

1 Top-Up by Voucher Data Contract

1.1 Overview

This section provides information on the flow of data between the Top-Up by Voucher product and the web service. The purpose of this REST-ful web service is to allow a back-end system to add a given amount of credit to a customer's account based on a pre-paid voucher number.

1.2 Request

Requests are made using HTTP POST requests, which pass the required caller details to the web service. For the Top-Up by Voucher product, the caller is identified using an Identification Module at the start of the callflow. The following parameters are then passed into the Query web service:

Table 1 Top-Up by Voucher web service request parameters

Parameter	Description	Example Values
AccountNumber	An arbitrary value retrieved by the Identification Module, to be passed to the web service to retrieve caller's account details.	"ABC12345", "123456789"
VoucherNumber	The long number assigned to the voucher being used to top up. This number should indicate the amount of credit to add to the customer's account.	"123456789012345"

A typical URL might look like this:

```
http://localhost:8080/fish-services/test/ActivateTopUp.jsp
```

And a typical HTTP POST body might look like this:

```
cli=02890571100&dnis=7896&sessionId=1234%2D3AAF%2D3372&AccountNumber=12345678&VoucherNumber=123456789012345
```

1.3 Response

The XML response specifies the overall status of the lookup, i.e. "success", or some other return code such as "agent", and provides a mechanism to set arbitrary variables in the call session.

When specifying variables in the response, you can cause some or all key-value pairs to be attached to the call via the CTI (where the platform supports it) by including an optional "attach" attribute with a value of "true" or to set them as the CLI data by including an optional "remember" attribute with a value of "true".

A typical XML response looks like this:

```
<activateTopUpResults>
  <status>success</status>
  <variables>
    <variable name="TopUpReference" value="1234567"/>
  </variables>
</activateTopUpResults>
```

A top up attempt where the caller's voucher number is not found might look like this:

```
<activateTopUpResults>
  <status>not found</status>
</activateTopUpResults>
```

A top up attempt where an error occurs might look like this:

```
<activateTopUpResults>
  <status>error</status>
</activateTopUpResults>
```

A top up attempt where there is an originator mismatch might look like this:

```
<activateTopUpResults>  
  <status>originator mismatch</status>  
</activateTopUpResults>
```

A top up attempt where the voucher has already been activated might look like this:

```
<activateTopUpResults>  
  <status>already activated</status>  
</activateTopUpResults>
```

A top up attempt where the voucher has expired might look like this:

```
<activateTopUpResults>  
  <status>expired</status>  
</activateTopUpResults>
```

A top up attempt where the account is locked out might look like this:

```
<activateTopUpResults>  
  <status>account locked</status>  
</activateTopUpResults>
```



HTTP response codes other than "200" will be treated as an error.

1.3.1 Statuses

The <status> element is the only mandatory element. The following statuses can be returned:

- “success” – indicates that the query was successful
- ‘not found’ – use this status if no details are found for the caller
- ‘error’ – the application’s error handling path will be followed in the callflow
- ‘orgid mismatch’ – this status should be used if the customer is trying to use a 3 Ireland voucher for 3 UK.
- ‘already activated’ – this status would indicate that the voucher number entered has already been used before
- ‘expired’ – the voucher which the customer has tried to use has passed the given expiry date
- ‘locked’ – the customer has made too many unsuccessful attempts and their account has been locked for 30 minutes
- any other status – causes the Module to return with that status.