

Zenoss Foglight Agent

Custom Integrations



Why Zenoss from LightSpeed PM?

Foglight requires custom services to integrate with Zenoss.
Without our custom cartridge these steps must be taken:

- Custom Event Driven Rule
 - Command Invocation
 - Actions
- Extensive Groovy Scripting for more control over sent events and other important metrics
- Defining Zenoss Fields in Foglight



Zenoss Integration With LightSpeed PM's Custom Cartridge



Zenoss – Highlights

Pattern Editor allows easy access to information related to the alarm

- No Groovy required
- Access to dozens of fields to easily provide values to parameters
- A dozen common operators available to easily avoid any coding

Use of Groovy scripts for power users

- Use of Groovy scripting with complete injected flow context
- Ability to reject/delay ticket generation

Persisted Queue so no events are lost

- Guaranteed delivery
- Keep history for rejected, failed and successful operations for total control
- Pause or delay control
- Consumer Thread settings/control to manage concurrency and volume



Zenoss Integration – Alarming Features

Auto Acknowledge Alarms

Alarm Flood Prevention

Alarm Transition Support

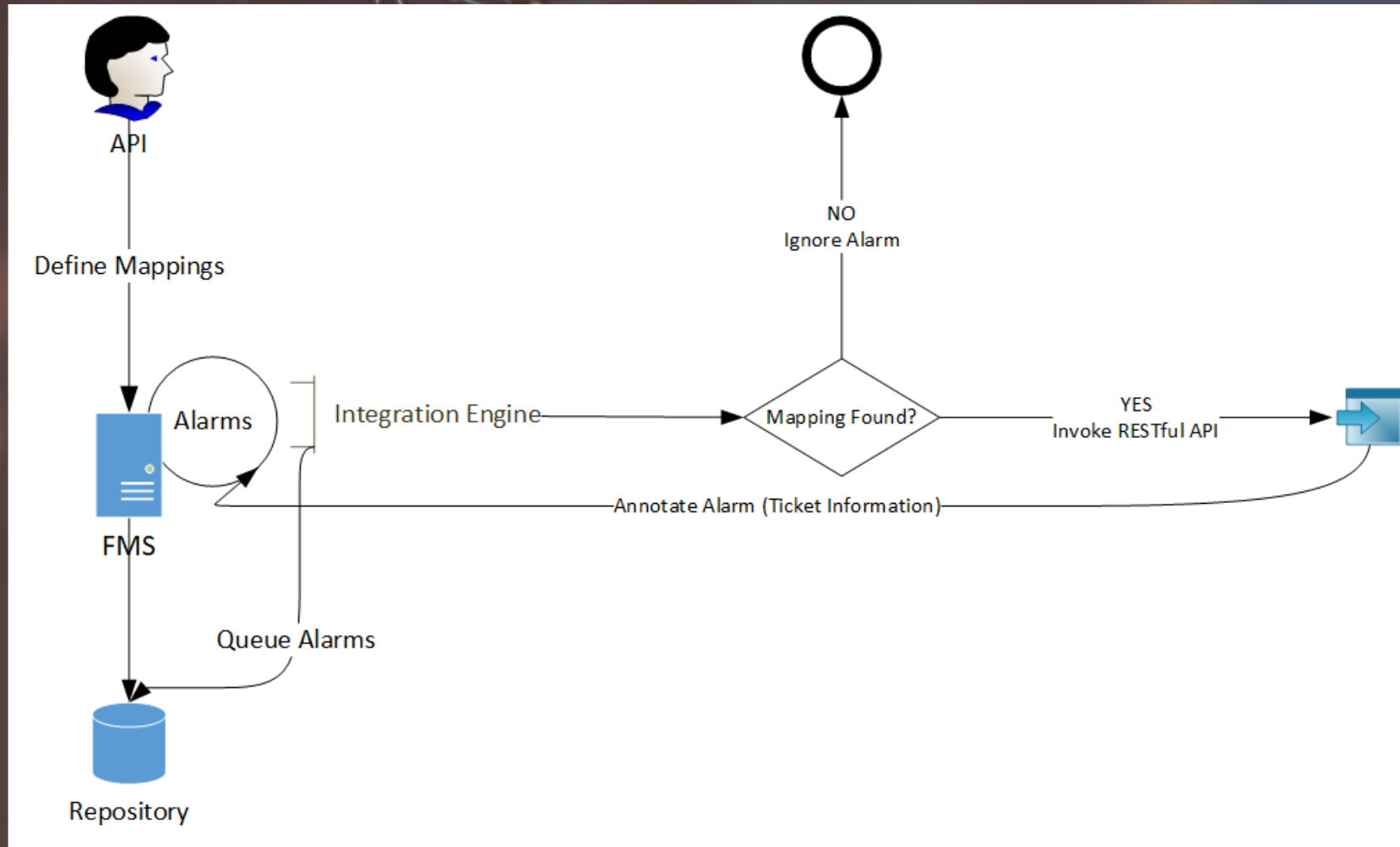
Can be Turned ON or
OFF

Update tickets when
alarm is transitioning
from one severity to
another.
Can be either to higher
severity only or any
transition

Close ticket only when
the last alarm in
transition has been
cleared

All transition information
is accessible from the
Pattern Editor

Zenoss Integration – RESTful API Flow



Zenoss Integration – Decision Engine

A rich UI within the solution presents the various mappings produced by the decision engine

Low Priority

No Service

High Priority

With Service

	Service	Host	Agent Type	Agent	Topology Object	Rule
0						
1		x				
2			x			
3				x		
4						x
5					x	
6		x				x
7			x			x
8				x		x
9					x	x
10	x					
11	x	x				
12	x		x			
13	x			x		
14	x					x
15	x				x	
16	x	x				x
17	x		x			x
18	x			x		x
19	x				x	x

Zenoss Integration – Changing Settings

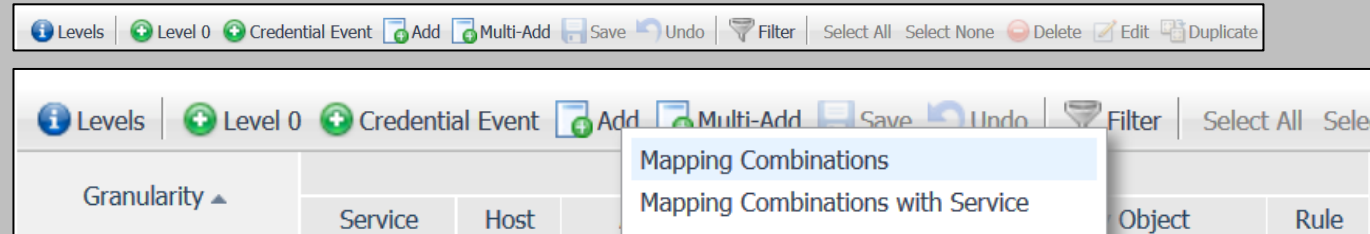
The screenshot displays the 'Integration Pack Registry Settings - Local FMS' window. The interface includes a navigation pane on the left with options like 'Expert View', 'Bookmarks', 'Homes', 'Dashboards', and 'Flight Servers'. The main area shows a table of settings for the 'Zenoss Integration' pack, categorized into sections like 'Open Ticket', 'Queue', 'Ignore', 'Delay', 'REST', and 'Authentication'. Each setting has a name, a value, and a description. The 'REST' section is currently expanded, showing settings for enabling REST, the URL, path, update path, update method, response properties, and trust certificates.

Name	Value	Scoping	Description
Zenoss Integration			
Open Ticket			
PSO.ZenossIntegration.OpenTicket.NoEmptyValue	true		If true any tag that returns a null will be replaced by a space. If false null values are simply not outputted.
PSO.ZenossIntegration.OpenTicket.TimeZone	GMT		Defines the time zone to use when outputting date and time in the Command.
PSO.ZenossIntegration.OpenTicket.DateTimeFormat	dd/MM/yyyy:dd:mm:ss		Defines the format for the date and time to use when outputting date and time in the Command.
Queue			
Ignore			
PSO.ZenossIntegration.Queue.Ignore.Normal	true		Should cleared alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load.
PSO.ZenossIntegration.Queue.Ignore.Warning	false		Should Warning alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load.
PSO.ZenossIntegration.Queue.Ignore.Critical	false		Should Critical alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load.
PSO.ZenossIntegration.Queue.Ignore.Fatal	true		Should Fatal alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load.
PSO.ZenossIntegration.Queue.Ignore.Rules			A list of comma separated rule names that the queue should ignore. Alarms generated by ignored rules are not queued.
Delay			
PSO.ZenossIntegration.Queue.Alarm.DelayPeriod.Warning	0		Time in minutes a warning alarm should be delayed in the queue before being forwarded to the target system if supported.
PSO.ZenossIntegration.Queue.Alarm.DelayPeriod.Critical	0		Time in minutes a critical alarm should be delayed in the queue before being forwarded to the target system if supported.
PSO.ZenossIntegration.Queue.Alarm.DelayPeriod.Fatal	0		Time in minutes a fatal alarm should be delayed in the queue before being forwarded to the target system if supported.
PSO.ZenossIntegration.Queue.AlarmTransition.Enable	false		Should alarm transition be supported. If enabled the integration will update a ticket vs creating a new one when an alarm transitions.
PSO.ZenossIntegration.Queue.Alarm.AcknowledgeUponSuccess	false		Should the alarm be acknowledged when the alarm has been successfully processed.
PSO.ZenossIntegration.Queue.Alarm.EnforceServiceFilter	true		When an alarm is being processed enforce the filters applied to the service. If true a service is not considered to be in a state of alarm until it has been processed.
PSO.ZenossIntegration.Queue.PausePeriod	0		Time in minutes no alarms should be forwarded to the target system. This is useful when the target system is down.
REST			
PSO.ZenossIntegration.REST.Enable	false		Should the integration use the REST method (true)
PSO.ZenossIntegration.REST.URL			The URL for the RESTful API the integration should use in order to open a ticket.
PSO.ZenossIntegration.REST.Path			The path following the URL.
PSO.ZenossIntegration.REST.Update.Path			The path following the URL to use when an alarm is cleared or if Alarm Transition is enabled when an alarm transitions.
PSO.ZenossIntegration.REST.Update.Method	POST		Which method to use when doing an update: POST or PUT
PSO.ZenossIntegration.REST.ResponseProperty			The name of the property to retrieve the Zenoss ticket ID after opening a ticket.
PSO.ZenossIntegration.REST.OtherResponseProperties			A comma separated list of properties to retrieve from the REST response. These properties are secondary to the ticket ID.
PSO.ZenossIntegration.REST.TrustAllSSL.Certs	false		If true all SSL certificates will be trusted. Should only be used in Development environments where certificates are not trusted.
Authentication			
PSO.ZenossIntegration.REST.AuthenticationMode	HTTP_HEADER		HTTP_HEADER, BASE64_ENCODING or URL_ENCODE authentication when not tokens are needed, or BASE64_ENCODING when tokens are needed.
PSO.ZenossIntegration.REST.Username			The username to gain access to the RESTful API the integration should use in order to open a ticket.
PSO.ZenossIntegration.REST.Password	*****		The password to gain access to the RESTful API the integration should use in order to open a ticket.
PSO.ZenossIntegration.REST.Authentication.URLEncodeFormat			The format in which to send the login information. property=(username) or (password). It is possible to add other properties as well.
PSO.ZenossIntegration.REST.Authentication.Token.URL			The URL to use for authentication. Should be used if a 3rd party login needs to be used. Use only if AuthenticationMode is set to TOKEN.

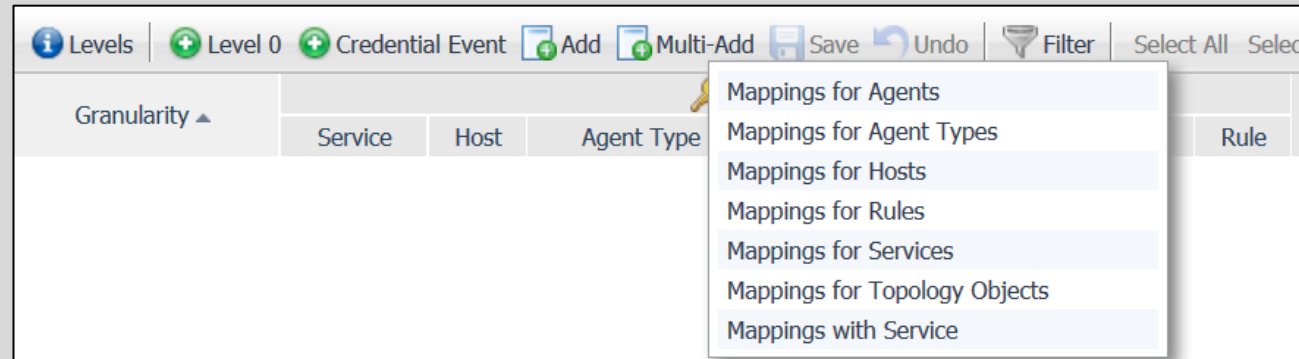


Zenoss Integration – Mappings Editor

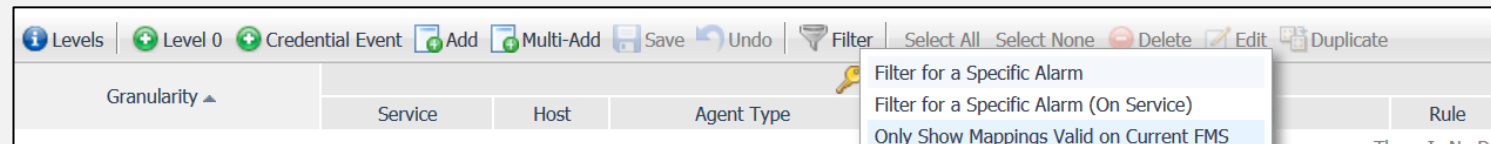
Combinations



Multiple Mappings



Filtering



Corrective Message :			
Duration :			
Expires On :			
Message :			
Severity :			
Source :			
Timestamp :			
Type :			
Uuid :			
Enable Normal : true			
Warning : true			
Critical : true			
Fatal : true			

Update

Update Selectively

Reset

Clear

Cancel

Zenoss Integration — Field Editor

Zenoss Integration — Pattern Editor

Alarm Severity Name Elements

Add

Insert

Edit Complete

Save

Undo

Pattern Element

@ALARM_SEVERITY_NAME

Remember to click Apply if you manually edit the pattern.
Spaces are automatically trimmed, please use Edit Complete to insert spaces and have them preserved.

@ALARM_SEVERITY_NAME

Apply

TAGS

Search

Tag ▲	Caption
AGENT_TYPE	Agent Type
AGENT_UID	Agent UID
ALARM_ACKNOWLEDGED_BY	Alarm Acknowledged By
ALARM_ACKNOWLEDGED_TIME	Alarm Acknowledged Time
ALARM_CLEARED_BY	Alarm Cleared By
ALARM_CLEARED_TIME	Alarm Cleared Time
ALARM_CREATED_TIME	Alarm Created Time
ALARM_ID	Alarm ID
ALARM_IS_ACKNOWLEDGED	Is Alarm Acknowledged
ALARM_IS_CLEARED	Is Alarm Cleared
ALARM_LINK	Alarm Link

OPERATORS

Operator	Short Description
?[]	If tag has a value.
:[]	If tag does not have a value.
?[]:[]	If tag has a value else.
^{}{}	Replace string with another.
=0	Equality
=0?[]:[]	If equal else
![[]]	Groovy Script
^U	Uppercase
^L	Lowercase
^T	Trim
^N	New Line Output
^R	Return

Zenoss Integration — System Requirements



Minimum required
FMS version

5.9.3



Supported
Database
Minimum
Version

MS SQL

2008
v 10.0.1600 or later

Oracle

9i R2

MySQL

5.1.45

PostgreSQL

9.4.0





Performance Monitoring customized to your unique environment

Ten plus years of providing Professional Services to Quest customers revealed these enhancements to be most requested modifications.

Let us take Foglight's out of the box capabilities and enhance for your unique environment.

Contact: Sales@LightSpeedPM.com

