00:00</a:ReconcileDate>
      <a:ReconcileReference>391</a:ReconcileReference>
      <a:ScheduledCollectionTotal>27392.7200</a:ScheduledCollectionTotal>
     </a:ClientPayment>
     <a:ClientPayment>
      <a:CollectedDate>2011-11-19T00:00:00</a:CollectedDate>
      <a:FailedCollectionTotal>362.4000</a:FailedCollectionTotal>
      <a:PaymentTotal>2672.5600</a:PaymentTotal>
      <a:ReconcileDate>2011-11-20T00:00:00</a:ReconcileDate>
      <a:ReconcileReference>396</a:ReconcileReference>
      <a:ScheduledCollectionTotal>3034.9600</a:ScheduledCollectionTotal>
     </a:ClientPayment>
     <a:ClientPayment>
      <a:CollectedDate>2011-12-10T00:00:00</a:CollectedDate>
      <a:FailedCollectionTotal>959.6000</a:FailedCollectionTotal>
      <a:PaymentTotal>27843.9100</a:PaymentTotal>
      <a:ReconcileDate>2011-12-11T00:00:00</a:ReconcileDate>
      <a:ReconcileReference>397</a:ReconcileReference>
      <a:ScheduledCollectionTotal>28803.5100</a:ScheduledCollectionTotal>
     </a:ClientPayment>
     <a:ClientPayment>
      <a:CollectedDate>2011-12-22T00:00:00</a:CollectedDate>
      <a:FailedCollectionTotal>0.0000</a:FailedCollectionTotal>
      <a:PaymentTotal>1998.7600</a:PaymentTotal>
      <a:ReconcileDate>2011-12-23T00:00:00</a:ReconcileDate>
      <a:ReconcileReference>398</a:ReconcileReference>
      <a:ScheduledCollectionTotal>1998.7600</a:ScheduledCollectionTotal>
     </a:ClientPayment>
    </a:Payments>
   </GetClientPaymentsResult>
```

```
    </GetClientPaymentsResponse>
  </s:Body>
</s:Envelope>
```

GetCustomersLastUpdated

This method will return a list of identifiers of customer records the most recent update to which took place within the specified date range. These updates will usually be prompted by instructions from BACS.

The GetCustomerDetails method offers a way to examine the current state of a Customer via its Customer Reference.

Request

Property	Type	Required	Remark
StartDate	Date	No	If the StartDate field is supplied then the returned data will contain only those identifiers of customer records that have been updated on or after the given date. If the StartDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.
EndDate	Date	No	If the EndDate field is supplied then the returned data will contain only those identifiers of customer records that have been updated on or before the given date. If the EndDate field is not supplied then it defaults to today's date. The date should be supplied in ISO 8601 format (YYYY-MM-DD). The date must be a date that falls within a time period beginning six years prior to today and ending one month from now.

Response

Property	Type	Remark
CustomerRef	String	The identifier of the customer record that has been updated.
LastUpdatedDate	Date	The date that the customer record was most recently updated.
LastCollectionDate	Date (Nullable)	The date of the most recent collection made from the customer. If no collections have been made from the customer then this value will be null.
SuspendedDate	Date (Nullable)	If this value is null then the customer is not suspended. If this value is not null then the customer is suspended and the value is the date that the suspension came into force.

Status	String	Possible values: A = Accepted: The customer's DDI has either not been submitted to BACS or has been submitted to BACS and accepted D = Declined: The customer's DDI has been submitted to and rejected by BACS

GetBankAndBranchName

The GetBankAndBranchName method allows the caller to request the names of a bank and branch to which a sort code refers. The returned bank name and branch name come from the Extended Industry Sort Code Directory.

(Note: Bank sub-branches exist, therefore a sub-branch cannot be fully individuated with just a bank sort code.)

The GetBankAndBranchName web service method takes a GetBankAndBranchNameRequest and returns a BankAndBranchNameResponse.

Request

GetBankAndBranchNameRequest extends RequestBase and includes a further string, SortCode, which it expects to be a representation of a sort code of a UK bank branch. A string which is "a representation of a sort code of a UK bank branch" is a string that is six characters long, with each of those six characters being a digit.

Property	Type	Required	Remark
SortCode	String	Yes	The sort code of a UK bank branch. Must be six characters long, each character being a digit.

If the SortCode that is given in the GetBankAndBranchNameRequest argument is not a string of six digits then the GetBankAndBranchName method returns a ResponseCode of 3 ("InvalidParameter"), with the returned SortCode, BankName and BranchName being undefined. In this case the returned Errors field will contain either the string "1: The value of the SortCode parameter must be a string of six digits." (when the request parameter RemoveErrorDescription was set to 'false'), or the string "1" (when the request parameter RemoveErrorDescription was set to 'true').

If the SortCode that is given in the GetBankAndBranchNameRequest argument is a string of six digits that does not represent a sort code existing in our EISCD database, then this method returns a ResponseCode of 0 ("Success") and the returned BankName and BranchName will be null.

Examples

An example SOAP request to the GetBankAndBranchName web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk"
xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetBankAndBranchName>
      <web:request>
        <web1:Credentials>
```

```
        <web:Group>1234</web:Group>
        <web:SecurityCode>mysecuritycode</web:SecurityCode>
      </web1:Credentials>
      <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
      <web1:SortCode>010085</web1:SortCode>
    </web:request>
  </web:GetBankAndBranchName>
 </soapenv:Body>
</soapenv:Envelope>
```

Response

BankAndBranchNameResponse extends ResponseBase and provides four additional strings; SortCode - the sort code from the request, BankName - the name of the bank that owns the given sort code, BranchName - the name of the branch that relates to the given sort code and BranchPrintedDirectoryName – a more user-friendly, printable form of the BranchName.

Property	Type	Remark
SortCode	String	The sort code of a UK bank branch, represented as a string of 6 digits.
BankName	String	A string, of up to 70 characters in length, representing the name of the bank that owns the SortCode, according to EISCD data.
BranchName	String	A string, of up to 27 characters in length, representing the title of the branch that is referred to by the SortCode, according to EISCD data.
BranchPrintedDirectoryName	String	A string, of up to 105 characters in length, representing the branch's name in a more user-friendly, presentable format than the shorter BranchName value. This value is sourced from EISCD "printed directory" data.

Note that it is possible for the returned ResponseCode to be 0 ("Success") while the BankName and/or BranchName are null. This would happen, for instance, in cases where the supplied SortCode is valid (a string of 6 digits) but is not present in the EISCD data source.

Examples

An example SOAP request to the GetBankAndBranchName web service method;

An example of the SOAP response from the GetBankAndBranchName web service method when the sought sort code was found;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetBankAndBranchNameResponse xmlns="https://webservices.landz.co.uk">
      <GetBankAndBranchNameResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:BankName>NATIONAL WESTMINSTER BANK PLC</a:BankName>
        <a:BranchName>BLACKBURN,35 KING WM ST</a:BranchName>
```

```
        <a:BranchPrintedDirectoryName>Blackburn, 35 King William Street</a:BranchPrintedDirectoryName>
        <a:SortCode>010085</a:SortCode>
      </GetBankAndBranchNameResult>
    </GetBankAndBranchNameResponse>
  </s:Body>
</s:Envelope>
```

An example response from the GetBankAndBranchName web service method when the sought sort code was *not* found;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetBankAndBranchNameResponse xmlns="https://webservices.landz.co.uk">
      <GetBankAndBranchNameResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:BankName i:nil="true"/>
        <a:BranchName i:nil="true"/>
    <a:BranchPrintedDirectoryName i:nil="true"/>
        <a:SortCode>123456</a:SortCode>
      </GetBankAndBranchNameResult>
    </GetBankAndBranchNameResponse>
  </s:Body>
</s:Envelope>
```

An example response from the GetBankAndBranchName web service method when the sought sort code was not a string of six digits, and the RemoveErrorDescription parameter was given a value of 'false';

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetBankAndBranchNameResponse xmlns="https://webservices.landz.co.uk">
      <GetBankAndBranchNameResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
          <b:string>1:  The value of the SortCode parameter must be a string of six digits.</b:string>
        </a:Errors>
        <a:Message>Please check the errors to identify the cause</a:Message>
        <a:ResponseCode>3</a:ResponseCode>
        <a:BankName i:nil="true"/>
        <a:BranchName i:nil="true"/>
    <a:BranchPrintedDirectoryName i:nil="true"/>
        <a:SortCode i:nil="true"/>
      </GetBankAndBranchNameResult>
    </GetBankAndBranchNameResponse>
  </s:Body>
</s:Envelope>
```

An example response from the GetBankAndBranchName web service method when the sought sort code was not a string of six digits, and the RemoveErrorDescription parameter was given a value of 'true';

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
```

```
  <s:Body>
    <GetBankAndBranchNameResponse xmlns="https://webservices.landz.co.uk">
      <GetBankAndBranchNameResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
          <b:string>1</b:string>
        </a:Errors>
        <a:Message>Please check the errors to identify the cause</a:Message>
        <a:ResponseCode>3</a:ResponseCode>
        <a:BankName i:nil="true"/>
        <a:BranchName i:nil="true"/>
    <a:BranchPrintedDirectoryName i:nil="true"/>
        <a:SortCode i:nil="true"/>
      </GetBankAndBranchNameResult>
    </GetBankAndBranchNameResponse>
  </s:Body>
</s:Envelope>
```

IsDdiReferenceAvailable

The IsDdiReferenceAvailable web service method allows the caller to ask whether a given DDI reference is acceptable and would be currently available for use on the eBacs system as the value of the DDIReference parameter in an AddCustomerRequest, were the caller to use the AddCustomer web service method (described elsewhere in this document).

For a DDI reference to be acceptable, it must pass both the Bacs AUDDIS core reference rules and the eBacs DDI reference rules as described in the documentation of the AddCustomer web service method. For a DDI reference to be currently available it must not be currently in use as the DDIReference of an existing Customer on the eBacs system that uses the same Group and/or SUN as this method's caller. (See the description of the DDIReference parameter of the AddCustomerRequest object used in a call to the AddCustomer web service method described elsewhere in this document for details.)

Note that this method returns information about whether the given DDI reference is *currently* available. It is possible for a given DDI reference, that this method reported as available, to subsequently be used and thus no longer be available during a later call to AddCustomer.

The IsDdiReferenceAvailable web service method takes an IsDdiReferenceAvailableRequest and returns an IsDdiReferenceAvailableResponse.

Request

IsDdiReferenceAvailableRequest extends RequestBase and includes a further string, DDIReference, which is the string that the caller wishes to be checked for availability as a DDI reference on the eBacs system.

Property	Type	Required	Remark
DDIReference	String	Yes	The string that the caller wishes to check for availability as a DDI reference.

Examples

An example of a call to the IsDdiReferenceAvailable web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:IsDdiReferenceAvailable>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
          <web:SecurityCode>mysecuritycode</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
```

```
        <web1:DDIReference>ABCDEFG</web1:DDIReference>
      </web:request>
    </web:IsDdiReferenceAvailable>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

IsDdiReferenceAvailableResponse extends ResponseBase and provides one additional Boolean value - DdiReferenceIsCurrentlyAvailable – which will contain the value 'true' if and only if the DDIReference that was supplied in the IsDdiReferenceAvailableRequest is currently available and acceptable to be used as the value of the DDIReference parameter in an AddCustomerRequest, were the caller to use it as a parameter sent to the eBacs system's AddCustomer() web service method (described elsewhere in this document).

Property	Type	Remark
DdiReferenceIsCurrentlyAvailable	Boolean	Contains the value ‘true’ if the supplied DDIReference is currently available for use as the value of the DDIReference parameter in an AddCustomerRequest by the caller, otherwise contains the value ‘false’.

Note that the response’s "Message" value, "Success", indicates that the web service method call completed successfully and that the given DDIReference was successfully checked for availability - it does *NOT* indicate that the DDI reference is currently available. Examine the DdiReferenceIsCurrentlyAvailable field of the response to see whether the DDI reference is currently available.

Examples

An example response indicating that the DDI reference that was supplied is currently available for use with the AddCustomer web service method;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <IsDdiReferenceAvailableResponse xmlns="https://webservices.landz.co.uk">
      <IsDdiReferenceAvailableResult
xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:DdiReferenceIsCurrentlyAvailable>true</a:DdiReferenceIsCurrentlyAvailable>
      </IsDdiReferenceAvailableResult>
    </IsDdiReferenceAvailableResponse>
  </s:Body>
</s:Envelope>
```

An example response indicating that the DDI reference that was supplied is *not* available for use with the AddCustomer web service method;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <IsDdiReferenceAvailableResponse xmlns="https://webservices.landz.co.uk">
      <IsDdiReferenceAvailableResult
xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:DdiReferenceIsCurrentlyAvailable>false</a:DdiReferenceIsCurrentlyAvailable>
      </IsDdiReferenceAvailableResult>
    </IsDdiReferenceAvailableResponse>
  </s:Body>
</s:Envelope>
```

GetRefunds

This method is not available for use.

SendCreateRefund

This method is not available for use.

SendUpdateRefund

This method is not available for use.

SendCancelRefund

This method is not available for use.

GetClientSettlements

Call this method to get list of all client settlements, when being paid via Net Settlements.

Request

GetClientSettlements data structure contains all the fields of the RequestBase data structure

Examples

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetClientSettlements>
        <web1:Credentials>
          <web:Group>4000</web:Group>
          <web:SecurityCode>12345gnuURHD6UN7Murz78V9IXXXXX</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
       <web1:ClientId>4000</web1:ClientId>
      </web:request>
    </web:GetClientSettlements>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

A GetClientSettlements data structure contains the fields of the ResponseBase data structure in addition to the ClientSettlements containing list of ClientSettlement entities.

ClientSettlement Entity

Property	Type	Remark
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
ClientSettlementId	Integer	System Generated Identifier of ClientSettlement.
CollectionAmount	Double	The Sum value of Collections included in this settlement.
ConnectedMerchantChargeAmount	Double	The Sum value of the Connected Merchant charges to Client.
ConnectedMerchantId	Integer	The Unique Identifier for the Connected Merchant.
ConnectedMerchantVatAmount	Double	Vat on ConnectedMerchantChargeAmount
CreatedDateTime	DateTime	DateTime on which this entry is created.

FailedCollectionAmount	Double	The sum value of Failed Collections included in this settlement.
IndemnityClaimAmount	Double	The sum value of Indemnity Claims included in this settlement.
NetdownAmount	Double	The sum value of Netdowns included in this settlement.
PaidDateTime	DateTime	The date when the Client Settlement payment is processed.
PaymentReferences	List	List of PaymentReference entities. Bank transfers/DirectCredits made to client.
RefundAmount	Double	The sum value of Refunds included in this settlement.
ServiceProviderChargeAmount	Double	The Sum value of Service Charges by London and Zurich to Client.
ServiceProviderVatAmount	Double	VAT on ServiceProviderChargeAmount.
SettlementAmount	Double	Final amount settled to Client after all deductions.
SettlementDate	DateTime	The date when the Client Payment is settled into clients account.
Status	String	New, Processed, Settled

PaymentReferences

Property	Type	Remark
Reference	String	This is unique reference generated by London and Zurich.
Amount	Double	Amount transferred in one bank transaction. Maximum amount is limited to 100000.

Examples

Following GetClientSettlementsResponse is return when GetClientSettlements request is sent:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetClientSettlementsResponse xmlns="https://webservices.landz.co.uk">
      <GetClientSettlementsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:ClientSettlements>
          <a:ClientSettlement>
            <a:ClientId>4000</a:ClientId>
            <a:ClientSettlementId>1</a:ClientSettlementId>
            <a:CollectionAmount>48.09</a:CollectionAmount>
            <a:ConnectedMerchantChargeAmount>1.41</a:ConnectedMerchantChargeAmount>
            <a:ConnectedMerchantId>1</a:ConnectedMerchantId>
            <a:ConnectedMerchantVatAmount>0</a:ConnectedMerchantVatAmount>
            <a:CreatedDateTime>2021-04-19T08:52:43.753</a:CreatedDateTime>
            <a:FailedCollectionAmount>0</a:FailedCollectionAmount>
            <a:IndemnityClaimAmount>0</a:IndemnityClaimAmount>
            <a:NetdownAmount>0</a:NetdownAmount>
```

```
        <a:PaidDateTime>2021-04-19T08:53:47.18</a:PaidDateTime>
        <a:PaymentReference>
          <a:PaymentReference>
            <a:Amount>46.68</a:Amount>
            <a:Reference>NSC0000000001</a:Reference>
          </a:PaymentReference>
        </a:PaymentReference>
        <a:RefundAmount>0</a:RefundAmount>
        <a:ServiceProviderChargeAmount>0</a:ServiceProviderChargeAmount>
        <a:ServiceProviderVatAmount>0</a:ServiceProviderVatAmount>
        <a:SettlementAmount>46.68</a:SettlementAmount>
        <a:SettlementDate>2021-04-22T00:00:00</a:SettlementDate>
        <a:Status>Processed</a:Status>
      </a:ClientSettlement>
      <a:ClientSettlement>
        <a:ClientId>4000</a:ClientId>
        <a:ClientSettlementId>11</a:ClientSettlementId>
        <a:CollectionAmount>252.5</a:CollectionAmount>
        <a:ConnectedMerchantChargeAmount>7.46</a:ConnectedMerchantChargeAmount>
        <a:ConnectedMerchantId>1</a:ConnectedMerchantId>
        <a:ConnectedMerchantVatAmount>0</a:ConnectedMerchantVatAmount>
        <a:CreatedDateTime>2021-04-20T09:16:51.567</a:CreatedDateTime>
        <a:FailedCollectionAmount>160</a:FailedCollectionAmount>
        <a:IndemnityClaimAmount>0</a:IndemnityClaimAmount>
        <a:NetdownAmount>0</a:NetdownAmount>
        <a:PaidDateTime>2021-04-20T09:17:31.353</a:PaidDateTime>
        <a:PaymentReference>
          <a:PaymentReference>
            <a:Amount>85.04</a:Amount>
            <a:Reference>NSC0000000011</a:Reference>
          </a:PaymentReference>
        </a:PaymentReference>
        <a:RefundAmount>0</a:RefundAmount>
        <a:ServiceProviderChargeAmount>0</a:ServiceProviderChargeAmount>
        <a:ServiceProviderVatAmount>0</a:ServiceProviderVatAmount>
        <a:SettlementAmount>85.04</a:SettlementAmount>
        <a:SettlementDate>2021-04-23T00:00:00</a:SettlementDate>
        <a:Status>Processed</a:Status>
      </a:ClientSettlement>
    </a:ClientSettlements>
  </GetClientSettlementsResult>
 </GetClientSettlementsResponse>
</s:Body>
</s:Envelope>
```

GetClientSettlementsByPaidDate

Call this method to get list of all client settlements which are paid to client between provided date range.

Request

GetClientSettlementsByPaidDate data structure contains all the fields of the RequestBase data structure in addition to following:

Property	Type	Required	Remark
StartDate	Date	Yes	The returned data will contain only those settlements with a Client Settlement paid on or after the given date. The date should be supplied in ISO 8601 format (YYYY-MM-DD).
EndDate	Date	Yes	The returned data will contain only those settlements with a Client Settlement paid on or before the given date. The date should be supplied in ISO 8601 format (YYYY-MM-DD).
ClientId	Integer	Yes	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.

Examples

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetClientSettlementsByPaidDate>
      <web:request>
        <web1:Credentials>
          <web:Group>4000</web:Group>
          <web:SecurityCode>1234PgnuURHD6UN7Murz78V9IFXXXX</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
       <web1:ClientId>4000</web1: ClientId >
        <web1:EndDate>2021-04-20</web1:EndDate>
        <web1:StartDate>2021-04-19</web1:StartDate>
      </web:request>
    </web:GetClientSettlementsByPaidDate>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

A GetClientSettlementsByPaidDateResponse data structure contains the fields of the ResponseBase data structure in addition to the ClientSettlements containing list of ClientSettlement entities. See ClientSettlement entity.

GetClientSettlementsBySettlementDate

Call this method to get list of all client settlements which are settled in client account between provided date range.

Request

GetClientSettlementsBySettlementDate data structure contains all the fields of the RequestBase data structure in addition to following:

Property	Type	Required	Remark
StartDate	Date	Yes	The returned data will contain only those settlements with a Client Settlement paid on or after the given date. The date should be supplied in ISO 8601 format (YYYY-MM-DD).
EndDate	Date	Yes	The returned data will contain only those settlements with a Client Settlement paid on or before the given date. The date should be supplied in ISO 8601 format (YYYY-MM-DD).
ClientId	Integer	Yes	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.

Examples

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web: GetClientSettlementsBySettlementDate>
      <web:request>
        <web1:Credentials>
          <web:Group>4000</web:Group>
          <web:SecurityCode>1234PgnuURHD6UN7Murz78V9IFXXXX</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
       <web1:ClientId>4000</web1: ClientId >
        <web1:EndDate>2021-04-25</web1:EndDate>
        <web1:StartDate>2021-04-19</web1:StartDate>
      </web:request>
    </web:GetClientSettlementsBySettlementDate>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

A GetClientSettlementsBySettlementDateResponse data structure contains the fields of the ResponseBase data structure in addition to the ClientSettlements containing list of ClientSettlement entities. See ClientSettlement entity.

GetClientSettlementDetails

Call this method to get details of a settlement. Returned result includes list all the Collections, Failed collections, Indemnity Claims, Netdowns and Refunds associated with this settlement and various charges applicable for each.

Request

GetClientSettlementDetails data structure contains all the fields of the RequestBase data structure in addition to following:

Property	Type	Required	Remark
SettlementID	Integer	Yes	This is system generated unique identifier of each client settlement.

Examples

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetClientSettlementDetails>
      <web:request>
        <web1:Credentials>
          <web:Group>4000</web:Group>
          <web:SecurityCode>1234PgnuURHD6UN7Murz78V9IFXXXX</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:SettlementID>1</web1:SettlementID>
      </web:request>
    </web:GetClientSettlementDetails>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

A GetClientSettlementDetailsResponse data structure contains the fields of the ResponseBase data structure in addition to the ClientSettlementDetails entity.

ClientSettlementDetails Entity

Property	Type	Remark
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
ClientSettlementId	Integer	System Generated Identifier of Client Settlement.
CollectionAmount	Double	The sum value of Collections included in this settlement.

ConnectedMerchantChargeAmount	Double	The sum value of the Connected Merchant charges to Client.
ConnectedMerchantId	Integer	The Unique Identifier for the Connected Merchant.
ConnectedMerchantVatAmount	Double	Vat on ConnectedMerchantChargeAmount
CreatedDateTime	DateTime	DateTime on which this entry is created.
FailedCollectionAmount	Double	The sum value of Failed Collections included in this settlement.
IndemnityClaimAmount	Double	The sum value of Indemnity Claims included in this settlement.
NetdownAmount	Double	The sum value of Netdowns included in this settlement.
PaidDateTime	DateTime	DateTime when Client Settlement payment is processed.
PaymentReferences	List	List of PaymentReference entities.
RefundAmount	Double	The sum value of Refunds included in this settlement.
ServiceProviderChargeAmount	Double	The Sum value of Service Charges by London and Zurich to Client.
ServiceProviderVatAmount	Double	VAT on ServiceProviderChargeAmount.
SettlementAmount	Double	Final amount settled to Client after all deductions.
SettlementDate	DateTime	DateTime when payment is settled into Client's account.
Status	String	New, Processed, Settled.
SettlementComponentAndCharges	List	This contains list of ClientSettlementComponentAndCharges entities

ClientSettlementComponentAndCharges Entity:

This entity mainly contains 5 main components used to generate settlement and aggregated charges. The components are collections, failed collection, indemnity claims, netdowns and refunds.

Property	Type	Remark
CollectionSettlements	SettlementByCollection	This contains List of SettlementByCollection entities.
Collections	Collections	Collections contains List of CollectionClientSettlementComponentAndCharges
ConnectedMerchantChargeAmount	Double	The sum value of the Connected Merchant charges to Client.
ConnectedMerchantVatAmount	Double	Vat on ConnectedMerchantChargeAmount
FailedCollections	FailedCollections	FailedCollections contains List of FailedCollectionClientSettlementComponentAndCharges
IndemnityClaims	IndemnityClaims	IndemnityClaims contains List of IndemnityClaimClientSettlementComponentAndCharges

Netdowns	Netdowns	Netdowns contains List of NetdownClientSettlementComponentAndCharges
Refunds	Refunds	Refunds contains List of RefundClientSettlementComponentAndCharges.
ServiceProviderChargeAmount	Double	The Sum value of Service Charges by London and Zurich to Client.
ServiceProviderVatAmount	Double	VAT on ServiceProviderChargeAmount.

XXXClientSettlementComponentAndCharges

The data structure of following entities is similar. Only Settlement Component varies depending upton type of entity.

- CollectionClientSettlementComponentAndCharges
  - SettlementComponent - Collection
- FailedCollectionClientSettlementComponentAndCharges
  - SettlementComponent - FailedCollection
- IndemnityClaimClientSettlementComponentAndCharges
  - SettlementComponent - IndemnityClaim
- NetdownClientSettlementComponentAndCharges
  - SettlementComponent - Netdown
- RefundClientSettlementComponentAndCharges
  - SettlementComponent - Refund

Property	Type	Remark
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
ConnectedMerchantChargeAmount	Double	The value of the Connected Merchant charge to Client calculated based on collection amount and collection status.
ConnectedMerchantId	Integer	This is the unique identifier assigned to a Connected Merchant.
ConnectedMerchantVatAmount	Double	Vat on ConnectedMerchantChargeAmount
ServiceProviderChargeAmount	Double	The value of Service Charge by London and Zurich to Client calculated based on amount and collection status.
ServiceProviderVatAmount	Double	VAT on ServiceProviderChargeAmount

SettlementComponent	XXXClientSettlementComponentAndCharges	For CollectionClientSettlementComponentAndCharges see Collection entity. For FailedCollectionClientSettlementComponentAndCharges see FailedCollection entity. For IndemnityClaimClientSettlementComponentAndCharges see IndemnityClaim entity. For NetdownClientSettlementComponentAndCharges see Netdown entity. For RefundClientSettlementComponentAndCharges see Refund entity.

Collection entity

Property	Type	Remark
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
CollectionId	Integer	The unique identifier for the collection
CollectionScheduleId	Integer	The unique identifier for the Collection Schedule
CustomerId	String	This is the unique identifier for a Customer. It is formed from a concatenation of the Client Id, a colon, and the customer's DDI Reference also referred as Account Reference or End Customer Reference
CollectionDates	List	collectionDate submissionDate settlementDate paymentDate
Amount	Double	The value of the Collection
StatusId	Integer	
Status	String	1. Successful 2. Unprocessed 3. Due 4. Indemnity 5. Failed 6. Collected 7. Unknown

FailedCollection entity

Property	Type	Remark
FailedCollectionId	Integer	The unique identifier for the Failed Collection.
CollectionId	Integer	The unique identifier for the related Collection.

ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number
CustomerId	String	This is the unique identifier for a Customer. It is formed from a concatenation of the Client Id, a colon, and the customer's DDI Reference also referred to as Account Reference or End Customer Reference
CollectionDate	DateTime	The Date of the Collection
DatePaid	DateTime	The date of the when
PaidBeforeFail	Boolean	A True or False Value showing whether the collection failed before it had paid.
FailedReason	String	The reason why the Collection failed
FailedReasonCode	String	The reason code for why the Collection Failed
Amount	Double	The value of the Failed Collection

IndemnityClaim entity

Property	Type	Remark
IndeminityClaimId	Integer	The unique identifier for the Indemnity Claim.
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
CollectionId	Integer	The unique identifier of the related Collection.
CustomerId	String	This is the unique identifier for a Customer. It is formed from a concatenation of the Client Id, a colon, and the customer's DDI Reference. also referred to as Account Reference or End Customer Reference.
CollectionDate	DateTime	The date of the original collection that relates to this indemnity claim.
DatePaid	DateTime	The Date when the Indemnity was paid
IndemnityClaimReason	String	The reason for the Indemnity
IndemnityClaimReasonCode	String	The reason code for the Indemnity.
Amount	Double	The value of the Indemnity Claim.
IsCancelled	Boolean	If Indemnity Claim is cancelled, then value is True otherwise false.
PayingBankReference	String	The value Bank reference, If Indemnity is paid.
CustomerName	String	Name of Customer who raised Indemnity Claim.

Netdowns entity

Property	Type	Remark
NetdownId	Integer	This unique identifier for the netdown.

ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number.
Amount	Double	Value of the netdown.
CollectionDate	String	The date of the original collection.
NetDownType	Integer	1. Unknown Deduction 2. Guarantee Payment 3. Indemnity Claim 4. Sales Ledger
Completed	Boolean	

Refund entity

Property	Type	Remark
CreatedDateTime	DateTime	The date the Refund was created.
CustomerPaidDate	DateTime	The date of when Refund payment is processed for Customer. Value is null If not processed.
CustomerSettlementDate	DateTime	The date of when the RefundAmount is settled to customer. Value is null If not processed.
CancelledDate	DateTime	The date of when the Refund was cancelled. If not cancelled value will be Null.
ClientPaidDate	DateTime	The date of when Refund payment is processed for Client. Value is null If not processed.
ClientSettlementDate	DateTime	The date of when the Refund is settled with client to recoup RefundAmount. Value is null If not processed.
RefundStatusId	Integer	
RefundStatus	String	1. New 2. Processed 3. Settled 4. Cancelled
CollectionDate	DateTime	The date of the original collection.
CollectionAmount	Double	The value of the original collection
RefundId	integer	This is the unique identifier for this Refund.
ClientId	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a group number
RefundAmount	Double	Value of the Refund.
CustomerId	String	This is the unique identifier for a Customer. It is formed from a concatenation of the Client Id, a colon, and the customer's DDI Reference, also referred to as Account Reference or End Customer Reference. This is the customer who has requested Refund.
CollectionId	Integer	This is the unique identifier for the collection that this Refund relates to.

SettlementByCollection entity

Property	Type	Remark
CollectionId	Integer	This is the unique identifier for a collection.
FailedCollectionId	Integer	This is the unique identifier for a failed collection. Value exists only if Status is failed.
CollectionScheduleId	Integer	This is the unique identifier of a collection schedule created when settlement amount is processed to recoup amount from Client.
ProcessingDates	List	collectionDate submissionDate settlementDate paymentDate
CollectionAmount	Double	Amount to recoup from client.
Status	String	Processing status of collection schedule.

PaymentReferences

Property	Type	Remark
Reference	String	This is unique reference generated by London and Zurich.
Amount	Double	Amount transferred in one bank transaction. Maximum amount is limited to 100000.

Examples

Following GetClientSettlementDetailsResponse is return when GetClientSettlementDetails request is sent:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetClientSettlementDetailsResponse xmlns="https://webservices.landz.co.uk">
      <GetClientSettlementDetailsResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:ClientSettlementDetails>
          <a:ClientId>4000</a:ClientId>
          <a:ClientSettlementId>3</a:ClientSettlementId>
          <a:CollectionAmount>94.5</a:CollectionAmount>
          <a:ConnectedMerchantChargeAmount>2.79</a:ConnectedMerchantChargeAmount>
          <a:ConnectedMerchantId>1</a:ConnectedMerchantId>
          <a:ConnectedMerchantVatAmount>0</a:ConnectedMerchantVatAmount>
          <a:CreatedDateTime>2021-04-20T17:14:48.15</a:CreatedDateTime>
          <a:FailedCollectionAmount>5</a:FailedCollectionAmount>
          <a:IndemnityClaimAmount>0</a:IndemnityClaimAmount>
          <a:NetdownAmount>0</a:NetdownAmount>
          <a:PaidDateTime>2021-04-20T17:15:27.3</a:PaidDateTime>
          <a:PaymentReference>
            <a:PaymentReference>
              <a:Amount>58.26</a:Amount>
              <a:Reference>NSC0000000003</a:Reference>
```

```
  </a:PaymentReference>
</a:PaymentReference>
<a:RefundAmount>27.25</a:RefundAmount>
<a:ServiceProviderChargeAmount>1</a:ServiceProviderChargeAmount>
<a:ServiceProviderVatAmount>0.2</a:ServiceProviderVatAmount>
<a:SettlementAmount>58.26</a:SettlementAmount>
<a:SettlementDate>2021-04-22T00:00:00</a:SettlementDate>
<a:Status>Processed</a:Status>
<a:SettlementComponentAndCharges>
  <a:ClientSettlementComponentAndCharges>
    <a:CollectionSettlements xmlns:b="http://schemas.datacontract.org/2004/07/HephaestusContract"/>
    <a:Collections xmlns:b="http://schemas.datacontract.org/2004/07/HephaestusContract">
      <b:CollectionClientSettlementComponentAndCharges>
        <b:ClientId>4000</b:ClientId>
        <b:ConnectedMerchantChargeAmount>1.36</b:ConnectedMerchantChargeAmount>
        <b:ConnectedMerchantId>1</b:ConnectedMerchantId>
        <b:ConnectedMerchantVatAmount>0.27</b:ConnectedMerchantVatAmount>
        <b:ServiceProviderChargeAmount>0</b:ServiceProviderChargeAmount>
        <b:ServiceProviderVatAmount>0</b:ServiceProviderVatAmount>
        <b:SettlementComponent>
          <b:Amount>46</b:Amount>
          <b:ClientId>4000</b:ClientId>
          <b:CollectionDates>
            <b:CollectionDate>2021-01-04T00:00:00</b:CollectionDate>
            <b:PaymentDate>2021-01-06T00:00:00</b:PaymentDate>
            <b:SettlementDate>2021-01-08T00:00:00</b:SettlementDate>
            <b:SubmissionDate>2020-12-30T00:00:00</b:SubmissionDate>
          </b:CollectionDates>
          <b:CollectionId>29063349</b:CollectionId>
          <b:CollectionScheduleId>0</b:CollectionScheduleId>
          <b:CustomerId>4000:TS4CYLK02</b:CustomerId>
          <b:Status>Successful</b:Status>
          <b:StatusId>1</b:StatusId>
        </b:SettlementComponent>
      </b:CollectionClientSettlementComponentAndCharges>
      <b:CollectionClientSettlementComponentAndCharges>
        <b:ClientId>4000</b:ClientId>
        <b:ConnectedMerchantChargeAmount>1.43</b:ConnectedMerchantChargeAmount>
        <b:ConnectedMerchantId>1</b:ConnectedMerchantId>
        <b:ConnectedMerchantVatAmount>0.29</b:ConnectedMerchantVatAmount>
        <b:ServiceProviderChargeAmount>0</b:ServiceProviderChargeAmount>
        <b:ServiceProviderVatAmount>0</b:ServiceProviderVatAmount>
        <b:SettlementComponent>
          <b:Amount>48.5</b:Amount>
          <b:ClientId>4000</b:ClientId>
          <b:CollectionDates>
            <b:CollectionDate>2021-01-04T00:00:00</b:CollectionDate>
            <b:PaymentDate>2021-01-06T00:00:00</b:PaymentDate>
            <b:SettlementDate>2021-01-08T00:00:00</b:SettlementDate>
            <b:SubmissionDate>2020-12-30T00:00:00</b:SubmissionDate>
          </b:CollectionDates>
          <b:CollectionId>29063350</b:CollectionId>
          <b:CollectionScheduleId>0</b:CollectionScheduleId>
          <b:CustomerId>4000:TS4CYLK02</b:CustomerId>
          <b:Status>Successful</b:Status>
          <b:StatusId>1</b:StatusId>
```

```
    </b:SettlementComponent>
  </b:CollectionClientSettlementComponentAndCharges>
</a:Collections>
<a:ConnectedMerchantChargeAmount>2.79</a:ConnectedMerchantChargeAmount>
<a:ConnectedMerchantVatAmount>0</a:ConnectedMerchantVatAmount>
<a:FailedCollections xmlns:b="http://schemas.datacontract.org/2004/07/HephaestusContract">
  <b:FailedCollectionClientSettlementComponentAndCharges>
    <b:ClientId>4000</b:ClientId>
    <b:ConnectedMerchantChargeAmount>0</b:ConnectedMerchantChargeAmount>
    <b:ConnectedMerchantId>1</b:ConnectedMerchantId>
    <b:ConnectedMerchantVatAmount>0</b:ConnectedMerchantVatAmount>
    <b:ServiceProviderChargeAmount>0</b:ServiceProviderChargeAmount>
    <b:ServiceProviderVatAmount>0</b:ServiceProviderVatAmount>
    <b:SettlementComponent>
      <b:Amount>5</b:Amount>
      <b:ClientId>4000</b:ClientId>
      <b:CollectionDate>2021-01-07T00:00:00</b:CollectionDate>
      <b:CollectionId>0</b:CollectionId>
      <b:CustomerId>4000:LZ40641</b:CustomerId>
      <b:DatePaid>2021-04-20T17:14:48.15</b:DatePaid>
      <b:FailedCollectionId>872465</b:FailedCollectionId>
      <b:FailedReason>0</b:FailedReason>
      <b:FailedReasonCode>0</b:FailedReasonCode>
      <b:PaidBeforeFail>true</b:PaidBeforeFail>
    </b:SettlementComponent>
  </b:FailedCollectionClientSettlementComponentAndCharges>
</a:FailedCollections>
<a:IndemnityClaims xmlns:b="http://schemas.datacontract.org/2004/07/HephaestusContract"/>
<a:Netdowns xmlns:b="http://schemas.datacontract.org/2004/07/HephaestusContract"/>
<a:Refunds xmlns:b="http://schemas.datacontract.org/2004/07/HephaestusContract">
  <b:RefundClientSettlementComponentAndCharges>
    <b:ClientId>4000</b:ClientId>
    <b:ConnectedMerchantChargeAmount>0</b:ConnectedMerchantChargeAmount>
    <b:ConnectedMerchantId>1</b:ConnectedMerchantId>
    <b:ConnectedMerchantVatAmount>0</b:ConnectedMerchantVatAmount>
    <b:ServiceProviderChargeAmount>1</b:ServiceProviderChargeAmount>
    <b:ServiceProviderVatAmount>0.2</b:ServiceProviderVatAmount>
    <b:SettlementComponent>
      <b:CancelledDate i:nil="true"/>
      <b:ClientId>4000</b:ClientId>
      <b:ClientPaidDate>2021-04-20T17:14:48.15</b:ClientPaidDate>
      <b:ClientSettlementDate>2021-04-23T00:00:00</b:ClientSettlementDate>
      <b:CollectionAmount>27.25</b:CollectionAmount>
      <b:CollectionDate>2020-12-15T00:00:00</b:CollectionDate>
      <b:CollectionId>28859793</b:CollectionId>
      <b:CreatedDateTime>2021-01-04T10:35:01.727</b:CreatedDateTime>
      <b:CustomerId>4000:LZ40024</b:CustomerId>
      <b:CustomerPaidDate>2021-01-05T10:35:01.727</b:CustomerPaidDate>
      <b:CustomerSettlementDate>2021-01-08T00:00:00</b:CustomerSettlementDate>
      <b:RefundAmount>27.25</b:RefundAmount>
      <b:RefundId>2</b:RefundId>
      <b:RefundStatus>Settled</b:RefundStatus>
      <b:RefundStatusId>2</b:RefundStatusId>
    </b:SettlementComponent>
  </b:RefundClientSettlementComponentAndCharges>
</a:Refunds>
```

```
            <a:ServiceProviderChargeAmount>1</a:ServiceProviderChargeAmount>
            <a:ServiceProviderVatAmount>0.2</a:ServiceProviderVatAmount>
          </a:ClientSettlementComponentAndCharges>
        </a:SettlementComponentAndCharges>
      </a:ClientSettlementDetails>
    </GetClientSettlementDetailsResult>
  </GetClientSettlementDetailsResponse>
 </s:Body>
</s:Envelope>
```

GetCollectionSummary

The GetCollectionSummary method is used to obtain summary information about all collections that have been made, and all collections are due to be made, from a customer within a supplied date range.

Request

Property	Required	Remarks
CustomerRef	Yes	The identifier of the customer whose collections are being queried.
FromDate	No	The earliest collection date that you want the query to report on – ISO 8601 format (YYYY-MM-DD) If this value is not supplied then today's date is used. The date must be a date that falls within a time period beginning two years prior to today and ending two years from today. If you select a date prior to two years before today then the value of two years before today will be used instead. If you select a date further than two years after today then the value of two years after today will be used instead.
ToDate	No	The latest collection date that you want the query to report on – ISO 8601 format (YYYY-MM-DD). If this value is not supplied then today's date is used. The date must be a date that falls within a time period beginning two years prior to today and ending two years from today. If you select a date prior to two years before today then the value of two years before today will be used instead. If you select a date further than two years after today then the value of two years after today will be used instead.

Response

Property	Type	Remark
Schedules	Data Structure	Contains 0 or more Schedule data structures – see below.

Schedule

Property	Type	Remark
ScheduledPaymentID	Integer	The identifier of the schedule.
CustomerRef	String	The customer reference.

FirstPaymentAmount	Decimal	As described in AddScheduledPayment.
FirstPaymentDate	Date (nullable)	As described in AddScheduledPayment.
RegularPaymentAmount	Decimal	As described in AddScheduledPayment.
RegularPaymentStartDate	Date (nullable)	As described in AddScheduledPayment.
RegularPaymentFrequency	Integer	See below.
NumberOfRegularPayments	Integer	See below.
NumberOfPaymentsRecieved	Integer	How many payments have been processed so far on this schedule.
ScheduleSetupDate	Date	Date when the schedule was set up.
ScheduleStatus		See below.
Collections	Data Structure	Contains 0 or more Collection data structures - see below.

The possible options for the value of the ScheduleStatus field are

ScheduleStatus field value	Description
1	Active
2	Completed
3	Cancelled
0	Unknown

Collection

Property	Type	Remark
Amount	Decimal	The amount of the collection.
CollectionDate	Date	The due date of the collection.
Status	String	See below.
CustomerRef	String	The identifier of the customer from whom the collection was/will be made
ScheduleID		The identifier (SchedulePaymentID) of the schedule from which this collection request was generated

The possible options for the value of the RegularPaymentFrequency field are:

RegularPaymentFrequency field value	Description
1	Monthly
3	Quarterly

6	Semi-annually (every 6 months)
12	Annually
101	Weekly
102	Every 2 weeks
104	Every 4 weeks

The NumberOfRegularPayments field is used to show how many same-valued payments are to be made (the first payment may be for a different amount). Any number above 0 indicates the number of same-valued payments that will be made. A value of 0 represents a continuous schedule without a defined end-point, so payments will be collected at the specified interval until the continuous authority is cancelled.

Collection Status

The Status of a Collection indicates whether a collection request has been sent to Bacs and, if so, whether the request resulted in a successful collection or a failed collection attempt.

Status	Interpretation
Successful	The collection request was sent to Bacs and did not fail.
Failed	The collection request was sent to Bacs and failed (typically for lack of funds in the payer's account although there may be another reason).
Indemnity	The collection request was sent to Bacs but payment was refunded due to some error under the Direct Debit Guarantee.
Projected	This collection will be requested in future.
Cancelled	This collection was not sent to Bacs.
Unprocessed	No collection request was/will be sent to Bacs.

Examples

An example of a call to the GetCollectionSummary web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:web="https://webservices.landz.co.uk"
            xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetCollectionSummary>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
          <web:SecurityCode>mysecuritycode</web:SecurityCode>
```

```
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:CustomerRef>1234:SBM015204</web1:CustomerRef>
        <web1:FromDate>2023-06-01</web1:FromDate>
        <web1:ToDate>2023-08-31</web1:ToDate>
      </web:request>
    </web:GetCollectionSummary>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetCollectionSummary web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetCollectionSummaryResponse xmlns="https://webservices.landz.co.uk">
      <GetCollectionSummaryResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
                                  xmlns:i=" http://www.w3.org /2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:Schedules>
          <a:Schedule>
            <a:CustomerRef>1234:SBM015204</a:CustomerRef>
            <a:FirstPaymentAmount>0</a:FirstPaymentAmount>
            <a:FirstPaymentDate i:nil="true"/>
            <a:NumberOfPaymentsReceived>23</a:NumberOfPaymentsReceived>
            <a:NumberOfRegularPayments>0</a:NumberOfRegularPayments>
            <a:RegularPaymentAmount>6.9900</a:RegularPaymentAmount>
            <a:RegularPaymentFrequency>1</a:RegularPaymentFrequency>
            <a:RegularPaymentStartDate>2021-10-01T00:00:00</a:RegularPaymentStartDate>
            <a:ScheduleSetupDate>2021-09-02T09:03:39.533</a:ScheduleSetupDate>
            <a:ScheduleStatus>1</a:ScheduleStatus>
            <a:ScheduledPaymentID>18128133</a:ScheduledPaymentID>
            <a:Collections>
              <a:Collection>
                <a:CustomerRef>1234:SBM015204</a:CustomerRef>
                <a:Amount>6.9900</a:Amount>
                <a:CollectionDate>2023-06-01T00:00:00</a:CollectionDate>
                <a:ScheduleID>18128133</a:ScheduleID>
                <a:Status>Successful</a:Status>
              </a:Collection>
              <a:Collection>
                <a:CustomerRef>1234:SBM015204</a:CustomerRef>
                <a:Amount>6.9900</a:Amount>
                <a:CollectionDate>2023-07-03T00:00:00</a:CollectionDate>
                <a:ScheduleID>18128133</a:ScheduleID>
                <a:Status>Successful</a:Status>
              </a:Collection>
              <a:Collection>
                <a:CustomerRef>1234:SBM015204</a:CustomerRef>
                <a:Amount>6.9900</a:Amount>
                <a:CollectionDate>2023-08-01T00:00:00</a:CollectionDate>
                <a:ScheduleID>18128133</a:ScheduleID>
                <a:Status>Projected</a:Status>
```

```
            </a:Collection>
          </a:Collections>
        </a:Schedule>
      </a:Schedules>
    </GetCollectionSummaryResult>
  </GetCollectionSummaryResponse>
 </s:Body>
</s:Envelope>
```

GetMaxRetryCount

Note: This method only has a use if the Xero integration is being used. There is no functionality in the system for retries.

If using the Xero integration, and a collection fails with the reason code “0: Refer to Payer”, then the collection may be re-attempted by the integration.

The value returned by this function gives the maximum number of times that a collection may be re-attempted.

Request

No additional properties over and above those of the base request are required in a request to GetMaxRetryCount. The value of the Group field of the Credentials element in the base request defines the group whose maximum rety count will be returned.

Response

Property	Type	Remark
MaxRetryCount	Integer	This value indicates how many times a collection may be retried after first failing with the reason code “0: Refer to Payer”. For example, a value of 0 indicates that the collection may not be retried. A value of 1 indicates that the collection may be retried once. A value of 2 indicates that the collection may be retried twice. A value of -1 indicates that the collection may be retried an unlimited number of times.

Examples

An example of a call to the GetMaxRetryCount web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:web="https://webservices.landz.co.uk"
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetMaxRetryCount>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
```

```
        <web:SecurityCode>mysecuritycode</web:SecurityCode>
      </web1:Credentials>
      <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    </web:request>
  </web:GetMaxRetryCount>
 </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetMaxRetryCount web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetMaxRetryCountResponse xmlns="https://webservices.landz.co.uk">
      <GetMaxRetryCountResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
                              xmlns:i=" http://www.w3.org /2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:MaxRetyCount>0</a:MaxRetyCount>
      </GetMaxRetryCountResult>
    </GetMaxRetryCountResponse>
  </s:Body>
</s:Envelope>
```

GetScheduleStatus

The GetScheduleStatus method returns an integer which represents the current status of the schedule with a given ScheduleID.

Request

Property	Required	Remarks
ScheduleID	Yes	The Integer value which is the identifier of the schedule the status of which we are querying. If the schedule does not pertain to your group then an error will be returned.

Response

The only field in the response, over and above those in the base Response class, is an integer ScheduleStatus field.

ScheduleStatus field value	Description
1	Active
2	Completed
3	Cancelled
0	Unknown

Examples

An example of a call to the GetScheduleStatus web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:web="https://webservices.landz.co.uk"
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetScheduleStatus>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
          <web:SecurityCode>mysecuritycode</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:ScheduleID>767920</web1:ScheduleID>
      </web:request>
    </web:GetScheduleStatus>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetScheduleStatus web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetScheduleStatusResponse xmlns="https://webservices.landz.co.uk">
      <GetScheduleStatusResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
                               xmlns:i=" http://www.w3.org /2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:ScheduleStatus>1</a:ScheduleStatus>
      </GetScheduleStatusResult>
    </GetScheduleStatusResponse>
  </s:Body>
</s:Envelope>
```

GetScheduleStatuses

The GetScheduleStatuses method allows the caller to query the status of multiple schedules in a single call. The method returns the requested list of schedule IDs each with their current status.

Request

Property	Required	Remarks
ScheduleIDs	Yes	A List of Integer values, each of which is the identifier of a schedule the status of which we are querying. The List may contain between 1 and 100 schedule ID values. If a given schedule ID does not pertain to a schedule in your group then no corresponding status will be returned.

Response

The only field in the response, over and above those in the base Response class, is a Data Structure called ScheduleIDAndStatus.

Property	Type	Remark
ScheduleIDAndStatus	Data Structure	Contains 0 or more ValueTupleOfintint data structures – see below.

Each ValueTupleOfintint contains a pair of integers representing a ScheduleID and a Status. The first integer represents the ScheduleID and the second that schedule's Status.

Property	Type	Remark
Item1	Integer	A ScheduleID matching one the requested ScheduleIDs.
Item2	Integer	A schedule Status integer, the possible values of which are described below.

The possible Status integers are as follows.

Item2 field (Status) value	Description
1	Active
2	Completed
3	Cancelled
0	Unknown

Additional Errors

Value	Meaning
130	A List of ScheduleIDs is required
131	The List of ScheduleIDs must contain at least 1 element
132	The List of ScheduleIDs may contain at most 100 elements

Examples

An example of a call to the GetScheduleStatuses web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:web="https://webservices.landz.co.uk"
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi"
                  xmlns:arr="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <soapenv:Header/>
  <soapenv:Body>
    <web:GetScheduleStatuses>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
          <web:SecurityCode> mysecuritycode </web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:ScheduleIDs>
          <arr:int>1833124</arr:int>
          <arr:int>1594289</arr:int>
          <arr:int>445032</arr:int>
        </web1:ScheduleIDs>
      </web:request>
    </web:GetScheduleStatuses>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetScheduleStatuses web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetScheduleStatusesResponse xmlns="https://webservices.landz.co.uk">
      <GetScheduleStatusesResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
                                 xmlns:i=" http://www.w3.org /2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
        <a:ScheduleIDAndStatus xmlns:b="http://schemas.datacontract.org/2004/07/System">
          <b:ValueTupleOfintint>
            <b:Item1>1833124</b:Item1>
            <b:Item2>1</b:Item2>
```

```
          </b:ValueTupleOfintint>
          <b:ValueTupleOfintint>
            <b:Item1>1594289</b:Item1>
            <b:Item2>1</b:Item2>
          </b:ValueTupleOfintint>
          <b:ValueTupleOfintint>
            <b:Item1>445032</b:Item1>
            <b:Item2>2</b:Item2>
          </b:ValueTupleOfintint>
        </a:ScheduleIDAndStatus>
      </GetScheduleStatusesResult>
    </GetScheduleStatusesResponse>
  </s:Body>
</s:Envelope>
```

GetWebhooks

The GetWebhooks method allows a caller to get a list of all available Webhooks in the system and indicates which ones the calling Client is subscribed to. The method returns a list of Webhook details each with their current callback URL.

Request

No additional properties over and above those of the base request are required in a request to GetWebhooks. The value of the GroupNo field of the Credentials element in the base request defines the Client whose webhook subscription information will be returned.

Response

The response, over and above those in the base Response class, returns a WebhookList object. Webhooks to which you are not subscribed will be empty.

Property	Type	Remark
WebhookList	Data Structure	Contains 0 or more Webhook data structures – see below.

Webhook

Property	Type	Remark
CallbackUrl	String	This is callback URL that was provided by client if this is an active subscription for this Webhook. If this is not populated, the client is not subscribed to this Webhook. Webhooks will be pushed to this URL when the related webhook is fired.
GroupNo	Integer	This is the unique identifier assigned to a Client of London & Zurich. Also known as a Client Identifier.
InactiveDate	Date	The timestamp of the moment that the webhook was deactivated.
SubscriptionKey	String	This is key/token/secret value provided by client. This is optional and if provided can be used as additional verification layer (for verification of received notification is from L&Z).
WebHookName	String	Unique Webhook name given to the events supported by system (see the Webhooks section for a list of available Webhooks).

Examples

An example of a Request call to the GetWebhooks web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:GetWebhooks="">
   <web:request>
    <web1:Credentials>
     <web:Group>1234</web:Group>
     <web:SecurityCode> mysecuritycode </web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
   </web:request>
  </web:GetWebhooks>
 </soapenv:Body>
</soapenv:Envelope>
```

An example response to a GetWebhooks web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <GetWebhooksResponse xmlns="https://webservices.landz.co.uk">
   <GetWebhooksResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <a:Message>Success</a:Message>
    <a:ResponseCode>0</a:ResponseCode>
    <a:WebhookList>
     <a:Webhook>
      <a:CallbackUrl i:nil="true"/>
      <a:GroupNo>0</a:GroupNo>
      <a:InactiveDate i:nil="true"/>
      <a:SubscriptionKey>r7whrrh3r73hr</a:SubscriptionKey>
      <a:WebHookName>CollectionCreated</a:WebHookName>
     </a:Webhook>
     <a:Webhook>
      <a:CallbackUrl i:nil="true"/>
      <a:GroupNo>0</a:GroupNo>
      <a:InactiveDate i:nil="true"/>
      <a:SubscriptionKey>r7whrrh3r73hr</a:SubscriptionKey>
      <a:WebHookName>CollectionUpdated</a:WebHookName>
     </a:Webhook>
     <a:Webhook>
      <a:CallbackUrl>https://www.londonandzurich.co.uk/</a:CallbackUrl>
      <a:GroupNo>4000</a:GroupNo>
      <a:InactiveDate i:nil="true"/>
      <a:SubscriptionKey>123r7whrrh3r73hr</a:SubscriptionKey>
      <a:WebHookName>CustomerCreated</a:WebHookName>
     </a:Webhook>
     <a:Webhook>
      <a:CallbackUrl>https://www.londonandzurich.co.uk/</a:CallbackUrl>
      <a:GroupNo>4000</a:GroupNo>
      <a:InactiveDate i:nil="true"/>
      <a:SubscriptionKey>123r7whrrh3r73hr</a:SubscriptionKey>
```

```
        <a:WebHookName>CustomerSuspended</a:WebHookName>
       </a:Webhook>
       <a:Webhook>
        <a:CallbackUrl>https://www.londonandzurich.co.uk/</a:CallbackUrl>
        <a:GroupNo>4000</a:GroupNo>
        <a:InactiveDate i:nil="true"/>
        <a:SubscriptionKey>123r7whrrh3r73hr</a:SubscriptionKey>
        <a:WebHookName>CustomerUpdated</a:WebHookName>
       </a:Webhook>
       <a:Webhook>
        <a:CallbackUrl i:nil="true"/>
        <a:GroupNo>0</a:GroupNo>
        <a:InactiveDate i:nil="true"/>
        <a:SubscriptionKey i:nil="true"/>
        <a:WebHookName>ScheduleCancelled</a:WebHookName>
       </a:Webhook>
       <a:Webhook>
        <a:CallbackUrl i:nil="true"/>
        <a:GroupNo>0</a:GroupNo>
        <a:InactiveDate i:nil="true"/>
        <a:WebHookName>ScheduleCompleted</a:WebHookName>
       </a:Webhook>
       <a:Webhook>
        <a:CallbackUrl i:nil="true"/>
        <a:GroupNo>0</a:GroupNo>
        <a:InactiveDate i:nil="true"/>
        <a:SubscriptionKey i:nil="true"/>
        <a:WebHookName>ScheduleCreated</a:WebHookName>
       </a:Webhook>
       <a:Webhook>
        <a:CallbackUrl i:nil="true"/>
        <a:GroupNo>0</a:GroupNo>
        <a:InactiveDate i:nil="true"/>
        <a:SubscriptionKey i:nil="true"/>
        <a:WebHookName>SettlementCreated</a:WebHookName>
       </a:Webhook>
      </a:WebhookList>
     </GetWebhooksResult>
    </GetWebhooksResponse>
   </s:Body>
</s:Envelope>
```

RemoveWebhookSubscription

The RemoveWebhookSubscription method allows the caller to remove subscription of a webhook for a group from L&Z system. After removing a webhook successfully, the L&Z system will stop sending notification data for that Webhook. This method returns the base response, indicating success or failure with failure reason.

Request

Only a valid Webhook Name is required over and above those of the base request field values in a request to RemoveWebhookSubscription.

Property	Type	Required	Remark
WebHookName	String	Yes	The valid name of webhook that need to be removed. The related event notification will be stopped after successful removal of webhook.

Response

No extra data beyond the ResponseBase data structure.

Examples

An example of a Request call to the RemoveWebhookSubscription web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:web="https://webservices.landz.co.uk"
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:RemoveWebhookSubscription="">
   <web:request>
    <web1:Credentials>
     <web:Group>1234</web:Group>
     <web:SecurityCode> mysecuritycode </web:SecurityCode>
    </web1:Credentials>
    <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
    <web1:WebHookName>ScheduleCompleted</web1:WebHookName>
   </web:request>
  </web:RemoveWebhookSubscription>
 </soapenv:Body>
</soapenv:Envelope>
```

An example response to a RemoveWebhookSubscription web service method call. This is failure response where the supplied Webhook name does not have an active subscription. A response when a subscription is found and removed would yield a base response with success message.

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <RemoveWebhookSubscriptionResponse xmlns="https://webservices.landz.co.uk">
   <RemoveWebhookSubscriptionResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
     <b:string>Webhook Subscription not found.</b:string>
    </a:Errors>
    <a:Message>Please check the errors to identify the cause</a:Message>
    <a:ResponseCode>4</a:ResponseCode>
   </RemoveWebhookSubscriptionResult>
  </RemoveWebhookSubscriptionResponse>
 </s:Body>
</s:Envelope>
```

AddWebhookSubscription

The AddWebhookSubscription method allows the caller to add new subscription to a Webhook or update an existing Webhook subscription's callback URL for a group in L&Z system. After adding a webhook successfully, L&Z system will start sending notification information for the webhook. Webhooks will only be triggered after a subscription is made – the system will not retro-actively generate Webhooks. The method returns the base response with success or failure with failure reason.

Request

A valid Webhook name and CallbackURL is required over and above those of the base request field values in a request to AddWebhookSubscription.

Property	Type	Required	Remark
WebHookName	String	Yes	The valid name of Webhook that the caller wishes to subscribe to
SubscriptionKey	String	No	This is key/token/secret value provided by client. This is optional and if provided can be used as additional verification layer (for verification of received notification is from L&Z).
CallbackUrl	String	Yes	A URL that can receive Webhook notifications. A unique URL can be used for each Webhook. If the same callback URL is used in all webhook subscriptions, then the received notification can be identified using 'Webhook' key in the header of the received notification. See Webhook Notification section for more details. This is limited to 100 characters

Response

No extra data is required in the request, beyond the ResponseBase data structure.

Examples

An example of a Request call to the AddWebhookSubscription web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="https://webservices.landz.co.uk" xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
 <soapenv:Header/>
 <soapenv:Body>
  <web:AddWebhookSubscription="" >
   <web:request>
```

```
      <web1:Credentials>
        <web:Group>1234</web:Group>
        <web:SecurityCode> mysecuritycode </web:SecurityCode>
      </web1:Credentials>
      <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
      <web1:CallbackUrl>https://www.londonandzurich.co.uk/</web1:CallbackUrl>
      <web1:SubscriptionKey>123qwe456</web1:SubscriptionKey>
      <web1:WebHookName>CustomerUpdated</web1:WebHookName>
    </web:request>
  </web:AddWebhookSubscription >
 </soapenv:Body>
</soapenv:Envelope>
```

An example response to a AddWebhookSubscription web service method call. This is a failure response given when the supplied webhook name already has a subscription with same callback url. A response when a subscription is found and removed would yield a base response with success message.

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
 <s:Body>
  <AddWebhookSubscriptionResponse xmlns="https://webservices.landz.co.uk">
   <AddWebhookSubscriptionResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
     <b:string>Webhook Subscription already exists.</b:string>
    </a:Errors>
    <a:Message>Please check the errors to identify the cause</a:Message>
    <a:ResponseCode>4</a:ResponseCode>
   </AddWebhookSubscriptionResult>
  </AddWebhookSubscriptionResponse>
 </s:Body>
</s:Envelope>
```

Webhooks

A Push notification is created if a valid subscription for the Webhook and Client exists in the system. Notifications will only be sent once.

The header will contain a key-value pairs

1. Name of webhook (Event Name) that the notification data relates to, e.g. “Webhook”: “CollectionCreated”.
2. Signature is unique value calculated by L&Z for every notification using the subscription key/secret/token value provided by client while creating webhook subscription. E.g. “Signature”:” 2ae1122f34s987654321x76657789cv50b654as852d56da66d778xx0fsi9d”
   How to use- C# example:

```
[HttpPost]
0 references
public ActionResult GetNotifications([FromBody] string webHookNotification)
{
    var webhookName = HttpContext.Request.Headers["Webhook"][0] ?? "";
    var signature = HttpContext.Request.Headers["Signature"][0] ?? "";
    var key = System.Configuration.ConfigurationManager.AppSettings["MyWebhookSubscriptionSecret"];

    if (!string.IsNullOrEmpty(webhookName) && !string.IsNullOrEmpty(signature))
    {
        var keyBytes = new System.Security.Cryptography.HMACSHA256(System.Text.Encoding.UTF8.GetBytes(key));
        var computedSignature = keyBytes.ComputeHash(System.Text.Encoding.UTF8.GetBytes(webHookNotification));
        var result = BitConverter.ToString(computedSignature).Replace("-", "").ToLower();

        if (!string.IsNullOrEmpty(result) && result == signature)
        {
            switch (webhookName)
            {
                case "CollectionCreated":
                    // Do CollectionCreated stuff
                    break;
                case "CollectionUpdated":
                    // Do CollectionUpdated stuff
                    break;
                case "ScheduleCancelled":
                    // Do ScheduleCancelled stuff
                    break;
                default:
                    break;
            }
            return Ok();
        }
        else
        {//Add code to handle Invalid or data from unknow source
            return new BadRequestResult();
        }
    }
    else
    {
        return  new BadRequestResult();
    }
}
```

The response contains a JSON formatted string.

Here is a list of all available Webhooks:

Key	Value
Webhook	• CollectionCreated • CollectionUpdated • CustomerCreated • CustomerSuspended • CustomerUpdated • ScheduleCancelled • ScheduleCompleted • ScheduleCreated • SettlementCreated

CollectionCreated

This Webhook will fire after a Collection is generated in the L&Z System and when Client has subscribed for this Webhook.

Example of Collection created notification:

{"CollectionId":50495751,"ReasonCode":null,"CustomerRef":"4000:1259JJF","Status":"Processing","CreatedDateTime":"2024-05-15T15:13:34.027","Id":6360,"GroupNo":4000,"Type":"Created","StatusCode":0}"

Property	Type	Remark
CollectionId	Integer	Unique Collection Id in L&Z system
ReasonCode	Integer	Value is 0.
CustomerRef	String	Unique Customer reference related to this collection.
Status	String	Collection Status. List of possible value: ▪ Unprocesssed ▪ Processing
CreatedDateTime	DateTime	Date and time when collection is created in L&Z system. The date value is in ISO 8601 format (YYYY-MM-DD).
Id	Integer	Unique identifier of Collection webhook.
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	This is collection Webhook type. Value is “Created”
StatusCode	String	Webhook processing status

CollectionUpdated

This Webhook will fire after a Collection is updated in the L&Z System and when Client has subscribed for this Webhook.

Example of Collection updated notification:

{"CollectionId":50495751,"ReasonCode":1,"CustomerRef":"4000:1259JJF","Status":"Failure","CreatedDateTime":"2024-06-15T15:13:34.027","Id":6390,"GroupNo":4000,"Type":"Updated","StatusCode":0}"

Property	Type	Remark
CollectionId	Integer	Unique Collection Id in L&Z system
ReasonCode	Integer	This field has value only if Collection is failed and Value indicated the standard reason code for failure.
CustomerRef	String	Unique Customer reference related to this collection.
Status	String	Collection Status. List of possible value: ▪ Processing ▪ Success ▪ Indemnity ▪ Failure
CreatedDateTime	DateTime	Date and time when collection is updated in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
Id	Integer	Unique identifier of Collection webhook
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Value is “Updated”
StatusCode	String	Webhook processing status

CustomerCreated

This Webhook will fire after a customer is created in the L&Z System and when Client has subscribed for this Webhook.

Example of Customer created notification:

"{"CustomerRef":"4000:1234FIF9TJ","Status":"Active","CreatedDateTime":"2024-05-15T15:13:44.28","Id":3303,"GroupNo":4000,"Type":"Created","StatusCode":0}"

Property	Type	Remark
CustomerRef	String	Unique Customer reference which is ‘<GroupNo>:<DdiReference>’.

Status	String	Customer Status. Value is: “Active”
CreatedDateTime	DateTime	Date and time when customer is created in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Webhook type. Value is “Created”
StatusCode	String	Webhook processing status

CustomerSuspended

This Webhook will fire after a customer is suspended in the L&Z System and when Client has subscribed for this Webhook.

Example of Customer Suspended notification:

"{"CustomerRef":"4000:1234FIF9TJ","Status":"Suspended","CreatedDateTime":"2024-05-16T13:13:44.28","Id":3350,"GroupNo":4000,"Type":"Updated","StatusCode":0}"

Property	Type	Remark
CustomerRef	String	Unique Customer reference which is ‘<GroupNo>:<DdiReference>’.
Status	String	Customer Status. Value is “Suspended”.
CreatedDateTime	DateTime	Date and time when customer is suspended in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Webhook type. Value is “Updated”.
StatusCode	String	Webhook processing status

CustomerUpdated

This Webhook will fire after a customer is updated for any information in the L&Z System and when Client has subscribed for this Webhook.

Example of Customer updated notification:

"{"CustomerRef":"4000:1234FIF9TJ","Status":"Active","CreatedDateTime":"2024-05-15T17:13:44.28","Id":3335,"GroupNo":4000,"Type":"Updated","StatusCode":0}"

Property	Type	Remark

CustomerRef	String	Unique Customer reference which is '<GroupNo>:<DdiReference>'.
Status	String	Customer Status. Value is "Active"
CreatedDateTime	DateTime	Date and time when customer is update in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Webhook type. Value is "Updated"
StatusCode	String	Webhook processing status

ScheduleCancelled

This Webhook will fire after a schedule is cancelled in the L&Z System and when Client has subscribed for this Webhook.

Example of Schedule cancelled notification:

"{"ScheduleId":2871234,"CustomerRef":"4000:1234AAAA","ExternalRef":null,"CreatedDateTime":"2024-05-15T17:20:50.127","Id":4065,"GroupNo":4000,"Type":"Cancelled","StatusCode":0}"

Property	Type	Remark
ScheduleId	Integer	Unique Schedule Id in L&Z system
CustomerRef	String	Unique Customer reference related to this schedule.
ExternalRef	String	Client provided schedule reference value.
CreatedDateTime	DateTime	Date and time when schedule is cancelled in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
Id	Integer	Unique identifier of schedule webhook.
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Schedule Webhook type. Value is "Cancelled"
StatusCode	String	Webhook processing status

ScheduleCompleted

This Webhook will fire after a schedule is marked as completed in the L&Z System and when Client has subscribed for this Webhook.

Example of Schedule Completed notification:

"{"ScheduleId":2871244,"CustomerRef":"4000:1234BBAA","ExternalRef":"ClientRef","CreatedDateTime":"2024-05-12T15:12:50.207","Id":4055,"GroupNo":4000,"Type":"Completed","StatusCode":0}"

Property	Type	Remark
ScheduleId	Integer	Unique Schedule Id in L&Z system
CustomerRef	String	Unique Customer reference related to this schedule.
ExternalRef	String	Client provided schedule reference value.
CreatedDateTime	DateTime	Date and time when schedule is marked as completed in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
Id	Integer	Unique identifier of schedule webhook.
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Schedule webhook type. Value is “Completed”
StatusCode	String	Webhook processing status

ScheduleCreated

This Webhook will fire after a schedule is created in the L&Z System and when Client has subscribed for this Webhook.

Example of Schedule created notification:

"{"ScheduleId":2871234,"CustomerRef":"4000:1234AAAA","ExternalRef":null,"CreatedDateTime":"2024-05-15T15:14:50.267","Id":4060,"GroupNo":4000,"Type":"Created","StatusCode":0}"

Property	Type	Remark
ScheduleId	Integer	Unique Schedule Id in L&Z system
CustomerRef	String	Unique Customer reference related to this schedule.
ExternalRef	String	Client provided schedule reference value.
CreatedDateTime	DateTime	Date and time when schedule is created in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
Id	Integer	Unique identifier of schedule webhook
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Schedule webhook type. Value is “Created”

StatusCode	String	Webhook processing status

SettlementCreated

This Webhook will fire after a client settlement is created in the L&Z System and when Client has subscribed for this Webhook.

Example of Settlement is created notification:

"{"SettlementId":212132,"SettlementAmount":55.5000,"CreatedDateTime":"2024-05-15T15:15:05.123","SettlementDate":"2024-05-14","Id":1,"GroupNo":5046,"Type":"Created","StatusCode":0}"

Property	Type	Remark
SettlementId	Integer	Unique Settlement Id in L&Z system
SettlementAmount	Money	Amount to be settled in client's account.
SettlementDate	Date	Date on which amount will be settled in client's account. The date value is in ISO 8601 format (YYYY-MM-DD).
CreatedDateTime	DateTime	Date and time when settlement is created in the system. The date value is in ISO 8601 format (YYYY-MM-DD).
Id	Integer	Unique identifier of settlement webhook
GroupNo	Integer	Unique 4-digit number assigned to client also known as clientId.
Type	String	Settlement webhook type. Value is “Created”
StatusCode	String	Webhook processing status

Test System Only

The following method calls are only available on the test system.

SetTestCollectionStatus

The SetTestCollectionStatus method is only available on the test system, not on live. The method can be used to set the status of a collection that is already on the test system.

Once the status of a collection has been updated the following web service method calls will reflect the new status;

GetCollectionsByDate()

GetFailedCollectionsByDate()

GetIndemnitiesByDate()

GetAllIndemnitiesByCreatedDate()

Request

Property	Required	Remarks
CollectionID	Yes	The Integer value which is the identifier of the collection the status of which we are requesting to be changed. If the collection does not pertain to your group then an error will be returned.
NewStatus	Yes	The new status that we wish the collection to take. Acceptable values are… "Successful", "Success", "S", "Failed", "Failure", "F", "Processing", "P", and "Unprocessed" or "U".
IsIndemnityClaim	No	A Boolean value indicating whether the collection failed because of an indemnity claim. If NewStatus is *not* "Failed" then this value is ignored. If NewStatus is "Failed" and this value is not supplied then it will be assumed to take the value false.
ReasonCode	Yes if NewStatus is "Failed" else No	The ReasonCode representing the reason that the collection failed. See below. If NewStatus is *not* "Failed" then this value is ignored.

ReasonCode values for collections which failed because of indemnity claims

FailureCode when MinorResult is "I"	FailureDescription
1	Amount and/or date of Direct Debit differs from advance notice
2	No advance notice was received by the payer
3	DDI cancelled by paying bank
4	Payer has cancelled DDI direct with service user
5	No instruction held. Payer disputes having given authority

6	Signature on DDI is fraudulent or not in accordance with account authorised signature(s).
7	Claim raised at service user's request after Direct Debit applied to payer's account.
8	Service user name disputed. Payer does not recognise service user collecting Direct Debit.

ReasonCode values for collections which failed for reasons *OTHER* than indemnity claims

FailureCode when MinorResult is NOT "I"	FailureDescription
0	Refer to Payer
1	Instruction Cancelled
2	Payer Deceased
3	A/C Transferred to another Bank
4	Advance Notice Disputed
5	No Account
6	No Instruction
7	Amount Differs
8	Amount Not Yet Paid
9	Presentation Overdue
A	Originator Differs
B	Account Closed
G	Group Request
X	CMS Request

Note

The live London and Zurich system will suspend the customer if a collection fails for any reason other than "0: Refer to Payer" (this includes all failures due to indemnity claims). To similarly suspend a customer on the test system use the SuspendCustomer() webservice call which is detailed elsewhere in this document.

The UpdateCustomer() webservice call, which is detailed elsewhere in this document, can be used to undo the suspension of a customer.

Response

No extra properties.

Additional Errors

Value	Meaning
130	Date range must span one year or less
131	You can only select dates on or after today

Examples

An example of a call to the SetTestCollectionStatus web service method;

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:web="https://webservices.landz.co.uk"
                  xmlns:web1="http://schemas.datacontract.org/2004/07/WebApi">
  <soapenv:Header/>
  <soapenv:Body>
    <web:SetTestCollectionStatus>
      <web:request>
        <web1:Credentials>
          <web:Group>1234</web:Group>
          <web:SecurityCode>mysecuritycode</web:SecurityCode>
        </web1:Credentials>
        <web1:RemoveErrorDescription>false</web1:RemoveErrorDescription>
        <web1:CollectionID>302817</web1:CollectionID>
        <web1:NewStatus>Failed</web1:NewStatus>
        <web1:IsIndemnityClaim>true</web1:IsIndemnityClaim>
        <web1:ReasonCode>4</web1:ReasonCode>
      </web:request>
    </web:SetTestCollectionStatus>
  </soapenv:Body>
</soapenv:Envelope>
```

An example response to a SetTestCollectionStatus web service method call;

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <SetTestCollectionStatusResponse xmlns="https://webservices.landz.co.uk">
      <SetTestCollectionStatusResult xmlns:a="http://schemas.datacontract.org/2004/07/WebApi"
                                     xmlns:i=" http://www.w3.org /2001/XMLSchema-instance">
        <a:Errors xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:Message>Success</a:Message>
        <a:ResponseCode>0</a:ResponseCode>
      </SetTestCollectionStatusResult>
    </SetTestCollectionStatusResponse>
  </s:Body>
</s:Envelope>
```

Glossary of Terms

Client (also referred to as a **Group**) - This is London & Zurich's client.

Group Number (also referred to as **Client Identifier**) - This is the unique identifier assigned to a **Client** of London & Zurich.

Customer (also referred to as **Account**, **Customer's Account**, **End Customer** or **Client's Customer**) - This term is used to refer to a **Client's** customer; an entity that has agreed to a DDI permitting the **Client** to request Direct Debits from their bank account.

Customer Reference (also referred to as **Account Reference** or **End Customer Reference**) - This is the unique identifier for a **Customer**. It is formed from a concatenation of the **Group Number**, a colon, and the customer's **DDI Reference**.

DDI Reference (also referred to as the **Customer Identifier** or **Account Only**) - This is the Direct Debit instruction identifier that is associated with a **Customer**. A DDI Reference must be unique for a given **Service User Number.**

Scheduled Payment (also referred to as a **Continuous Authority** or **Schedule**) – An instruction for London & Zurich to generate a series of one or more Direct Debit collection requests under the authority of a DDI.

Service Provider is entity (in this case London and Zurich) who provides services to Clients and Connected Merchants.

Service User (also referred to as the **Originator**) – This is a BACS term for an entity registered as a user of BACS services.

Service User Number or **SUN** (also referred to as **Originator's Identification Number** or **OIN**) - This is a six digit number allocated by BACS to identify a **Service User.**

Suspended (also referred to as **End-Dated** or **Ended**) – A **Customer** that has been suspended is sometimes referred to as "**Inactive**", as opposed to "**Active**". If a **Customer** is suspended then no Direct Debit collections will be attempted from that **Customer**'s account. A **Customer** may become suspended due to an instruction from Bacs (for example if the customer were to cancel their DDI with their bank) or due to a voluntary suspension at the request of the **Client** or London & Zurich. To remove the suspension of a **Customer**, a request should be made to "reinstate" the **Customer**.

Version History

10th May 2024

- Added GetCollectionByCollectionId method

20th March 2024

- Added AddCollectionSchedule and GetSchedulesByExternalReference methods

21st February 2024

- Added GetSchedulesByCustomer and GetScheduleByScheduleId methods

30th August 2023

- Wording clarification for GetMaxRetryCount method – only for use with Xero integration

14th August 2023

- GetScheduleStatuses method added

21st July 2023

- GetMaxRetryCount method added
- GetScheduleStaus method added

13th July 2023

- SetTestCollectionStatus method added

11th July 2023

- GetCollectionSummary method added

9th June 2023

- Added a Method Request Size Limit section to the Overview

22nd May 2023

- Updated GetCustomerDetails Collections and Failed Collections objects to correctly reflect what is being returned by the webservice call

24th February 2023

- Updated information on the test environment

9th September 2022

- Refund methods marked as unavaliable

11th August 2022

- GetEarliestCollectionDate method correction: Customer Ref not required
- GetLatestSubmissionDate method correction: Customer Ref not required

19th November 2021

- GetEarliestCollectionDate method details added
- GetLatestSubmissionDate method details added

06th October 2020

- GetRefunds method added
- SendCreateRefund method added
- SendUpdateRefund method added
- SendCancelRefund method added
- GetClientSettlements method added
- GetClientSettlementsByCreationDate method added
- GetClientSettlementsByPaidDate method added
- GetClientSettlementDetails method added

21st December 2016

- GetBankAndBranchName method added
- IsDdiReferenceAvailable method added
- AddCustomer additional errors list updated

10th June 2016

- GetCustomerStatus method added

18th May 2016

- Advance Notice timings updated
- Error code description modifications
- IsServiceAvailable ping method added
- BankModulusAndSortcodeDDEnabledCheck method added

18th April 2016

- Advance notice and account setup timings document reference added to the AddScheduledPaymentMethod section.

13th May 2015

- Changed lead time for AddScheduledPayment from seven days to five.

28th April 2015

- Added AUDDIS reason codes as part of the GetBacsCustomerUpdates.
- Added GetCustomersLastUpdated.

23rd April 2015

- Added new service SendIndemnityPayment
- Corrected data type definitions on some of the calls.
- Renamed PaymentID to ScheduledPaymentID where relevant.