

HYTE Console

HYTE Console

Messaging File Upload & Send

Quick User Guide

November 1, 2018

Table of Contents

<i>Zip file requirements</i> _____	3
<i>Supported message files</i> _____	3
<i>Required message file format</i> _____	3
<i>Known Limitations</i> _____	3
<i>Behavior</i> _____	4
<i>Message Metadata</i> _____	5
Message Metadata Template _____	5
Message Metadata with Embedded Text Body _____	6
Message Metadata with Embedded Binary Body _____	6
<i>How To Use - v4.0.13</i> _____	7
Upload a Zip File for Bulk Message Loading _____	7
Sending a Bulk Message Load _____	7
Review Status of an Uploaded Zip File _____	8
Delete (Remove) an Uploaded Zip File _____	8

ZIP FILE REQUIREMENTS

- File name must end in .zip
- Self-extracting zip files are not supported.
- Empty zip files are not supported.

SUPPORTED MESSAGE FILES

- HYTE messages are supported with the xml metadata file containing an embedded message body.
- HYTE messages are supported with the xml metadata file and an external message body file with matching index values in both filenames.
- Message bodies may be text or binary.

REQUIRED MESSAGE FILE FORMAT

- HYTE message metadata files must conform to the format:
msg- $\$$ index- $\$$ miscellaneous.xml
- HYTE message body files must conform to the format:
msg- $\$$ index- $\$$ miscellaneous.txt or *msg- $\$$ index- $\$$ miscellaneous.bin*
- Multiple message metadata files with matching index values will invalidate the zip file.
- Message filenames may only consist of the following characters:
0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ-_.

KNOWN LIMITATIONS

- HYTE JmsResend Messaging Task zip files should never be used for upload and send, since it contains both the original and the modified message files.
- HYTE JmsProduceBatch Messaging Task zip files are not supported for upload and resend, as all the messages reference the same message body file.

BEHAVIOR

- For security reasons, the HYTE Console will never unzip an uploaded zip file. All processing is done via extracting data directly from the zip file.
- Invalid zip files are automatically deleted from the server.
- Uploading a zip file of the same name is not supported. The second zip file will be invalidated and deleted. The first zip file must be deleted before a zip file of the same name is uploaded.
- If any file within the zip file fails the CRC check, the entire zip file will be marked as invalid.
- If any message xml metadata file within the zip file fails to parse correctly, the entire zip file will be marked as invalid.
- If any message body file within the zip file fails to load correctly, the entire zip file will be marked as invalid.
- The original JMSMessageID and JMSDestination, header values from the xml metadata file are not used. New JMSMessageID and JMSDestination values will be automatically created per the JMS specification.
 - The original JMSMessageID header value is copied to a string property: HYTE_ORIG_MSG_ID.
 - The original JMSDestination header value is copied to a string property: HYTE_ORIG_DEST_NAME.
- The original JMSTimestamp and JMSExpiration header values from the xml metadata file are not used since the expiration could have already passed.
- Zip files that exceed the max file size (default 128 MB) are marked as invalid and deleted.
- Zip files that exceed the max number of messages contained in the zip file (default 32,768 messages) cause the zip file to be marked invalid and deleted.
- By default, the full set of messages from the zip file will be sent in a single transaction. This ensures that if there were to be a problem during the send that a user may resend the entire zip file without having to be concerned with creating duplicate messages.

MESSAGE METADATA

Message Metadata Template

msg- $\$$ index- $\$$ misc.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<msg:Message xmlns:cfg="http://hyte.io/config/v4"
  xmlns:core="http://hyte.io/core/v4"
  xmlns:jce="http://hyte.io/crypto/ice/v4"
  xmlns:msg="http://hyte.io/messaging/v4"
  xmlns:auth="http://hyte.io/auth/v4"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="msg:JmsMessage"
  id="ID:414d5120514d4c414231202020202020202020202020207847865b0bcc0c20" type="TEXT">
  <JmsHeaders JmsCorrelationID="ID:CORRELATIONID-ABCDEFGH" JmsDeliveryMode="Persistent"
  JmsDestination="queue:///TEST.1" JmsExpiration="0"
    JmsMessageID="ID:414d5120514d4c414231202020202020202020202020207847865b0bcc0c20" JmsPriority="4"
  JmsRedelivered="false"
    JmsTimestamp="1535692436519"/>
  <JmsProperties>
    <Property xsi:type="core:BooleanProperty" value="" name=""/>
    <Property xsi:type="core:DoubleProperty" value="" name=""/>
    <Property xsi:type="core:FloatProperty" value="" name=""/>
    <Property xsi:type="core:IntProperty" value="" name=""/>
    <Property xsi:type="core:LongProperty" value="" name=""/>
    <Property xsi:type="core:StringProperty" value="" name=""/>
    <Property xsi:type="core:ShortProperty" value="" name=""/>
  </JmsProperties>
</msg:Message>
```

Message Metadata with Embedded Text Body

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<msg:Message xmlns:cfg="http://hyte.io/config/v4"
  xmlns:core="http://hyte.io/core/v4"
  xmlns:jce="http://hyte.io/crypto/jce/v4"
  xmlns:msg="http://hyte.io/messaging/v4"
  xmlns:auth="http://hyte.io/auth/v4"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="msg:JmsMessage" type="TEXT">
  <body xsi:type="msg:TextBody">
    <body>Sample text message body</body>
  </body>
  <JmsHeaders JmsCorrelationID="jmsCorrelationIdValue14" JmsDeliveryMode="Persistent"
    JmsExpiration="9223372036854775806" JmsMessageID="messageIdValue14"
    JmsPriority="4" />
</msg:Message>
```

Message Metadata with Embedded Binary Body

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<msg:Message xmlns:cfg="http://hyte.io/config/v4"
  xmlns:core="http://hyte.io/core/v4"
  xmlns:jce="http://hyte.io/crypto/jce/v4"
  xmlns:msg="http://hyte.io/messaging/v4"
  xmlns:auth="http://hyte.io/auth/v4"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="msg:JmsMessage" id="idValue14" type="HEXBINARY">
  <body xsi:type="msg:BinaryBody">
    <body>53616D706C652062696E617279206D65737361676520626F6479</body>
  </body>
  <JmsHeaders JmsCorrelationID="jmsCorrelationIdValue14" JmsDeliveryMode="NonPersistent"
    JmsExpiration="9223372036854775806" JmsMessageID="messageIdValue14"
    JmsPriority="4" />
</msg:Message>
```

Note: The binary body should be encoded in hexadecimal format.

How To Use - v4.0.13

Upload a Zip File for Bulk Message Loading

1. Select “Uploads” from the “Messaging” pull down menu at the top of the screen to enter the “Messaging Uploads” screen. The “Messaging Uploads” screen allows a user to upload Zip files and review status of previous file uploads.
2. Click the “Upload” icon at the top of the table.
3. Click the “Browse” button
4. Find and select the Zip file that you wish to upload from your computer hard drive.
5. Click the “Ok” button to begin the upload.
6. Click the “Reload” button to refresh the listing of uploaded Zip files.

Sending a Bulk Message Load

1. Select “Uploads” from the “Messaging” pull down menu at the top of the screen to enter the “Messaging Uploads” screen.
2. Click the box to the left of the Zip file that you wish to send. (Note: Only one file may be selected at a time)
3. Click the “Send” icon at the top of the table to enter the “Send Uploaded Messages” screen.
4. Enter the values for the “Server” and “Target” Queue that you wish to send the bulk messages.
5. Adjust the “Max Exec Time” to ensure that the process has long enough time to complete sending the number of messages, which are noted in the “Message Count” field.
6. Click the “Send” button to begin sending messages.
7. The “Sending Message(s)” pop-up provides status information regarding the send process and success.
8. Choose your next button depending on your desired task:
 - a. Select the “View Task” button to enter the “Messaging Tasks” screen to view statuses and details regarding your tasks within the HYTE Console.
 - b. Select the “Download Task” button to download a Zip file of the sent messages from the current Send task.
 - c. Select the “Close” button to exit the “Sending Message(s)” screen.

Review Status of an Uploaded Zip File

1. Select “Uploads” from the “Messaging” pull down menu at the top of the screen to enter the “Messaging Uploads” screen.
2. The “Status” field of the “Messaging Uploads” screen will indicate “Active” if the Zip file successfully passed all of the file upload verification tests.
3. If the “Status” field indicates “Invalid”
 - a. Click the box to the left of the Zip file that you uploaded
 - b. Click the “View Details” link next to the Status Messages on the right side of the screen to view the file upload error message(s).

Delete (Remove) an Uploaded Zip File

1. Select “Uploads” from the “Messaging” pull down menu at the top of the screen to enter the “Messaging Uploads” screen.
2. Click the box to the left of the uploaded file that you wish to delete.
3. Click the “Remove” icon at the top of the table to delete the highlighted uploaded file.
4. Click to “Ok” button on the screen asking to verify that you want to remove the highlighted uploaded file.