Higate Concepts

2015

V1.1

Table of Contents

1. Introduction

This document describes the common Higate concepts necessary for an understanding of the available API's.

2. Overview

The Higate system is a mobile services aggregation platform that provides client applications access to the various mobile services (bearers) offered by the local Mobile Network Operators of a country or region.

Higate clients (Service Providers) have a range of Higate API's to choose from, each of which offers the client access to a range of bearers available from the various Mobile Operators. Thus, the client is only required to implement a single (Higate) API rather than a multitude of API's, one for each bearer and Mobile Operator.

3. General Higate Concepts

3.1 Accounts

Higate allows for the creation of 'Higate Accounts', and under each account there may be one or more associated client 'Logins' (binds). A Login may also be referred to as a 'Client' – this arises from the convention of referring to external applications as 'clients' in the 'client-server' paradigm.

3.2 Service Codes

Higate Service Codes are unique, system-wide alphanumeric strings of up to 8 characters that allow for a finer level of control when configuring Higate Logins. They are in effect sub-logins, and all transaction routing rules are keyed on these names. Whenever a Login transacts with the Higate, there is an associated Service code, which may be explicitly defined or defaulted by the Higate (if it is not specified by the client application).

3.3 Higate Credits

Higate Credits are the basic unit of currency for all transactions on the Higate system, and client applications are unable to transact with the Higate unless they have sufficient credits to cover the cost of the requested transaction. The cost of a transaction (e.g. sending an SMS or MMS) is customized per Account, Bearer, and destination Mobile Network Operator.

The rate of exchange between Higate Credits and the local currency is fixed at site installation and remains constant for the life of the system. The exchange rate for the South African site was set as 1 Higate Credit = 1 Cent (Rands).

3.3.1 Credit Management

Credits are purchased from the Accounts Department of the relevant Higate site and are allocated to the customer's Higate Account. Note, however, that these credits are unusable until they are transferred down to the Login level.

Account credits are transferred to the Login level either manually, or automatically my means of a mechanism known as '**Automatic Drawdown**'.

3.3.2 Drawdown

Client Logins may be configured to allow automatic credit Drawdowns, for a fixed number of credits, whenever the Login runs out of credit. For such a Login, the Higate will automatically attempt to drawdown the configured amount, and if the full amount is not available at the Account level, it will drawdown what is available. Note that credits that have already been transferred to other Logins under this Account are unaffected by this process.

3.3.3 Credit Alarms

A Higate Account may be configured to automatically notify the relevant client authority via SMS and/or email when a pre-configured credit '**credit alarm level**' is reached.

Some of the Higate API's also provide real-time feedback on the Login's current credit availability.

3.3.4 Credit Limits

Under very rare circumstances, Higate Accounts may be allocated a credit limit, which will allow the Account to go into overdraft (negative).

3.3.5 Credits and Multiple Segment SMS's

Generally the maximum length of a single segment SMS is 160 7-bit characters or 140 8-bit characters (either way that is a total of 1120 bits), but it is nevertheless possible to send much longer messages – for example a message of 400 7-bit (ASCII) characters.

Even though a 400 character SMS constitutes a single Higate transaction, it is actually sent in three segments, and is priced accordingly in terms of Credits consumed. The reason for this is that the Mobile Network Operator will in effect receive three distinct SMSs, one for each segment.

3.4 Routing

One of the primary tasks of the Higate system is to ensure that Login MT transactions are correctly routed to their destination gates, and that MO transactions are correctly routed from the Operator gateway (gate) back to the correct Login. (See Section 4 for a brief description of the system nodes).

The main elements of transaction routing are as follows:

- MSISDN normalization (See Section 3.5)
- Service Codes (See Section 3.2)
- Mobile Network Operator prefixes
- Mobile Number Portability (MNP) (See Section 3.6)
- Source Numbers (See Section 3.7.1)
- Source Number Tagging (SMS, MMS, etc) (See Section 3.7)
- Keyword Routing (for MO traffic only) (See Section 3.8)
- Routing Rules

For an MT transaction (e.g. SMS) to be correctly routed, the following items must be uniquely identified:

- TOC (Bearer) (See Section 5)
- Normalized destination Number (MSISDN) (See Section 3.7)
- Source Number (MSISDN) (See Section 3.9)
- Source Number Tag (See Section 3.9.1)
- Higate Service Code (See Section 3.2)
- Destination Mobile Network Operator (NetworkID)
- Destination Gateway / Interface (GateID)

The following additional information must also be determined.

- Number of segments (SMS only)
- Credits Consumed

3.4.1 Routing Sheet

Example Routing Configuration Sheet

```
Login Details
-------------
ID                : 1234
Name              : USERNAME
Password          : PASSWORD
Created           : 05/03/17 12:49:21
Status            : Enabled
API Type          : HTTP
Default Service   : SERVICECODE
Default URL       : http://www.yourdomain.co.za/higate.php
Auto-drawdown     : Yes
Drawdown Amount   : 26000 Credits
Final Status Only : False

Service Codes  Vod Service Code  URL
-------------  ----------------  ---
SERVICECODE          INT00123        http://www.yourdomain.co.za/higate.php


Transmit (MT) routings by service code
--------------------------------------
C3T01
  SMS
    Rule: By Network
      MTN
        GateID: 272
        Source Address: 27839300365     PUBLIC   Tags ENABLED  1
    Rule: By Network
      CellC
        GateID: 92
        Source Address: 27840004683     PUBLIC   Tags ENABLED  1
Rule: By Network
      Vodacom
        GateID: 209
        Source Address: 27820048062     PUBLIC   Tags ENABLED  1
  USS
    Rule: Default
        GateID: 299
  OBS
    Rule: By Network
      CellC
        GateID: 203
    Rule: By Network
      Vodacom
        GateID: 211
Rule: By Network
      MTN
        GateID: 254
  VSR
    Rule: Default
        GateID: 316

Receiver (MO) SMS routings by service code
------------------------------------------
SERVICECODE
54321     by Default Number
```

Each account login has a routing sheet (see Listing 1) associated with it which contains all the necessary information to transact with the Higate system. The configuration in the example listing is used for all examples in this document.

It is very important to review this information to ensure proper operation of the services. The routing sheet is available on request from your account manager.

3.4.2 Login Details

This section contains the login details:

Parameter	Description
Name	The login name, to be used as the value for the "UserID" parameter.
Password	The password for the login, to be used as the value for the "Password" parameter.
Status	The status of the login. This should be "Enabled" to be able to transact.
API Type	The type of interface for the login. Should be set to HTTP to use the HTTP API described in this document.
Default Service	The default service code to use should a service code not be specified in the transaction.
Default URL	The default call back URL should a specific URL not be specified for the service.

3.4.3 Service Codes

Each login can have a number of service codes each having specific routing information.

A login would typically use more than one service code if different routes are required for different services, such as a bulk SMS service and a priority SMS service. If more than one service code is linked to an account that service code needs to be specified as the value of the Ticket.Service attribute, unless the service code is configured as the default service in which case this is optional.

Note: The "Vod Service Code" is included for historical reasons but can be ignored as its meaning is transparent for the user.

3.4.4 Transmit Routing

This section shows the routing configured for the different bearers enabled on your account. The available types are SMS, OBS, USSD and VSR (voucher). For a login that has only SMS configured only SMS routing would be displayed.

A section is included for each of the services associated with the login.

3.4.5 Receiver Routing

This sections shows routing configured for MO messages and lists the MSISDNs and short codes on which MO traffic is received. Note that MO traffic for the numbers configured for SMS MT routing are received by default.

3.5 Flags

The Higate system makes use of flags to define particular properties of transaction that cannot be achieved through the standard parameters and attributes.

Note: Number prefixed with "0x" denotes hexadecimal numbers (base 16)

Flag Name	Value	Description
HIGATE_FLAG_SMS_POPUP	0x00000001	Send SMS as a popup message if supported by operator
HIGATE_FLAG_SMS_FLASH	0x00000002	Send SMS as a flashing popup message if supported by operator
HIGATE_FLAG_SMS_8BIT	0x00000004	SMS contains 8 bit (binary) data [note1]
HIGATE_FLAG_SMS_UDH	0x00000008	SMS data includes a User Data Header (UDH)
HIGATE_FLAG_SMS_SEGMENT	0x00000010	SMS is a segment of a larger message
HIGATE_FLAG_SMS_DN_MASK	0x001F0000	SMS Delivery Receipt Mask
HIGATE_FLAG_SMS_DN_FINAL	0x00010000	SMS Delivery Receipt requested where final delivery outcome is success or failure
HIGATE_FLAG_SMS_DN_FAILED	0x00020000	SMS Delivery Receipt requested where final delivery outcome is failure
HIGATE_FLAG_SMS_DN_INTERM	0x00100000	SMS Delivery Receipt requested for Intermediate notifications
HIGATE_FLAG_OBS_LINKED	0x00000020	SMS pending OBS authentication
HIGATE_FLAG_OBS_TKT_ADDR	0x00000040	Charge the sms to the ticket charge address (OBS-linked only)
HIGATE_FLAG_OBS_AUTO_CONFIRM	0x00000200	Auto-confirm once authorized

HIGATE_FLAG_OBS_SUBSCRIBE_ONLY	0x00000400	OBS transaction is a subscription only transaction
HIGATE_FLAG_HEX_ENCODED	0x00000080	The source string is hex encoded text (eg "060BFC')
HIGATE_FLAG_MT_BILLED	0x00000100	MT Billed content
HIGATE_FLAG_USS_EXIT	0x00000001	Terminate this USSD Session

Note1: Although binary SMS is supported by all networks some networks forbid the use of WAP Push messages.

Multiple flags can be combined by bitwise OR (adding) the respective flags.

3.6 Configurable Login Settings

For each login on a user account a certain number of parameters can be configured that affects the behaviour of the system. The following options are available:

Option Name	Description
COPT_FINAL_STATUS_MSG (Default: Disabled)	Enabling this option will force the interface to only send status notifications when the transaction has reached a final state (i.e. cannot change again). Intermediate status notifications are not sent.
COPT_NO_UCS2_TO_ASCII (Default: Disabled)	SMS MO Message received with UCS2 encoding that contains only the first 256 characters as defined for ISO/IEC-8859-1 (Latin-1) are converted and transmitted as TEXT and the unused top UCS2 byte is discarded. This feature can be disabled by setting this option to 'enabled'.
COPT_NO_QUEUED_STATUS (Default: Disabled)	Enabling this option will force the interface to all send status change notifications but not the "QUEUED" status, which is implied automatically when successfully submitting a transaction.
COPT_HTTP_HOST_NO_PORT (Default: Disabled)	Enabling this option will prevent the destination port number to be included in the header.
COPT_RX_NORM_ERR (Default: Enabled)	This option forces the system to use a set of generic error codes that remaps system and operating specific errors to a set of generic codes. This option is enabled by default but can be disabled for existing clients on new logins to ensure compatibility with existing systems.

3.7 MSISDN Normalization

Client applications frequently submit transactions (e.g. MT SMS, OBS etc) that specify a destination number that are either not fully 'qualified' or contain imbedded non-numeric characters. Examples include the following…

- 083 652 1009
- +27 (83) 652 1009
- (+27) 83, 6521009
- Mobile: 083 652 1009
- 123.Garbage.Test

These must all be normalized into the form '27836521009', whilst also identifying the correct 'International Direct Dialling Code' (27) and the correct Mobile Network Operator Prefix (83).

To avoid difficulties, it is advisable to format the numbers as fully qualified MSISDNs.

3.8 Mobile Number Portability (MNP)

In some regions - such as South Africa - subscribers may 'port' from one Mobile Operator to another without losing their mobile number. This is certainly an advantage to the subscriber, but places a significant technical burden on any Mobile Content service, because it can no longer rely on using the Mobile Network Operator Prefix to identify the destination operator, and hence the correct destination interface. This means that every normalized number must also be checked against a 'Number Portability Database' to ensure that the correct Mobile Operator (and interface) has been identified.

The Number Portability Database only contains numbers that have been ported away from their default Operator, and is maintained by the Higate system based on daily updates from the 'Mobile Number Portability Company' - an independent body that was specifically set up to manage portability issues.

3.9 Source Numbers

When an SMS is sent to a phone, Higate is required to specify a Source Number (MSISDN) when submitting the transaction to a Mobile Network Operator interface (usually an SMPP interface). They appear on the destination phone as the number from which the message was sent.

These numbers are not arbitrary - they are allocated to Higate by the Mobile Network Operators for use from specific SMPP binds to the operator.

One of the functions of Higate routing is to identify which Source Number is applicable to a particular transaction (SMS) and destination Mobile Network Operator (gate).

Source numbers are important because they determine the destination number to which reply messages are sent when a subscriber answers a received SMS. Reply messages find their way back to the Higate, which must then route the (MO) reply back to the original Higate Client (Login). It does this based on the 'ownership' of that source number. This implies that in order for a client Login to receive any MO messages, it must have ownership of at least one MSISDN source number per Mobile Network Operator. The difficulty with this arrangement is that not only are Source Numbers a limited resource, they must also be obtained from the operators – an administrative process that is notoriously slow and cumbersome. The solution is to make use of 'Source Number Tagging' on selected so-called 'Public' source numbers.

3.9.1 Tagging

If a Source Number has been configured by the operator, it is possible to add trailing 'Tag digits' to the end of the number without affecting its behaviour. The advantage of this is that the tag digits remain a part of the source number all the way to the destination device (phone), and back to the Higate when the subscriber replies. This effectively allows the Higate to create many thousands of additional source numbers from the single operator-allocated base Source number.

For example, a Mobile Network Operator has allocated Higate the Source number +27827020303, and up to 5 additional tagging digits have been enabled on this number. The Higate has a large number of client Logins that require MT and MO SMS capability to that operator. In this case, each client Login may be allocated one or more Tag numbers on this source number:

Login Name	Allocated Tags
Client0	00000, 00002, 00003,…00009
Client1	10000, 10002, 10003… 10009
Client2	20000, 20002, 20003… 20009
…	
Etc.	

If Client1 sends an SMS, Higate would submit the SMS to the Mobile Operator using the extended source number +2782702030310000. If Client2 were to send an SMS, the extended source number would be +2782702030320000, etc.

When the subscriber replies to the message from Client1, they are in effect sending to the number +2782702030310000, and so when the message arrives at the Higate, it is able to identify the intended service (in this case Client1) by examining the additional tag digits (in this case **10000**).

Thus, all Higate Logins that require SMS (or MMS) functionality are allocated one or more Tags per source number. In almost all cases, the Login will receive the same set of Tags for all its relevant source numbers, but this is sometimes not possible.

Note that when a Login submits an MT SMS transaction it is not required to specify a Tag value (it will be defaulted to the first allocated Tag), but if one is specified then it must be a Tag that it 'owns'. If it attempts to specify a Tag that it doesn't own, the transaction will fail.

Note also that the following Tags are all different **0, 00, 001, 1, 100**.

3.9.2 Private Source Numbers

Unlike the 'Public' Source Numbers described above, the Higate will allow a Login to be allocated exclusive access to a Source Number. This implies that the Login also has ownership of the full Tag range.

3.9.3 Short Codes

A Short Code is much like any other MSISDN, but for the fact that it has fewer digits, and the number is defined on all the Mobile Network Operators for a given region.

Examples include 32900, 2727, etc. The number of digits in a Short Code is usually fixed in a given region.

Short Codes were introduced as a convenience – it's easier to remember a 5 digit number like 32900 that an 11 digit number like 27829012345 – but they often also have an associated 'Premium Value', which means that sending a message to such a number will often cost the subscriber considerably more than the cost of sending a standard SMS.

3.9.4 Spoofing

Spoofing occurs when the Mobile Network Operator allows any Source Number to be specified, including text, such as the word 'MAGPY'. For obvious reasons, very few operators allow Spoofing.

3.10 Keyword Routing (MO Only)

Some numbers cannot be 'Tagged'. This usually only applies to certain Source Numbers and Short Codes (although in some regions even Short Codes may be tagged).

For such numbers it is necessary to employ another technique to successfully route an MO SMS to its intended Login, and in this instance, it requires the cooperation of the subscriber who originated the (MO) message. In this case, the subscriber is required to prefix his/her with a **Keyword** that uniquely identifies the intended Login on that number. This means that Keywords need only be unique per destination number.

For example, suppose that the following keywords were defined on the number 27827020303:

Keyword	Client Login
RUGBY	SportChat
ROCK	MyMusic
SUP*	Comfort

Then the following messages received on this number would be routed as shown:

Message	Client Login
Rugby Go Sharks. What a game!	SportChat
Rugby	SportChat
ROCK Did you see the show last night?	MyMusic
ROCKY how are you?	<none> No match
SuPeR day today	Comfort
SUP Give me some love	Comfort
Supper was very good tonight.	Comfort
ROCKnROLL will lives for ever	<none> No match
SUPPOSE	Comfort

Note:

- Keywords are not case sensitive
- Message keywords must be followed by a blank (or nothing) in the message in order for the message to be correctly routed.
- A keyword that ends in '*' will match any message that starts with the preceding characters (in this case SUP)

3.11 Higate Queues

The Higate manages two important queues – the Transmit (TxQueue) and Receive (RxQueue) queues for MT and MO traffic respectively. Transactions generated by a client application enter the TxQueue and are transferred to the Mobile Network Operators, whilst transactions received from the Mobile Network Operators enter the RxQueue and are passed back to the client application (Login).

3.12 Sequence Numbers

The transactions of both *TxQueue* and *RxQueue* are separately indexed unique 64-bit unsigned numeric 'Sequence Numbers' – *TxSeqNo* for *TxQueue* and *RxSeqNo* for *RxQueue*

In addition, the client application may (and in some cases must) provide a unique 64-bit unsigned Reference number (*RefNo*), when submitting transactions to the Higate. Reference numbers are particularly important for OBS-linked transactions, and the developer is encouraged to make sure these numbers are unique where possible. In fact, OBS-linked transactions require Reference numbers to be unique over a period of at least 7 days.

Whenever the Higate notifies the client Login of an MT transaction status change or an MO transaction, the appropriate Sequence Number will be included, together with the Login-defined '*RefNo*' (where applicable), in a 'Result' message.

Sequence Numbers are particularly important for the Http interface where they must be used to identify duplicate transactions. This is explained in more detail in Section 4.3.3.

3.13 Transaction Statuses

All Higate transactions have an associated 'Transaction Status', which represents the state of the transaction as it passes through the system. Higate client applications (Logins) are notified in real-time of status changes, but although some 'Interim' statuses (statuses that still have the potential to change) are sent, client applications are only guaranteed to receive 'Terminal' / final statuses.

3.13.1 Result Messages – MT Status Notifications

The Higate APIs return MT Status changes in the form of 'Result' messages, which contain the following parameters:

- Higate-allocated (unique system wide) Sequence Number (TxSeqNo)
- Client-allocated Reference Number (must be unique within a 7 day period)
- TOC (bearer)
- MSISDN
- Local Network (1=Vodacom, 2=MTN, 3=CellC)
- International Mobile Network Code (MCC+MNC)
- Adjusted Currency Value (for MTB, OBS, or OBS-Linked transactions)

- Result Code (includes the transaction status) (See Section 3.13)
- Sub Code (32-bit unsigned number - Higate Error Code) (See Section 3.14)
- Result Text (a plain text description of the error – Higate Error Text)

Status	Final / Interim	Description
QUEUED	Interim	Received from the client
SUBMITTED	Interim	Submitted to gateway (network operator)
ACKNOWLEDGED	Interim	Acknowledged by gateway or authorized (OBS)
RECEIPTED	Terminal	Successfully delivered or Confirmed (OBS)
CANCELLED	Terminal	Cancelled (OBS only)
EXPIRED	Terminal	Validity period has expired
PENDING	Interim	Pending authentication (e.g. LBS)
DENIED	Terminal	Authorization denied
FAILED	Terminal	Generic Failure
ERROR	Terminal	Error condition

3.13.2 TxQueue Statuses

The complete list of transmit queue status values are as follows:

3.13.3 RxQueues Statuses

The complete list of transmit queue status values are as follows:

Status	Final / Interim	Description
QUEUED	Interim	Received from the gateway
DISPATCHED	Terminal	Dispatched to the client application
TRASHED	Terminal	Trashed / discarded
FAILED	Terminal	Error condition

3.14 Common Transaction Scenarios

In order to understand the status sequence, it will be useful to consider various common scenarios. In the examples described below, we will only consider an SMS transaction, but these cases are equally valid for other bearers such as MMS and LBS.

3.14.1 Standard non-billed SMS

The status sequence for a typical (free) MT SMS is as follows:

- If the client application correctly specifies all the necessary parameters then the MT SMS transaction enters the TxQueue with a status of QUEUED. Otherwise, the transaction will appear in the TxQueue with a final status of ERROR and will not be processed any further.
- QUEUED messages are read from the TxQueue, submitted to the operator (with the Delivery Notification flag set) and the status changed to SUBMITTED.
- Once the operator acknowledges receipt of the message, the status is changed to ACKNOWLEDGED. This usually happens almost immediately, however, very occasionally, the operator fails to acknowledge, and in this case, the Higate has no choice but to move on. The Higate will not resubmit the SMS because the local operators do not correctly support retries, and any attempted retry is simply treated as a new message.
- Upon receipt of a delivery notification from the operator the transaction status is changed to
    - RECEIPTED if the message was correctly received by the subscriber.
    - EXPIRED if the message was not delivered within the specified validity period.
    - FAILED if the message was not successfully delivered to the subscriber.

- Occasionally, the operator fails to return a Delivery Notification, in which case the transaction status remains ACKNOWLEDGED.

3.14.2 MT-billed Transactions

On Higate, MT-billed transactions apply to most bearers (SMS, MMS, LBS, WIG, etc.) and denote transactions for which there is a billing component. This means that before the transaction may be processed (sent) a billing request must be successfully completed for the destination MSISDN.

Consider the case of a typical MT-billed SMS:

- When the client application submits such a transaction, the Higate system actually creates two distinct transactions (each with its own unique Higate Sequence Number):

    1. An OBS (Online Billing Services) transaction that is immediately posted to the *TxQueue* with an initial transaction status of QUEUED.
    2. A standard (free) SMS transaction that is contained in a temporary holding queue (not the *TxQueue*) for processing later.
- Both transactions are allocated the same Reference Number (as defined by the client application). It is for this reason that client applications must allocate unique reference numbers to MT-billed transactions.
- The OBS transaction is submitted to the relevant operator for processing and the transaction status is changed to SUBMITTED.
- If the OBS transaction is not authorized (status FAILED) then the SMS contained in the temporary holding queue is simply deleted, and only the FAILED OBS transaction will remain.
- Otherwise, if the operator authorizes the OBS transaction, then there are two possibilities – depending on the operator.

    - For both MTN and Cell C, the authorized OBS transaction is effectively finalized with ‘final’ status RECEIPTED. In this case the SMS in the holding queue is transferred to the *TxQueue* where it is processed in exactly the same way as a non-bill SMS (described earlier).
    - For Vodacom the situation is a bit more complicated. In this case:

- The OBS transaction is first changed to the ‘interim’ status AUTHORIZED (not RECEIPTED).
- The SMS is transferred to the *TxQueue* as described above
- Once the SMS is successfully delivered (status RECEIPTED), the OBS transaction is again submitted to the operator for ‘Confirmation’. Only once this step is successfully completed will the OBS transaction status also be changed to RECEIPTED and processing ends.
- However if the SMS is not successfully delivered (status FAILED or EXPIRED), then Vodacom requires that the OBS then be ‘Cancelled’, and it’s status changes to the ‘final’ status of CANCELLED.

It is important to understand that the client application will receive status notifications for both transactions (SMS and OBS). It is a simple matter to correctly identify the transactions as, although they have different Higate Sequence numbers and TOC’s (Type of Content – bearer), they will share a common client-specified Reference Number.

Because of the association with OBS transactions, MT-billed SMS transactions are sometimes called “OBS-linked” SMSs.

3.15 Result Codes

Due to the historical evolution of the Higate system, the Result Code parameter combines both the status of a transaction (See Section 3.11) as well as an error code.

- Numbers in the range 1 to 11 inclusive denote the status of the transaction.
- Numbers from 128 and greater denote transactions that are in an ERROR status while the value also provides the reason for the error condition.

Name	Value	Description
SUCCESS	0	Success
QUEUED	1	Content queued for submission to the gateway
SUBMITTED	2	Content was submitted to the gateway
ACKNOWLEDGED	3	Content was confirmed as received by operator
RECEIPTED	4	Successful Delivery Receipt was received
EXPIRED	5	Expiry Delivery Receipt was received
FAILED	6	Transaction failed
DENIED	7	Authorization Denied by subscriber
PENDING	9	Pending authorization by subscriber
CANCELLED	11	Cancelled
REQUEST_DENIED	128	A requested action/method was denied
BAD_ADDRESS	129	Invalid destination address/MSISDN was specified
BAD_FORMAT	130	Badly formatted content
INSUFFICIENT_FUNDS	131	Deprecated
WML_PARSE_ERR	132	Deprecated
CREDIT_LIMIT	133	The account credit limit exceeded
FAILED_LINKED_OBS	134	Associated OBS transaction failed
NOT_SUPPORTED	135	Requested operation not supported
INVALID_PARAM	136	Invalid Parameter
BAD_MESSAGE_LEN	137	Message length was zero or too long
ROUTE_ERR	138	Unresolved routing information for specified prefix
INVALID_TAG	139	Attempt to use invalid address tag
INVALID_SERVICE	140	Attempt to use invalid service code
UNDEFINED_ROUTE	141	Undefined gate route
INVALID_ORIENTATION	142	Deprecated
DEQUEUED	143	Deprecated
GENERIC	199	Catch all error
UNKNOWN	255	Unknown error condition

Note that Result Code 133 (CREDIT_LIMIT) is a special case. When this Result Code is returned, the Higate system does NOT record the transaction to the database. The transaction is simply discarded.

3.16 Sub Codes

Sub codes must be interpreted within the context of the relevant TOC (bearer) and network operator.

Note that this value actually represents error codes from both the Higate system and the network operators, and does not necessarily denote an error.

Errors are identifiable by the fact that the high order bit (0x80000000) is set. If this error code comes from the network operator then the next highest bit (0x40000000) is also set.

For example, consider a failed OBS transaction to Vodacom with a SubCode value of 0xC0000258. The fact that both of the two high order two bits are set (0x80000000 + 0x40000000 = 0xC0000000) means that the remainder of the number (0x00000258) represents the Vodacom-specific error code.

3.17 Adult Rating Codes

These codes define the level of explicit sexual and or violent detail in the delivered content. The values in the table below represent the Higate scale, and these are converted to the operator specific values upon delivery. Consequently Higate client applications should use the Higate scale when submitting content to Higate.

The relationship between the Higate value and the respective operators are tabled below – where applicable.

Note that currently, Adult Rating Codes are only applicable to transactions that have an OBS component on the Vodacom network.

Higate Name	Code	Description
ADULT_NONE	0	All Ages
ADULT_G	2	General Audience
ADULT_PG	4	Parental Guidance Suggested
ADULT_PG13	6	Parents Strongly Cautioned
ADULT_R	8	Restricted-Under 17 requires accompanying parent or adult guardian
ADULT_NC17	10	No One 17 and Under
ADULT_X	13	Erotica
ADULT_XX	14	Explicit Pornography
ADULT_XXX	15	Hard Core Pornography

3.17.1 Vodacom Rating Values

Code	Description	Higate Equivalents
0	All Ages	0 to 6 inclusive
2	Age Classified	8 to 10 inclusive
4	Age Restricted	13 and above

3.18 Archiving

The Higate performs a daily archive of transactions in both the Transmit and Receive queues. This usually runs daily at 01:30 (local time) and transfers ‘old’ transactions into a remote ‘archive’ (or ‘History’) database, where they remain for a period of at least 5 year (often longer).

All transactions that have achieved ‘Terminal Status’ (see Section 3.11) are archived after 3 days, or alternatively after 7 days regardless of their status.

3.19 Statistics

Higate statistics are computed hourly, and form the basis of the computed account revenue figures.

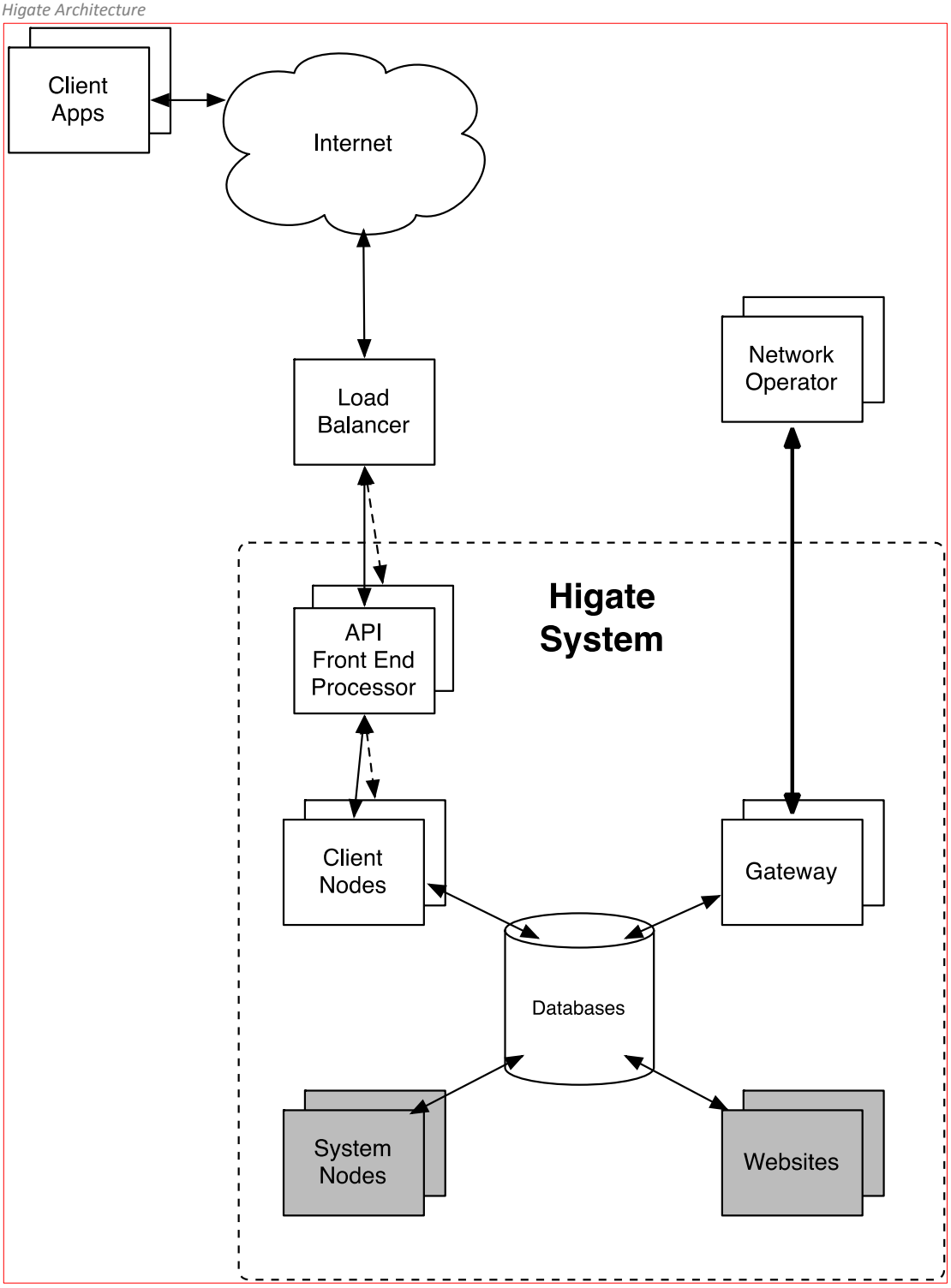
In some (rare) cases, the revenue earned is based on the total volume over a full calendar month, and in this case, the (hourly) revenue calculation may appear to be very low towards the beginning of the month. This is because assumed total monthly volume only includes number of transactions in the month so far.

4. Higate Architecture

Higate is implemented as a collection of cooperative 'Higate Nodes' (processes) that communicate with each other across multiple machine boundaries. This architecture provides both scalability and redundancy.

The following figure illustrates the basic structure.

Higate Architecture



4.1 Nodes

The Higate nodes may be broadly classified as follows:

- **Client Nodes** - These nodes manage communication between client applications and the Higate system. They are dedicated to the following functions:

    1. Login authentication
    2. Sending and receiving client transactions
    3. Management of available credit
    4. Transaction Routing

- **Gateway Nodes** - These nodes manage communication between the Higate system and the various local Mobile Network Operators. Each gateway node is dedicated to a specific bearer and Mobile Network Operator, and there may be multiple gateways configured even for a single bearer and Network Operator. These nodes are dedicated to the following functions:

    1. Operator Connectivity
    2. Sending and receiving of bearer transactions
    3. Routing of received transactions

- **System Nodes** - These nodes manage a multitude of internal functions, including but not limited to the following:

    1. System Supervision and State Of Health monitoring
    2. Client State Of Health monitoring
    3. Number Portability (where applicable)
    4. OBS-Linked Billing (where applicable)
    5. Remote System Management

4.2 Protocols

Higate client applications may use one or more of the following application-layer protocols over TCP/IP.

4.2.1 SMPP

Short Message Peer to Peer (SMPP) is an open, industry-standard protocol for sending text message (SMS) data over the Internet. It is the standard protocol for communication between a “Short Message Service Centre” SMSC (the role of the Higate system), and an “External Short Message Entity” (ESME), which is the role played by the Higate client application.

Despite the fact that SMPP is an industry standard, there are almost always small differences in implementation. These are usually confined to the many optional TLV (Tag Length Value) parameters, some of which are standard and some of which are implementation specific. The Higate makes extensive use of TLVs to carry as much of the functionality available in HGV200 over to the SMPP protocol as possible. The translation is complete for SMS and related billing transactions, but does not yet support the other bearers such as MMS, LBS, USSD, etc.

All SMPP clients connect to the core Higate system via the SHG process, which serves as a protocol translation gateway between SMPP and HGV200.

The Higate implementation supports the ability to send multiple SMS segments in a single transmission unit – this has important consequences for OBS, or MT-Billed messaging.

A full description of the Higate SMPP implementation details is documented separately in the Intarget SMPP API.

4.2.2 HTTP/XML

Unlike the other protocols, this does not constitute a persistent TCP/IP connection, in other words, it is "**not session orientated**". Instead, XML messages are exchanged by means of the HTTP POST mechanism.

The implications of this are that:

- All POST'ed messages must be individually authenticated
- Bandwidth utilization can be substantially greater than the other alternatives
- Clients are required to pre-register their destination URLs for return of the traffic
- Client applications must detect duplicate received messages (see below).

Nevertheless, despite these limitations, it is simple to implement, and is sometimes essential in a website scenario.

From the client's perspective, the HTTP Interface behaves as a webserver.

4.3 Higate APIs

The Higate system currently supports the following development environments:

- HTTP/XML clients from any environment
- SMPP clients from any environment

The detail of each API is documented separately, but the underlying functionality is essentially the same across all of them. It is important to understand the common Higate concepts described in this document before attempting to implement any of the available APIs.

4.4 Message Throughput

Message throughput on the Higate persistent API's can be affected by various factors.

- **Latency**: The amount of time that it takes a packet (datagram) to travel from the source to the destination. These transmission delays are due to the storage and disk access delays in the numerous intermediate devices such as switches and bridges. Clearly, the more intermediated devices (or 'hops'), the greater will be the latency.

- **Bandwidth**: The inherent data transfer rate (capacity) of the underlying network segments measures in bits per second. The total bandwidth of a sequence of network segments in obviously limited by the segment with the lowest bandwidth.

- **Port (traffic) Shaping**: This is an Internet traffic management practice that switches and bridges may implement so as to optimize the usable bandwidth by delaying some packets that meet certain criteria. One such strategy is to give preference to traffic on the TCP/IP port 80. Unfortunately, since the Higate servers operate (listen) on a variety of different ports (not port 80), Higate client traffic can be adversely affected.

- **Network Reliability**: If a network link drops messages (possibly due to congestion), the session partners will be required to resend messages – thereby reducing overall throughput.

The overall impact of these factors is that it can sometime take a significant period of time for a single Higate client (application) message to actually reach the server. Furthermore, since the underlying Higate protocol (HGV200) requires each massage to be acknowledged by the session partner, the delay between sending a message and receiving the subsequent acknowledgement can sometimes be as much as 400 ms (or more) – depending on the underlying network conditions.

4.5 Sliding Window

In order to overcome these inevitable transmission delays, the HGV200 protocol was designed to have an adjustable sliding window protocol. Essentially what this means is that each session partner may send multiple sequential messages without waiting for their associated acknowledgement messages to arrive, and may continue to do so until the number of un-acknowledged messages reaches some predefined limit – called the sliding window size. It is important to understand here that we are referring to the HGV200 protocol messages not the application layer messages.

When network latency is high, throughput can be improved by increasing the sliding window size.

Note that system will not allow the application to send at a rate that is faster than the limits imposed by the sliding window size. If the application ever attempts to send message after the number of un-acknowledged message sent has already reached the defined window size the API will return an error code indicating that the transaction rate has exceeded the limit.

5. Subscription Services

It is common practice in the mobile market to attract subscribers to a service which is paid for either once-off or periodically (daily, weekly, monthly), by means of OBS transactions. These are knows as subscription services and are subject to a number of industry rules that try to ensure that subscribers are not exploited. These rules include:

- The requirement for regular messages (SMSs) to remind the subscriber that they are registered for such a service
- Subscribers must be informed as to how to unsubscribe from the service. This usually takes the form of a so-called 'STOP' message (SMS) that the subscriber is required to send to a pre-defined number.
- Subscribers have to opt-in to the subscription.

The complete set rules for 'subscription services' is fairly extensive and is documented separately, or available on the WASPA website at www.waspa.org.za/code/codeconduct.shtml

6. Product Portal

6.1 Overview

This document describes the procedure to create and maintain product subscriptions via the Redbox Product Portal Interface.

The Redbox Product Portal is a system used by Integrat as well as clients to manage all product related information. A "product" is defined as any specific service offered through the Integrat aggregation platform.

Due to new and future expected requirements related to mobile billing requests clients are required to provide a number of details of all their products that have a billing component. The Product Subscription Interface described here is the means by which these details are recorded. It is the responsibility of all clients to ensure that the details are accurate as these details are used by the network operators to manage billing requests.

The procedure described below is a tutorial on how to add and update the product subscription details.

6.2 Permissions

All clients are provided with access to the Redbox system to manage components of their accounts. The product portal interface requires the user to have the specific permission "Product portal" to be enabled for that user. This permission can be set from the "Manage users" option available from the "My account" menu option.

Product Portal Permission

Product portal	Edit and view product descriptions	☑

Since incorrect information supplied here can have severe financial implications it is very important to manage the assigning of this permission carefully.

6.3 Viewing Existing Products

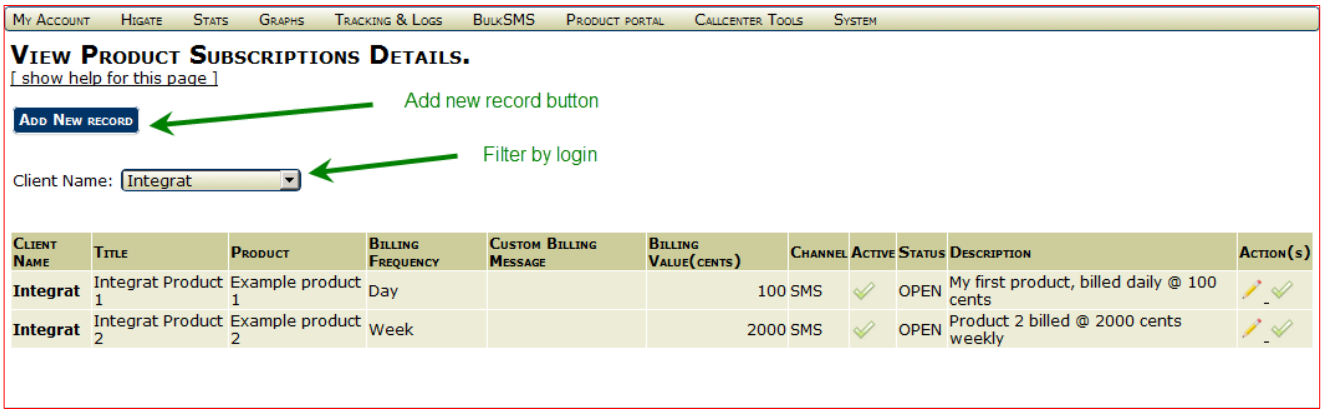
The product subscriptions main page is accessed from the "View Product Subscriptions" option available from the "Product Portal" main menu option.

After the menu option is selected the user is presented with a list of existing product subscriptions.
The following fields form part of each product subscription record:

- **Client Name** – The login name (also known as Client Name) under the main account to which the product subscription has been linked.
- **Title** – The title of the product that will be presented to network operators. This field can contain only alphanumerical characters and the "_" and "-" characters. Examples of the contents of this field will be "News headlines", "Daily Jokes" etc. The title must be unique for the login name. The maximum length of this field is 64 characters.
- **Product** – The product refers to the name specified in the "Category" field in the OBS subscription information ticket that should be submitted with all OBS transactions. The maximum length of this field is 32 characters.
- **Billing Frequency** – The possible values can be selected from a dropdown list, they are "Once" (for ad-hoc billing requests),"Day", "Week", "Month", "Year"
- **Billing Value** – The value at which the subscriber will be billed at, at the indicated frequency, in cents (ZA)
- **Channel** – The channel through which users subscribe to this service. If more than one channel is used include a list of all channels, for example "SMS" or "SMS, USSD, WEB".
- **Active** – Products are set active by default. Products can be deactivated by clients, which would prevent the product details from being used for new billing transactions.
- **Status** – The status of all products are 'OPEN" by default when created. Integrat can change the status to either "SUSPENDED" which will prevent any further subscriptions for this product until the suspension is lifted, or "CLOSED" which is a final state which cannot be reversed and that will prevent any new subscriptions on the product.
- **Description** – This field should be used to provide a short description of the service in maximum 256 characters.

Initially the product subscription records for all logins linked to the account is shown, ordered alphabetically. The results can be filtered by login by selecting a specific login from the "Client name" dropdown box.

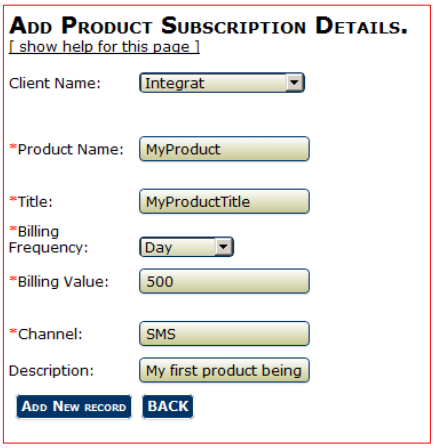
View Product Subscription View



A new product can be added by selecting the "Add new Record" button. Selecting this function fill present the user with the "Add Product Subscription Details" form. The use of this form is discussed in detail in the next section.

6.4 Creating a New Product

Upon selecting the "Add New Record" button the "Add Product Subscription Details" form is shown. To create a new product record follow the following steps:



Step1: Select the login (Client Name) from the dropdown box. This box contains all logins related to the main account. Until this selection has been made the other fields will remain read-only.

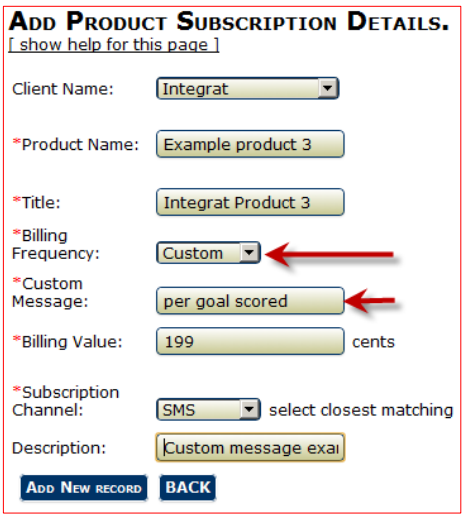
Step2: Add the Product Name, Title, Billing Frequency, Billing Value and subscription Channel fields. (Compulsory fields are indicated by *). Optionally add a product description.

Step3: Select the "Add New Record" button the commit the changes.

Take care when creating a new product, only the Product Title, Channel and Description can be edited once a new product has been created.

6.5 Setting a Custom Billing Frequency

The latest double-opt-in specification allows for a custom billing frequency to be entered instead of a fixed frequency per calendar period. For example it is possible to create a billing frequency of "per game played" or "per goal scored" etc.



To set up a custom billing frequency - select the "Custom" option from the Billing Frequency drop down box. A "Custom Message" field will appear where you can enter your custom message.

Because the opt-in message is limited to 160 characters take care to keep this field as short as possible. Also be sure that there is no ambiguity in the custom message.

6.6 Opt-In Message

The format of the opt-in message that is send to subscriber the will be as follows:

If the selected Billing Frequency is "CUSTOM" then the opt-in message will be:

"Confirm your request for (Title)@ (Billing Value) (Custom Message).Reply "YES" to accept/"NO" to cancel, free SMS"

If the selected Billing Frequency is not "CUSTOM" then the opt-in message will be:

"Confirm your request for (Title)@(Billing Value) per (Billing Frequency).Reply "YES" to accept/"NO" to cancel, free SMS"

If the selected Billing Frequency is “ONCE” then the opt-in message will be:

"Confirm your request for (Title)@(Billing Value) once-off.Reply "YES" to accept/"NO" to cancel, free SMS"

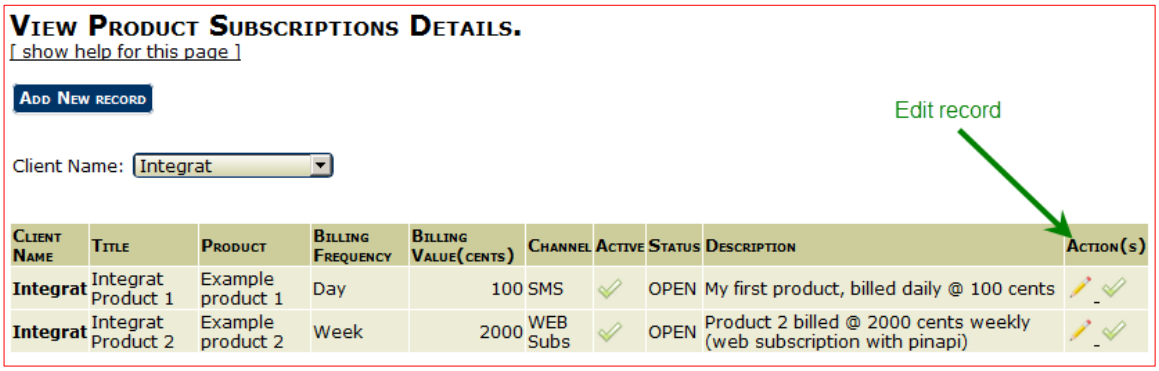
Important: If the length of the message exceeds 160 characters the “Title” field will be shortened by Vodacom to allow the message to fit into a single message segment. A preview function is provided when the product is created to ensure the message is formatted correctly. Please select the “View OptIn Message” button to trigger the preview.

The exact format of this message might change without warning and is controlled by Vodacom.

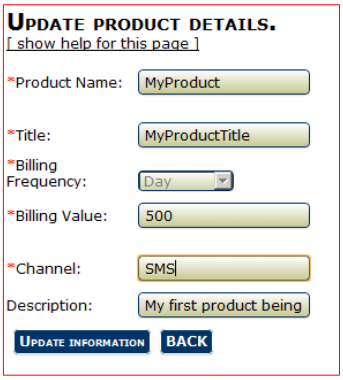
6.7 Editing a Product

At any point after a product subscription records has been created can some of its parameters be edited. The parameters that can be edited are limited to Product Title, Channel and Description due to technical and safety reasons. If a product has been created with incorrect parameters that cannot be changed rather deactivate that product and create a new product from scratch.

Invoking Product Editing



To edit a product select the Edit function from the Actions column that will load the “Update Product Details” form.

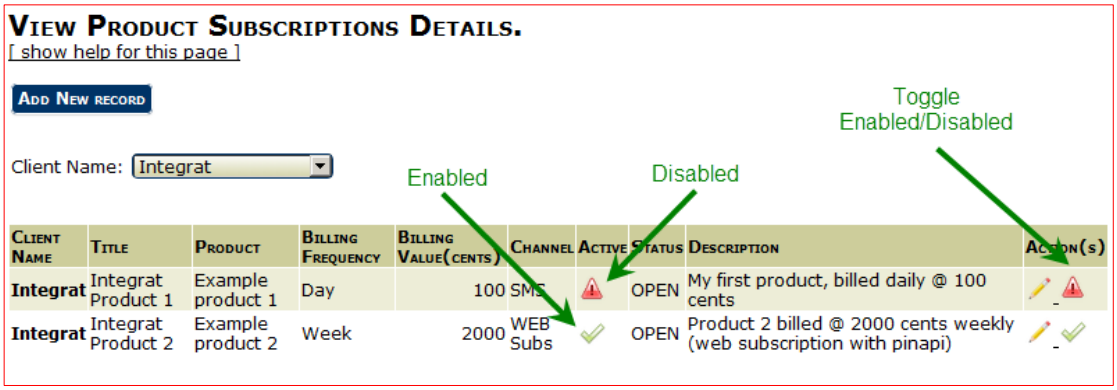


Similarly to creating a new product the values are entered into the relevant fields followed by selecting the “Update Information” button. To cancel the update select the “Back” button or navigate to the main page using the menu option.

Note: The product subscription record details currently cater for a single network operator’s new OBS billing requirements. As other operators implement similar systems the fields may be expanded and what is editable might also change depending on future requirements. This situation is unfortunately unavoidable.

6.8 Changing the Enabled Status of a Product

Products can be enabled or disabled from the "View Product Subscription Details" form. To toggle the status of a product select the enabled/disabled status icon in the actions column next to the appropriate product. A dialogue will pop up to confirm the action.



Note that the OPEN/SUSPENDED/CLOSED Status overrides the Enabled/Disabled activity status of a product. Thus the activity status only has effect when the status is OPEN.

7. Bearers

On Higate, bearers are denoted by the acronym TOC ("Type of Contents"), and all Higate transactions have an associated TOC.

TOC	Value	Description	Supported
SMS	1	Short Message Service	Yes
USS	4	Unstructured Supplementary Services Data (USSD)	Yes
OBS	7	Online Billing Service (Was MNU now deprecated)	Yes
VSR	9	Voucher Services	Yes

8. SMS

8.1 Encoding

The content of SMS messages should usually be submitted as plain ASCII (ANSI X3.4) but there are exceptions to this rule when the content contains characters that don't overlap between ASCII and GSM7 (see SMS3.38).

SMPP 3.4 "suggests" that the ESME submits the short message in 8 bit ASCII (includes extended characters) to the SMSC and that the SMSC performs the character conversion based on the data coding scheme specifies. This is however just a suggestion and not a rule, and mobile operators are inconsistent in applying these rules. In GSM7 for example submitting "}" would lead to an escape sequence being sent to the phone as this character is in the extended part of the alphabet table.

MTN does this correctly for example but Vodacom does not, it just drops the top bit off the 8 bit ASCII submitted. Some other characters affected are "**{|~ [\]^_**" (all in the extended part). Another example is latin-1 encoding which should map 1:1 to ASCII, but here CellC sends this to the phone as GSM7. There can also be differences between different SMSCs at the same operator, some using GSM7 by default and others using CCITT T.50.

When a message contains characters that don't overlap it is important to test the resulting message on a physical device for each network to ensure the message is displayed correctly.

9. USSD

A USSD session between a handset and a mobile application consists of nothing more than an exchange of simple ASCII text strings. USSD has no inherent 'understanding' of a so called menu, it is just an ASCII text string that happens to be formatted so as to appear as a menu on the subscriber's phone. It is also important to understand that USSD sessions are half-duplex – any message from one side must be answered before the other side may send a new message.

The following event play an important role in the management of a USSD session.

9.1 Events

This event is invoked whenever one of the following USSD events (dialogueID's) occur.

- OPEN: A new session has started. At this point it is usually convenient to do any pre-processing necessary to handle the pending session. No response can be sent on an OPEN event.
- CLOSE: A session has just terminated. No response can be sent on a CLOSE event.
- TRACE: This event will only be generated for clients that have a lodged USSD script on the Higate platform. It reflects the execution of a script 'Trace' method. Replies are routed back to the calling script.
- ERROR: Denotes a scripting error. No response can be sent on a CLOSE event.
- REQUEST: This event indicates one of two possibilities...
    - If there is a lodged USSD script then this represents a request from the script for information.
    - If there is no lodged script then this denotes reply text that was entered by the subscriber, or the word 'REQ' for the first such event (ie the subscriber has just started the session).
    -

At this point the application MUST return a text string to the handset. The application may also terminate the session at this point by specifying a flag value of FL_EXIT_DIALOGUE.

9.2 Important

- The application must only reply to REQUEST events.
- All REQUEST events must be answered – even if it is only to close the session using the FL_EXIT_DIALOGUE flag.
- Many USSD sessions may occur simultaneously, so the application must manage the state of each session independently.

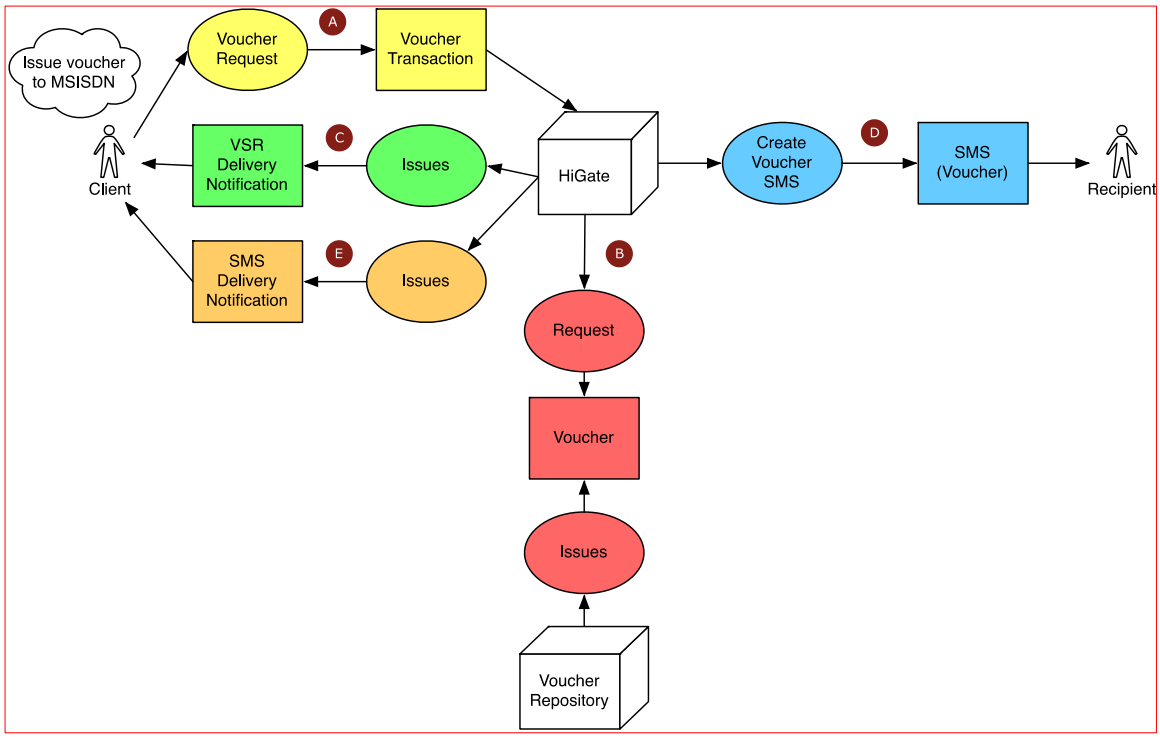
10. VSR

The Higate system offers a voucher repository through which almost any type of voucher can be issued.

10.1 System Overview

The content flow diagram below illustrates the steps in in the voucher issuing process.

Voucher Transaction Flow



A. The client application submits a VSR transaction containing a voucher request XML document to the Higate system which creates a voucher transaction. If successful it provides the client application with a Higate sequence number.

B. The Higate system requests a voucher from the voucher repository. If the voucher is network dependant it uses the supplied destination address to determine for which network the voucher should be issued.

C. The Voucher transaction status is returned to the client application.

D. If the voucher was successfully issued an SMS transaction is generated containing the voucher details as specified in the request document.

E. The status of the SMS delivery is returned to the client. Note that this delivery notification will have a different sequence number from the originating VSR transaction but will contain the same reference number as supplied by the originating VSR transaction.

10.2 Services

The following voucher services are currently available:

Higate API:

- Issue voucher with SMS delivery.
- Issue voucher and return to client.

10.3 Setup

Before any vouchers can be issued by an account the account has to be configured for the particular type of vouchers. The setup process is as follows:

1. Your Integrat account manager will arrange access to the voucher repository.
2. Your service code(s) will be configured to accept VSR transactions
3. 3The application may only utilize pre-authorized Service codes. Depending on how the Voucher Repository has been configured for this Account/Voucher/Network combination.

Note that test vouchers that have no value can be loaded for testing purposes.

10.4 API

There are two methods of issuing a voucher:

1. By submitting a VSR transaction directly to the Higate API which will issue the voucher and send it to the receiver via SMS according to a user customised template consisting of a number of keyword fields.
2. By submitting a VSR transaction directly to the Higate API which will issue the voucher and send it back to the client application which is then responsible for submitting the voucher.

10.4.1 Higate API

Vouchers are issued by submitting a VSR transaction to the Higate system. If issuing via SMS is configured (default) then a SMS transaction will be generated by the system automatically upon successful issuing of the voucher. The status of the SMS shall be send to the client application. The reference number used when requesting the voucher can be used to link the SMS transaction to the originating voucher transaction.

10.4.1.1 XML Template – Submit

Vouchers are requested by submitting an XML document containing the following elements:

- **Action** – (Optional). If this field is omitted the voucher is issued to the recipient via SMS. To return the voucher to the calling application instead include this field with the value set to “Get” instead.
- **Payload** – (Optional). Depends on the nature of the product.
- **Source** – (Optional). This is for audit purposes and may include any meaningful string denoting the source of the original query. For instance it may include the USSD string that was dialled to initiate the request.
- **FaceValue** – The value of the associated mobile product. If this represents a monetary amount then it must be expressed in cents. Note that it may alternatively be a Volume amount depending on the nature of the product. It must lie in the range defined by the values ‘*ValueMin*’ and ‘*ValueMax*’ returned by ‘*QueryNetworkProducts*’.
- **Instruction** – An SMS template for the message to be sent to the recipient. This may be ignored if the application is expected to deliver the Voucher redemption details (instructions). See ‘SMS Templates’.

XML Voucher Submit Document Format

```
<Voucher>
   <Action></Action>
   <Name></Name>
   <FaceValue></FaceValue>
   <Instruction></Instruction>
   <Source></Source>
   <Payload></Payload>
</Voucher>
```

10.4.1.2 XML Template – Response

When vouchers are requested with the "**Get**" action the voucher is returned to the application in the form of a VSR transaction with the voucher details in an XML structure.

XML Voucher Response Document Format

```
<Txn>
  <RefNo></RefNo>
  <Instruction></Instruction>
</Txn>
```

- **RefNo** – The reference number of the VSR transaction that requested the voucher.
- **Instruction** – The voucher details formatted according to the SMS Template rules. Note that this means that the client application needs to specify the template for the format of the reply message. See 14.1.3 Instruction SMS Template

10.4.1.3 Instruction SMS Template

SMS templates are text strings defining the format of voucher redemption instructions to the subscriber. A typical example of such a template is as follows...

Please dial {**RDCODE**} to redeem your {**FACEVALUE**} airtime. Ref ({**V_REF**}).

Note that text enclosed by braces {} represent place-holder text that will be replace by the Voucher Repository once the associated values are known. Note that the onus is on the application to ensure that valid parameters are used. If the Voucher Repository fails to identify a parameter, then it will remain in the message 'as is'.

Currently defined parameters include...

- {**RDCODE**} – The network specific USSD redeem code
- {**PIN**} – the assigned redemption PIN for this voucher
- {**FACEVALUE**} – the Face Value of the voucher in format :<currency symbol><rands>.<cents> where rands uses the comma currency format to denote thousands, millions etc. (E.g. R 15.00)
- {**V_REF**} – the unique Voucher Repository Reference number for this Voucher/Transaction
- {**A_REF**} – the client application specified Reference Number.

10.4.1.4 Higate HTTP/XML

The examples below shows the minimum required elements needed to request a voucher via the HTTP interface. The voucher request content can be sent in plain text, hex encoded or base 64 encoded format.
Hex Encoding Example:

HTTP Hex Encoded Example

```
<?xml version="1.0"?>
<Message>
  <Request Type="SendVSR" RefNo="2">
    <UserID>USERNAME</UserID>
    <Password>PASSWORD</Password>
    <SendVSR ToAddr="0821234567">
<Content
Type="HEX">3c566f75636865723e3c4e616d653e41697274696d65546573743c2f4e616d653e3c4661636556616c75653e353
0303c2f4661636556616c75653e3c496e737472756374696f6e3e5468616e6b7320666f72207573696e67205768697a7a21205
46f2072656465656d20796f7572207b4641434556414c55457d2061697274696d6520766f7563686572206469616c207b5244
434f44457d3c2f496e73747756374696f6e3e3c536f757263652f3e3c5061796c6f61642f3e3c2f566f75636865723e</Content>
    </SendVSR>
  </Request>
</Message>
```

When a VSR transaction is submitted in plain text format it is important to substitute the XML reserved characters with XML safe escape codes. The substitutions are listed in Table 1 below.

Character	Substitute
&	&amp;
'	&apos;
"	&quot;
<	&lt;
>	&gt;

Text Encoding Example:

Note the character substitutions shown in **red font.*

HTTP Plain Text Example

```
<?xml version="1.0"?>
<Message>
  <Request Type="SendVSR" RefNo="1">
    <UserID>myuser</UserID>
    <Password>mypass</Password>
    <SendVSR ToAddr="0821234567">
<Content
Type="TEXT">&amp;lt;Voucher&amp;gt;&amp;lt;Name&amp;gt;AirtimeTest&amp;lt;/Name&amp;gt;&amp;lt;FaceValue&amp;gt;500&amp;lt;/FaceValue&amp;gt;&amp;lt;Instruction
&amp;gt;Thanks for using Whizz! To redeem your {FACEVALUE} airtime voucher dial
{RDCODE}&amp;lt;/Instruction&amp;gt;&amp;lt;Source/&amp;gt;&amp;lt;Payload/&amp;gt; &amp;lt;/Voucher&amp;gt;</Content>
    </SendVSR>
  </Request>
</Message>
```

Successful Reply Example

```
<?xml version="1.0"?>
<Response token="" status_code="0">
  <Data name="msg_generic_rsp">
    <field name="msg_no" value="1"/>
    <field name="seq_no" value="12345678"/>
  </Data>
</Response>
```

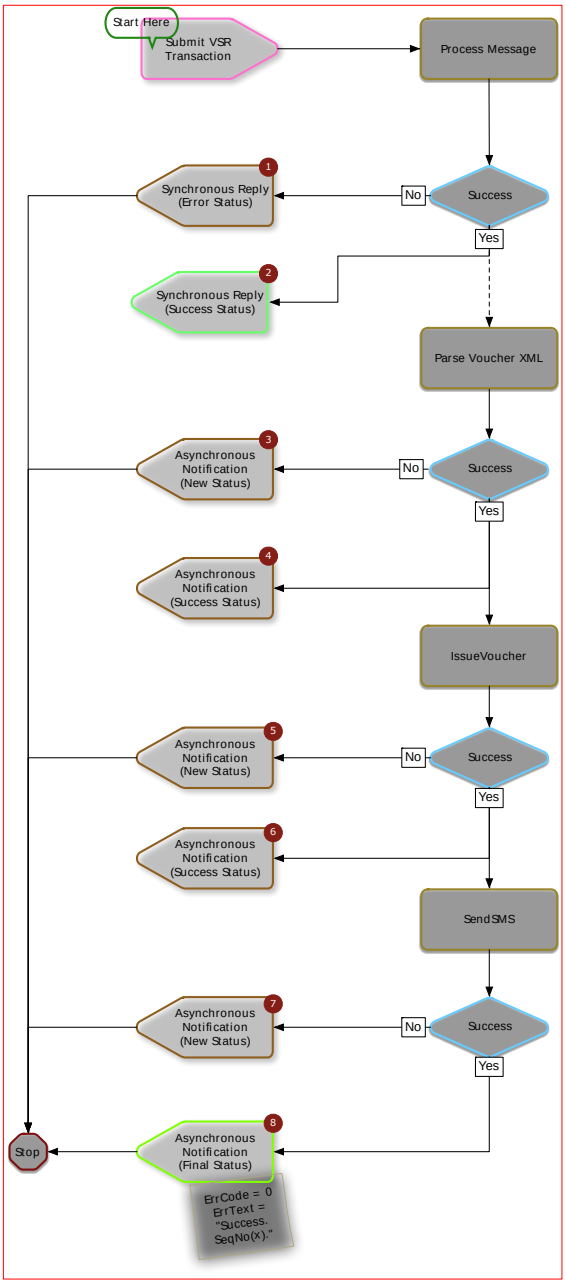
Failure Reply Example

```
<?xml version="1.0"?>
<Response token="" status_code="-1">
  <Data name="msg_nak">
    <field name="status" value="1"/>
    <field name="error_text" value="<error message>"/>
  </Data>
</Response>
```

10.5 Error Management

The flow chart in Figure 6 shows the steps in typical voucher transactions.

Voucher Transaction Flow



There are four possible points of failure in a VSR transaction:

- Failure on submit (Refer to 1 in Figure 6).
- Failure on voucher XML parsing (Refer to 3 in Figure 6).
- Failure on voucher issuing (Refer to 5 in Figure 6).
- Failure on voucher delivery (Refer to 7 in Figure 6).

10.5.1 Failure on Submit

In this case the failure code is returned directly in reply to submit transaction. The transaction is never accepted or recorded by the Higate system.

10.5.2 Failure on Voucher XML Parsing

In this case the XML parsing was successful but the voucher could not be issued. The error response will be in the form of a call back (HTTP interface) or delivery notification (SMPP interface)

10.5.3 Failure on Voucher Issuing

In this case the XML parsing was successful but the voucher could not be issued. The error response will be in the form of a call back (HTTP interface) or delivery notification (SMPP interface)

10.5.4 Failure on Voucher Delivery

In this case the voucher was successfully issued but delivery via SMS failed.

11. Subscriber Billing

11.1 MT Billing

A true MT-Billed transaction (e.g. SMS) is passed on to the relevant operator together with an associated currency value. The operator charges the destination subscriber the prescribed amount and sends the transaction on to the subscriber. If the subscriber has insufficient funds to cover the cost of the transaction, it will fail and the subscriber will not receive the transaction (SMS).

11.2 Online Billing Services

In some regions (such as South Africa), the local operators do not support true MT-Billing, but do provide so-called 'Online Billing Services' (OBS). This bearer provides for the ability to issue direct (silent) billing requests against a mobile subscriber's account, without necessarily delivering any service (such as an MT SMS).

OBS is a billing mechanism, and has no direct association with any other bearer such as SMS or MMS.

11.2.1 OBS by Operator / Telco

The implementation of OBS can be different, depending on the operator.

11.2.1.1 Vodacom OBS

The Vodacom OBS implementation is characterized by the following features:

- Valid transaction values are on a continuous scale between some minimum and maximum value.
- The transactions are completed in two steps:
    - An authorization step

- A confirmation or cancellation

The Vodacom rules dictate that the confirmation step may only be requested once the full service (e.g. a content SMS) has been successfully delivered to the subscriber. For example, in the case where the service is an MT content SMS, the OBS transaction may only be confirmed once the SMS has been successfully delivered to the subscriber (i.e. a successful Delivery Receipt has been received). If the content could not be delivered, then Vodacom requires that the OBS transaction to be 'Cancelled'.

- An OBS transaction is not deemed to be successful until it is confirmed.
- There are very strict requirements in terms of percentage failure rates. (See Section 7.2.2)

11.2.1.2 MTN OBS

The MTN OBS implementation is characterized by the following features:

- Valid transaction values are on a discrete scale between some minimum and maximum value. The Higate automatically adjusts the Login-defined value to the largest valid (discrete) value less than or equal to the specified value.
- The transaction is completed in a single step.
- There is no mechanism to cancel a transaction.

11.2.1.3 CellC OBS

The Cell C OBS implementation is characterized by the following features:

- Valid transaction values are on a discrete scale between some minimum and maximum value. The Higate automatically adjusts the Login-defined value to the largest valid (discrete) value less than or equal to the specified value.
- The transaction is completed in a single step.
- There is a cancellation mechanism, but this is not implemented by Higate.

11.2.2 OBS Controls

In some regions (such as South Africa), the operators place very stringent requirements and penalties upon the WASPs and content providers.

11.2.2.1 OBS and Percentage Failures

Some operators (e.g. Vodacom) have limited capacity to process the volume of OBS transactions that are issued on a daily basis. To address this, they have imposed penalties on WASPs and content providers by increasing the imposed transaction fee on a sliding scale that depends on the percentage of OBS transactions that fail.

11.2.2.2 OBS and Recycled Numbers

MSISDNs are periodically recycled - local network operators remove allocated MSISDN numbers from circulation, only to reintroduce them again some three months later, at which point they are allocated to new subscribers.

The challenge is to ensure that OBS Subscription Services (See Section 6) that applied to the first subscriber are not carried over to the second subscriber. This scenario should not occur because while a number is out of circulation, all OBS billing attempts on this number will fail, and the content provider is obliged (according to WASPA rules) to automatically unsubscribe a number after three months of unsuccessful billing attempts.

Unfortunately, many content providers do not honour this rule, leading to bad publicity for the operators.

In an attempt to remedy this, some operators (currently only Vodacom) have introduced the requirement for all OBS transactions to be accompanied by a parameter that defines the "Subscription Start Date" (the date and time the subscription was purchased) for the content service for that MSISDN. The operator is then able to identify (and hence fail) any OBS transaction that applied to a previous owner of the specified MSISDN by comparing this date with the date on which the number was last recycled.

Because the above solution does not apply to all operators, Higate also implements an OBS control mechanism (See Section 9).

12 OBS-Linked Transactions

OBS-linked transactions are a Higate mechanism that emulates MT-billing. The Higate system achieves this by creating two transactions – a pure OBS transaction and a second one for the 'Linked transaction' (e.g. SMS). Note however that the linked transaction is actually only created if the OBS is at least 'Authorized' (Vodacom) or just successful (MTN & Cell C).

A Login may submit an OBS-linked transaction by:

- Setting the FL_OBS_LINKED bit in the 'Flags' parameter.
- Specifying a local currency value that is within the valid range.
- Specifying a unique Reference number (RefNo)
- Submitting the linked transaction of the required type (SMS, MMS, etc.)
- If this OBS is for a "Subscription Service", then the following additional parameters are required (see Section 6 for more on Subscription Services):
    - A short Login-defined service name (less than 32 characters)
    - A "Subscription Start Date". This is the date and time upon which the subscription commenced expressed in UTC (or GMT).

It is important to appreciate that for a successful OBS-linked transaction there are always two transactions, and that the status changes for both transactions are returned to the client application (in real-time). If the OBS is NOT 'Authorized' (Vodacom) or not successful (MTN & Cell C), then only the OBS transaction is created. This can lead to confusion, but is easily managed if the following points are understood:

- The status changes for both transactions are returned to the client application via Result Messages.
- In the Result Messages…
    - The OBS transaction will have a TOC value of TOC_OBS.
    - The TOC of the Linked transaction will depend on the specified transaction type.
    - Each transaction will have a unique (different) SeqNo.
    - Both transactions will have the same Login-defined RefNo.
    - The OBS transaction value may be adjusted downward to match a valid discrete value (MTN and Cell C).

13. Double Opt-In (DOI)

OBS charge transactions can only be performed on subscriber accounts that have successfully opted in to a recurring subscription or to a once off (ad hoc) charge. The subscription process is triggered by the FIRST keyword being included in the Started parameter in the "*higate_subscription_data_2*" TLV in the case of SMPP API and the Started attribute in the "*Subscr*" parameter of a request in the case of the HTTP API.

Each network specifies a set of business rules that must be adhered to. For the details of each network specific rules and requirements refer to the separately available DOI documentation.

14. Higate OBS Control Mechanism

The Higate OBS control mechanism attempts to ensure that client applications adhere to the rules regarding OBS transactions. The purpose is to both limit the transaction failure rates (percentages) and to safeguard against billing of recycled numbers.

A central component of the OBS control system is a table ('obsbarred') that maintains a running total of consecutive OBS failures for every MSISDN. If an OBS transaction is successful, then the associated MSISDN will be removed from the list. Note that barring a specific MSISDN relates to all services and not a specific one. Thus all OBS failures for an MSISDN are taken into account, and when barred will affect all services trying to perform OBS billing on that number.

14.1 Status

- Unbarred: An Unbarred status denotes an MSISDN number for which there have been one or more consecutive failures, but for which the 'FailCount' has not yet reached the value of FAIL_LIMIT (see Configuration Items).
- Barred: The number has been barred from further OBS transactions. Note that this is not necessarily a permanent condition (See Heuristic Rules)
- Suspended: Once that it has been determined that a number has probably been deactivated (pending recycle), the status will be set to 'Suspended'. This is essentially a permanent condition until there is hard evidence that the number has been recycled – at which point the number will be removed from the 'obsbarred' table.

14.2 Client Rules

The usual client rules apply, with one exception.

If the client application receives an OBS transaction 'Failed' result that indicates that the MSISDN number has been 'Suspended', then the application should treat such a condition as a permanent error and should immediately remove this number from its subscription lists. In this case there is no need to send an 'unsubscribe' notification message.

14.3 Configuration Items

FAIL_LIMIT - (Currently set to 1000) - This is the number of consecutive failures after which the MSISDN will be 'Barred'. Note that a 'Barred' number may be subsequently 'Unbarred'.

DAYS_LIMIT - (Currently set to 92 days) - The number of days after which a 'Barred' MSISDN will be flagged as 'Suspended'