NetYCE 5.3.1

> YCE 5.3.1 is a maintenance release of 5.3 and includes mainly fixes and minor enhancements.

Front-end enhancements

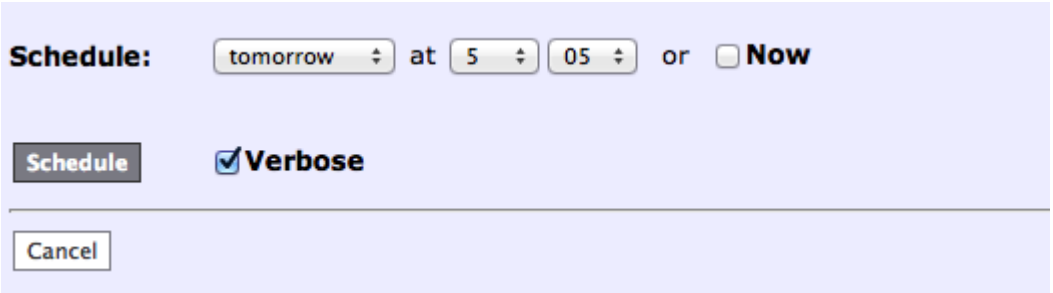
Scheduled jobs

Any job scheduled and listed under the scheduler tool has a number of job details associated with it. These include the scenario, the commands and optionally other details. These details have up to now only be accessible after the job had started from the “Job logs” tool.

To be able to review the scheduled changes and their scenario’s, the scheduler tool has been modified to link into these details.

By clicking the JobID the same details window opens as with the Job logs to review the job’s data but also its progress when it starts.

##2.2 Schedule for ‘now’## Scheduling a job to execute ‘now’ resulted in the job being scheduled at the first available slot in the queue starting at the first full minute-mark that that is a t least 10 seconds in the future. This resulted in a waiting time of 20 to 90 seconds.



The reason for the delay was mainly to allow the system time to create and schedule a large amount of jobs before the first actually had to start. Analysis learned however, that the ‘now’ option is rarely used when scheduling a large number of jobs. The most common occurrence is just one or two jobs.

The ‘now’ option has been re-implemented to start within a few seconds. The majority of jobs will now start between 5 and 7 seconds. The minimal delay of 5 seconds is for the system to create the job and schedule it. The actual schedule time depends on the queue settings. The job interval, normally set to 2 seconds, can be added to those initial 5.

For the ‘evpn’ queue (central core pushes) that interval is set at 20 seconds resulting in a job start delay of 5 to 25 seconds.

##2.3 Config diff## The Node configuration tools have been extended to include the new tool ‘Diff node config’. This tool allows the operator to compare the differences between two versions of a configuration from a node.

Strictly speaking, the tool allows the operator to compare the results of any two ‘show’ commands that were executed on the node using command jobs of scenarios. The results of any ‘show’, ‘ping’ or ‘dir’ command is stored by YCE for later retrieval. The ‘show config’ command is probably the most used.



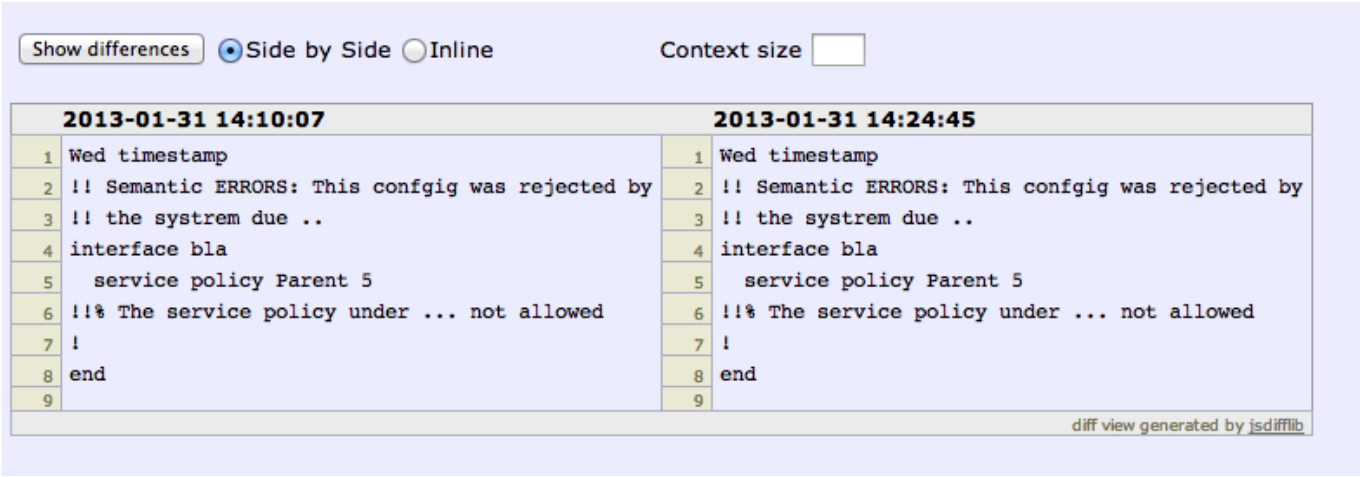
The ‘show startup configuration’ command is stored when the scenario includes the ‘Save_config’ task.

While typing a name of a node, the database is searched for matching nodes and presented for selection.

Click on ‘Get Commands’ after the nodename was selected or typed to see for which commands results were stored. The number of different stored results is shown between brackets.

Select one of more of these commands and click ‘Get Revisions’ to see when these results were collected.

By selecting two revisions from the list and clicking ‘Show Differences’, the two versions are compared and listed. The default format is ‘Site-by-side’ where both results are listed alongside and background colors indicate where differences are found.



The option ‘Inline’ shows one version with the same color coding highlighting the differences.

To find in a long configuration the few lines that actually were different, the option ‘Context size’ is very useful. This option defines the number of lines that will be shown around any block of differences. These lines just provide a few lines of context for the blocs where the differences were found.

##2.4 Command-jobs## Since command-jobs can have any type of command they can potentially be harmful. Previously these jobs could be executed on any number of nodes provided the user had ‘engineering’ level access to the node.

To allow the YCE manager more control over the scope of these command-jobs for engineering level users, a restriction has been built into the scheduler when submitting command jobs. This restriction limits the number of command jobs that can be scheduled at once. When the engineering level user submits up to five jobs, operation is granted and the changes will be scheduled as normal. No authorization is required.

But when the same user submits six or more jobs simultaneously, the behavior changes: he first receives a warning that the limit was reached and his jobs will be accepted but only in suspended state should he continue.



When he does continue, the jobs are scheduled and he is informed that approval is required before these jobs can be executed.

At that stage his jobs are in suspended state. The ‘Level’ column shows that at minimum a “modeler’ level user can approve these jobs.

The job list such a modeler (or manager) is presented with allows for the selection and “Resume” options that the approval consists of.

The limit of 5 nodes was not chosen arbitrarily feeling that it minimizes the risk of large scale disruptions while still providing a fair amount of freedom to execute average size changes. If the limit needs modification or needs to be disabled altogether, change the customization section of the file: /opt/yce/engineer/command_job.pl

```
#--- customization

# Lower-level (level<=3) users may submit 5 jobs at a time.
# when more jobs are submitted, all are suspended
# set to 0 if no limit applies
$auth_limit = 5;

#----
```

##2.5 Port configuration tool##

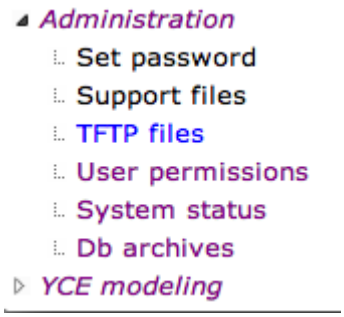
Previously, users with ‘operator’ level permissions could also modify port templates and port settings of interfaces with topology, potentially ruining the operation of the entire device, not just the interface.

This behavior was modified to allow these low-level operators to only modify the interfaces without

topology. The relevant options for these interfaces are now disabled. Generating the commands and updating these interfaces with the existing settings is still allowed though.

##2.6 Database restore##

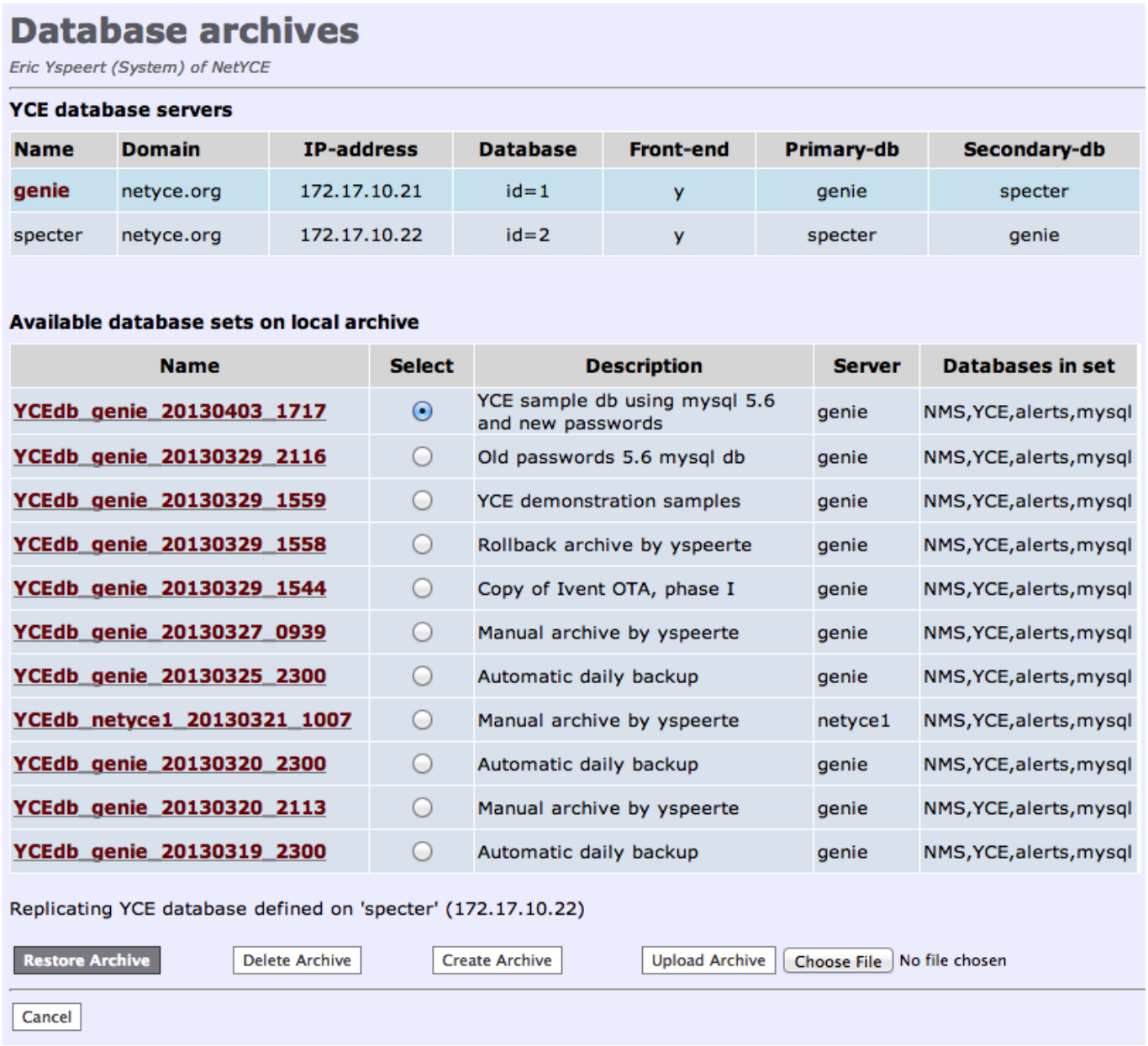
A new web-tool for the creation and restore of YCE database archive sets was created. It was added to the “Administration” section and required ‘manager’ level permissions.



This tool allows a manager to:

* create a new archive on the fly (although it will affect other users’ sessions), * download an archive for safekeeping, * upload an archive from another system, * review the daily automatic archives, * review content and test integrity of archives, * restore archives in full or partial, * setup database replication between two YCE database servers

Important. This tool currently has the limitation that is can only manipulate the LOCAL server YCE database. In architectures where the YCE database server is not running the web-front-end services, this tool cannot function. As a workaround, these services could be configured for this purpose without making them part of the general architecture.



Database archives

Eric Yspeert (System) of NetYCE

YCE database servers

Name	Domain	IP-address	Database	Front-end	Primary-db	Secondary-db
genie	netyce.org	172.17.10.21	id=1	y	genie	specter
specter	netyce.org	172.17.10.22	id=2	y	specter	genie

Available database sets on local archive

Name	Select	Description	Server	Databases in set
YCEdb_genie_20130403_1717	◉	YCE sample db using mysql 5.6 and new passwords	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130329_2116	○	Old passwords 5.6 mysql db	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130329_1559	○	YCE demonstration samples	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130329_1558	○	Rollback archive by yspeerte	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130329_1544	○	Copy of Ivent OTA, phase I	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130327_0939	○	Manual archive by yspeerte	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130325_2300	○	Automatic daily backup	genie	NMS,YCE,alerts,mysql
YCEdb_netyce1_20130321_1007	○	Manual archive by yspeerte	netyce1	NMS,YCE,alerts,mysql
YCEdb_genie_20130320_2300	○	Automatic daily backup	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130320_2113	○	Manual archive by yspeerte	genie	NMS,YCE,alerts,mysql
YCEdb_genie_20130319_2300	○	Automatic daily backup	genie	NMS,YCE,alerts,mysql

Replicating YCE database defined on 'specter' (172.17.10.22)

Restore Archive | Delete Archive | Create Archive | Upload Archive | Choose File | No file chosen

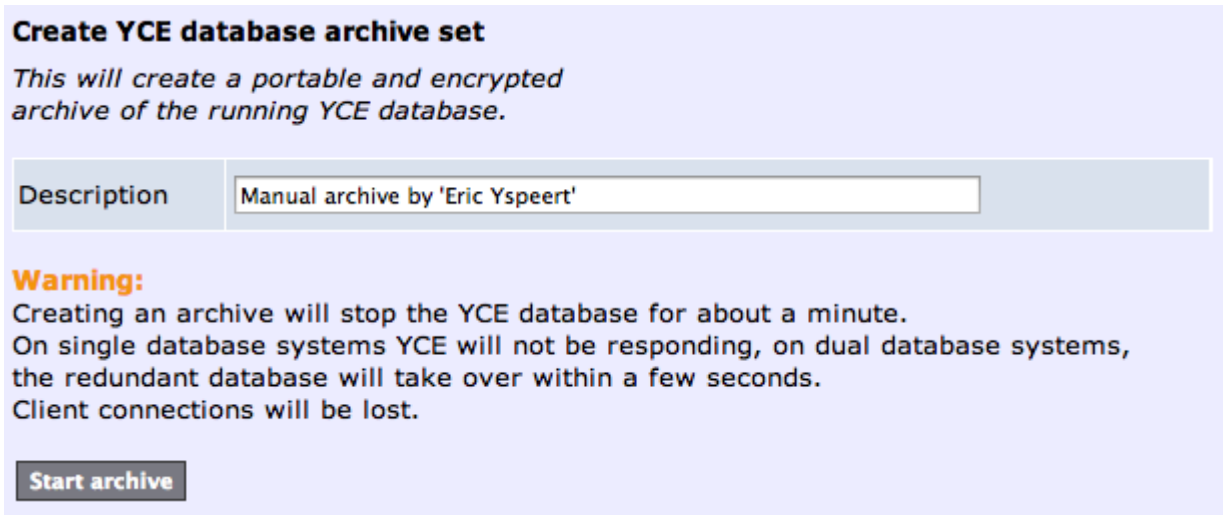
Cancel

When opening the tool, the database setup of the YCE architecture is displayed. This is for reference only. Below it, the list of current archives is shown. The list is maintained daily automatically by the archiving job (dbarchive.pl using ‘yce’ crontab). This job creates an archive and then removes the oldest archives exceeding 15 archives.

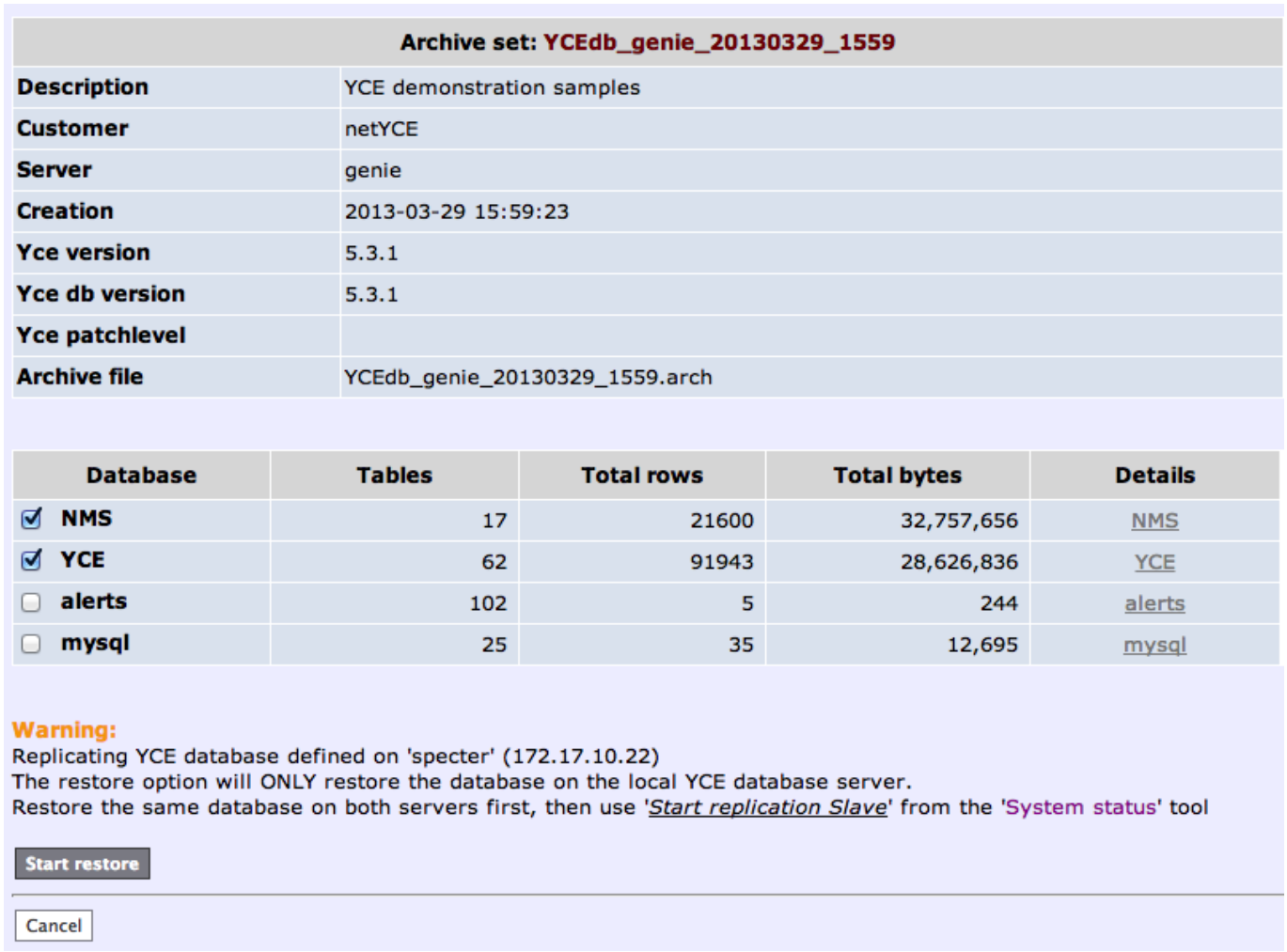
The list specifies the name of the archive based on the server name and the timestamp. A brief description states the cause for the archive. Automatic archives list “Automatic daily backup” by default. The server name is listed to show where the archive was created.

By clicking the archive name it can be downloaded. Special attention went into securing this archive. Apart from including a manifest listing the databases, sizes and record-counts, each database is encrypted. This encryption uses a triple-DES algorithm and a key that is based on the YCE license file of the customer. This ensures that archives cannot be accessed by any other than the rightful owner, the customer. And only on the YCE database server.

A manual archive is started by clicking the ‘Create Archive’ button. The manager can change the description message to reflect the reason for the archives creation. Please take note of the warning message since the database will be briefly unavailable.



To review the details of an archive, select the archive using the radio-button in the 'select' column and click the 'Restore Archive' button. The resulting page lists the archive details and a list of the included databases.



Archive set: YCEdb_genie_20130329_1559	
Description	YCE demonstration samples
Customer	netYCE
Server	genie
Creation	2013-03-29 15:59:23
Yce version	5.3.1
Yce db version	5.3.1
Yce patchlevel	
Archive file	YCEdb_genie_20130329_1559.arch

Database	Tables	Total rows	Total bytes	Details
☑ **NMS**	17	21600	32,757,656	NMS
☑ **YCE**	62	91943	28,626,836	YCE
☐ **alerts**	102	5	244	alerts
☐ **mysql**	25	35	12,695	mysql

A checkmark before the database name will restore it. It is recommended to include only the YCE and NMS databases unless instructed otherwise by NetYCE support.

Additional details on the archived database are shown when clicking on the name in the details column. It brings up a window with a list of the tables making up the database including the number

of records and their size in bytes (data only).

YCE details of archive set YCEdb_genie_20130329_1559		
Table name	**Table rows**	**Table bytes**
-- Totals --	**91943**	**28,626,836**
Action_log	14583	2,741,132
City_codes	4039	157,856
Client	31	2,928
Client_types	93	4,792
Config_log	70	341,640
Domain	56	14,624
Images	131	16,115,332
Ios_images	7	648
Ios_upgrades	2945	480,980
Ios_upgrades_13031903	2945	480,980
Ios_upgrades_13031906	2945	528,132

The databases selected can be restored to the local database server by clicking ‘Start restore’. A rollback archive will be created first so there is no need to do that manually. The restore will result in a cleanup of all database statuses and history. The replication logs, both master and relay, will be removed as will the replication setup itself (if any). The database will not be available to any user.

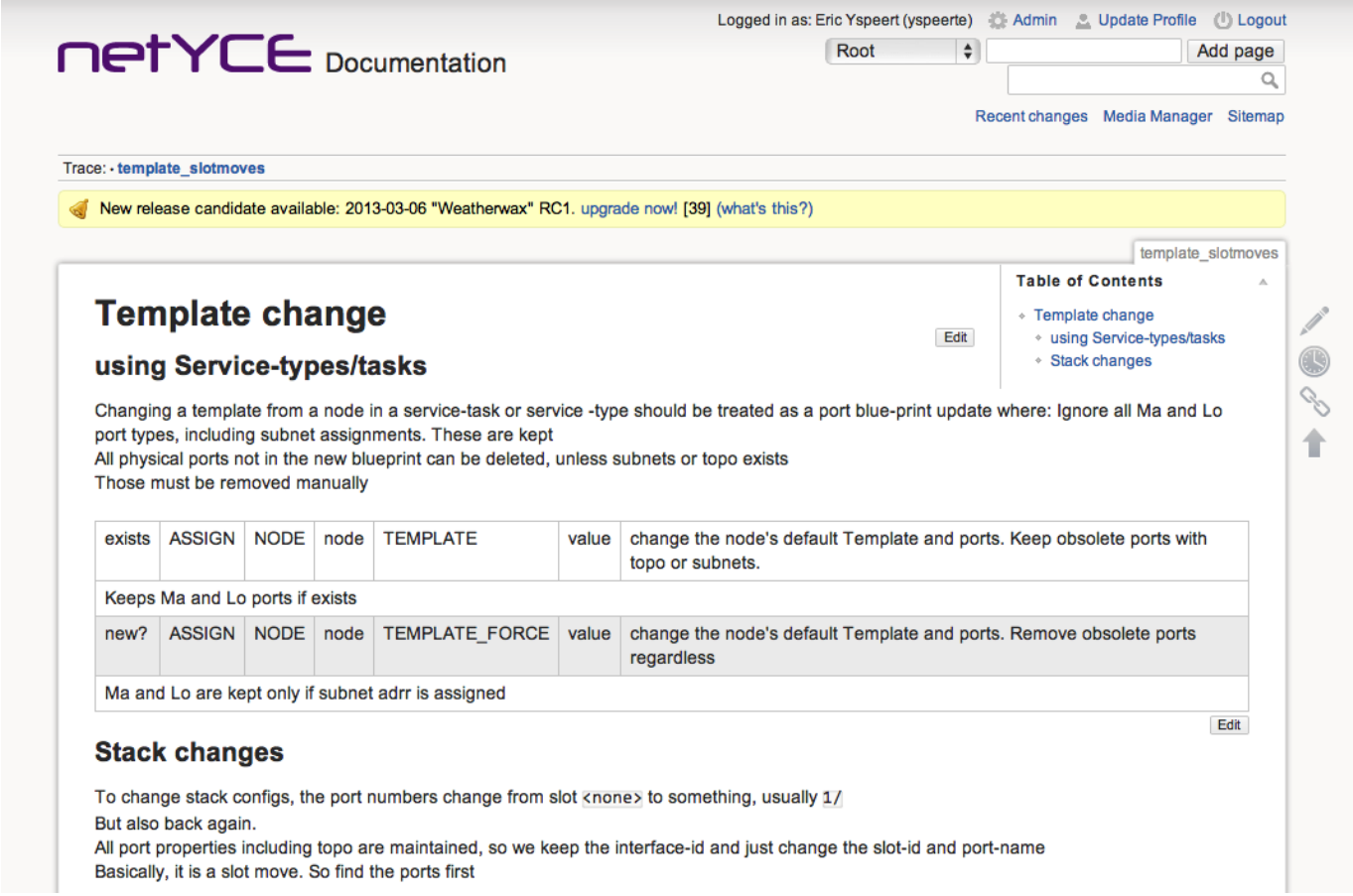
Once restored, the database is restarted and the replication is reconfigured (if setup for replication). But the replication is not started yet. This has to wait until both databases are restored. Even when one database is copied over (as an archive set) from a running YCE server and restored, it is a REQUIREMENT that both databases have that SAME set restored near simultaneously! Only when both are cleanly restored can de replication be started.

Replication is started using the “System status” tool by clicking the ‘Start Replication Slave” button for the database server. Each database replication needs to be started in succession. First start one end and monitor for several minutes if conflicts are reported.

Errors are flagged in the tool which also provides a 'Skip synchronization error' button for SQL errors causing synchronization conflicts. Reported SQL errors pertaining to the 'Server_setup' table can be skipped safely but should number no more than about 6 per operational server. Counters on the number of SQL updates and inserts pending on the current error is provided and updated after each 'skip'. If errors were encountered on one server that were resolved using this 'skip' procedure, then the same errors will have to be skipped when the second server has it synchronization enabled.

In the end, both database status reports should be clean.

##2.7 Documentation Wiki## The NetYCE web front-end includes a “Wiki” button in the header. This button links to the on-line NetYCE Wiki where the documentation can be found.



We are in the middle of moving our documentation to this Wiki to ultimately provide our users with up-to-date documentation at the various levels of expertise. Please consult the Sitemap for the available pages.

Experienced users who are willing to contribute can apply for an account at support@netyce.com. Help is greatly appreciated.

##2.8 Modeling##

On the modeling side, support for the service-tasks has been extended to allow for standardized changes to an existing YCE ‘service’. These YCE ‘services’ consist of a container for nodes and/or subnets that were initially created using ‘service-types’.

Once created, these services could only be modified manually that frequently involved much clicking and a bit of hard thinking. Service-tasks are the equivalent of service-types to do these modifications.

YCE 5.3 introduced the service-tasks which have now been extended to perform more versatile tasks. Amongst these are commands for managing address reservations in subnets to determine if they should be considered ‘full’ (conditionally creating a new subnet of the same type), but also commands for moving slots in devices and moving (topology) links between nodes or ports.

The following commands have been added or improved:

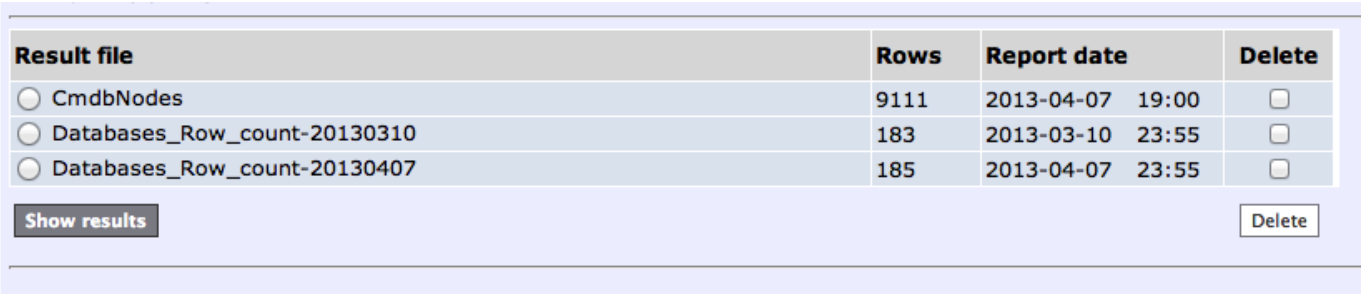
Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
ASSIGN	NODE	node	TEMPLATE	value		change Template and ports. Keeps obsolete topo/subnets, keeps port-details
ASSIGN	NODE	node	TEMPLATE_FORCE	value		change Template and ports. Drops obsolete topo/subnets, updates port-details
ASSIGN	NODE	node	VRF	vrf		
ASSIGN	PORT	port	PORT_NAME_OF	port		move the port to the other port's location. Drops the target port along with topo/subs
ASSIGN	SERVICE	srv	HIERARCHY_ID	value		set the service hierarchy-id to the desired value
ASSIGN	SERVICE	srv	SERVICE_CLASS	value		set the service class to the desired value
ASSIGN	SERVICE	srv	SERVICE_TYPE	value		set the service type to the desired value
ASSIGN	SUBNET	net	ADDR_RESERVATION	value		
ASSIGN	SUBNET	net	ADDR_RESERVATION_INC	value		increase the subnets address reservation count to value
ASSIGN	SUBNET	net	ADDR_RESERVATION_DEC	value		decrease the subnets address reservation count to value
ASSIGN	SUBNET	net	VRF	vrf		
ASSIGN	VRF	vrf	NODE	node		
ASSIGN	VRF	vrf	SUBNET	net		

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
ASSIGN	VRF	vrf	VRF_TEMPLATE	value		Value is Vrf template name
ASSIGN	SLOT	slot	SLOT_ID	value		Move all interfaces to a new slot_id. Value format = slot[/module]. You can use '/' to indicate blank slot/module
LOCATE	NODE	SERVICE	CURRENT		node	find the selected node in the current service (tasks only)
LOCATE	SERVICE	SERVICE	CURRENT		srv	find the current selected service (tasks only)
LOCATE	SUBNET	SERVICE	CURRENT		net	find the current selected subnet in the service (tasks only)
LOCATE	LINK	node	NODE	node	portlist	find the ports involved linking these two nodes
LOCATE	NODE	portlist	UPLINK_NODE		node	returns the node with higher hierarch-id in the link(s)
LOCATE	NODE	portlist	DOWNLINK_NODE		node	returns the node with the lower hierarch-id in the link(s)

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
LOCATE	SLOT	node	SLOT_ID	value	slot	find all interfaces using slot_id. Value format = slot[/module]. You can use '/' to indicate blank slot/module

The full list of supported service-type and service-task commands is shown in Appendix A. They now number 124 in total.

##2.9 Query results## The query results overview page now includes a column with the report date and time.



The report date indicates the age of the report but also allows confirmation that the report is still updated daily should that have been its setup. **#3 Back-end#**

##3.1 NetYCE distribution using patch levels## With 5.3.1 we introduce patch levels with our distributions. This mechanism allows us to distribute incremental patches for customers requiring a quick-response fix or custom extension without having to hold out to a major or minor release. The patches will be cumulative so any update will always bring the system to the latest available level without the chance of skipping a modification or extension.

Separate patch level registrations are maintained by the YCE environment to keep tabs on the current status. Each server maintains its own level with a separate one for the (replicating) databases.

##3.2 New archive mechanism## The daily raw database backup is replaced by a new format that creates portable and encrypted database sets. The existing format was considered too sensitive to allow for off-system storage or ordinary maintenance.

The crontab for the 'yce' user will be modified to replace mydbarch.pl for dbarchive.pl. See paragraph 2.6 for details how to use these archive sets.

##3.3 MySQL 5.6 support## Recent changes and vulnerabilities in the database engine that NetYCE deploys, MySQL, require an update of the MySQL version. Several vulnerabilities were uncovered in the previous version of MySQL, 5.5.x that were not (fully) addressed in the latest release of 5.5.x.

The MySQL 5.6.x release gas been GA for some time now and have no unaddressed vulnerabilities. Reason enough to extend NetYCE support and upgrade to MySQL 5.6.x where advisable.

Several internal changes to MySQL required some changes to NetYCE, all dealing with enhanced

security. The changes will also affect the existing 5.5.x installations, introducing the same modifications for both 5.5.x and 5.6.x installations.

#5 Fixes#

##5.1 ASR 9000 eVPN scan job failure## Each time an operator executes an eVPN scan job, the Job-log shows this job had failed. Few persons would take notice since the results of the scan were unaffected.

The reason the job was reported a failure is that although only a series of show commands were issued, the tooling nevertheless did start a configuration session which ultimately required a 'commit' since it is a Cisco XR device. Having really nothing to commit, this step fails and hence the job.

The commit handling has been improved to detect these empty commits.

##5.2 Password expiry warning## When a password is about to expire or grace logins are permitted, the window to change the password failed to appear when logging in using the YCE Client. This caused unexpected password expired denials.

The problem was fixed by correcting the call to the password update form.

##5.3 No permissions without Full name## The required permissions for the web-tools were denied for a user although access was granted based on these permissions. The reason proved that when the user's full name was omitted, the web-tools failed to retrieve the specific permissions.

By rewriting the routine that validate permissions for the web-tools was this problem resolved.

##5.4 Relation-test tool## Although context functions were introduced in 5.3.0, there was no corresponding way to review their results in a given (node) context as with the 'relation-test' tool.

This shortcoming has been corrected and enhanced with error handling to validate the code that creates these context functions.

##5.5 Port config## The order of the ports shown in the port config tool is non-numeric when viewing an un-stacked Avaya switch. Since the port-names of un-stacked Avaya switches have a blank slot-id, the resulting port name failed to sort in sensible way.

This problem has been corrected.

##5.6 Cisco 6500 startup-config## When pushing a new startup configuration to a Cisco 6500 device, the job fails with a verification error. This problem proved to be related to a timing issue with this type of device. After the tftp upload to the device, the next command - to verify the file is complete – is to list the mandatory "end" at the end of the file. This command fails because NO response is received from the device.

This problem was seen earlier, during the development of the Cisco module. At that time it was resolved by issuing a dummy command (show privileges) before trying to list the file. However, at this stage it proved necessary to repeat these dummy commands several times and each time to wait a while, before the device could comply properly too the command.

##5.7 Avaya switch banners## Including banners in Avaya switches is common practice, but after its deployments prevented YCE to login. The length of the banner and the time required caused the

generic vendor module to no longer detect prompt for the <control>-Y that typically starts all Avaya CLI sessions.

The problem proved to be hard to fix since timing was involved. And because the timing and format of the banner changes after a device reload, finding a reliable solution proved even harder.

##5.8 Nexus TFTP transfer## The Cisco Nexus 5000 series failed to perform file transfers with the YCE tftp server. A few hundred frames into the transfer, the tftp stream stagnated and retransmits fail to recover.

Using the default retransmission timers of 5 seconds caused the Nexus to misinterpret the frame sequencing. Actively retransmitting after a much shorter timeout prevents this lockup. #6 Appendix A – Service type syntax#

The full set of service type and service task commands now number 124 in total. The full list is shown below.

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
ADD	SERVICE	CLIENT	CURRENT		srv	create a new service container on the current site
ADD	SERVICE	SITE	CURRENT		srv	create a new service container on the current site
ADD	NODE	srv	NODE_TYPE	value	node	create new node in a service of specified node-type
ADD	PORT	node	FAST_ETHERNET	value	port	value format= slot[/module]/port
ADD	PORT	node	GIGABIT_ETHERNET	value	port	value format= slot[/module]/port
ADD	PORT	node	LOOPBACK	value	port	value = loopback-id number of Lo interface (Lo0)
ADD	PORT	node	MANAGEMENT	value	port	value = management-id number of Ma interface (Ma0)
ADD	PORT	node	PORT_CHANNEL	value	port	value = portchannel-id number of Po interface (Po1)
ADD	PORT	node	PORT_CHANNEL_NEW	value	port	value = lowest portchannel-id number to start search for new id
ADD	PORT	node	TENGIGABIT_ETHERNET	value	port	value format= slot[/module]/port
ADD	PORTS	node	FAST_ETHERNET	value	portlist	value format= slot[/module]/port[-port]
ADD	PORTS	node	GIGABIT_ETHERNET	value	portlist	value format= slot[/module]/port[-port]
ADD	PORTS	node	TENGIGABIT_ETHERNET	value	portlist	value format= slot[/module]/port[-port]
ADD	LINK	port	PORT	port	portlist	connect two ports using topology
ADD	SUBNET	srv	CUSTOM	value	net	add new CUSTOM subnet. No IP-plan is used
ADD	SUBNET	srv	NET_NAME	value	net	locate and add subnet from an IP-plan by name

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
ADD	SUBNET	srv	NET_NAME_NOTFULL	value	net	locate and conditionally add subnet from an IP-plan by name should the existing be 'full'
ADD	SUBNET	srv	NET_TYPE	value	net	add subnet from an IP-plan by name
ADD	VRF	node	VRF_ID	value	vrf	create vrf on node using its id. Note: node must have management / loopback address assigned first!
ADD	VRF	node	VRF_NAME	value	vrf	create vrf on node using its name. Note: node must have management / loopback address assigned first!
ASSIGN	ADDRESS	addr	PORT	port		set a single IP-address on a Management or Loopback interface
ASSIGN	EVPN_VLAN	node	EVPN_CORE	value		assign eVPN to a core-node by name
ASSIGN	EVPN_VLAN	node	EVPN_CORE	node		assign eVPN to a core-node by node-alias
ASSIGN	NODE	node	NODE_POSITION	value		assign a node to a position in the reference-model
ASSIGN	NODE	node	TEMPLATE	value		change Template and ports. Keeps obsolete topo/subnets, keeps port-details
ASSIGN	NODE	node	TEMPLATE_FORCE	value		change Template and ports. Drops obsolete topo/subnets, updates port-details
ASSIGN	NODE	node	VRF	vrf		
ASSIGN	PORT	port	PORT_CHANNEL	value		make the port a port-channel member of specified channel-id
ASSIGN	PORT	port	PORT_CHANNEL_OF	port		make the port a member of the channel-group of the other port
ASSIGN	PORT	port	PORT_MODE	value		set the duplex mode of the port
ASSIGN	PORT	port	PORT_SHUT	value		enable or disable the port (shut=Y → disabled)
ASSIGN	PORT	port	PORT_SPEED	value		set the port speed in Mbps
ASSIGN	PORT	port	PORT_TEMPLATE	value		set the port's port-template
ASSIGN	PORT	port	PORT_TEMPLATE_OF	port		set the port's port-template to that of the other port
ASSIGN	PORT	port	PORT_NAME_OF	port		move the port to the other port's location. Drops the target port along with topo/subs
ASSIGN	PORT	port	SUBNET	net		

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
ASSIGN	PORTS	portlist	PORT_CHANNEL	value		make the ports a port-channel member of specified channel-id
ASSIGN	PORTS	portlist	PORT_CHANNEL_OF	port		make the ports a member of the channel-group of the other port
ASSIGN	PORTS	portlist	PORT_MODE	value		set the duplex mode of the ports
ASSIGN	PORTS	portlist	PORT_SHUT	value		enable or disable the ports (shut=Y → disabled)
ASSIGN	PORTS	portlist	PORT_SPEED	value		set the ports speed in Mbps
ASSIGN	PORTS	portlist	PORT_TEMPLATE	value		set the ports port-template
ASSIGN	PORTS	portlist	PORT_TEMPLATE_OF	port		set the port port-template to that of the other port
ASSIGN	PORTS	portlist	SUBNET	net		
ASSIGN	SERVICE	srv	HIERARCHY_ID	value		set the service hierarchy-id to the desired value
ASSIGN	SERVICE	srv	SERVICE_CLASS	value		set the service class to the desired value
ASSIGN	SERVICE	srv	SERVICE_TYPE	value		set the service type to the desired value
ASSIGN	SUBNET	net	LINK	portlist		
ASSIGN	SUBNET	net	PORT	port		
ASSIGN	SUBNET	net	PORTS	portlist		
ASSIGN	SUBNET	net	VLAN_ID	value		assign this vlan-id to the subnet
ASSIGN	SUBNET	net	VLAN_ID_INCLIENT	value		find and assign the first free vlan-id in client starting at type-value
ASSIGN	SUBNET	net	VLAN_ID_INSERVICE	value		find and assign the first free vlan-id in the service starting at type-value
ASSIGN	SUBNET	net	VLAN_ID_INSITE	value		find and assign the first free vlan-id in site starting at type-value
ASSIGN	SUBNET	net	ADDR_RESERVATION	value		set the subnets address reservation count to value
ASSIGN	SUBNET	net	ADDR_RESERVATION_INC	value		increase the subnets address reservation count to value
ASSIGN	SUBNET	net	ADDR_RESERVATION_DEC	value		decrease the subnets address reservation count to value
ASSIGN	SUBNET	net	VRF	vrf		
ASSIGN	VRF	vrf	NODE	node		
ASSIGN	VRF	vrf	SUBNET	net		
ASSIGN	VRF	vrf	VRF_TEMPLATE	value		Value is Vrf template name

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
ASSIGN	SLOT	slot	SLOT_ID value			Move all interfaces to a new %%slot_id%%. Value format = slot[/module]. You can use '/' to indicate blank slot/module
LOCATE	ADDRESS	net	ADDRESS_FIRSTFREE	value	addr	find the first free address in a named range of a subnet
LOCATE	NODE	SERVICE	CURRENT		node	find the selected node in the current service (tasks only)
LOCATE	NODE	CLIENT	NODE_POSITION	value	node	find a node given its position in the client
LOCATE	NODE	CLIENT	NODE_TYPE	value	node	find a node given its type in the client
LOCATE	NODE	SITE	NODE_POSITION	value	node	find a node given its position on the site
LOCATE	NODE	SITE	NODE_TYPE	value	node	find a node given its type on the site
LOCATE	NODE	srv	NODE_POSITION	value	node	find a node given its position in its service
LOCATE	NODE	srv	NODE_TYPE	value	node	find a node given its type in its service
LOCATE	PORT	node	PORT_NAME	value	port	find a port by its name (internal name e.g.: Gi01/12)
LOCATE	PORT	node	PORT_TEMPLATE	value	port	find first port using this template. Wildcard optional
LOCATE	PORT	node	PORT_TEMPLATE_FIRST	value	port	find first port using this template. Wildcard
LOCATE	PORT	node	PORT_TEMPLATE_FIRSTNOTOPO	value	port	find first port without topology using this template. Wildcard optional
LOCATE	PORT	node	PORT_TEMPLATE_LAST	value	port	find last port using this template. Wildcard optional
LOCATE	PORT	node	PORT_TEMPLATE_LASTNOTOPO	value	port	find first port without topology using this template. Wildcard optional
LOCATE	PORT	portlist	PORT_TEMPLATE_FIRST	value	port	find first port in list with this template
LOCATE	PORT	portlist	PORT_TEMPLATE_FIRSTNOTOPO	value	port	find first port in list with this template without topology
LOCATE	PORT	portlist	PORT_TEMPLATE_LAST	value	port	find last port in list with this template
LOCATE	PORT	portlist	PORT_TEMPLATE_LASTNOTOPO	value	port	find last port in list with this template without topology
LOCATE	PORTS	node	PORT_TEMPLATE_ALL	value	portlist	find all ports in list with this template
LOCATE	PORTS	node	PORT_TEMPLATE_ALLNOTOPO	value	portlist	find all ports in list with this template without topology

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
LOCATE	PORTS	node	FAST_ETHERNET	value	portlist	value format= slot[/module]/port[-port]. Use '*' for any slot, module or port
LOCATE	PORTS	node	GIGABIT_ETHERNET	value	portlist	value format= slot[/module]/port[-port]. Use '*' for any slot, module or port
LOCATE	PORTS	node	TENGIGABIT_ETHERNET	value	portlist	value format= slot[/module]/port[-port]. Use '*' for any slot, module or port
LOCATE	PORTS	node	LOOPBACK	value	portlist	value format= slot[/module]/port[-port]. Use '*' for any slot, module or port
LOCATE	PORTS	node	PORT_CHANNEL	value	portlist	value format= slot[/module]/port[-port]. Use '*' for any slot, module or port
LOCATE	SERVICE	CLIENT	SERVICE_CLASS	value	srv	
LOCATE	SERVICE	CLIENT	SERVICE_TYPE	value	srv	
LOCATE	SERVICE	SITE	SERVICE_CLASS	value	srv	
LOCATE	SERVICE	SITE	SERVICE_TYPE	value	srv	
LOCATE	SERVICE	SERVICE	CURRENT		srv	find the current selected service (tasks only)
LOCATE	SUBNET	SERVICE	CURRENT		net	find the current selected subnet in the service (tasks only)
LOCATE	SUBNET	srv	NET_NAME	value	net	find the subnet by name in the service
LOCATE	SUBNET	srv	NET_NAME_NOTFULL	value	net	find and conditionally add subnet from an IP-plan by name should the existing be 'full'
LOCATE	SUBNET	srv	VLAN_ID	value	net	find the subnet by vlan-id in the service
LOCATE	VRF	node	VRF_ID	value	vrf	find a vrf on the node using its id
LOCATE	VRF	node	VRF_NAME	value	vrf	find a vrf on the node using its name
LOCATE	VRF	SITE	VRF_ID	value	vrf	find a vrf on the site using its id
LOCATE	VRF	SITE	VRF_NAME	value	vrf	find a vrf on the site using its name
LOCATE	LINK	node	NODE	node	portlist	find the ports involved linking these two nodes
LOCATE	LINK	node	PORT_TEMPLATE	value	portlist	find the ports involved in a link using a local port-template
LOCATE	NODE	GLOBAL	NODE_NAME	value	node	find node using its name. Supports wildcard
LOCATE	NODE	portlist	UPLINK_NODE		node	returns the node with higher hierarch-id in the link(s)
LOCATE	NODE	portlist	DOWNLINK_NODE		node	returns the node with the lower hierarch-id in the link(s)
LOCATE	SERVICE	GLOBAL	SERVICE_CLASS	value	srv	
LOCATE	SERVICE	GLOBAL	SERVICE_TYPE	value	srv	

Type_exec	Type_class	Type_scope	Type_match	Type_value	Type_alias	Type_notes
LOCATE	PORT	node	PORT_TEMPLATE_FIRSTSHUT	value	port	find first disabled port with this template. Wildcard optional
LOCATE	PORT	node	PORT_TEMPLATE_LASTSHUT	value	port	find last disabled port with this template. Wildcard optional
LOCATE	SLOT	node	SLOT_ID	value	slot	find all interfaces using slot_id. Value format = slot[/module]. You can use '/' to indicate blank slot/module
DELETE	NODE	node	COMPLETE			delete node including topology
DELETE	NODE	node	VRF	vrf		remove node from vrf
DELETE	PORT	port	COMPLETE			delete port including topology
DELETE	PORT	port	TOPOLOGY			remove topology from port
DELETE	PORT	port	SUBNET	net		remove subnet from port
DELETE	PORTS	portlist	COMPLETE			delete ports including topology
DELETE	PORTS	portlist	TOPOLOGY			remove topology from ports
DELETE	PORTS	portlist	SUBNET	net		remove subnet from ports
DELETE	SUBNET	net	COMPLETE			delete subnet, remove from all ports
DELETE	SUBNET	net	VRF	vrf		remove subnet from vrf
DELETE	SUBNET	net	PORTS	portlist		remove subnet from ports
DELETE	SUBNET	net	PORT	port		remove subnet from port
DELETE	VRF	vrf	NODE	node		remove node from vrf
DELETE	VRF	vrf	SUBNET	net		remove subnet from vrf

[eof]

From:
https://wiki.netyce.com/ - **Technical documentation**

Permanent link:
https://wiki.netyce.com/doku.php?id=maintenance:releases:historical:releasenotes_531

Last update: **2024/07/03 12:31**