

# Target Jitterbit Variables

## Introduction

This page covers Jitterbit variables that are available for activities used as targets, organized by informational variables that you get (read), and settings variables that you set (write):

## Informational

- `jitterbit.target.file_count`
- `jitterbit.target.http.status_code`

## Settings

- `jitterbit.target.chunk_node_name`
- `jitterbit.target.chunk_size`
- `jitterbit.target.chunking_uncombined`
- `jitterbit.target.db.character_encoding`
- `jitterbit.target.db.commit_chunks`
- `jitterbit.target.db.include_null_in_sql_statement`
- `jitterbit.target.db.no_data_action`
- `jitterbit.target.db.pre_target_sql`
- `jitterbit.target.db.schema_name_delimiter`
- `jitterbit.target.db.transaction`
- `jitterbit.target.file_share.create_directories`
- `jitterbit.target.ftp.transfer_timeout`
- `jitterbit.target.http.form_data`
- `jitterbit.target.http.form_data.ContentType`
- `jitterbit.target.http.form_data.filename`
- `jitterbit.target.http.form_data.name`
- `jitterbit.target.http.max_redirs`
- `jitterbit.target.http.remove_trailing_linebreaks`
- `jitterbit.target.http.response.header.HeaderName`
- `jitterbit.target.http.ssl_cert_id`
- `jitterbit.target.http.transfer_timeout`
- `jitterbit.target.sftp.ssh_key_id` and `jitterbit.source.sftp.ssh_key_id`
- `jitterbit.target.text.character_encoding`
- `jitterbit.target.wave.json`
- `jitterbit.target.xml.include_empty_xml`
- `jitterbit.target.xml.include_null_xml`
- `jitterbit.target.xml.nsprefix`
- `jitterbit.target.xml.num_for_bool`
- `jitterbit.target.xml.prettify`

## Informational

### `jitterbit.target.file_count`

<b>Variable</b>	<code>jitterbit.target.file_count</code>
<b>Data Type</b>	Integer
<b>Description</b>	After all the input files have been transformed, this variable is set to the number of target files that will be transferred.

### `jitterbit.target.http.status_code`

<b>Variable</b>	<code>jitterbit.target.http.status_code</code>
<b>Data Type</b>	Integer
<b>Description</b>	This is the response status code returned from a HTTP target when used in an operation or <code>WriteFile()</code> script function. Available as of Jitterbit Harmony <a href="#">version 8.20</a> .

## Settings

### `jitterbit.target.chunk_node_name`

## Related Articles

- [API Jitterbit Variables](#)
- [Global Variables](#)
- [Global Variable Target](#)
- [Hosted Endpoint Jitterbit Variables](#)
- [Operation Jitterbit Variables](#)
- [Scripting Jitterbit Variables](#)
- [SFDC Jitterbit Variables](#)
- [Source Jitterbit Variables](#)
- [Specifying Source and Target Fields Dynamically](#)
- [Target Jitterbit Variables](#)
- [Text Jitterbit Variables](#)
- [Transformation Jitterbit Variables](#)
- [Web Service Jitterbit Variables](#)

## Related Topics

- [Sources and Targets](#)

Last updated: Feb 11, 2020

<b>Variable</b>	<code>jitterbit.target.chunk_node_name</code>
<b>Data Type</b>	String
<b>Description</b>	<p>If chunking (splitting a file apart) is enabled, a node is the repeating record (or loop node) at which it is possible to split the file.</p> <p>For example, in an XML file, the named levels (repeating records) could be root, organization, department, and employee. You could choose to chunk (split) the file on department or on employee.</p> <p>It is also possible to set the node in <a href="#">Operation Options</a>.</p>

**`jitterbit.target.chunk_size`**

<b>Variable</b>	<code>jitterbit.target.chunk_size</code>
<b>Data Type</b>	Integer
<b>Description</b>	<p>Two possible cases where <code>jitterbit.target.chunk_size = 2000</code>:</p> <ul style="list-style-type: none"> <li>• <b>Case 1:</b> Source chunking is turned off, perform target chunking with size = 2000.</li> <li>• <b>Case 2:</b> Source chunking is turned on, it will combine the target first and then chunk the combined target to size = 2000.</li> </ul>

**`jitterbit.target.chunking_uncombined`**

<b>Variable</b>	<code>jitterbit.target.chunking_uncombined</code>
<b>Data Type</b>	Boolean
<b>Description</b>	If set to <code>true</code> , and source chunking is turned on, target chunk files are not combined. This allows for 1 target file per record.

**`jitterbit.target.db.character_encoding`**

<b>Variable</b>	<code>jitterbit.target.db.character_encoding</code>
<b>Data Type</b>	String
<b>Description</b>	Specify the character encoding for ODBC target database character/text columns. If no encoding is specified, Latin-1 (ISO-8859-1) is assumed. For supported encoding, see <a href="#">Character Encoding</a> .

**`jitterbit.target.db.commit_chunks`**

<b>Variable</b>	<code>jitterbit.target.db.commit_chunks</code>
<b>Data Type</b>	Boolean
<b>Description</b>	If set to <code>true</code> and chunking is enabled for an operation with a database source, each chunk is committed to the database as it is created. This setting can make the transformation run faster as chunks can be committed in parallel.

**`jitterbit.target.db.include_null_in_sql_statement`**

<b>Variable</b>	<code>jitterbit.target.db.include_null_in_sql_statement</code>
<b>Data Type</b>	Boolean
<b>Default</b>	<code>false</code>
<b>Description</b>	If set to <code>true</code> , INCLUDE and UPDATE statements generated by Jitterbit for database targets will contain explicit null values for columns that are nullable and have a mapping with a null result.

**jitterbit.target.db.no\_data\_action**

<b>Variable</b>	<code>jitterbit.target.db.no_data_action</code>
<b>Data Type</b>	String
<b>Description</b>	<p>Option for what to do if an empty database target is created. These values are valid:</p> <ul style="list-style-type: none"> <li>• OK</li> <li>• Warning</li> <li>• Error</li> </ul> <p>The default (for when this variable is not set) can be changed in <code>jitterbit.conf</code> with the option in the <code>[ErrorHandling]</code> section called <code>NoTargetDataGenerated</code>. If nothing is specified there either, the default is to issue a warning.</p>

**jitterbit.target.db.pre\_target\_sql**

<b>Variable</b>	<code>jitterbit.target.db.pre_target_sql</code>
<b>Data Type</b>	String
<b>Default</b>	" "
<b>Description</b>	Run this SQL statement before starting to populate the target database table(s). The default is an empty string.

**jitterbit.target.db.schema\_name\_delimiter**

<b>Variable</b>	<code>jitterbit.target.db.schema_name_delimiter</code>
<b>Data Type</b>	String
<b>Default</b>	.
<b>Description</b>	The default is a single period (.). Character used by the target database to delimit the schema name from the table name, e.g. <code>Schema.Tab</code> . Almost all databases use a period for the delimiter.

**jitterbit.target.db.transaction**

<b>Variable</b>	<code>jitterbit.target.db.transaction</code>
<b>Data Type</b>	Boolean
<b>Default</b>	true
<b>Description</b>	Set to <code>false</code> if you don't want database changes to be made in a transaction. The default is to commit all of the database changes in a transaction.

**jitterbit.target.file\_share.create\_directories**

<b>Variable</b>	<code>jitterbit.target.file_share.create_directories</code>
<b>Data Type</b>	Boolean
<b>Default</b>	false
<b>Description</b>	Set to <code>true</code> if you want file share target directories to be checked and created. If the directory is known to exist then don't use this option.

**jitterbit.target.ftp.transfer\_timeout**

<b>Variable</b>	<code>jitterbit.target.ftp.transfer_timeout</code>
-----------------	--

<b>Data Type</b>	Integer
<b>Description</b>	Set the transfer time-out in seconds for FTP target transfers. The default transfer time-out for FTP is one hour (3,600 seconds). Set to zero to disable.

**jitterbit.target.http.form\_data**

<b>Variable</b>	<code>jitterbit.target.http.form_data</code>
<b>Data Type</b>	Boolean
<b>Description</b>	If set to <code>true</code> , each target file is uploaded using RFC 1687 form upload.

**jitterbit.target.http.form\_data.ContentType**

<b>Variable</b>	<code>jitterbit.target.http.form_data.ContentType</code>
<b>Data Type</b>	String
<b>Description</b>	If RFC 1687 form upload is used, this sets the Content-Type of the file.

**jitterbit.target.http.form\_data.filename**

<b>Variable</b>	<code>jitterbit.target.http.form_data.filename</code>
<b>Data Type</b>	String
<b>Description</b>	If RFC 1687 form upload is used, this sets the name of the uploaded file.

**jitterbit.target.http.form\_data.name**

<b>Variable</b>	<code>jitterbit.target.http.form_data.name</code>
<b>Data Type</b>	String
<b>Description</b>	If RFC 1687 form upload is used, this sets the name of the form.

**jitterbit.target.http.max\_redirs**

<b>Variable</b>	<code>jitterbit.target.http.max_redirs</code>
<b>Data Type</b>	Integer
<b>Default</b>	0
<b>Description</b>	Maximum number of redirects to follow when using an HTTP target. Set this variable to a negative number to follow any number of redirects. The default is 0 (zero); no redirects are followed.

**jitterbit.target.http.remove\_trailing\_linebreaks**

<b>Variable</b>	<code>jitterbit.target.http.remove_trailing_linebreaks</code>
<b>Data Type</b>	String
<b>Description</b>	If set to <code>true</code> , trailing line breaks will be removed before the target data is posted.

**jitterbit.target.http.response.header.HeaderName**

<b>Variable</b>	<code>jitterbit.target.http.response.header.HeaderName</code> (shown in the script editor as <code>jitterbit.target.http.response.header.*</code> )
-----------------	--

<b>Data Type</b>	String
<b>Description</b>	<p>Set when handling a hosted web service post or an HTTP endpoint. HTTP response headers are not exposed, just the payload. When substituted by the name of an HTTP header for <i>HeaderName</i>, this variable holds the corresponding value.</p> <p>For example, the variable <code>\$jitterbit.networking.http.response.header.Proxy-Authenticate</code> would hold the value of the Proxy-Authenticate response-header.</p>


### `jitterbit.target.http.ssl_cert_id`

<b>Variable</b>	<code>jitterbit.target.http.ssl_cert_id</code>
<b>Data Type</b>	String
<b>Description</b>	<p>An identifier for a configuration entry in <code>jitterbit.conf</code> for the SSL certificate to use for HTTP targets. See also <a href="#">Customizations &gt; Client Certificates</a> and <a href="#">Adding Certificates to Keystore for Private Agents</a>.</p>

### `jitterbit.target.http.transfer_timeout`

<b>Variable</b>	<code>jitterbit.target.http.transfer_timeout</code>
<b>Data Type</b>	Integer
<b>Description</b>	<p>Set the transfer time-out in seconds for HTTP target transfers. The default transfer time-out for HTTP is one hour (3,600 seconds). Set to zero to disable.</p>

### `jitterbit.target.sftp.ssh_key_id` and `jitterbit.source.sftp.ssh_key_id`

<b>Variable</b>	<code>jitterbit.target.sftp.ssh_key_id</code> and <code>jitterbit.source.sftp.ssh_key_id</code>
<b>Data Type</b>	String
<b>Default</b>	" "
<b>Description</b>	<p>For configuring multiple SSH keys. By default, these are both empty, meaning the configuration in the [SSH] section of the <code>jitterbit.conf</code> configuration is used for all SFTP sources and targets with no password defined:</p> <pre>[SSH] PublicKeyFile='Z:/ssh/rsa2048.pub' PrivateKeyFile='Z:/ssh/rsa2048' PrivateKeyPassphrase=secret</pre> <div style="border: 1px solid #ffc107; padding: 5px; margin: 10px 0;"> <p> <b>NOTE:</b> When editing the configuration file, note that a line that is commented out with a leading # must not be left blank, or the next line will be read as a comment even if it does not begin with #.</p> </div> <p>If either of these variables is set to a string (such as "RSA"), then these configuration options will be used for a source or target (depending on which variable is defined):</p> <pre>PublicKeyFileRSA='Z:/ssh/id_rsa.pub' PrivateKeyFileRSA='Z:/ssh/id_rsa' PrivateKeyPassphraseRSA=verysecret</pre> <p>This allows the picking of different key sets when using public/private SSH key authentication for different SFTP sources and targets.</p>

### `jitterbit.target.text.character_encoding`

<b>Variable</b>	<code>jitterbit.target.text.character_encoding</code>
-----------------	---

<b>Data Type</b>	String
<b>Description</b>	Specify the character encoding for target text documents. If no encoding is specified, Latin-1 (ISO-8859-1) is assumed. For supported encoding, see <a href="#">Character Encoding</a> .

**jitterbit.target.wave.json**

<b>Variable</b>	<code>jitterbit.target.wave.json</code>
<b>Data Type</b>	String
<b>Description</b>	Used to override the JSON in a Salesforce Einstein (Wave) JSON metafile. See <a href="#">Overriding JSON Metadata in Salesforce Einstein Analytics</a> .

**jitterbit.target.xml.include\_empty\_xml**

<b>Variable</b>	<code>jitterbit.target.xml.include_empty_xml</code>
<b>Data Type</b>	Boolean
<b>Default</b>	<code>false</code>
<b>Description</b>	If a mapping to a target node results in an empty value, include the node. The default is to exclude this node. Setting this option to <code>false</code> removes the node from the target XML. Setting this option to <code>true</code> includes the node in the target XML.

**jitterbit.target.xml.include\_null\_xml**

<b>Variable</b>	<code>jitterbit.target.xml.include_null_xml</code>
<b>Data Type</b>	Boolean
<b>Default</b>	<code>true</code>
<b>Description</b>	If a mapping to a target node results in null or an empty value, include the node but make it nil. The default is to include this node. Setting this option to <code>false</code> removes the node from the target XML.

**jitterbit.target.xml.nsprefix**

<b>Variable</b>	<code>jitterbit.target.xml.nsprefix</code>
<b>Data Type</b>	String
<b>Default</b>	<code>"NS"</code>

<b>Description</b>	<p>The default XML namespace prefix is "NS". If your XML requires a different namespace prefix, this variable can be used to override the default namespace prefix with your existing namespace prefix name.</p> <p>For example, with the default XML namespace, your XML target output may look like this:</p> <pre>&lt;ns:Body xmlns:ns="http://www.marketo.com/mktows/"&gt; &lt;ns:successGetMultipleLeads&gt; &lt;result&gt;</pre> <p>When you set this global variable equal to a different prefix, for example <code>\$jitterbit.target.xml.namespace="tp"</code>, then the output in this example would change to this:</p> <pre>&lt;tp:Body xmlns:tp="http://www.marketo.com/mktows/"&gt; &lt;tp:successGetMultipleLeads&gt; &lt;result&gt;</pre>
--------------------	---

**jitterbit.target.xml.num\_for\_bool**

<b>Variable</b>	<code>jitterbit.target.xml.num_for_bool</code>
<b>Data Type</b>	Boolean
<b>Default</b>	<code>true</code>
<b>Description</b>	If set to <code>true</code> , XML boolean target values will be represented with "0" or "1" for false and true values respectively. This is the default so to get the string values "true"/"false" instead you need to set this parameter to <code>false</code> .

**jitterbit.target.xml.prettify**

<b>Variable</b>	<code>jitterbit.target.xml.prettify</code>
<b>Data Type</b>	Boolean
<b>Default</b>	<code>true</code>
<b>Description</b>	If set to <code>true</code> , XML targets are written with line breaks and tab indentation. This is easier to read but the resulting file will be larger.