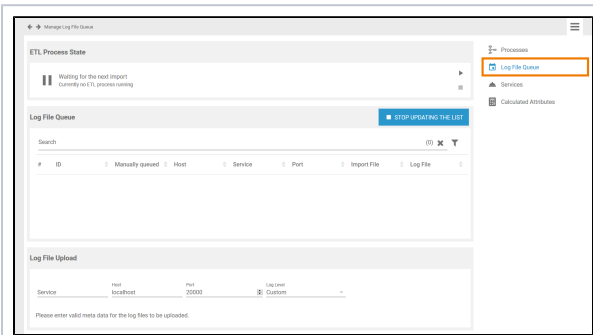


Monitoring the ETL Process State



The **Log File Queue** menu allows you to monitor the log file import from your services to Process Mining.

On this Page:

- [Starting and Stopping the ETL](#)



You have to other options in the **Log File Queue** menu:


- Go to [page Uploading Log Files](#) to learn how to upload log files manually.
- Go to [page Monitoring the Log File Queue](#) for further information about the contents of the log file queue.

Pages:







- [Adminstrating Process Mining](#)
 - [Collecting Logs](#)
 - [Uploading Log Files](#)
 - [Adding Diagrams](#)
 - [Setting a Retention Time for Instance Data](#)
 - [Using Calculated Metrics](#)
 - [Monitoring the Log File Queue](#)

Documentation:

- [Process Mining Installation Guide](#)
- [Process Mining BRIDGE Integration Platform User's Guide](#)
 - [Transaction Log Levels of an xUML Service](#)

<div>ETL Process State</div> <div>  Importing data ID: 10, File: transaction_2020-03-31.log </div>	<p>Section ETL Process State shows the current state of the ETL service.</p> <p>The following states can occur:</p> <table> <tr> <th>State</th><th>Ongoing Action</th></tr> <tr> <td>Deleting</td><td>The ETL is deleting obsolete data.</td></tr> <tr> <td>Importing</td><td>Data is currently imported.</td></tr> <tr> <td>Loading</td><td>Status of the service is requested.</td></tr> <tr> <td>Running</td><td>The ETL processes the import data.</td></tr> <tr> <td>Stopped</td><td>The import process is completed.</td></tr> <tr> <td>Unavailable</td><td>The service is not available.</td></tr> <tr> <td>Waiting</td><td>The ETL service is waiting for the next import.</td></tr> </table>	State	Ongoing Action	Deleting	The ETL is deleting obsolete data.	Importing	Data is currently imported.	Loading	Status of the service is requested.	Running	The ETL processes the import data.	Stopped	The import process is completed.	Unavailable	The service is not available.	Waiting	The ETL service is waiting for the next import.
State	Ongoing Action																
Deleting	The ETL is deleting obsolete data.																
Importing	Data is currently imported.																
Loading	Status of the service is requested.																
Running	The ETL processes the import data.																
Stopped	The import process is completed.																
Unavailable	The service is not available.																
Waiting	The ETL service is waiting for the next import.																

Starting and Stopping the ETL

<div data-bbox="164 159 349 184">ETL Process State</div> <div data-bbox="206 245 237 285"></div> <div data-bbox="259 243 503 289"><p>Waiting for the next import Currently no ETL process running</p></div> <div data-bbox="662 210 721 321"></div>	<p>In section ETL Process State administrators have also the option to start and stop the current action of the ETL service:</p> <ul style="list-style-type: none">• Press  to (re-)start the ETL service.• Press  to stop the ETL service. The service will stop when the current import is finished.
<div data-bbox="164 1253 266 1268">ETL Process State</div> <div data-bbox="178 1297 209 1327"></div> <div data-bbox="214 1297 678 1327"><p>Import is finished An error occurred: ERL_NO_REFERENCED_ROW_2: Cannot add or update a child row: a foreign key constraint fails</p></div> <div data-bbox="688 1281 727 1344"></div>	<p>If an error occurs during the import, this import process is terminated with an error.</p> <p>The erroneous import is <u>not discarded</u> to block the start of the next import. This forces the administrator to correct the error or its cause (for example the database down or else).</p>

In case an error occurred, two new buttons will appear in the **ETL Process State** window:

- Press **Retry**



to move the process back to the step where the error occurred. This is useful, for example, if the process encountered the error when deleting old data but the data should definitely be deleted before new data is imported.

- Press **Ignore**



to discard the erroneous import. A new import process is started with the next file waiting to be imported.