CISCO™

MATE Live Configuration Guide

Release 6.1
February 2015

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number:

CONTENTS

Overview

This guide is for administrators who are configuring and managing the MATE Live application. You must have an admin role assigned to configure MATE Live from the web UI.

- Datastore—Describes how to install, back up, and restore a MATE Live datastore.
- General Settings—Describes how to configure MATE Live to get plan files from the Collector server, Continuous Poller server, or an external archive, and how to set default interface types to use in reports.
- Map Settings—Describes how to configure the Map page, including its panels.
- User-Defined Properties—Describes how to create user-defined properties for use in Explore and Analytics.

Network Data

MATE Live data is collected from Collector, which can include the creation and addition of a simulated demand traffic matrix. The data is stored in *plan files*. MATE Live uses two internal mechanisms, a time-series optimized datastore and an internal archive, to store this data and make available for use in the application. The Map component shows the visualization of these plan files in a weathermap layout. The Explore and Analytics components access the datastore.

To enhance the visualization of these plan files, you can create and edit multiple plan file *layouts* using the MATE GUI. After saving the plan file as a template back to the MATE Live server, it is applied to the most recently collected plan file and stored for use in the Map.

Related Topics

- *WAE Platform Configuration Guide*
- *System Administration Guide*
- *MATE GUI Visualization Guide*
- *MATE Integration and Development Guide*
- *MATE Live User Guide*
- *Table Schema and CLI Reference*

CHAPTER 2

Datastore

This chapter describes how to install a new MATE Live datastore, how to back up a datastore, and how to restore it. If you need to migrate a MATE Live datastore from 5.4 or a prior release, or if you need to upgrade a 6.1 MATE Live datastore, contact your support representative.

Many references in the documentation explicitly identify directories in which the software is installed. Otherwise, references to where the software resides are as follows.

- `$CARIDEN_ROOT`—Location of both the MATE installation.

  If the defaults were used during installation, then `$CARIDEN_ROOT` is the same as `/opt/cariden`.

- `$CARIDEN_HOME`—Sub-directory of `$CARIDEN_ROOT` that contains the MATE software.

  If the defaults were used during installation, then `$CARIDEN_HOME` is the same as `/opt/cariden/software/mate/current`.

New Datastore Installation

From the `$CARIDEN_HOME/bin` directory, use the `mld` tool to install a new MATE Live datastore. This tool installs both the mld server and an empty datastore directory.

- The `-size` option is required. The values are D (demo), S (small), M (medium), and L (large). For assistance in which value to use, contact your support representative.
- Best practices
  - Use the `-backup` option to change the default backup directory to one that is on a different physical disk.
  - For better performance, create a separate ext2 partition for the directory specified with the `-datastore` option.

The following command uses default directories that are listed in Table 2-1. To change any of the defaults, use the options as described in this table.

Minimal requirement: `mld -action install -size <D,S,M,L>`

Best practice: `mld -action install -size <D,S,M,L> -backup <backup_directory>`

mld Options

Table 2-1   mld Options

Options	Description	Default
-version	Display the datastore version	
-action	start —Start the mld server. Note that to start the mld server, you must use the same username as when you installed it. stop —Stop the mld server status —Show the status of the mld server restart —Stop and then restart the mld server install —Install a new mld server and datastore, and start the mld server. upgrade —Update an existing mld server and start the mld server. If you have a previous MATE Live version and are installing a new version, you must run mld -action upgrade.	Default installation directory $CARIDEN_ROOT/software/mld/current
Use only with -action install		
-size=[DSML]	Size of the datastore. Required if using -action install.	
-mldata <directory>	Directory where all application data is stored. This includes the datastore, report output, and other application data.	$CARIDEN_ROOT/data/mldata
-datastore <directory>	Directory where the datastore is initialized. Once set, this directory cannot be changed. You can, however, use symbolic links.	$CARIDEN_ROOT/data/mldata/datastore
-cpus <#>	Number of CPUs reserved for the datastore and the mld server. If using -size D, this must be set to 1.	Half of the total CPUs
Use only with -action install or -action upgrade		
-mld <directory>	Directory where the mld server is installed. Once set, this directory cannot be changed. You can, however, use symbolic links.	$CARIDEN_ROOT/software/mld/current
-backup <directory>	Directory for saving datastore backups. You can override this for a single backup. See the Backups section.	$CARIDEN_ROOT/data/mldata/backup

Backups

Back Up a Datastore

MATE Live backs up the time-series derived data from plan files. It does not back up transaction logs or other MATE Live data, such as application data and report data.

Backups require approximately 1 TB of space.

Best Practices

- Perform the backup to a different disk drive, or copy the backup to a different physical device once you finish the backup.

- Perform backups outside of peak traffic hours.

- Use a backup directory that is on a different physical disk, and set this when you first install the mld server and datastore. Doing so sets the default backup directory for all backups.

  ```
  mld -action install -backup <backup_directory>
  ```

- The backup process makes a copy of the datastore, but it does not back up other MATE Live data, such as application data and report data. Therefore, with some regularity, copy this other data to a safe location, such as to a different physical disk.

- Perform a full backup at least weekly or monthly, with numerous incremental backups in-between them.

- Rather than running manual backups, call `ml_backup` from a cron job.

- Perform only one backup at a time so that their schedules do not overlap. Ensure there is at least one hour between each backup. Once completed, verify that the backup was completed within the hour.

Backup Steps

> **Note** Keep both the datastore (`mld`) and the `wae-web-server` running.

You can execute `ml_backup` to run a manual backup at any time.

The default backup directory is `$CARIDEN_ROOT/data/mldata/backup` unless this was changed when installing the datastore.

The `ml_backup` tool enables you to perform multiple levels of backups to save disk space, though the first time using these levels, you must perform backups in this sequence.

Sequence	Enter	Description

1	`ml_backup`
	or
	`ml_backup -L 0`
		Back up everything.

Sequence	Enter	Description
2	`ml_backup -L 1`	Back up everything since the most recent level 0 backup was performed.
3	`ml_backup -L 2`	Back up everything since the most recent level 1 backup was performed.

To run a backup using all defaults, you need only enter the following. This uses the default backup directory, and creates a full backup.

```
ml_backup
```

- To override the default backup directory, use the `-directory` option.
- To set a different backup level, use the `-L` option.

Example: This example sets the backup directory to `$CARIDEN_ROOT/data/matelive/backups` and backs up only data that is new since the last level 0 backup was run. This assumes that you have run `ml_backup` one time using the default level of 0.

```
ml_backup -directory /data/matelive/backups -L 1
```

Additional Backups

A best practice is to back up the following directories by making copies of them. By default, these are located in `$CARIDEN_ROOT/data/mldata`. Otherwise, they are stored in the directory specified upon installation (`mld -mldata <directory>`).

> **Note** Before copying, stop the mld server using `mld -action stop`. Remember to immediately restart the mld server when you are finished copying using `mld-action start`.

- archive—Stores the template (template.pln) and plan files used in the Map. Note that files sizes can be quite large, depending on the size of the network and how long the collection has been running.
- appdata—Application data, such as report definitions, user parameters, and report history.
- datastore—MATE Live datastore, which contains network measurements accumulated over time.
- jobs—Error log files for plan file insertions.
- plans—Queue of plan files waiting to be inserted.
- reports—Every report that can be run caches its output in this directory for quick retrieval.

Additionally, it is a best practice to make copies of the following.

- `config.xml` file, which is located in one of these three places: `~/.cariden/etc/config`, `$CARIDEN_ROOT/etc/config`, or `$CARIDEN_HOME/etc/config`
- `$CARIDEN_HOME/etc/user_manager`, which is the directory in which users are defined

Restore a Datastore

> ✎
>
> **Note**   This section references collection methods. For information on collecting data from the Collector Module, see the *WAE Platform Configuration Guide*.

Prerequisites

- To restore a datastore, you must have a backup of it. See the Backups section.

- Ensure you have a proper disk and disk space. For example, if your data was corrupted, you would need a new disk. If the restoration is due to a space issue, add more space to the existing disk.

- If you have a single-device configuration, the collection of data will be interrupted during the restoration of a MATE Live datastore. Note that this affects the MATE Design Archive application if you are running it in addition to MATE Live.

- If the backup datastore resides on a different device, ensure the following pre-requisites are met.

    - The username and user ID (uid) of both devices must be the same.

    - The backup datastore name uses a hostname as a portion of its name. This hostname portion of the backup datastore name must be changed to be the same as the hostname on the device to which it is being restored.

       Example: The backup datastore name is akdobi.acme.com_1_L0. The hostname on the device on which the datastore is being restored is akgudei.acme.com. In this case, change the backup datastore name to akgudei.acme.com_1_L0.

Restoration Steps

Figure 2-1   Datastore Restoration Workflow
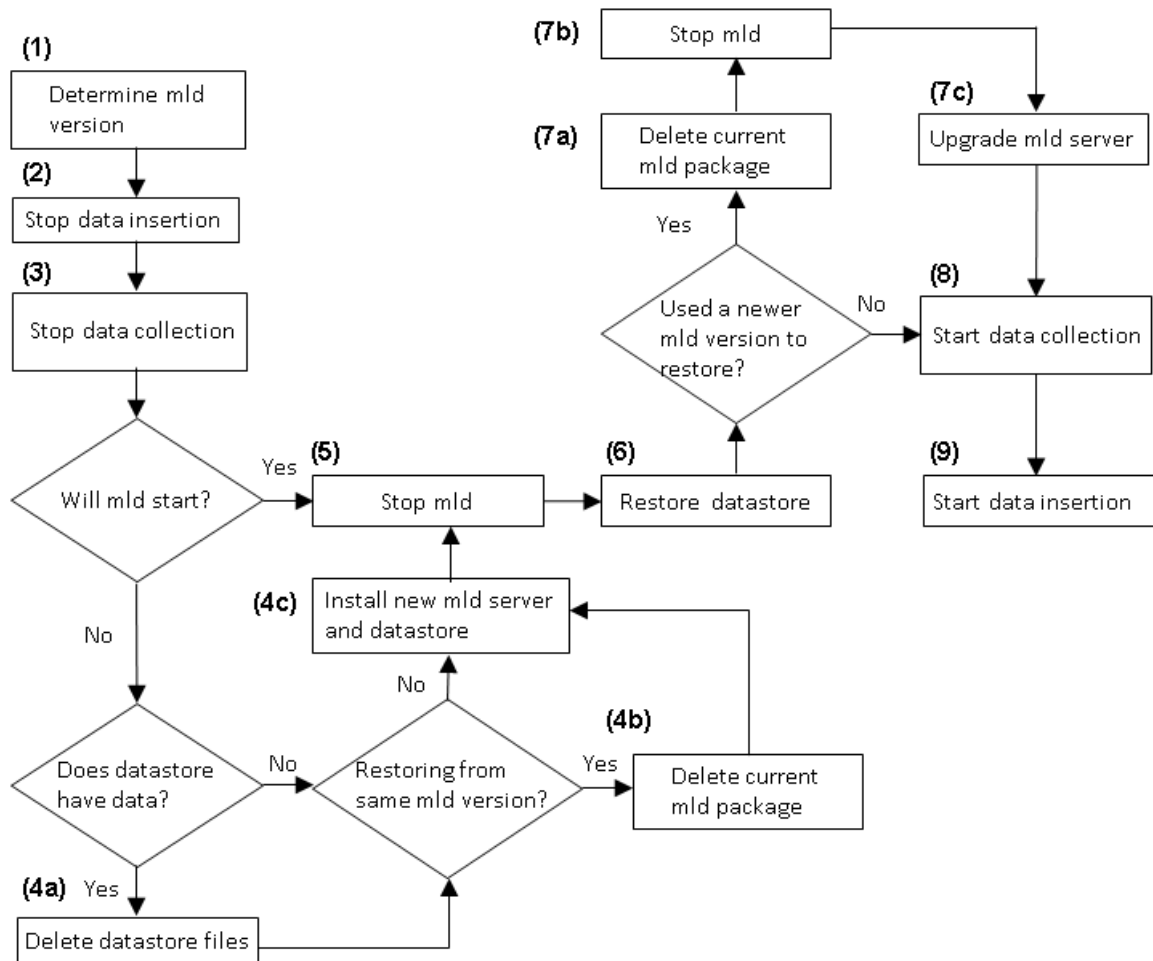
Step 1   Determine the version of the mld software package. If you have only one version, it is listed in `/opt/cariden/software/mld`. If you have multiple versions, determine which one is currently being used by entering the following from `/opt/cariden/software/mld`.

```
ls -l | grep current
```

Step 2   Stop the insertion of data.

- If using augmented or manual snapshots, disable the insertion using the following command within the snapshot. Note that currently running insertions continue until completion.

```
ml_insert_ctl -disable-scheduler
```

- If insertion is configured through the MATE Live UI, open this UI and stop the insertion from the Settings page. Set the Data Collection option to None, and click Apply.

Step 3   Stop the collection of data.

- If using augmented or manual snapshots, after the most recent snapshot has finished, stop further snapshots from running by disabling the cron job (using comments #). To determine if a snapshot has finished, check the system process table or check the log file, which by default is in $CARIDEN_ROOT/logs.

- If collection is configured solely through the Collector UI or if using augmented snapshots, open the Collector UI. Stop the collection from the Collection->Schedule page by clicking the Stop button. Then stop the wae-web-server.

```
service wae-web-server stop
```

If the mld server will start, skip to step 5.

Step 4   **If the mld server will not start**, follow these steps.

- **a.** **If you have data**, delete the data files. If there is no data, such as when restoring to a new disk, skip to step 4b.

```
rm -rf /opt/cariden/data/mldata/datastore/*
```

- **b.** **If using the same mld version that was used to generate the restored datastore**, delete the current mld package. If using a newer mld version than was used to generate the datastore, skip to step 4c.

```
rm -rf /opt/cariden/software/mld/<package_name>
```

- **c.** Install a new mld server and datastore. This action also starts the mld server.

```
mld -action install -size <D,S,M,L>
```

Step 5   Stop the mld server.

```
mld -action stop
```

Step 6   Restore the datastore data.

- If you used the default backup directory when setting up the backups, you do not need to give -directory a value.

- If you did not use the default backup directory, the -directory value must be the same as configured when installing the datastore or the same as configured in the config.xml file (in $CARIDEN_ROOT/etc/configs).

```
ml_restore -directory <backup_datastore_directory>
```

Step 7   **If you used a newer mld version than was used to generate the datastore**, follow these steps.

- **a.** Delete the current mld package.

```
rm -rf /opt/cariden/software/mld/<package_name>
```

- **b.** Stop the mld server.

```
mld -action stop
```

- **c.** Upgrade the mld server. See Table 2-1 for information on the two available options: -mld <directory> and -backup <directory>.

```
mld -action upgrade
```

Step 8   Restart the collection of data.

- If using augmented or manual snapshots, test the snapshot process by running it as a single instance.

  Augmented: `snapshot -config-file /opt/cariden/etc/snapshot_augment_collector.txt`

  Manual: `snapshot -config-file /opt/cariden/etc/snapshot.txt`

  If you are satisfied that the data is valid, restart the cron job. To determine if a snapshot has finished, check the system process table or check the log file, which by default is in `$CARIDEN_ROOT/logs`.

- If collection is configured solely through the Collector UI or if using augmented snapshots, start the `wae-web-server`.

  ```
  service wae-web-server start
  ```

  Open the Collector UI, and start the collection from the Collection->Schedule page.

Step 9   Restart the insertion of data.

- If using augmented or manual snapshots, enable the insertion using the following command within the snapshot.

  ```
  ml_insert_ctl -enable-scheduler
  ```

- If insertion is configured through the MATE Live UI, open this UI and stop the insertion from the Settings page. Set the Data Collection option to the appropriate server or archive, and click Apply.

Related Topics

- General Settings
- Map Settings

General Settings

Access: MATE Live->Settings and select General Settings tab

The General Settings page enables you to configure where the MATE Live data and templates are stored, how often to collect the data and from where, status of the latest data set collected, and which interface types are used in reports.

Application Data

Use the Application Data section to configure where to put plan files so that MATE Live can extract them.

Required Fields

* Map Archive—The local directory in which collected plan files and the template file (template.pln) are stored for display in the Map. This field is required only if using the Map component.

    Note that MATE Live automatically creates the template.pln file from the most recently collected plan file if no template exists. To create the visual elements of a template file and save it to this location, use the MATE GUI. (For information, see the *MATE GUI Visualization Guide*.) If visual elements have not been applied, the plan file is visualized in the default, unprocessed, state.

Note If you are using an external archive for the collection method, the map archive must be a different location than that of the external archive.

Datastore Location

By default, the datastore is located in `$CARIDEN_ROOT/data/mldata/datastore`, where `$CARIDEN_ROOT` is `/opt/cariden` if the default location was used during installation. By default, the datastore data, which is used in the Explore and Analytics components, is `$CARIDEN_ROOT/data/mldata`. These locations are only configurable during a datastore installation or upgrade. For information, see the Datastore chapter.

Data Collection

This area identifies the source, frequency, and status of data collection. Data collection is the process of getting the plan files from Collector, storing them in the internal datastore (for Explore and Analytics features) and in the internal plan file archive (for the Map feature), and applying the template for visualizing the network in the weathermap.

Required Fields

You must select one of these source options.

- Server—Use a server to collect the data. This can be the Collector server or the Continuous Poller server. The server must be running.

  - Server URL—The location of the server. The format is https://<server_IP>:<port>.

    Example: https://172.23.123.456:8443

  - Username—An administrative username for the server.

  - Password—The password for the username of the server.

  For information on configuring collection through the Collector UI, including continuous polling, see the *WAE Platform Configuration Guide*.

  The Collector server username and password are set in the System->User Management page and require an admin role. For information, see the *System Administration Guide*. For information on changing the Continuous Poller username and password, see the `/opt/cariden/software/wae-collector/WAECollectorAuth_README.txt` file.

- External Archive—Select this option if you are collecting data through an augmented snapshot or a manual snapshot. The archive must exist, and it must be a different location than the map archive. For information on creating an archive and on configuring augmented or manual snapshots, see the *WAE Platform Configuration Guide*.

- None—Do not collect data. If you are collecting data through manual snapshots, and if you prefer to manually insert data into MATE Live, select this option. For manual snapshots, however, the recommendation is to use the External Archive option.

Additionally, you must identify the frequency to check for and receive plan files, and whether you are using the Continuous Poller server.

- Collection Frequency—The frequency, in seconds, in which MATE Live checks for a plan file. The recommended value is 30 seconds.

  - If using the Collector server, MATE Live receives the plan file at this interval only if it has been updated with changes.

  - If using the Continuous Poller server, MATE Live receives the plan file based on the Frequency setting.

- Continuous—Identifies whether to collect plan files from the Continuous Poller server or not.

- Frequency—The frequency, in minutes, in which MATE Live receives the plan file from the Continuous Poller server. MATE Live receives the plan file regardless of whether it has been updated. The value must be a factor of 60 (1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60). The recommended value is 15 minutes.

Optional Fields

- Check Status—Shows the status of the most recently collected data by MATE Live.

- Advanced Config—This is for advanced configuration editing only. Consult your support representative for assistance.

Reports

In this section, specify the default interface type showing on the Explore Interface page. These can then be used in reports on selected interfaces of one or more types.

- Individual physical interface—An interface that is not in a LAG and does not have an IP address. These interface names do not have a decimal.

    Example: so-0/2/2, g2/0/0

- Individual logical interface—An interface with an IP address that is not in a LAG. These names might have a decimal.

    Example: et-0/1/2.0, g2/0/0

- LAG logical interface—An interface with an IP address that contains member interfaces.

    Examples: ae0.0, Bundle-Ether1, Port-Channel7

- LAG physical interface—The physical representation of an aggregated interface to which a LAG logical interface (such as ae0.0) is assigned.

    Example: ae0 for Juniper interfaces

- LAG member interface—An interface that is in a LAG.

    Example: g2/0/0

- Unknown—None of the above interface types

Related Topics

- Datastore
- Map Settings
- User-Defined Properties

Map Settings

Access: MATE Live->Settings and select Map Settings tab

The Map Settings page enables you to configure the MATE Live Map panels, as well as set a default layout for the weathermap.

Configure Traffic Panel

The Traffic Widget section enables you to configure the Peering panel that appears on the Map page. This panel shows the highest traffic utilization per peer and per site. The far left column lists the peers, while each top column lists the associated sites. The icons in the panel represent the traffic flowing between the two.

For this panel to have data, you must configure the collection of BGP. For information on how to configure the collection of data, see the *WAE Platform Configuration Guide* if.
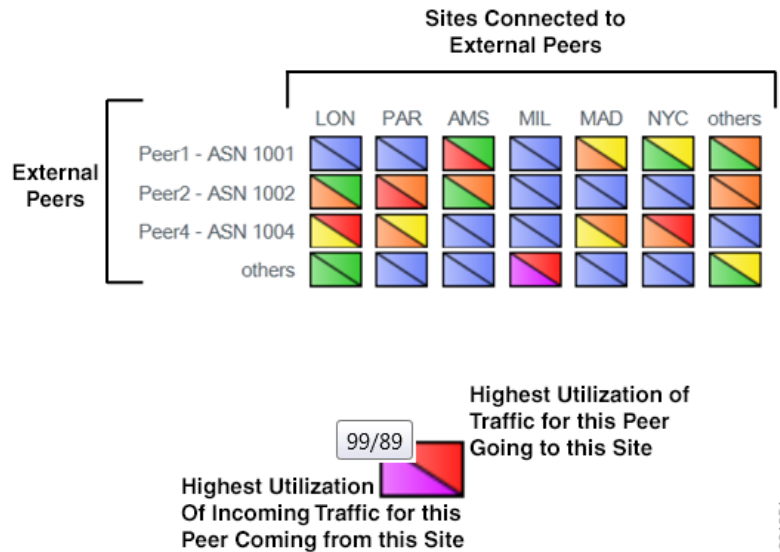
Traffic Widget Fields

- Enabled—If selected, the panel shows in the Map page. If not selected, the panel does not appear.

- Title—Name of the panel to appear on the Map page.

- Rows—Configure the ASNs (external peers) used in the panel. Their names appear on the left side of the panel. Each ASN represents a row in the panel.

  – Top ___ ASNs by Traffic Utilization—The ASNs with the highest traffic utilization. Enter a whole number 1-20.

– Except—To exclude specific ASNs that would be captured by the highest utilization, enter one or more ASNs separated by commas.

– Only These—To use only a specific set of ASNs, regardless of their traffic utilization, enter one or more ASNs separated by commas.

- Columns—Configure the sites used in the panel. Their names appear across the top of the panel. Each site represents a column in the panel.

– Top ___ Sites by Traffic Utilization—The sites with the highest traffic utilization. Enter a whole number 1-20.

– Only These—To use only a specific set of sites, regardless of their traffic utilization, enter one or more site names separated by commas.

- Advanced Config—This is for advanced configuration editing only. Consult your support representative for assistance.

Configure Events Panel

The Events Widget section enables you to configure which events appear on the Map page. This panel tracks a set of current events and the number of changes since the last plan file was received.

Configuration of this panel is a combination of this UI and the `default_timeplot_summary_format.txt` file located in the `$CARIDEN_HOME/etc/matelive` directory. (`$CARIDEN_HOME` is the location of where the MATE software package is installed, which is by default `/opt/cariden/software/mate/current`.)

For an event to appear, it must be configured in both locations. Combining the examples in this section would result in two events in the panel: Circuits down and Nodes down.

- `default_timeplot_summary_format.txt` file—Defines the metrics used for the events. Each metric is listed in the "Queries" section. For information on how to configure this file, see the comments at the end of the file, or refer to the Archive chapter in the *WAE Platform Configuration Guide*.

Example: This is an excerpt from the default file.

```
----------Queries----------
# Circuit Down
CircuitsDown SELECT Old_Name AS Details
CircuitsDown FROM Joined_Circuits
CircuitsDown WHERE ((Old_Name IS NOT NULL) AND (Name IS NULL)) OR ((Old_Active = 'T') AND
(Active = 'F'));

#Node Down
NodesDown SELECT Old_Name AS Details
NodesDown FROM Joined_Nodes
NodesDown WHERE ((Old_Name IS NOT NULL) AND (Name IS NULL)) OR ((Old_Active = 'T') AND
(Active = 'F'));
```

- Events Widget UI—This section lists the metrics and their display names. Each metric must match an event metric (query) that appears in the `default_timeplot_summary_format.txt` file.

Example:

Metric	Display Name
CircuitsDown	Circuits down
NodesDown	Nodes down

Events Widget Fields

- Enabled—If selected, the panel shows in the Map page. If not selected, the panel does not appear.

- Title—Name of the panel to appear on the Map page.

- Add Metric—Click to add a new event for display on the Map page.

  – Metric—The name of the event metric that appears in the `default_timeplot_summary_format.txt` file.

  – Display Name—The event name that appears in the panel on the Map page.

- Edit (pencil icon)—Edit the event metric name or the display name.

- Delete (trash can icon)—Delete a metric from the UI so that it no longer appears in the Events panel. There is no undo.

- Advanced Config—This is for advanced configuration editing only. Consult your support representative for assistance.

Related Topics

- `default_timeplot_summary_format.txt` configuration, see comments in the file or see the Archive chapter in the *WAE Platform Configuration Guide*

- Datastore

- General Settings

- User-Defined Properties

User-Defined Properties

Access: MATE Live->Settings and select User-Defined Properties tab

MATE Live acts on a set of objects (network elements) that are either discovered by Collector or added via the snapshot process. For example, interfaces are discovered while demands can be created as part of the snapshot. Each object has a set of default properties (attributes) that are tracked in and retrieved from the datastore. For instance, an LSP's Shortest TE Path, a node's CPU, or an interface's Traffic In are all properties that are collected and stored by default. For a complete list of objects and properties, see the *MATE Live User Guide*.

Additionally, you can create user-defined properties to fit your specific needs. These user-defined properties are defined by and thus, derive their values from other properties. For example, rather than tracking incoming traffic (Traffic In) and outgoing traffic (Traffic Out), you might need to track the ratio of incoming to outgoing traffic (Figure 5-1).

As with default properties, these user-defined properties are available in the Explore and Analytics pages.

Add Property Fields

Each user-defined property consists of an object, a user-specified name, and a user-created definition.

- Object—Select the object type to which the newly created property applies.
- Name—Enter a property name. Once the property is saved, this name appears as a selectable property for the given object in Explore and Analytics pages.
- Definition—Each definition is an expression using one or more default properties. A few rules apply, as follows.
  - Properties used in the definition must be default properties. That is, you cannot use user-defined properties in defining other user-defined properties.
  - Property names in the Definition field are not case-sensitive.
  - A user-defined property can be an integer, a double (floating point number), string, or boolean.
  - You can use any combination of the following in the definition. You can group any of these with parenthesis, which are executed in the order in which they appear.
    - Arithmetic expressions: +, -, *, /, and % (Figure 5-1)
    - Relational operators: >, <, =, <=, and >=
    - Logical operators: AND, OR, and NOT
    - Regular Expression Functions

- CASE Statements
- Edit (pencil icon)—Edit the user-defined property.
- Delete (trash can icon)—Delete a user-defined property. There is no undo.

- Preview—View the results that the object and definition together return.

Note For two-word properties, use one word in the definition. For example, for the Remote Node property, use "remotenode." For Traffic In and Traffic Out properties, use "traffin" and "traffout" in the definition.

Figure 5-1 **Example of a User-Defined Property Definition Using Simple Arithmetic Expression**

Regular Expression Functions

In the Definition field you can use regular expressions to filter results using regexp_extract and regexp functions.

regexp_extract

regexp_extract Syntax

regexp_extract(formatted_result, identifier1, pattern1, identifier2, pattern2, ...)

This searches a string with a list of regular expressions to produce a formatted string result compiled from back references. Back references are the results of capturing groups from the regular expressions and referencing them in order by $N, where N is a number that is greater than or equal to 1. This N identifies the captured string returned by a back reference in the order it appears in the matching pattern. At a minimum, only three arguments are needed for regexp_extract. You can use any number of pairs of attributes and expressions.

Example: In this example, both captured groups ($1 and $2) come from the regular expression for Description. The expression for Node does not have any capturing groups. It is used only as an additional filter. If any of the regular expressions fail to match, then the overall regexp_extract fails to match.

"regexp_extract("$1 ($2)", Description, ";interconnect to ([^\(]*) \(([^\)]*)\)", Name, "^(?!.*lo)")

regexp

boolean regexp Syntax
regexp(identifier1, pattern1)

This regexp syntax returns true, false, or null.

Example: If Node starts with 'palt', then sum the Traffic In and Traffic Out values. If the Node starts with something else, or is null, then no value is displayed.

case when regexp(node, '.*PALT.*') then traffin + traffout else null end

Example: If your node names include site names, this regexp syntax is useful for creating site properties for interfaces.

regexp_extract('$1', node, '([^-]+)-(.*)')

CASE Statements

The CASE statement allows conditional values based on a set of expressions. All result-expressions must be of the same type. That is, they can all be numeric or all be string, but they cannot be a mixture of both string and numeric.

CASE Statement Syntax
CASE WHEN <boolean-expression> THEN <result-expression>
[WHEN ...]
[ELSE <result-expression>]
END

Example: This example creates a user-defined property if a node name matches a remote node and if neither name is null.

case

when regexp_extract('$1', node, '(....).*\.(..)\..*') isnull then null

when regexp_extract('$1', remotenode, '(....).*\.(..)\..*') isnull then null

when regexp_extract('$1', node, '(....).*\.(..)\..*') = regexp_extract('$1', remotenode, '(....).*\.(..)\..*') then true else false end

Expression Syntax

Two syntaxes are used for the supported expressions: EBNF (Extended Backus–Naur Form) and a simpler query. Note that you can use one or multiple words for isNull and isNotNull.

- "isNull" can be represented by "is null"

- "isNotNull" can be represented by "is not null"

Simple Query

Following is the syntax for a simple query.

Simple Syntax
expression := value { +, -, *, /, %, IN, is [not] null, case } [expression]
when-expression := expression [{AND, OR, NOT } when-expression]
case := CASE WHEN when-expression THEN expression [ELSE expression] END
value := { object-attribute, number, string, regexp(), regexp_extract() }

EBNF

Table 5-1 is the grammar in EBNF format.

Table 5-1   SELECT Syntax in EBNF Format

SELECT Rule	Syntax
expression	case |
	logical
logical :=	relational {(OR, AND, SEMICOLON, COMMA) relational }
relational :=	additive {(<, <=, > , >=, =, ==, !=) additive }
additive :=	multiplicative {(+, -, ||) multiplicative }
multiplicative :=	unary {(*, /, %) unary }
unary :=	(!, not, ~, +, -) factor |
	factor isnull |
	factor

SELECT Rule	Syntax
factor :=	NUMERIC |
	STRING |
	NULL |
	TRUE |
	FALSE |
	ID |
	regexp |
	regexp_extract |
	in_expr |
	'(' expression ')'
regexp :=	REGEXP '(' literal COMMA STRING ')' |
regexp_extract :=	REGEXP_EXTRACT '(' STRING COMMA reg_arg_list ')' opt-as
reg_arg_list :=	literal COMMA STRING |
in_expr :=	IN '(' string_list ')'
string_list :=	STRING COMMA string_list
	STRING
case :=	CASE when when_list opt_else END
when_list :=	when when_list |
when :=	WHEN expression THEN expression
	empty
opt_else :=	ELSE expression |
	empty
opt_as :=	AS ID |
literal :=	ID | STRING

Related Topics

- General Settings
- Map Settings

MATE GUI and Remote MATE Live

Plan files and templates are stored in archive repositories, which can be installed on a different device than the one on which the MATE GUI is used. From the MATE GUI, you can extract MATE Live .pln plan files and templates from remote archive directories and make whatever modifications are necessary. You can re-insert them as template files into an archive directory used by the MATE Live application.

Note The recommendation is to open the remote plan file, modify it to create the visual template, and save it to the remote archive as a template. When uploading a template to a remote archive, you are overwriting the template that is in the archive.

Prerequisites

- You must have an admin role assigned to you through the web System UI, and you must know that username and password.

- To save templates to a MATE Live archive, that archive must first be configured through the MATE Live UI, Settings->General Settings page.

Step 1 In the MATE GUI, use the File menu to start the process.

- To download a MATE Live plan file or template from a remote archive, select the File->Open from->MATE Live menu.

- To upload a MATE Live template to a remote archive, select the File->Save to->MATE Live menu. You must have an administrative username and password to upload templates.

Location	
Server	172.14.4.180
Protocol:	HTTPS
Port:	8443
Username:	buda
Password:	••••••••••••••••

Step 2 After opening the dialog box, enter the hostname or IP address of the server.

Step 3 Identify how to connect to the server by selecting the appropriate protocol and entering the port number (for example, HTTPS 8443).

Step 4 Enter the username and password that gives you access to the server. Both are case sensitive. If you do not know the password, contact your system administrator. You must have an admin role assigned to you.

Step 5 Optional: Save the data you entered for future use.
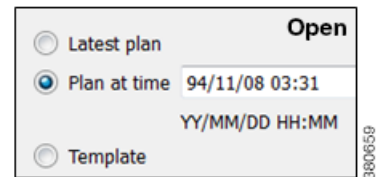- To save all settings except the password, select Save Settings.
- To save the password for future use, select Save Password.

Step 6   If saving a template, click OK and stop here. If opening a plan file or template, continue to step 6.

Step 7   If opening a template, select the Template option.

If opening a plan file, identify which one by selecting the latest or by entering a specific time using a YY/MM/DD HH:MM format.

Click OK.

Related Topics

- *MATE Integration and Development Guide*
- *Table Schema and CLI Reference*