



Cisco WAE Release 6.2

Contents

- How to Upgrade3
 - MATE API Upgrades to Design API4
- WAN Automation Engine4
- Release Highlights4
- Application Enhancements.....4
 - WAE Design4
 - Design API5
- WAE Platform Enhancements6
 - WAE Collector6
 - Collector Server and WAE Network Interface Server6
 - Snapshots6
 - WAI NI API7
 - WAE Core7
- WAE System Enhancements8
- Schema Changes.....9
 - New Schema9
 - New Tables9
 - New Columns11
- CLI Changes12
 - Removed or Deprecated CLI Tools12
 - Removed CLI Options12
 - New CLI Tools12
 - New CLI Options.....13
 - New Values for CLI Options16
 - Changed CLI Behavior16
- Service Changes.....17
 - Removed Services.....17
 - New Services.....17
- Open Source18
- Issues Fixed18
 - Issues Fixed Since 6.1.3.....18
 - WAE Live18
 - WAE Collector.....18
 - WAE System19
 - Issues Fixed Since 6.1.2.....19
 - WAE Design.....19
 - WAE Collector.....19
 - Issues Fixed Since 6.1.1.....19
 - WAE Design.....19

WAE Live	19
WAE Collector.....	19
WAE Core	19
Issues Fixed Since 6.1.....	20
WAE Design.....	20
WAE Live	20
WAE Collector.....	20
WAE System.....	20
Known Limitations	21
Applications	21
WAE Design.....	21
WAE Live	21
WAE Platform	21
WAE Collector.....	21
Collector Server	21
WAI NI Server	21
Snapshots	21
SAM-OSS Integration with Snapshots.....	23
Deployer Module	23
OSC Controller.....	23
NSO Controller.....	23
WAE System.....	23
Installation and Startup	23
Web Server	24
WAE Statistics UI.....	24
Web User Management.....	24
License Check Failures on Newer Linux Distributions.....	24
Java Memory.....	25
Documentation.....	25

How to Upgrade

- Plan files—Plan files from previous versions are read and upgraded automatically on opening in release 6.2.
- WAE Live datastore—To upgrade the datastore from a release prior to 5.6, contact your support representative. To upgrade from a 5.6+ release, execute a backup (`mld -action backup` or `ml_backup`). Then stop all services and run the upgrade (`mld -action upgrade`). For assistance, refer to the `-help` output or to the *WAE Live Configuration Guide*. Note that copying files, rather than using a backup or upgrade tool, is not sufficient.
- WAE Platform
 - Backup files—Prior to performing an upgrade, make copies of the following files in case you need them after the installation.
 - `/opt/cariden/software`
 - `wae/core/etc`
 - `wae-messaging/conf`
 - `wae-osc/etc` (In 6.1, this was `wae-cdl`)
 - `wae-db/conf`
 - `wae-ni/etc` (In 6.1, this was `wae-collector/etc`)
 - `/opt/cariden`
 - `etc`
 - `data`
- Collector server—If using the default installation directory and username, a database upgrade is automatically performed. To upgrade the database and to migrate Collector server files when the installation directory and username are not the same, answer "Yes" when prompted as such during the installation process.
- Collector snapshots—If using an upgraded snapshot process, run `archive_init -upgrade` on individual archive repositories before adding them to the server. This ensures that the internal archive database schema is current, but does not upgrade the underlying plan files. The `archive_insert` tool does not automatically run a simulation before inserting a plan file. To check in a plan file containing a simulation, run `mate_sim` on the plan file before running `archive_insert`.
- Add-ons and scripts—Add-ons and scripts may be affected by the schema changes and CLI tool changes that are described in their respective sections.
- FlexNet Publisher License Server—Whether this is a new or an upgraded installation, if using the FlexNet Publisher License server for WAE Design users, you must download the latest version from the customer download portal.
- Package names have changed to be either WAE Planning or WAE Automation. For Linux, these packages are `wae-planning-k9-6.2.0` and `wae-automation-k9-6.2.0.bin`, respectively. The WAE Automation package requires a dual-server installation. Contact your support representative for information on what these packages contain and for dual-server installation details.

MATE API Upgrades to Design API

The MATE API has been renamed to Design API. As a result, the API namespaces have changed to reflect this name change. These changes are not backwards compatible. If you are using these APIs, you must change your code to reflect these changes.

- The `mateapi` namespace has been replaced by `wae.design`.
- The `pln` namespace has been renamed to `model`.

Python Examples: Note that similar changes are required for Java code that uses the API.

Replace	By
<code>import com.cisco.mateapi</code>	<code>import com.cisco.wae.design</code>
<code>com.cisco.mateapi.pln.net.Circuit</code>	<code>com.cisco.wae.design.model.net.Circuit</code>

This API is now accessed from `/opt/cariden/software/mate/current/docs/api/design`.

WAN Automation Engine

As of WAE Release 6.2, all products are subsumed under the WAE suite of products. This consolidation creates a total solution from WAE planning to automated modeling, deployment, and control.

As such, all previously named “MATE” products are now “WAE.” Rebranding the actual products is an incremental process that will occur over time. For now, you may still see the term “MATE,” such as in messages.

Release Highlights

The 6.2 release focuses on the following major enhancements. The Enhancement sections list their details, as well as many other enhancements.

- Segment Routing (SR) collection, optimization, and deployment,
- BGP LS collection
- NSO integration
- Inventory collection
- Serviceability

Application Enhancements

WAE Design

- New SR TE Optimization Tool—Creates or updates segment lists to optimize the routes of selected SR LSPs using the fewest number of segment list hops as possible. One use case is to minimize the delay of SR LSPs. Another is to enforce that LSPs avoid routing through specified nodes.
- Segment Routing Enhancements
 - Segment lists can contain anycast groups. When a segment list hop corresponds to an anycast group, the SR LSP routes through the node in the anycast group that has the shortest IGP path to it. This

-
- enables SR LSPs to choose amongst the possible next segment list hops, potentially reducing latency and improving load balancing.
- Simulations support loops so that segment-routed traffic can be forwarded to the next segment list hop without regard for whether previous segment list hops were reached.
 - Note that SR simulations do not yet support tunnel binding SIDs.
 - New RSVP TE Optimization Tool—This tool calculates explicit routes for RSVP-TE LSPs. This tool has several key use cases.
 - Optimizing the latency of all LSPs subject to bandwidth constraints.
 - Creating a new LSP or set of LSPs while minimizing their latency without exceeding a bandwidth bound on each interface. There is flexibility to specify whether existing LSPs must remain fixed or can move if needed.
 - Performing tactical congestion mitigation.
 - Layer 1 Enhancements
 - L1 link waypoints are available to model network devices that are not L1 nodes and that reside in the L1 link path. This enables you to more accurately model geographic fiber paths without creating L1 nodes. Waypoints are used only for visualization purposes in the network plot and for calculations executed by the Latency and Distance initializer.
 - Simultaneous viewing of both L1 and L3, thus improving the ability to see their relationships, particular in the event of failures.
 - Visualization of Failure Impact in the L1 view when L1 links are failed through Simulation Analysis.
 - Additional Simulation Enhancements
 - Overload bit provides an easy way to prevent traffic from transiting the node without affecting traffic sourced from or destined for the node itself.
 - EIGRP calculations were updated to ensure more accurate EIGRP simulations. This includes the addition of an EIGRPDelay column in the <Interfaces> table.
 - To improve performance and better take advantage of available resources, you can specify the maximum number of threads in the Simulation Analysis and Metric Optimization tools via the GUI.
 - Miscellaneous Enhancements
 - A new `copy_from_template` option enables you to identify whether to map all nodes in the plan file based on the <SiteMapping> table or not. This removes the need to manually intervene if a node is assigned to a site in the template.
 - A new `trim_nodes` option enables you to specify nodes to include in a plan file, whereas in the past, you could only exclude nodes.
 - The feature for opening from and saving to a remote WAE Design Archive server was improved to ease the manner in which you can enter relevant information, such as the ability to enter an archive name.

Design API

- Object Enhancements
 - The ability to create objects in bulk enables you to more efficiently create large numbers of an object type with a single API method call rather than for each one. For instance, you can "bulk" create

multiple SRLGs for use in failure analysis, and you can bulk create thousands of L1 objects rather than individually creating them.

- Add, edit, and delete L1 link waypoints to support more geographically accurate fiber paths without creating fully modeled L1 nodes.
- Read-only access the content of NetInt* tables. These are treated much like user tables via the API. Content is returned in the form of two-dimensional array of strings.
- Ability to add and remove ActualPathHops for LSPs that are discovered through login or XML configuration files.
- Ability to include L1 links and L1 nodes in failure scenarios to support multi-layer failure analysis.
- The ability to determine the shortest IGP path between two nodes and total IGP cost without having to create a demand reduces manual effort when planning IGP networks.
- NetworkFailureScenario was added so that you can set and get a failure scenario associated with the plan, including L3 and L1 objects. Objects specified here as failed will have a failed state of True when viewed in the WAE Design GUI. Note that this removes the setInactive() attribute in the existing SimAnalysis, ExplicitOptimizer, and ExplicitOptimizerTactical APIs.
- The ability to compute routes for temporary demands that are not added to the network enables you to quickly determine potential paths and their properties for what-if analyses.
- InterfaceManager and NodeManager can find interfaces and nodes by IP address, thus simplifying their lookup.
- Ability to create and edit reports to include in plan files enables a purely API-based method for report generation.
- Optionally send the Design API log messages to a log file, rather than stdout.
- SSL connections with the Design API are available for use with C++ and Java API clients.

WAE Platform Enhancements

WAE Collector

Collector Server and WAE Network Interface Server

- LSPs that are managed by WAE (PCE) can be delegated to WAE. The WAE Network Interface (NI) server is used to continuously collect PCEP RSVP-TE LSPs. You can configure this continuous collection either through the WAE Collector UI or through snapshot configuration files.
- Continuous polling performance was improved to enhance system scalability.

Snapshots

- `get_inventory` enables you to collect hardware information.
 - Playback recorder for SNMP sessions improves deployment and debugging.
 - Juniper: Can put the NETCONF/login XML session information into a file for debugging purposes.
- `build_inventory` enables you to process the collected hardware inventory for inclusion in the WAE Live application.
- New `find_bgpls` tool supports a single IGP domain.

- Enables you to use BGP LS to actively collect and listen to IGP topology and Traffic Engineering state changes, thereby presenting a near real-time view of network topology. The OSC controller is used to collect this information.
- Ability to discover unreserved BW from BGP LS (for IS-IS).
- Segment Routing (IOS XR) support for a single IS-IS domain.
 - `login_find_igp_db` and `parse_igp` collect SIDs from the IS-IS database, including the collection of anycast node groups and their members.
 - `parse_configs` collects the tunnels, paths, and hops (non-PCEP).
 - `import_lsp` can be used to import PCEP SR tunnels managed by WAE.
- Tools Enhancements
 - `snmp_find_interfaces`
 - Juniper: Accounts for LACP state so now the `-lag-port-match exact` value can exclude a port that is up, but whose LACP state indicates it is not a LAG member.
 - A new `-lag-port-match complete` value matches ports that are down based on LACP, as well as on the `guess heuristic`.
 - `parse_configs` identifies LSP auto-bandwidth enablement.
 - `login_find_igp_db` and `parse_igp` collect overload indicators used to prevent transit traffic from traversing a router.
 - `find_bgp` collects BGP peer data, including eBGP multi-hop, from NX-OS devices.

WAI NI API

- Ability to get a list of PCEP LSPs through a REST interface.

WAE Core

The following new capabilities were added via new APIs or changes to existing APIs.

- Exposing Design APIs through REST—New `wae-appenginecore` and `wae-designapiserver` services allow REST access to Design APIs in a staging area. This enables you to develop applications that take advantage of the Design simulation and optimization APIs. With these services, you can make changes to a staged model of the network using Design APIs, and then download the plan file back to the working (current) network model.

Note that this new set of APIs has an overlap with existing REST APIs such that over time, these overlapping REST APIs will be deprecated.

- API Deployment Enhancements
 - NSO was integrated so that RSVP-TE LSPs can be deployed to greenfield networks through an NSO controller. NED types include CLI (Cisco) and NETCONF (Juniper). For assistance integrating WAE with NSO, contact your support representative.
 - Note that the deployment of LSPs with affinities requires the following.
 - The existence of affinities in the imported plan file.
 - Network nodes must be pre-configured w/ affinity groups.
 - For assistance integrating WAE with NSO, contact your support representative.
 - Ability to create, modify, delete, and deploy PCEP SR LSPs within a single IGP area.

- New PCEP deployment configuration options for improved integration with OSC.
- API Model Enhancements
 - Ability to get information about staging areas, such as per-stage ID time created and staging policy, which makes it easier to manage staging areas.
 - Ability to get per-interface traffic simulation, capacity simulation, and utilization simulation for the working plan.
 - Complete parity in API support between working and staging areas, which eases and improves application development. Note that this had the effect of changing these APIs from GET to POST since they now work on staging IDs. The working plan is identified as stage ID 0.
Example: The `/wae/network/modeled/entities/tunnel/from-model/get-all-tunnels` API can now be used to get tunnel information for specified staging areas.
 - Ability to execute a dry-run deployment as a simulation in a staging area. This allows you to determine if deploying LSP changes would conflict with or cause problems in the network before deploying them.
- API Calendaring Enhancements
 - Ability to list all calendared demands in a specified time frame without defining the source and destination.
 - Ability to customize the projection of scheduling bandwidth calendaring, as well as the window in which it occurs.

WAE System Enhancements

- Service Enhancements
 - New system services are available for monitoring the WAE system. These services enable the monitoring through the new WAE Statistics UI.
 - The WAE Live datastore is now started and stopped as the `wae-mlld` service. Note that installation and upgrade are still performed using the `mlld` tool.
 - All services except Automation services and `wae-mlld` are started automatically upon the completion of installation. (As in prior releases, Automation services can be automatically started if you so choose.)
 - The term “SDN services” was renamed to “Automation services.”
 - The `wae-collector` service was renamed to `wae-ni`.
- WAE Statistics UI—A new WAE Statistics web UI enables you to monitor and troubleshoot all services and servers used in the deployment.

Note: The `wae-mlld` service is not monitored and thus, is not included in the following statistics.

- Process Status—Shows the status of WAE processes. From here you can determine if a service is operational, unreachable, not initialized, or not being monitored. With an admin role, you can also start, stop, and restart services, as well as enable or disable their monitoring.
- Event Logs—Shows detailed log information for each service and thus, can be used for troubleshooting the system or simply better understanding it. For instance, you could find logs that identify why a collection failed or find warnings applicable to PCEP LSP deployments. This UI also

includes graphs that show the percentage of total logs by service, log level counts, and log event timelines.

- Platform Diagnostics—Shows trends of diagnostics for all components (hosts), which is useful for analyzing trends over time and for being alerted to sudden changes in the services.
- Installation Enhancements
 - Creates an installation directory so that you do not have to create one as a preliminary step.
 - Stops all services so that you do not have to stop them prior to installing.
 - Optionally migrates Collector server files if using different installation directories in an upgrade installation.
 - Sets the file capabilities for all binaries in `/opt/cariden/software/mate/current/lib/ext/pmacct/sbin`, which enables you to collect flow data using `flow_manage` and `flow_get` without having to change the file capabilities for the flow collection server.
 - Sets file permissions for the WAE user and others to improve control over who can write to and execute the files.
- Miscellaneous Enhancements
 - Users can now borrow, rather than check out, floating licenses, alleviating the need to be connected to the internet when using floating licenses.
 - `mate_tech_support` was updated to include `wae-ni` and `wae-core` log files, thus easing troubleshooting efforts.
 - `wae-web-server` has a configuration file for specifying the behavior when this service is started (`/opt/cariden/etc/sysconfig/wae-web-server.cfg`). For upgrades, you can modify these parameters prior to installing release 6.2 to alter the initial startup behavior of `wae-web-server`.
 - Each package containing customer-facing documentation has a `docs` directory and if applicable, a `docs/api` directory. The package's `docs` directory is symbolically linked to the `/opt/cariden/software/docs` directory so that you can find all documentation, including API documentation, in one location.
 - The `mate` package name was changed to `wae-dlc`. The `mate` directory name remains the same.

Schema Changes

Consult the `/opt/cariden/software/mate/current/docs/table_schema.html` file for a complete reference.

New Schema

New Tables

New Table	Column	Type	Description
<AnycastGroupMembers>			Lists the nodes within each anycast group. This table does not appear in the WAE Design GUI.
	AnycastGroup	Key	Anycast group name.
	Node	Key	Name of the node in the anycast group.

New Table	Column	Type	Description
<AnycastGroups>			
			Lists the anycast groups.
	Name	Key	Anycast group name.
	NodeList	Derived	List of nodes in the anycast group.
	NumNodes	Derived	Number of nodes in the anycast group.
	SID	Plan	Anycast group segment identifier.
<InterfaceTEPriorities>			
			For each interface, lists the measured unreserved bandwidth per TE priority.
	Node	Key	Node name.
	Interface	Key	Interface name.
	TEPriority	Key	TE priority, which is an integer from 0 to 7 where 0 is the highest priority.
	UnreservedBWMeas	Plan	Measured unreserved bandwidth.
<L1LinkWaypoints>			
			List of the L1 link waypoints.
	Description	Plan	Description of the waypoint.
	L1NodeA	Key	L1 node A on the waypoint.
	L1NodeB	Key	L1 node B on the waypoint.
	L1LinkName	Key	Name of the L1 link on which the waypoint resides.
	Latitude	Plan	Latitude of the waypoint.
	Longitude	Plan	Longitude of the waypoint.
	Order	Key	Order of the waypoint starting with L1 node A. The waypoint closest to L1 node A is 1, the second closest is 2, and the numbering continues through to L1 node B.
<NetIntAdjacencySIDs>			
			Lists the adjacency SID associated with the interfaces in Segment Routing. The adjacency SID is mapped to the source and the destination nodes associated with the interface.
	AdjacencySID	Plan	Adjacency segment identifier.
	Flags	Plan	SID type identification flag.
	Node	Key	Node name associated with the SID.
	Prefix	Key	Adjacency prefix.
	RemoteNode	Plan	Remote node name associated with the SID.
<NetIntPrefixSIDs>			
			Lists the Prefix SID associated with the node in Segment Routing. It also has a flag to indicate if it is an anycast SID or not.
	Anycast	Plan	Identifies whether the SID belongs to an anycast group.
	Flags	Plan	SID type identification flag.

New Table	Column	Type	Description
	Node	Key	Node name associated with the SID.
	Prefix	Key	Node prefix (loopback address).
	PrefixSID	Plan	Prefix SID.
<NodeGroupMembers>			Lists the nodes within each node group.
	Node	Key	Name of the node in the node group.
	NodeGroup	Key	Node group name.
<NodeGroups>			List of node groups.
	Name	Key	Node group name.
	NodeList	Derived	List of nodes in the node group.
	NumNodes	Derived	Number of nodes in the node group.

New Columns

Table	New Column	Type	Description
<Interfaces>	EIGRPDelay	Plan	Configured delay for EIGRP in microseconds.
	SID	Plan	Adjacency segment identifier.
<L1Links>	FailureImpact	Derived	Maximum increased utilization over network under failure of this L1 link.
	FIInterface	Derived	Interface with highest utilization if this L1 link fails.
<L1Nodes>	FailureImpact	Derived	Maximum increased utilization over network under failure of this L1 node.
	FIInterface	Derived	Interface with highest utilization if this L1 node fails.
<LSPPaths>	NetIntPathType	Plan	Type of LSP path discovered: RSVP or SR.
<LSPs>	FRRDestinationGroup	Plan	Destination group for SR Fast Reroute LSPs.
	SID	Plan	Binding segment identifier.
<Nodes>	AvoidTransit	Plan	Node is not used for transit traffic.
	SID	Plan	Node segment identifier.
<Ports>	LACPMuxState	Plan	Multiplexing state of the aggregation port.
<SegmentListHops>	SegmentAnycastGroup	Plan	Name of the anycast group for the segment.

CLI Changes

WAE Design, WAE Live, and WAE Collector CLI tools are located in `/opt/cariden/software/mate/current/bin`. For more information on any CLI tool, execute it with the `-help` option.

Removed or Deprecated CLI Tools

CLI Tool	Removed or Deprecated	Replacement
<code>embedded_web_server</code>	Deprecated	<code>wae-web-server</code> , which is a service
<code>login_find_bgp</code>	Deprecated	<code>find_bgp</code>
<code>mateapid</code>	Removed	<code>designapid</code>
<code>mate_api_python</code>	Removed	<code>design_api_python</code>
<code>snmp_find_bgp</code>	Deprecated	<code>find_bgp</code>

Removed CLI Options

CLI Tool	Removed Option	Replacement
<code>collector_getplan</code>	<code>-out-plan-file</code>	<code>-out-file</code>
<code>copy_from_template</code>	<code>-include-extras</code>	
<code>rsvp_te_opt</code>	<code>-keep-affinities</code>	
	<code>-keep-bandwidth</code>	
	<code>-keep-setup-pri</code>	
	<code>-lsp-tag</code>	
	<code>-lsp-table</code>	
	<code>-path-length-metric</code>	
	<code>-reroute-non-opt</code>	
	<code>-setup-pri</code>	

New CLI Tools

CLI Tool	Description
<code>add_nodes_to_nso</code>	Initializes the NSO server with plan file nodes and authenticates them.
<code>build_inventory</code>	This processes the <code>NetIntHardware*</code> tables so that inventory collection can be made available for use in WAE Live. As input, it uses the plan file generated from <code>get_inventory</code> and a number of template configuration files.
<code>collector_migrate</code>	<p>Migrate Collector server files from the previous release to the current one. Files in the following directories are migrated.</p> <ul style="list-style-type: none"> <code>/opt/cariden/data/collector/server/file-persistence</code> <code>/opt/cariden/data/collector/server/snapshots</code> <code>/opt/cariden/etc/collector/server/configs</code> <code>/opt/cariden/etc/collector/server/db-persistence</code> <p>Note: You are presented this option by the installer.</p>

CLI Tool	Description
designapid	Starts the <code>designapi</code> service with an endpoint accessible at <code><hostname>:<port></code> that uses communication protocol <code><protocol></code> . This tool replaces <code>mateapid</code> .
design_api_python	Executes the given python script with the correct environment for Design API scripts. These scripts were previously called "MATE API" scripts. This tool replaces <code>mate_api_python</code> .
find_bgpls	Collects BGP LS from an OSC controller.
get_inventory	Use SNMP to access a network and collect inventory information. For Juniper routers, use NETCONF to log in and get SFP transceiver information that is not available via SNMP.
license_borrow	Borrow a floating license for up to 30 days. This enables you to use the license when you do not have connectivity with the license server.
license_return	Return a license to a floating license server.
sr_te_opt	Optimize routings of SR LSPs by minimizing the TE metric, delay (latency), or IGP metric of the path. You can also optionally avoid specific nodes.

New CLI Options

CLI Tool	New Option	Description
collector_getplan	<code>-flume-port</code>	Port number to which the flume agent is bound. Default is 41417.
	<code>-flume-server</code>	Name or address for the server running a flume agent. Default is localhost.
	<code>-out-file</code>	A local path to a file in where to write the latest network plan file. This replaces <code>-out-plan-file</code> .
collector_pushplan	<code>-flume-port</code>	Port number to which the flume agent is bound. Default is 41417.
	<code>-flume-server</code>	Name or address for the server running a flume agent. Default is localhost.
copy_from_template	<code>-update-assigned-sites</code>	If <code>false</code> (default), only nodes that are in the plan file but are not in the template are mapped to sites based on the <code>-site-mapping-table</code> option or the <code><SiteMapping></code> table. If <code>true</code> , all nodes in the plan file are mapped to sites based on the <code>-site-mapping-table</code> option or the <code><SiteMapping></code> table.
flow_get	<code>-flume-port</code>	Port number to which the flume agent is bound. Default is 41417.
	<code>-flume-server</code>	Name or address for the server running a flume agent. Default is localhost.
	<code>-number-of-threads</code>	Maximum number of simultaneous threads to be used in parallel computations. If a percentage symbol is present by the number, then the number of threads is computed over the number of cores and then rounded down. Default is 1.
flow_manage	<code>-flume-port</code>	Port number to which the flume agent is bound. Default is 41417.
	<code>-flume-server</code>	Name or address for the server running a flume

CLI Tool	New Option	Description
		agent. Default is localhost.
login_find_igp_db	-get-segments	Specify whether to collect Segment Routing information from IS-IS database. Only valid for IS-IS on CISCO IOS XR routers. Default is false.
parse_configs	-flume-port	Port number to which the flume agent is bound. Default is 41417.
	-flume-server	Name or address for the server running a flume agent. Default is localhost.
parse_igp	-get-segments	Specify whether to collect Segment Routing information from IS-IS database. Only valid for IS-IS on CISCO IOS XR routers. Default is false.
rsvp_te_opt	-fit-lsps-table	Reroute LSPs in this table only to satisfy bandwidth constraints. If a valid route does not exist, the LSP is routed or rerouted to optimize path length subject to bandwidth constraints. The default is empty.
	-fix-lsps-table	Reroute LSPs in this table only to satisfy bandwidth constraints. If a valid route does not exist, the LSP is routed or rerouted to optimize path length subject to bandwidth constraints. The default is empty.
	-init-int-bwbound	Specifies how to initialize the BWBound column in the <Interfaces> table. Values can be one of the following. <ul style="list-style-type: none"> resvbw (default) - Use reservable bandwidth (ResvBWSim) capacity - Use simulated capacity (CapacitySim) capavailsim - Use available capacity, excluding simulated traffic none - Use existing column entries
	-init-int-metric	Specifies how to initialize the Metric column in the <Interfaces> table. Values can be one of the following. <ul style="list-style-type: none"> temetric (default) - Use the interface TE metric (TEMetric) delay - Use circuit delay none - Use existing column entries
	-init-lsp-bwreq	Specifies how to initialize the BWReq column in the <LSPs> table. Values can be one of the following. <ul style="list-style-type: none"> setupbw (default) - Use setup bandwidth (SetupBW) traffmeas - Use measured traffic (TraffMeas) traffsim - Use simulated traffic (TraffSim) none - Use existing column entries
	-init-lsp-bwreq-sec	Specifies how to initialize the SetupBW column of secondary LSP paths in the <LSPPaths> table. Values can be one of the following. <ul style="list-style-type: none"> zero (default) - Use 0 (zero) for the required bandwidth prim - Use the same bandwidth as required for

CLI Tool	New Option	Description
		the primary LSP path
	<code>-init-lsp-groups</code>	<p>If <code>true</code>, use the tables specified by these options to classify the LSPs in the Group column of the <LSPs> table.</p> <ul style="list-style-type: none"> <code>-opt-lsps-table</code> <code>-fit-lsps-table</code> <code>-fix-lsps-table</code> <p>WAE Design uses these tables in the order they are listed. All LSPs not specified are marked as ignored.</p> <p>If <code>false</code> (default), use existing column entries.</p>
	<code>-parameter-namespace</code>	<p>Initializes and expects columns in plan file under this namespace. This means that all parameters used by the tool follow the specified namespace. For example, by default the required LSP BW can be specified by the column <code>TEOpt::BWRReq</code> in the <LSPs> table.</p> <p>Default is <code>TEOpt</code>.</p>
	<code>-queue</code>	Queue to use when the <code>-init-lsp-bwreq</code> or <code>-init-int-bwbound</code> option needs traffic.
	<code>-service-class</code>	Service class to use the <code>-init-lsp-bwreq</code> or <code>-init-int-bwbound</code> option needs traffic. Default is the undifferentiated class.
	<code>-set-bw-prim</code>	<p>How to set the LSP SetupBW property in the output file.</p> <ul style="list-style-type: none"> <code>init-lsp-bwreq</code> - Set the LSP SetupBW property based on the required LSP bandwidth. <code>keep</code> - Keep existing LSP SetupBW properties. <code>zero</code> (default) - Set the LSP SetupBW property to 0.
	<code>-traffic-level</code>	Traffic level to use when the <code>-init-lsp-bwreq</code> or <code>-init-int-bwbound</code> option needs traffic.
<code>sam_getplan</code>	<code>-flume-port</code>	Port number to which the flume agent is bound. Default is 41417.
	<code>-flume-server</code>	Name or address for the server running a flume agent. Default is localhost.
<code>trim_nodes</code>	<code>-exclude-node-table</code>	If <code>true</code> (default), node-table entries are excluded (trimmed). If <code>false</code> , node-table entries are included (kept), and all other nodes are excluded (trimmed).
<code>sam_getplan</code>	<code>-flume-port</code>	Port number to which the flume agent is bound. Default is 41417.

New Values for CLI Options

CLI Tool	Option	New Value	Description
build_topology	-lag-port-match	complete	Match deterministically based on LACP first and then try to match as many as possible.
parse_configs	-include-objects	sr_lsps	Segment Routing LSP paths.
snmp_find_interfaces	-lag-port-match	complete	Match deterministically based on LACP first and then try to match as many as possible.

Changed CLI Behavior

CLI Tool	Option	Value	New Behavior
copy_from_template	-visualL1	true	Additionally copy the <L1Waypoints> table.
	-missingL1	replace	If both -missingL1 is replace and if -method is missing, additionally copy the <L1Waypoints> table.
	-method	missing	
dmd_mesh_creator	-service-class		If omitted, use the default class. (Previously, this was use the default BE class.) You can now specify multiple commas-separated service classes to create multiple demand meshes, one for each service class.
flow_get	-list-extra-aggregation-keys	true/ false	Previously, you listed the keys. Now you set true to act on the keys defined by -extra-aggregation. The default is false.
import_layer1			Additionally copy the <L1Waypoints> table
latency_distance_init			Use L1 waypoints in the calculation of L1 distance.
mate_tech_support			Includes wae-ni and wae-core log files, where in the past it only included Collector server and WAE Live log files.
rsvp_te_opt	-opt-lsps-table	<file>	<p>The -opt-lsps-table is no longer tied to -lsps-table since that option was removed. Whether -opt-lsps-table is used is now defined by the -init-lsp-groups option.</p> <p>While the purpose of the option is the same, it now optimizes LSPs based on the new options. See the "New CLI Options" section.</p>
snmp_find_interfaces	-lag-port-match	exact	Now excludes every port that is up, but whose LACP state indicates that it is not a LAG member
trim_nodes	-node-table	<file>	File containing <Nodes> table of nodes to either exclude (trim) or include (keep),

CLI Tool	Option	Value	New Behavior
			as determined by <code>-exclude-node-table</code> option. Previously, nodes could only be excluded. Default is that no table is used.

Service Changes

Removed Services

Removed Service	Replacement
<code>wae-collector</code>	<code>wae-ni</code>
<code>wae-cdl</code>	<code>wae-osc</code>

New Services

Service	Description
<code>wae-appenginecore</code>	Service API that manages and routes requests to load the network model to the appropriate <code>wae-designapiserver</code> .
<code>wae-designapiserver</code>	Service that enables <code>wae-appenginecore</code> to use the Design APIs. This wraps the Design APIs and controls Design API instances.
<code>wae-mls</code>	Service that enables the WAE Live datastore server.
<code>wae-ni</code>	Service that enables the northbound WAE NI APIs. WAE NI is used for continuous polling and for continuous PCEP LSP collection. This replaces <code>wae-collector</code> .
<code>wae-osc</code>	Service that enables the OSC (Open SDN Controller) server. This replaces <code>wae-cdl</code> .
<code>wae-svcs-dashui</code>	Service that controls the UI dashboard used for displaying logs and diagnostics in the Statistics UI.
<code>wae-svcs-db</code>	Service that enables the datastore that stores statistic information.
<code>wae-svcs-logagent</code>	Service that forwards log entries from client applications to <code>wae-svcs-db</code> .
<code>wae-svcs-metricsbkr</code>	Service that receives the collected diagnostic entries from client applications to <code>wae-svcs-db</code> .
<code>wae-svcs-metricsd</code>	Service that collects diagnostic entries from client applications.
<code>wae-svcs-mon</code>	Service that monitors all services and automatically restarts them in the event of ungraceful terminations (such as with a <code>kill</code> command).
<code>wae-svcs-ui</code>	Service used to enable the Statistics UI.

Open Source

This product includes the following.

- Software developed by MetaStuff (<http://www.dom4j.org>)
- Cryptographic software written by Eric Young (eay@cryptsoft.com)
- Software developed by the OpenSSL project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)
- Software written by Tim Hudson (tjh@cryptsoft.com)
- Software developed by the University of California, Berkeley and its contributors

Issues Fixed

Issues Fixed Since 6.1.3

WAE Live

Key	Summary
CSCus63504	Errant failure message when network is successfully added
CSCus75675	Performance issues when inserting QoS data
CSCus79594	Unable to run reports based on filters in text fields
CSCut02726	Inventory "Go to Interface" does not work with Chrome
CSCut27859	Errant snapshot and view option messages in Map
CSCuu70260	CSV export does not wrap general text data in a single field text within quotes (")
CSCuu81534	Inventory is not working due to conflicts between cache building and cache clean-up tasks

WAE Collector

Key	Summary
CSCur90796	<code>poll_ldp</code> uses IPManage in the <Nodes> table instead of the system IP address to match the LSPID in the <NetIntLdp> table
CSCus32202	<code>snmp_find_vpn</code> does not discover ASR 9000 routers in L2 VPNs
CSCus81204	<code>parse_igp -igp-protocol isisv6</code> does not discover multi-topology IS-IS IPv6 adjacencies
CSCut70452	<code>parse_configs</code> does not populate management IP address on IOX-XR
CSCut72707	<code>snmp_find_multicast</code> is not giving accurate results
CSCuu36698	<code>parse_configs</code> after running <code>login_find_igp</code> creates duplicate entries
CSCuu68564	<code>parse_configs</code> creating incorrect SRLGs on Juniper routers
Collector Server	
CSCur89815	Interface queue measurements are collected without WAE Collector being configured to do so
CSCus50384	Snapshot directory is deleted and no plan files are available
WAE NI Server (In 6.1, this was the "Continuous Poller server")	
CSCut19158	Temporary files created are not removed from <code>~/ .cariden/tmp/continuous_poller/temporary_plan_files</code>
CSCut19275	Temporary files are saved in <code>~/ home ~/ .cariden/tmp</code>

WAE System

Key	Summary
CSCus63082	A vulnerability in Apache CXF could allow an unauthenticated, remote attacker to cause a denial of service (DoS) condition
CSCut12319	Exception errors occur when creating a new bookmark
CSCut46092	OpenSSL vulnerabilities released by the OpenSSL Group
CSCut91029	Installer binary modifies the shell user's python path variable
CSCuu72229	Web server vulnerability to SSLv3 POODLE (Padding Oracle On Downgraded Legacy Encryption)

Issues Fixed Since 6.1.2**WAE Design**

Key	Summary
CSCuu37447	Incorrect simulation of RSVP TE LSPs when switching from actual path hops to loose hops
CSCuu42438	<code>lsp_diagnostics</code> does not report reason simulation is not following shortest TE path
CSCuu48851	When running <code>exp_opt</code> , the ending report and actual results differ
CSCuu51718	Layer 1 circuit text is corrupted when L1 circuits are not horizontal

WAE Collector

Key	Summary
CSCuu37196	<code>parse_configs</code> is not populating LSP destinations

Issues Fixed Since 6.1.1**WAE Design**

Key	Summary
CSCuu12804	Compare Traffic reports returns "na" for Util Sim values on all interfaces

WAE Live

Key	Summary
CSCut16685	Map log files rapidly growing with "TabPlanParser.parse invalid file format" errors

WAE Collector

Key	Summary
CSCuu15994	Enabling <code>find_bgp</code> causes fatal error in snapshot process

WAE Core

Key	Summary
CSCut06750	PCEP tunnels are set to autoroute disabled in XR 5.3.0
CSCut70387	Unable to upload large plan files
CSCuu12960	Primary and secondary paths are not correct for PCEP tunnels

Issues Fixed Since 6.1

WAE Design

Key	Summary
CSCus69128	Timestamp on LSP disjointness report is inaccurate
CSCus87792	Nodes are crowded on top of each other when opening a plan file from a previous release
CSCut25481	L1 circuit's Lambda Blocking inaccurately set to "na" when it should be "T" (true)
CSCut51807	Changes are not deployed to the network

WAE Live

Key	Summary
CSCur83136	Inaccurate aggregate calculations over raw data when there are measurement gaps
CSCus52134	Exception error occurs when going to Settings page
CSCus95849	Adding a network fails when the web server is started as a service
CSCut02942	Data collection from external archive is not working
CSCut14286	Network is not fully deleted by Network Manager
CSCut45783	Unable to add notes for demands

WAE Collector

Key	Summary
CSCut60246	Update netaccess.txt to poll for Alcatel-Lucent nodes
CSCut63202	PCEP LSPs are not discovered for JunOS
Collector Server	
CSCut19296	Unable to remove unique suffix from node names
WAE NI Server	
CSCut21594	Generates incorrect traffic statistics

WAE System

Key	Summary
CSCut10956	Improper check for Archive license

Known Limitations

Applications

WAE Design

In some Linux installations with Xfce desktop installed, the documentation does not open from the WAE Design GUI Help menu. The work around is to do one of the following.

- Open the help files from a terminal in the `$CARIDEN_HOME/docs` directory.
- Install a default browser.
- Install the following packages.

```
yum install evince
yum groupinstall "X Window System"
yum groupinstall "Desktop"
yum groupinstall "General Purpose Desktop"
```

WAE Live

- L2 interface types are categorized incorrectly as "individual physical interfaces" on the Explore Interfaces page.
- The "Unknown" interface type is not used.

WAE Platform

WAE Collector

Due to vendor MIB limitations, Collector cannot represent QoS traffic on interfaces that have more than one VLAN configured. If a network contains such interfaces, their queue traffic statistics are omitted from the collection. The total traffic on these interfaces is still measured. As a result, per class-of-service demands estimated through Demand Deduction are less accurate. Estimates of traffic totals over all classes of services, however, are not affected.

Due to lack of MIB support, SR tunnel type is not collected for IOS XR routers through SNMP.

Collector Server

- If upgrading the Collector server from release 5.6x to 6.1x or 6.2, the `$CARIDEN_ROOT/etc/collector/server/db-persistence/DiscoveryEngineImplementation.db` file must be removed prior to starting the web server. Since installation automatically starts the web server, the recommendation is to remove this prior to installation.
- OSPFv3 and IPv6 IS-IS databases cannot be collected. The workaround is to use a manual snapshot.
- SNMPv3 device access is not supported. The workaround is to use a manual snapshot and `mate_auth_init`.

WAI NI Server

- Continuous PCEP LSP collection does not support SR tunnels. Note that you can collect SR tunnels through PCEP using the Collector server or snapshots.

Snapshots

- `snmp_find_interfaces`
 - Does not support association of GRE tunnel with the physical interface it uses to reach the tunnel destination since the IP-Tunnel MIB lacks this information.

- Does not update LAG port status if LAGs are discovered using both `parse_configs` and `snmp_find_interfaces`. The workaround is to use only `snmp_find_interfaces`.
- Juniper routers: Signaled standby LSP path option is not available from the standard MPLS-TE MIB for Juniper routers. Only the active path option name is collected.
- IOS XR routers
 - IGP topology collected through `parse_igp` and `login_find_igp_db`
 - IS-IS link-state database with TE extensions contains incorrect interface admin-weights (TE metric) on Intel-based routers.
 - IPv6 IS-IS link-state database does not contain IPv6 interface addresses or parallel interfaces. This information is only available when IOS XR supports IS-IS IPv6 TE extensions. The `snmp_find_interfaces` tool collects this information.
 - MAC Accounting is not supported.
 - `snmp_find_rsvp`
 - Does not set the Standby value in the <LSPPaths> table for signaled backup paths.
 - Does not collect named affinities configured with affinity-maps.
- BGP peers
 - `find_bgp` does not build BGP pseudo-nodes among internal ASNs.
 - `find_bgp` does not collect BGP peers under PE-CE VRFs.
- `parse_configs` does not accurately detect the bandwidth of some Juniper 'ge' interfaces that have a capacity of 10 Gbps.
- TE Extended Admin Groups (EAGs), also known as extended affinities, are not supported.
- Port circuits are not built for LAG/bundle members whose nodes are not within the same IGP instance as the AS.
- There is no support for building port circuits for LAG members that are not within the same IGP (inter-AS circuits)
- It is not possible to distinguish between physically connected and unconnected LAG ports that are down for LAG port matching.
- `parse_configs` collects POS bundles, but has limitations due to unavailability of the port `OperStatus` property.
- `snmp_find_ospf_db` cannot be used when routers have a large number of links that cannot fit into a single PDU.
- `find_bgpls` does not support multi-area OSPF or multi-level IS-IS, non-TE-enabled interfaces, and pseudo-nodes. The workaround is to use SNMP- or login-based discovery.
- `get_inventory` does not collect Juniper multi-chassis router hardware inventory.
- Segment routing
 - SR LAN adjacency SIDs and pseudo-node creation are not supported.
 - SR protected adjacency SIDs are not supported.
 - Concurrent RSVP-TE and SR-TE paths are not supported on the same LSP.

SAM-OSS Integration with Snapshots

- `sam_getplan` does not populate the <NodeTraffic> table. This table is derived and populated when `sam_getplan` and SNMP tools are used together.
- `sam_getplan` does not populate the NetIntActivePath column in the <LSPs> table.
- If `sam_getplan` and SNMP tools are used together in the snapshot process for multi-vendor network collection, then Alcatel-Lucent traffic measurements cannot be aligned with those collected from other router platforms.

Deployer Module

OSC Controller

- During detailed PCEP tunnel creation or when modifying PCEP tunnels, affinity values are misinterpreted if multiple affinities are specified. This limits you to specifying one affinity for IncludeAffinity, IncludeAnyAffinity, and ExcludeAffinity, and each of these values must be a number within [0,31].

NSO Controller

- LSPs that exist in the network by another controller cannot be updated.
- IOS XR: WAE client specifies the XR LSP signaled-name, while NSO service and device use tunnel-id. The workaround is to deploy all IOS XR LSPs using the tunnel-id and to make sure that existing LSPs are not redeployed.
- LSPs that have a named-path used by another LSP cannot be deleted.
- NEDs (NSO console)
 - IOS XR
 - No option to give the IP address of the LSP directly, you can only specify a loopback address.
 - No option to give tunnel affinity values directly; you can only specify an affinity-map name.
 - Junos: Does not have the inter-domain keyword, which is used only when an inter-area LSP is created.

WAE System

Installation and Startup

- The WAE NI server and the WAE Core server cannot reside on the same device or on the same VM. Note that the *WAE Server Installation Guide* assumes that they are on the same device. If needed, please contact your support representative for further installation details.
- If the OS is using an old CA certificate to verify the integrity of the EPEL repository, you might see this error from the OS vendor.

Error: Cannot retrieve metalink for repository: epel. Please verify its path and try again.

- One possible workaround is to perform an offline installation. For instructions, refer to the Offline Installation chapter in the *WAE Server Installation Guide*.
- Another possible workaround is to change `https` to `http`.

Note: This is not a secure solution. For information on how to resolve OS security issues, contact your OS vendor.

1. In the `/etc/yum.repos.d/epel.repo` file, change the first instance of `https` to `http`.

```
sudo vim /etc/yum.repos.d/epel.repo
```

Change `https` to `http` in the following line.

```
mirrorlist=[https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch]
```

2. Execute `yum` to clean up `makecache`.

```
sudo yum clean all && yum makecache
```

3. Re-run the installer. For more detailed installation instructions, see the *WAE Server Installation Guide*.

```
sudo bash wae-k9-<version>.bin
```

- The `$CARIDEN_HOME` directory is not automatically added to `$PATH` (only `$CARIDEN_HOME/bin` is). If not in `$CARIDEN_HOME/bin`, to start the WAE Design GUI from the command line, you must specify its full path.

```
/opt/cariden/software/mate/current/mate
```

Web Server

- The `embedded_web_server` tool is being deprecated. The recommendation is to use the `wae-web-server` service, which is constantly monitored to be brought up automatically.

By default, this web service starts upon installation completion. Therefore, if you stop the web server using the `embedded_web_server` tool (`embedded_web_server -action stop`), the web server does not stop. The workaround is the following.

```
service wae-svcs-mon stop
embedded_web_server -action stop
```

WAE Statistics UI

- The WAE Statistics page does not open in all browsers. The workaround is the following.

1. Click the WAE Statistics link. The URL format is <https://<server IP>:8443>.

Example: <https://171.71.147.14:8443>

2. Copy the URL of this page to another browser window.
3. In this new browser, change the URL's port from 8443 to 8843.

Example: <https://171.71.147.14:8843>

4. Follow the browser's messages to accept the connection and add it as an exception.

Web User Management

Both the System UI and the WAE Design Archive UI have local user management capabilities. If both are used to configure users, WAE uses the most recently updated information. The recommendation is to use only the System UI to manage local users.

License Check Failures on Newer Linux Distributions

Some newer Linux distributions have started using a new way (via `biosdevname`) of naming hardware devices, including the network interfaces. This causes some software that depends on the traditional naming (for example, `eth0`, `eth1`) to fail on license checks, including WAE.

The workaround is to append `biosdevname=0` to the kernel line of the grub configuration file and reboot. (Syntax varies among distributions.)

After reboot, you should be able use `ifconfig` to verify that the NIC are named `eth0` (or `eth1`, ...) instead of the `biosdevname` names (such as `p34p1`).

Java Memory

Certain tools (such as `sam_getplan` and `parse_configs`, for example) may require more memory to start than what is available. The symptom is an error message similar to the following.

```
Error occurred during initialization of VM.  
Could not reserve enough space for object heap.  
Error: Could not create the Java Virtual Machine.  
Error: A fatal exception has occurred. Program will exit.
```

The workaround is to set the maximum memory to a low enough value in the `CARIDEN_JAVA_OPTIONS` variable before calling the tool. An example setting is as follows.

```
set CARIDEN_JAVA_OPTIONS=-Xmx1000m
```

Documentation

- *WAE Server Installation Guide*
 - Does not identify package names as being either WAE Automation or WAE Planning. As such, it identifies the Linux package as `wae-k9`, when Linux packages are either `wae-planning-k9-6.2.0` or `wae-automation-k9-6.2.0.bin`, depending on what you are installing.
 - Does not identify that the WAE Automation package requires a dual-server installation.
 - Contact your support representative for information on what these packages contain and for dual-server installation details.
- The *WAE Design GUI Installation Guide* does not identify WAE Design as being part of the WAE Planning package. For the Linux package, `wae-k9` should be `wae-planning-k9-6.2.0`.
- WAE Collector inventory collection configuration is documented accurately, but it is not based on the use of the shipped default hardware inventory configuration files.
- The documentation is missing the following. For information, contact your support representative.
 - WAE Live 6.1 inventory features
 - WAE Design RSVP TE Opt tool
 - New or changed WAE Core configurations (`.cfg` file configurations) for OSC extensions, LSP deployment through NSO controller, `wae-appenginecore` (which exposes Design APIs through REST), WAE REST API credentials, and calendaring projections
 - SR LSP collection
 - NSO integration

August 2015, Version: 3

For further details, please visit www.cisco.com/go/wae.




Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

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.

 Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)