

Tungsten Replicator Master Class

Intermediate: Filter Like a Pro!

Chris Parker

VP of Customer Success, EMEA

Topics

In this short course, we will discuss filtering:

- Recap replication stages and pipelines
- Look at available filters
- Example configurations and use cases
- Enabling filters
- Custom filters

The Replicator

Tungsten Replicator Stages

Extractor

Each stage passes through:

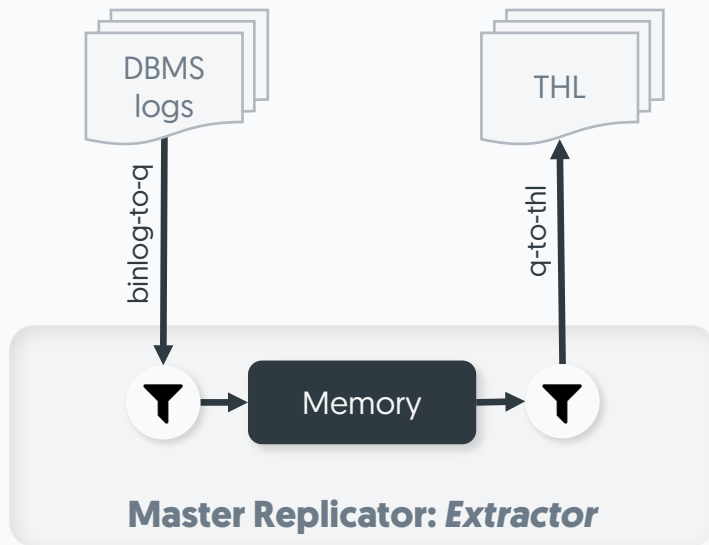


`binlog-to-q`

Handles extraction of transactions from binary log and places into in-memory queue

`q-to-thl`

Takes transactions from in-memory queue and writes down to disk into thl log file



Tungsten Replicator Stages

Applier

`remote-to-thl`

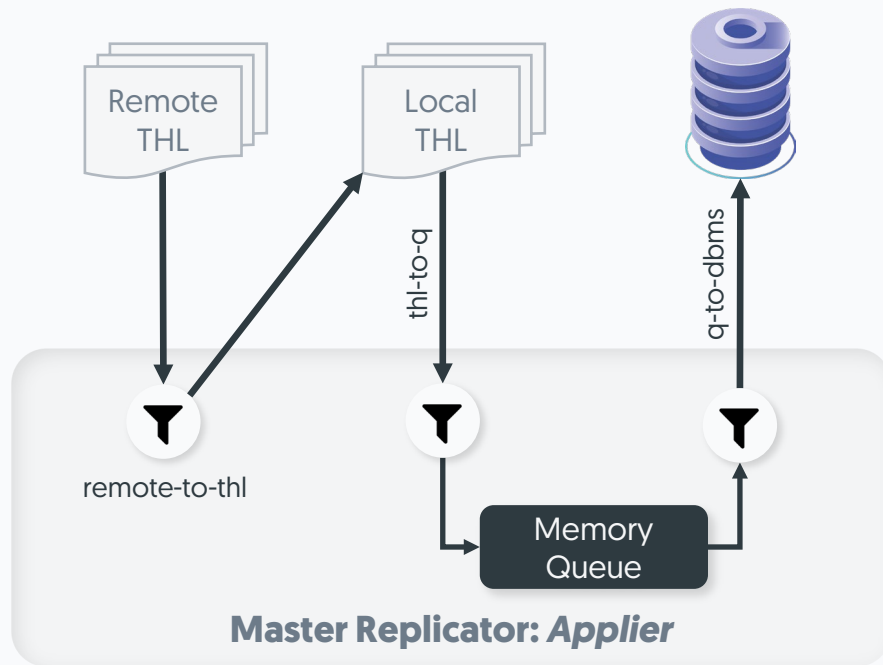
Streams thl log files across network socket, from configured source. Writes out THL to local disk.

`thl-to-q`

Reads transactions from thl and places into in-memory queue

`q-to-dbms`

Reads transactions from in-memory queue and writes out to target database



Filtering

Filtering

- Remove THL entries
 - For example, filter by schema/table name
 - Filter by data
 - Filter by type (statement, row, DDL, DML)
- Change Content
 - Remove a row or column
 - Change the contents of rows – for example string formats, or
 - Change the datatypes of rows – for example, an Int into Float, SET types into strings, or date types
 - Data Masking (Coming soon)
- Add Data
 - Introduce entirely new transactions
 - Insert new data
 - Duplicate data into other tables

colnames

- Adds column names to THL
- Essential for Heterogeneous Environments and pre-requisite for some filters
- Can increase size of THL on disk
- No config required, just enable at binlog-to-q stage

replicate

- Filters out at schema or table level
- .do = Only replicates objects listed
- .ignore = Replicates everything except objects listed
- Can be applied at extract or apply stage
 - Think carefully! If filter at extract – gone forever! But will reduce THL on disk

dropcolumn

- Filters out at column level
- Apply at Extract or Apply stage
- Requires colnames filter!
- Requires additional JSON configuration file

droprow

- Filters out data at row level
- Apply at Extract or Apply stage
- Requires colnames filter!
- Requires additional JSON configuration file

rename

- Remaps object names between source/target
- Remaps Schema, Table and even columns
- Requires colnames filter for column remapping
- Require additional CSV configuration file

rowaddddbname

- Adds a new column to THL containing schema name
- Useful for Fan-In topologies
- Requires colnames filter

Custom Filters

- You can write your own filters!
 - JavaScript
 - Full templates and examples available online
 - <https://www.continuent.com/resources/docs/latest/tr/filters-javascript.html>
- Full information along with a list of all available filters in our documentation:
 - <https://www.continuent.com/resources/docs/latest/tr/filters.html>

Summary

What we have learnt today:

- Reviewed replication stages and pipelines
- Looked at available filters
- Viewed configurations and use cases
- How to enable filters
- Custom filters

Thank you for listening

continuent.com

Chris Parker

VP of Customer Success, EMEA