



# TOTSCo Hub Developer Guide

## API Specifications

Version Draft V0.2  
31/03/2023



## Contents

<b>1. Introduction.....</b>	<b>4</b>
1.2 Change Log .....	4
1.3 Contributing Authors.....	4
<b>2. TOTSCo Hub Integration Specification.....</b>	<b>4</b>
1.4 Letterbox API Specification .....	5
1.5 Directory API Specification .....	12
<b>3. Security Implementation .....</b>	<b>16</b>
<b>4. Routing ID Summary .....</b>	<b>16</b>
<b>Appendix A – Sample Directory Listing.....</b>	<b>17</b>



## Figures

Figure 1 – Post Office JSON Message Envelope Structure .....	9
Figure 2 – Directory JSON Structure.....	14



## 1. Introduction

The TOTSCo Hub Developer Guide API Specification document complements and supports the One Touch Switch (OTS) Technical Design and the OTS Industry Process document.

This document contains a draft of the message envelope structure as well as a draft Directory and Letterbox API specification.

For a definition of the OTS process, please refer to the OTS Industry Process document.

### 1.1.1 Proposed Features

Any features described in this document which are awaiting change control approval are highlighted in yellow and included to ensure implementations take these into consideration in design.

### 1.1.2 Open Issues

Any aspects of this document not yet defined are highlighted in green. They are included to identify that changes are going to be needed to the design but the details have not yet been established to any level or maturity to include for information and will be added in subsequent versions once those specifications have been finalised.

## 1.2 Change Log

<b>Version Date Changed By</b>	<b>Reason for change</b>
V0.1 First draft 27/03/2023 Dave Stubbs	API Specification for the TOTSCo hub. First release for hub vendor.
V0.2 Updated draft 31/03/2023 Dave Stubbs	Added specification of the directory API.

## 1.3 Contributing Authors

<b>Author</b>	<b>Organisation</b>
Dave Stubbs	Virgin Media
Niall Gillespie	BT

## 2. TOTSCo Hub Integration Specification



The TOTSCo hub provides the mechanisms to deliver messages from one party to another in an environment where it is impractical for all parties to talk to each other directly.

The analogy of a post office is appropriate as TOTSCo are the agent who you will accept a senders message and be responsible for delivering it on their behalf to the intended recipient. Senders do not need to find a way to hand deliver the message directly. In architecture parlance, this is a hub and spoke mechanism as opposed to point to point.

Any IT based messaging system requires standards to ensure interoperability, and to that end all messages sent via the TOTSCo hub will be represented in JSON format and delivered using REST APIs.

Messages are made up of an envelope containing the delivery data needed for the TOTSCo hub to route the message to the correct destination, including a return address for replies and failures, and a message body.

The hub vendor does not need to know anything about the message body, that information is only for the sender and recipient to know and understand.

#### **1.4 Letterbox API Specification**

The TOTSCo hub letterbox API specification defines how messages will be sent to and received from the TOTSCo hub. The API specification is separate from the industry processes message formats as the hub does not need to know anything about the message format itself. The hub acts on a routingID which may or may not be related to the message format. Examples of current routing IDs are described in §3 of this document.

The requirement is that both the TOTSCo hub vendor, and the TOTSCo hub users all implement the same API specification. This makes it simpler to implement the messaging protocols in a uniform way as well as supporting the ability to perform peer to peer testing.

A sample definition of the letterbox API specification can be found at the following URL.

<https://app.swaggerhub.com/apis/DAVE.STUBBS/letterbox/0.4.0>

The letterbox provides a mechanism to deliver a message via the TOTSCo hub to a named recipient. The TOTSCo hub will only process the message envelope, to understand what it is for and be able to process it correctly, and not the message body.

For example, certain messages sent to the hub will be subject to a delivery timer with backoff and fallout policies. Others will require that the hub delivers the message no matter what. The policies for delivery messages will be industry-agreed, and will be defined in the hub and not specified by the sender.

Please note, the Letterbox API specification is not specific to any one messaging or industry process. It is designed as a standard, reusable interface to facilitate a hub and spoke message



distribution framework to support the adoption of near real time message processing and guaranteed message delivery where peer to peer interfaces and mechanisms are impractical, or lack the security or functionality the hub provides, e.g. email, SFTP etc.

### 1.4.1 The post API Interface

The post API takes a JSON message from an authorised source and delivers it to an identified destination. The information needed to route that message is defined within the messages envelope contained within the JSON message.

For the post API, the OAuth credentials of the sender of the message will be matched against the source information in the envelope to ensure the messages are not being spoofed.

The API URI format provided by the TOTSCo hub will conform to the following convention.

**https://{fqdn}/letterbox/{version}/post**

The elements of the URI are as follows.

URI Element	Description	Format
FQDN	The Fully Qualified Domain Name of the provider of the letterbox API. The TOTSCo hub FQDN will be provided by TOTSCo. every consumer of the TOTSCo hub will need to provide their FQDN as part of their hub endpoint configuration.	Compliant with standard RFC 1035
Version	This is the version number of the letterbox API. This version will only ever change if there is a substantial update in the way messages are processed by the TOTSCo hub. If a new version is introduced, the previous versions will remain in service for compatibility with existing processes.	n.n (e.g. 1.0)

The post API will use http POST as the method of request and the content type and encoding of the messages must be set to “Content-Type: text/plain; charset=UTF-8”.

### 1.4.2 The forward API Interface

The forward API takes a JSON message from an authorised destination and redelivers it to an alternative destination using an override routingID. The destination information is defined within the messages envelope contained within the JSON message, and the routingID will be defined within the URI.

For the forward API, the OAuth credentials will be matched against the sender ID to ensure the messages are not being spoofed.

The API URI format provided by the TOTSCo hub will conform to the following conventions.

**https://{fqdn}/letterbox/{version}/forward?redirect={routingID}**

The elements of the URI are as follows.

URI Element	Description	Format
FQDN	The Fully Qualified Domain Name of the provider of the letterbox API. The TOTSCo hub FQDN will be provided by TOTSCo. every consumer of the TOTSCo hub will need to provide their FQDN as part of their hub endpoint configuration.	Compliant with standard RFC 1035
Version	This is the version number of the letterbox API. This version will only ever change if there is a substantial update in the way messages are processed by the TOTSCo hub. If a new version is introduced, the previous versions will remain in service for compatibility with existing processes.	n.n (e.g. 1.0)
routingID	On the forward API, for use when a message recipient needs to redirect the message to another location for processing, the routingID identifies that new location	e.g. OTSPortal, GPLBPortal etc.

The forward API will use http POST as the method of request and the content type and encoding of the messages must be set to “Content-Type: text/plain; charset=UTF-8”.

The forward API contains a URI parameter to override the envelopes routingID.

The forward function is provided in situations where a message is delivered to a default location for a provider, but after processing it they identify they need to send it to another system, for example a portal, and therefore need to forward that message on. The message as originally received at the primary destination stays the same, there is no need to modify the message.

### 1.4.3 Envelope Elements

Every message sent through the hub REST API must have an envelope and a message body. The envelope contains the addressable information needed to identify and authenticate the messages originator, the “source”, and the intended recipient, the “destination”.

The envelope also contains a “routingID” which the hub will use to determine how and where to send the message. Every message specification will define how the routingID should be populated and what with. The hub will also use the routingID to determine the delivery policy for the message as well.

Finally there is an array element called “auditData” that should be used to provide information to TOTSCo for reporting to OFCOM and Industry.

```
{
  "envelope": {
    "source": {
      "type": "RCPID",
      "identity": "ABCD",
      "correlationID": "XYZ987"
    },
    "destination": {
      "type": "RCPID",
      "identity": "DCBA",
      "correlationID": "ABC123"
    }
  },
  "auditData": [
    {
      "type": "RCPID",
      "identity": "ABCD",
      "correlationID": "XYZ987"
    }
  ]
}
```



```

"routingID": "messageRoutingInformation",
"auditData": [{
  name: "auditFieldName": value: "auditFieldValue",
  name: "auditFieldName": value: "auditFieldValue"
}
],
"messageBody": {
}
}

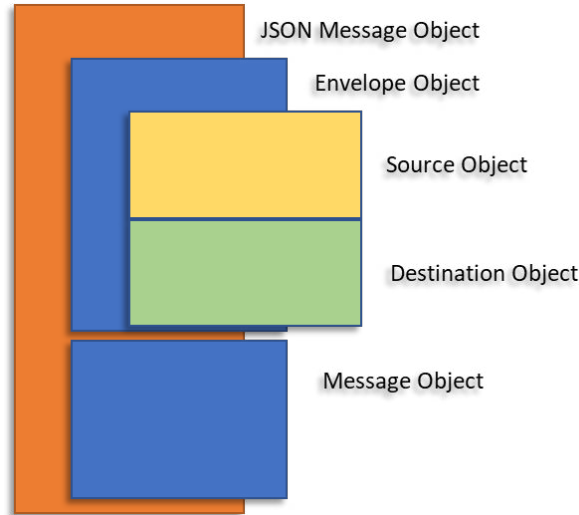
```

The example above shows a completed envelope, including audit information. The messageBody would be where the specific message content will be defined, that is not processed by the TOTSCo hub.

JSON element	Description	Format	Notes
envelope	A container defining the delivery information for any associated message.	Object	Required
source	A container defines the originator of the message and represents the return address for any message requiring a response.	Object	Required
destination	A container representing the destination of the message and used by the Post Office to identify the correct letter box to deliver it to.	Object	Required
source/type destination/type	The name of the directory list where the identity can be found and validated.	String	Required
source/identity destination/identity	The identity of the sending or receiving entity for the message as defined in the directory list selected.	String	Required
source/correlationID destination/correlationID	A string of characters that the message originator will recognise and allow matching of a reply to a request message.  In a source element, the correlationID must always be provided, the format can be anything the originator chooses to support their messaging process.  In a destination element, the correlationID would only be populated when the message is being sent in response to a message previously sent to you. In that case the correlationID will be the value that was sent by the original sender of the message so is being reflected back to them.	String	Required /Optional
routingID	The routingID that the hub will use to route the message to the recipients desired destination. Each messaging specification may have its own requirements for how this value is populated, but the value must be supported by the hub and published in the directory.	String	Required
auditData	A list of name value pairs that TOTSCo will use for auditing and reporting.	Array	Optional
auditData/name	The text name of the property being provided for auditing	String	Required
auditData/value	The value associated to the named entity above.	String	Required



<p><code>messageBody</code></p>	<p>This is the message to be sent to the recipient. The actual element name should be replaced based on the message being sent. Please refer to §3 for the messages supported for the process described in this document.</p>	<p>String</p>	<p>Required</p>
---------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------	-----------------



**Figure 1 – Post Office JSON Message Envelope Structure**

The container structure of a Post Office message is displayed above, the message object is separated from the envelope to allow changes in the content of either structure without affecting each other.

The reason every message must have a source correlation ID is that, as well as the recipient of the message being able to reply to you, in the event of a failure to deliver a message the post office can return a delivery failure notification as well, even if the message sent was a reply. So you can get failures to request messages as well as response messages.

#### 1.4.4 Message Formats

The industry hub API specification defines the envelope of the JSON message only, and the API for sending the messages. These are the only parts of the JSON message the hub will be responsible for understanding. With each envelope, there will be a message body, and this is the element that the recipient of the message must understand. Both parts together form the entire JSON message.

There may be hundreds of supported message bodies the hub will support, those defined in this document in section §3 relate only to the process described by this document.

Should any change be made to the API or envelope specification, that change will apply to every messaging process, so must be agnostic of those processes.

## 1.4.4.1 Letterbox Synchronous Replies

The letterbox API REST interfaces are synchronous, meaning that when a message is sent to the letterbox will reply within the same communication session. That reply does not contain a JSON message as its purpose is only to acknowledge receipt of the message being delivered to the letter box.

The letterbox APIs will acknowledge message receipt with a HTTP 202 response code, the definition of which is as follows: *“The request has been accepted for processing, but the processing has not been completed. The request might or might not be eventually acted upon, and may be disallowed when processing occurs.”*

If the message fails to be accepted by the API, various 400 errors may result subject to the OAuth processing of the API, or validation of the received message. For example, 401 – Unauthorised, or 403 – Forbidden for authentication errors, or in the event of a JSON format failure, error 400 – Bad request will be returned.

The synchronous acceptance of the message must extract the source element as a minimum to determine if the sender is valid and authorised to be sending messages into the hub. An invalid source will result in a 401 or 402 error.

## 1.4.4.2 Post Office Faults and Messages

In the event of the post office being unable to deliver a message to its intended recipient, or if status update messages are sent as a result of a delivery policy, the post office will create a message to the originator of the message in the following format.

```
{
  "envelope": {
    "source": {
      "type": "RCPID",
      "identity": "TOTSCO"
    },
    "destination": {
      "type": "RCPID",
      "identity": "DCBA",
      "correlationID": "ABC123"
    },
    "routingID": "messageRoutingInformation",
    "auditData": [{
      name: "originalDestinationType": value: "RCPID",
      name: "originalDestination": value: "XYZ3211",
      name: "originalRoutingID": value: "residentialSwitchMatchRequest"
    }
  ]
},
  "postOfficeMessage": {
    "code": "9005",
    "text": "Unable to deliver the message to the destination, no valid route.",
    "severity": "failure"
  }
}
```



The postOfficeMessage body describes a notification to the sender of the original message of a failure to deliver the message. The source information will represent the TOTSCO hub, and the audit data will contain the original intended message recipient and destination. The originators correlation ID will be returned in the destination information. Note that the source does not contain a correlationID, the postOfficeMessage cannot be replied to as it is a notification, and therefore no correlationID is required.

The content of this message then describes the notification information.

JSON element	Description	Format
postOfficeMessage	Container for messages from the post office	Object
code	A number that represents the nature of the fault and can be used by the message originator to determine remedial action.	Integer
text	A description of the associated response code	String
severity	An indicator of the nature of the message about the processing of the originators' message. Values will include, "information", "warning", "failure".	String

#### 1.4.4.3 Response Codes

The following table defines the list of response codes the post office will generate in the event of an error processing a message.

Code	Text	Severity
9000	Unknown or missing destination Type	Failure
9001	Unknown or missing destination ID	Failure
9002	Unknown or invalid source Type	Failure
9003	Unknown or invalid source ID	Failure
9004	Source type and ID not permitted from originating location	Failure
9005	Unable to deliver the message to the destination, no valid route.	Failure
9006	Unable to deliver the message to the destination, rejected, invalid message format	Failure
9007	Recipient rejected message	Failure
9008	Unable to deliver the message to the destination, timed out.	Failure
9009	The message has not been delivered to the destination but will be retried.	Information



Note, errors 9002 to 9004 cannot be routed back to the originator of the message as by their nature the source information in the message is determined to be invalid. However, based on the credentials of the sender of the message the hub should make attempts at establishing a valid source ID to route the failure message back to.

### 1.5 Directory API Specification

The TOTSCo hub maintains a central directory of all entities involved in the sending and receiving of messages using the TOTSCo hub. To be able to send message via the hub, all users need access to the directory to obtain the directory list.

The directory API takes two optional parameters to return the directory information for one specified party on the hub, a list type for all parties of that specific list type, or all parties documented on the hub.

For the directory API, the OAuth credentials will be used to verify access.

The API URI format provided by the TOTSCo hub will conform to the following convention.

**https://{fqdn}/letterbox/{version}/directory?list={listID}&identity={identityID}**

The elements of the URI are as follows.

URI Element	Description	Format/example
FQDN	The Fully Qualified Domain Name of the provider of the letterbox API. The TOTSCo hub FQDN will be provided by TOTSCo. every consumer of the TOTSCo hub will need to provide their FQDN as part of their hub endpoint configuration.	Compliant with standard RFC 1035
Version	This is the version number of the letterbox API. This version will only ever change if there is a substantial update in the way messages are processed by the TOTSCo hub. If a new version is introduced, the previous versions will remain in service for compatibility with existing processes.	n.n (e.g. 1.0)
listID	This optional value specifies the entity types to be included in the results. For example RCPID.	Text (e.g. RCPID)
identityID	This optional value specifies a specific identity to obtain the details for. If this value is specified, the listID MUST also be specified.	Text (e.g. BT001)

The directory API will use http POST as the method of request and the content type and encoding of the messages must be set to “Content-Type: text/plain; charset=UTF-8”.

The API will return a JSON document containing all of the public directory information to be shared with other users of the hub. A larger example of this JSON document is included in Appendix A.

```
{  
  "directory": {  
    "listType": "RCPID",
```



```

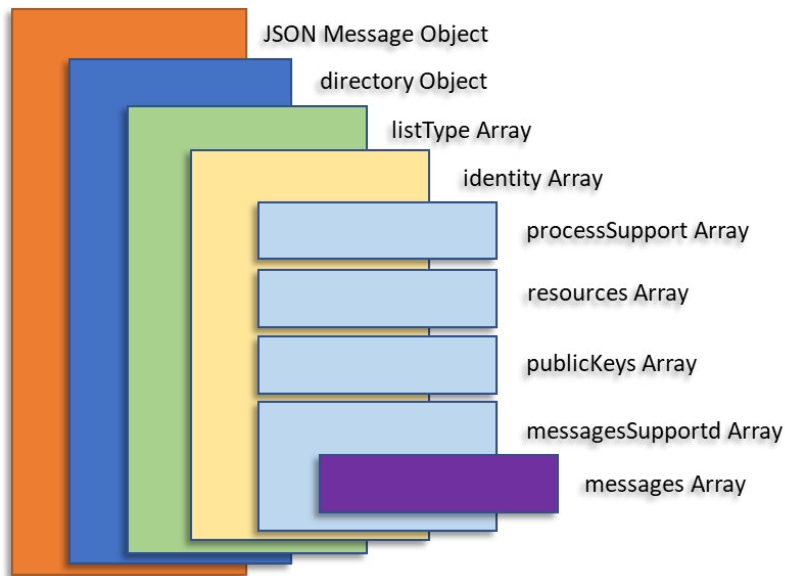
    "identityList": [{
      "id": "ID001",
      "processSupport": [{
        "process": "OTS",
        "status": "live"
      }],
      "resources": [{
        "name": "OTSAssistURL",
        "type": "URL",
        "value": "https://vm.com/ots"
      }],
      "publicKeys": [{
        "publicKeyID": "OTS1",
        "publicKey": "1654654effe654cf654f54c654b6a54c65e41a"
      }],
      "messagesSupported": [{
        "publicKeyID": "OTS1",
        "encryptionRequired": "Y",
        "messages": [
          "residentialSwitchMatchRequest",
          "residentialSwitchMatchConfirmation",
          "residentialSwitchMatchFailure"
        ]
      }],
    }],
  }
}

```

The elements of the JSON document will be as follows.

JSON element	Description	Format	Notes
directory	An array of the lists for the directory information	Object array	Required
directory/listType	The list type associated to the contained identities.	String	Required
directory/identityList	An array of all of the identity objects applicable to the list type	Object array	Required
identity/id	The value assigned to an entity for the purposes of messaging via the TOTSCo hub.	String	Required
Identity/ProcessSupport	An array of objects defining what industry processes this identity supports.	String	Optional
processSupport/process	The name of the industry process. Current supported values are "OTS" and "GPLB"	String	Required
processSupport/status	A value indicating the production status of this process. Current supported values are "live" and "test".	String	Required
identity/resources	A list of objects containing resources applicable to the identity.	Array	Optional
resources/name	The names of the resources will be a set of industry agreed values to represent a use or function. Samples include "OTSAssistURL", "residentialSwitchSupportPhone", "OTSTradingName", "OTSAssistURL" etc. A full list of the values and their uses will be described in a future update.	String	Required

resources/type	This value provides a type to represent the resource supplied. Many resources may only support a single type, others may support multiple. This value specifies how to interpret the value provided.	String	Required
resources/value	This is the value of the named resource.	String	Required
identity/publicKeys	This is an array of public key information related to the identity.	Object array	Optional
publicKeys/publicKeyID	An identifier for the public key	String	Required
publicKeys/publicKey	The public key, held in standard pem format.	String	Required
identity/messagesSupported	An object list of the messages supported by the identity.	Object array	Optional
messagesSupported/publicKeyID	The identity of the public key that should be used to encrypt the provided message formats. The key must exist in the list of public keys for this identity.	String	Optional
messagesSupported /encryptionRequired	An indicator, if the publicKeyID is supplied, of whether the recipient requires encryption, or if it is optional. Valid values are "Y" and "N".	String	Optional
messagesSupported/messages	A list of the message names supported and for which the selected encryption applies.	String Array	Required



**Figure 2 – Directory JSON Structure**

The container structure of the directory is displayed above. How many identities and lists you receive in your result depends on your selection criteria.

The recommendation is to use this service nightly, or at the most weekly, to request a full list of all the latest information.



If you have received a message from an unknown source ID, or that message contains a signing key that you do not know, then you would request that single identity from the directory to update your local cache.



### 3. Security Implementation

The API will be considered release suitable when it is published at version 1.0. The current API is lacking the final security elements that will be defined in conjunction with the solution provider, specifically the OAuth 2.0 implementation.

#### 1.5.1 API Authentication

This section to be update with input from the selected Vendor and in consultation to agree on the acceptable API authentication mechanisms, for example OAuth 2.0, that the hub will support.

### 4. Routing ID Summary

The TOTSCo hub will be capable of routing messages of any kind from many different industry processes. Below are a sample list of the current known and proposed routing IDs.

The TOTSCo hub will allow configuration of adding new routing IDs as and when required.

Industry Process / Function	routingID	routingID
Post Office	messageDeliveryFailure	
One Touch Switch	residentialSwitchMatchRequest	residentialSwitchOrderUpdateRequest
	residentialSwitchMatchConfirmation	residentialSwitchOrderUpdateConfirmation
	residentialSwitchMatchFailure	residentialSwitchOrderUpdateFailure
	residentialSwitchOrderRequest	residentialSwitchOrderTriggerRequest
	residentialSwitchOrderConfirmation	residentialSwitchOrderTriggerConfirmation
	residentialSwitchOrderFailure	residentialSwitchOrderTriggerFailure
		residentialSwitchOrderCancellationRequest
		residentialSwitchOrderCancellationConfirmation
		residentialSwitchOrderCancellationFailure
Business GPL Switching	businessSwitchMatchRequest	businessSwitchOrderUpdateRequest
	businessSwitchMatchConfirmation	businessSwitchOrderUpdateConfirmation
	businessSwitchMatchFailure	businessSwitchOrderUpdateFailure
	businessSwitchOrderRequest	businessSwitchOrderTriggerRequest





	businessSwitchOrderConfirmation	businessSwitchOrderTriggerConfirmation
	businessSwitchOrderFailure	businessSwitchOrderTriggerFailure
		businessSwitchOrderCancellationRequest
		businessSwitchOrderCancellationConfirmation
		businessSwitchOrderCancellationFailure

## Appendix A – Sample Directory Listing

```
{
  "directory": [{
    "listType": "RCPID",
    "identityList": [{
      "id": "TOTSCO",
      "publicKeys": [{
        "publicKeyID": "SIG1",
        "publicKey": "4cf654f54c654b6a51654654effe654c65e4fa"
      }
    ]
  }
],
  {
    "identity": "XM001",
    "processSupport": [{
      "process": "OTS",
      "status": "live"
    }
  ],
  "resources": [{
    "name": "OTSAssistURL",
    "type": "URL",
    "value": "https://xm.com/ots"
  },
  {
    "name": "residentialSwitchSupportPhone",
    "type": "TN",
    "value": "01234111111"
  },
  {
    "name": "OTSTradingName",
    "type": "businessName",
    "value": "Xenon Telecoms"
  },
  {
    "name": "OTSTradingName",
    "type": "businessName",
    "value": "Xenon Group"
  }
],
  "publicKeys": [{
    "publicKeyID": "OTS1",
    "publicKey": "1654654effe654cf654f54c654b6a54c65e4fa"
  },
  {
    "publicKeyID": "OTS2",
    "publicKey": "4cf654f54c654b6a54c65e4fa1654654effe65",
  },
  {
    "publicKeyID": "OTS3",
```

```

        "publicKey": "f54c654b6a54c65e4cf6544fa1654654effe65"
    },
    {
        "publicKeyID": "SIG1",
        "publicKey": "a16546544cf654f54c654b6a54c65e4feffe65",
    }
],
"messagesSupported": [{
    "publicKeyID": "OTS1",
    "encryptionRequired": "N",
    "messages": [
        "residentialSwitchMatchRequest",
        "residentialSwitchMatchConfirmation",
        "residentialSwitchMatchFailure"
    ]
},
{
    "publicKeyID": "OTS2",
    "encryptionRequired": "Y",
    "messages": [
        "businessSwitchMatchRequest",
        "businessSwitchMatchConfirmation",
        "businessSwitchMatchFailure"
    ]
},
{
    "publicKeyID": "OTS3",
    "encryptionRequired": "Y",
    "messages": [
        "businessBusinessRedirect"
    ]
}
]
},
{
    "id": "XM002",
    "processSupport": {
        "process": "GPLB",
        "status": "test"
    }
},
"resources": [
    {
        "name": "businessSwitchAssistURL",
        "type": "URL",
        "value": "https://xmb.com/gplb"
    },
    {
        "name": "businessSwitchSupportPhone",
        "type": "TN",
        "value": "01234111111"
    },
    {
        "name": "GPLBTradingName",
        "type": "businessName",
        "value": "Xenon Business"
    },
    {
        "name": "GPLBTradingName",
        "type": "businessName",
        "value": "Xenon Group (Business)"
    }
],
"publicKeys": [
    {
        "publicKeyID": "GPLB1",
        "publicKey": "f54c654b6a54c65e4fa14cf654654654effe65",
    }
]
}

```

```

    ],
    "messagesSupported": [{
      "publicKeyID": "GPLB1",
      "encryptionRequired": "Y",
      "messages": [
        "businessSwitchMatchRequest",
        "businessSwitchMatchConfirmation",
        "businessSwitchMatchFailure"
      ]
    }
  ]
},
{
  "id": "BT001",
  "processSupport": [{
    "process": "OTS",
    "status": "live"
  },
  {
    "process": "GPLSB",
    "status": "live"
  }
],
  "resources": [{
    "name": "OTSAssistURL",
    "type": "URL",
    "value": "https://bt.com/ots"
  },
  {
    "name": "residentialSwitchSupportPhone",
    "type": "TN",
    "value": "01444111111"
  },
  {
    "name": "businessSwitchAssistURL",
    "type": "URL",
    "value": "https://bt.com/ots"
  },
  {
    "name": "businessSwitchSupportPhone",
    "type": "TN",
    "value": "01444222222"
  },
  {
    "name": "OTSTradingName",
    "type": "businessName",
    "value": "BT"
  },
  {
    "name": "GPLBTradingName",
    "type": "businessName",
    "value": "BT"
  },
  {
    "name": "OTSTradingName",
    "type": "businessName",
    "value": "EE"
  },
  {
    "name": "OTSTradingName",
    "type": "businessName",
    "value": "Plusnet"
  }
],
  "publicKeys": [{
    "publicKeyID": "OTS1",

```

```

        "publicKey": "54cf654f54c1654654effe6654b6a54c65e4fa",
    },
    {
        "publicKeyID": "OTS2",
        "publicKey": "4c654b6a54c65e1654654effe654cf654f54fa",
    },
    {
        "publicKeyID": "OTS3",
        "publicKey": "65e4fa14cf6546f54c654b6a54c54654effe65",
    }
],
"messagesSupported": [{
    "publicKeyID": "OTS1",
    "encryptionRequired": "Y",
    "messages": [
        "residentialSwitchMatchRequest",
        "residentialSwitchMatchConfirmation",
        "residentialSwitchMatchFailure"
    ]
},
{
    "publicKeyID": "OTS2",
    "encryptionRequired": "Y",
    "messages": [
        "residentialSwitchMatchRequest",
        "residentialSwitchMatchConfirmation",
        "residentialSwitchMatchFailure"
    ]
},
{
    "publicKeyID": "OTS3",
    "encryptionRequired": "Y",
    "messages": [
        "businessSwitchMatchRequest",
        "businessSwitchMatchConfirmation",
        "businessSwitchMatchFailure"
    ]
}
]
},
{
    "id": "TC001",
    "processSupport": [{
        "process": "OTS",
        "status": "live"
    }
],
"resources": [{
    "name": "OTSAssistURL",
    "type": "URL",
    "value": "https://telco.com/ots"
},
{
    "name": "residentialSwitchingSupportPhone",
    "type": "TN",
    "value": "01333111111"
},
{
    "name": "OTSTradingName",
    "type": "businessName",
    "value": "Telco"
}
],
"publicKeys": [{
    "publicKeyID": "OTS1",
    "publicKey": "654effe654cf6515464f54c654b6a54c65e4fa",

```

```

    }
  ],
  "messagesSupported": [{
    "publicKeyID": "OTS1",
    "encryptionRequired": "Y",
    "messages": [
      "residentialSwitchMatchRequest",
      "residentialSwitchMatchConfirmation",
      "residentialSwitchMatchFailure"
    ]
  }
  ],
  {
    "publicKeyID": "OTS2",
    "publicKey": "f654f54c654b6a54c651654654effe654ce4fa",
    "routingID": "SKYEndPoint2",
    "encryptionRequired": "Y",
    "messages": [
      "residentialSwitchMatchRequest",
      "residentialSwitchMatchConfirmation",
      "residentialSwitchMatchFailure"
    ]
  }
  ],
  {
    "id": "FR001",
    "processSupport": [{
      "process": "OTS",
      "status": "live"
    }
  ]
  },
  "resources": [
    {
      "name": "residentialSwitchingSupportPhone",
      "type": "TN",
      "value": "0145611111"
    },
    {
      "name": "OTSTradingName",
      "type": "businessName",
      "value": "Freds Communications"
    }
  ]
  },
  "messagesSupported": [{
    "messages": [
      "residentialSwitchMatchRequest",
      "residentialSwitchMatchConfirmation",
      "residentialSwitchMatchFailure"
    ]
  },
  {
    "messages": [
      "residentialSwitchMatchRequest",
      "residentialSwitchMatchConfirmation",
      "residentialSwitchMatchFailure"
    ]
  }
  ],
  {
    "listType": "CUPID",
    "identityList": [{
      "id": "001",
      "processSupport": [{
        "process": "NumberPorting",
        "status": "live"
      }
    ]
  }
  ],

```



```
"resources": [{
  "name": "portingSupportURL",
  "type": "URL",
  "value": "https://xm.com/porting"
},
{
  "name": "portingSupportDeskPhone",
  "type": "TN",
  "value": "0123433333"
}
],
"publicKeys": [{
  "publicKeyID": "OTS1",
  "publicKey": "54c654b6a54c651654654effe654cf654fe4fa",
}
],
"messagesSupported": [{
  "publicKeyID": "PORT1",
  "encryptionRequired": "Y",
  "messages": [
    "numberPortOrder",
    "numberPortOrderConfirmation",
    "numberPortOrderFailure",
    "expressPortQuery",
    "expressPortOrder",
    "expressPortOrderConfirmation",
    "expressPortOrderFailure"
  ]
}
]
}
]
}
}
}
```

**End of Document**