

ServiceNow (Advanced) Integration



Custom
Integrations

Why ServiceNow (Advanced) from LightSpeed PM?

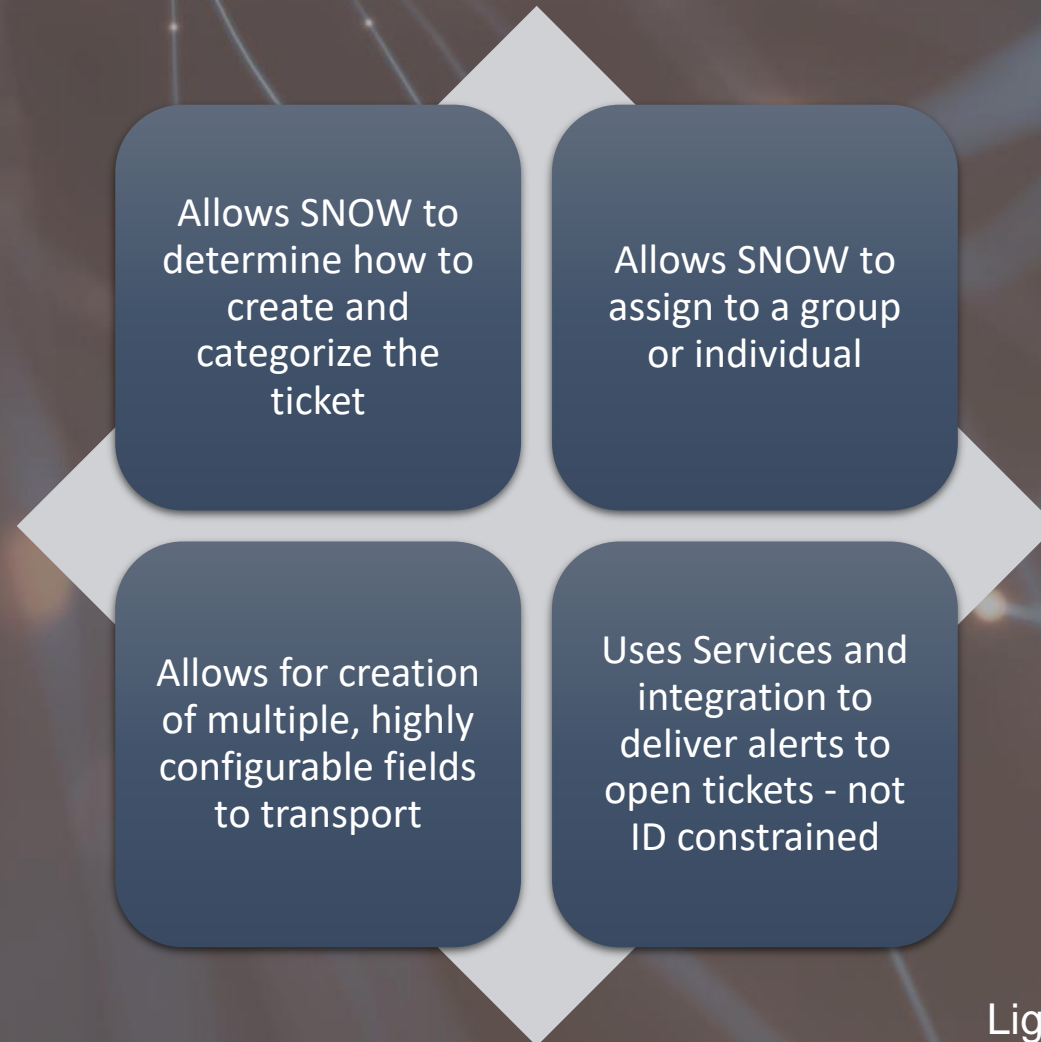
Foglight's recently released out of the box solution is available at no cost when you purchase Foglight. Our custom integration services remove these additional challenges:

- Ticket assignment settings must be manually updated on each Foglight server as users and groups change in ServiceNow
- Minimal flexibility in ticket creation and categorization
- Ticket fields are restricted to a fixed set of values defined by the Foglight alarm
- Matrix comparison of features found on pages 15 and 16



ServiceNow (Advanced) Integration

Expand the capabilities of ticket creation and updates



Simplify ServiceNow Integration With LightSpeed PM's Custom Cartridge

Advanced Rich UI for All Functionalities

Customizable REST Invocation Support

- Create/Update tickets
- Ability to export data
- Flexible parameter definitions
- Auto Annotate Alarms with returned information

Customizable Command Line Invocation Support

Remote Administration of solution across servers

Decision Engine for total / precise control

- 20 levels of granularity
- Severity Level control



ServiceNow (Advanced) Integration Highlights

Pattern Editor: easily access the 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 as to manage concurrency and volume.

ServiceNow (Advanced) Integration Alarming Features

Auto Acknowledge Alarms

Alarm Flood Prevention

Alarm Transition Support

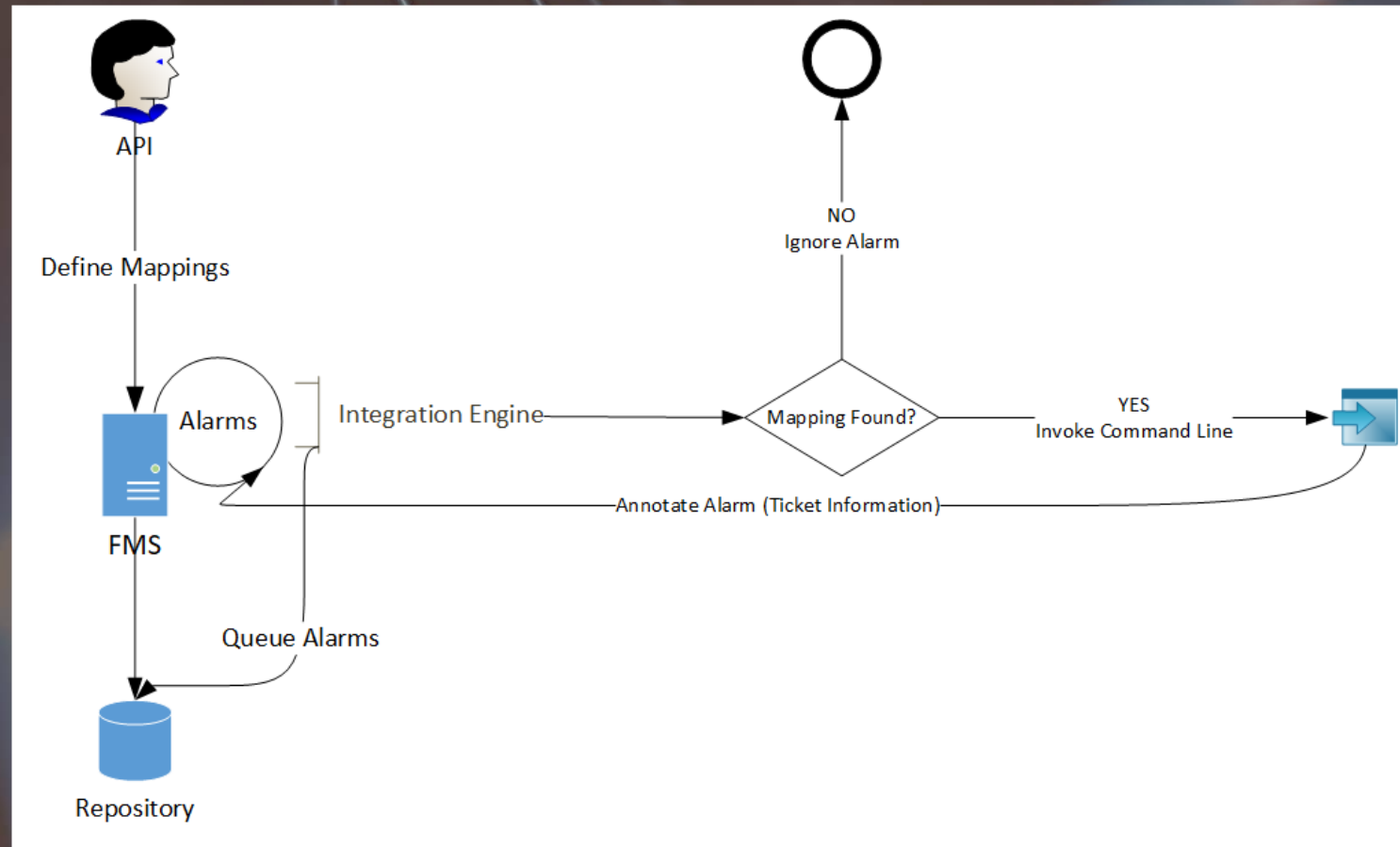
Can be Turned ON or
OFF

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

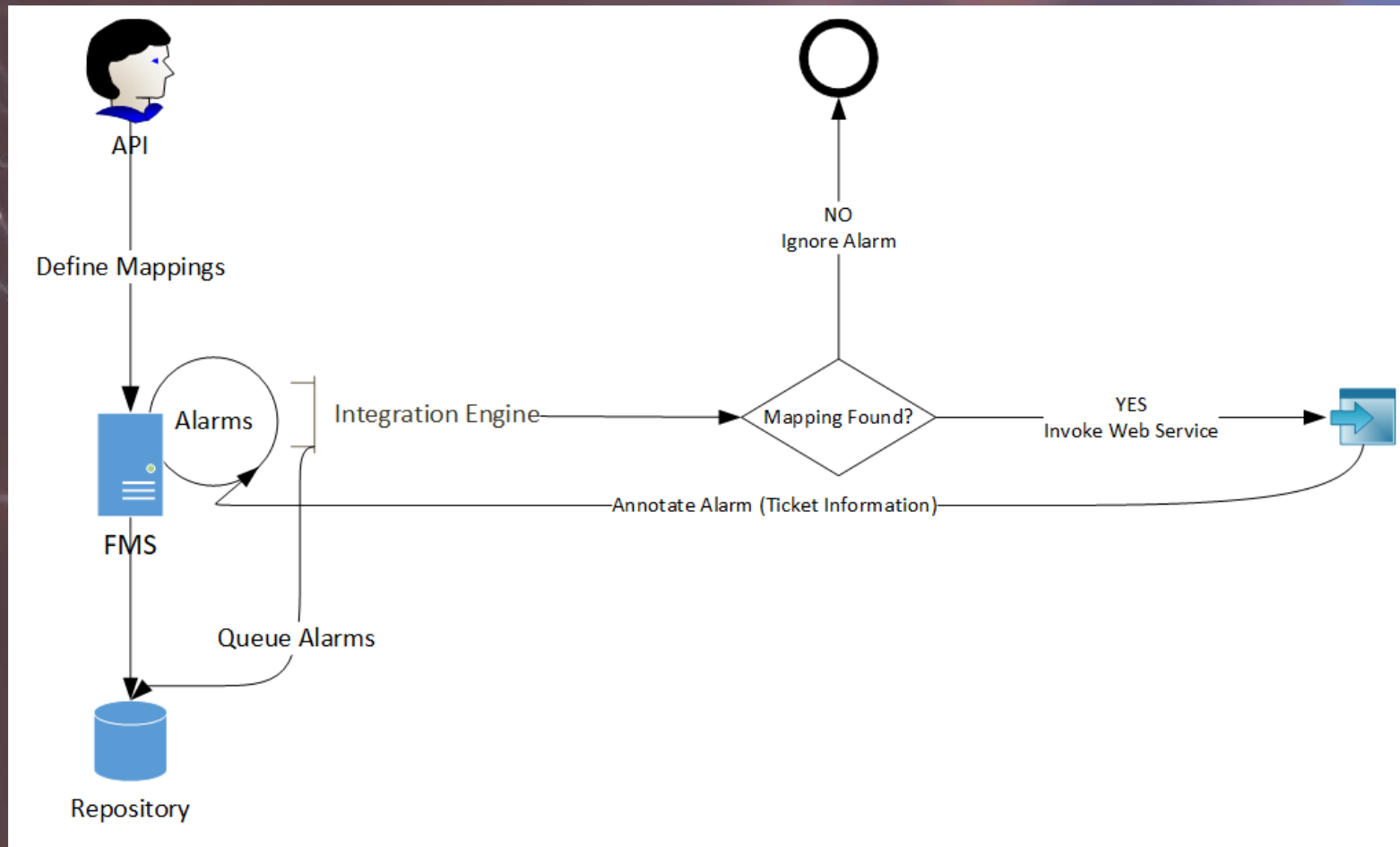
Close ticket only when
the last alarm in
transition has been
cleared

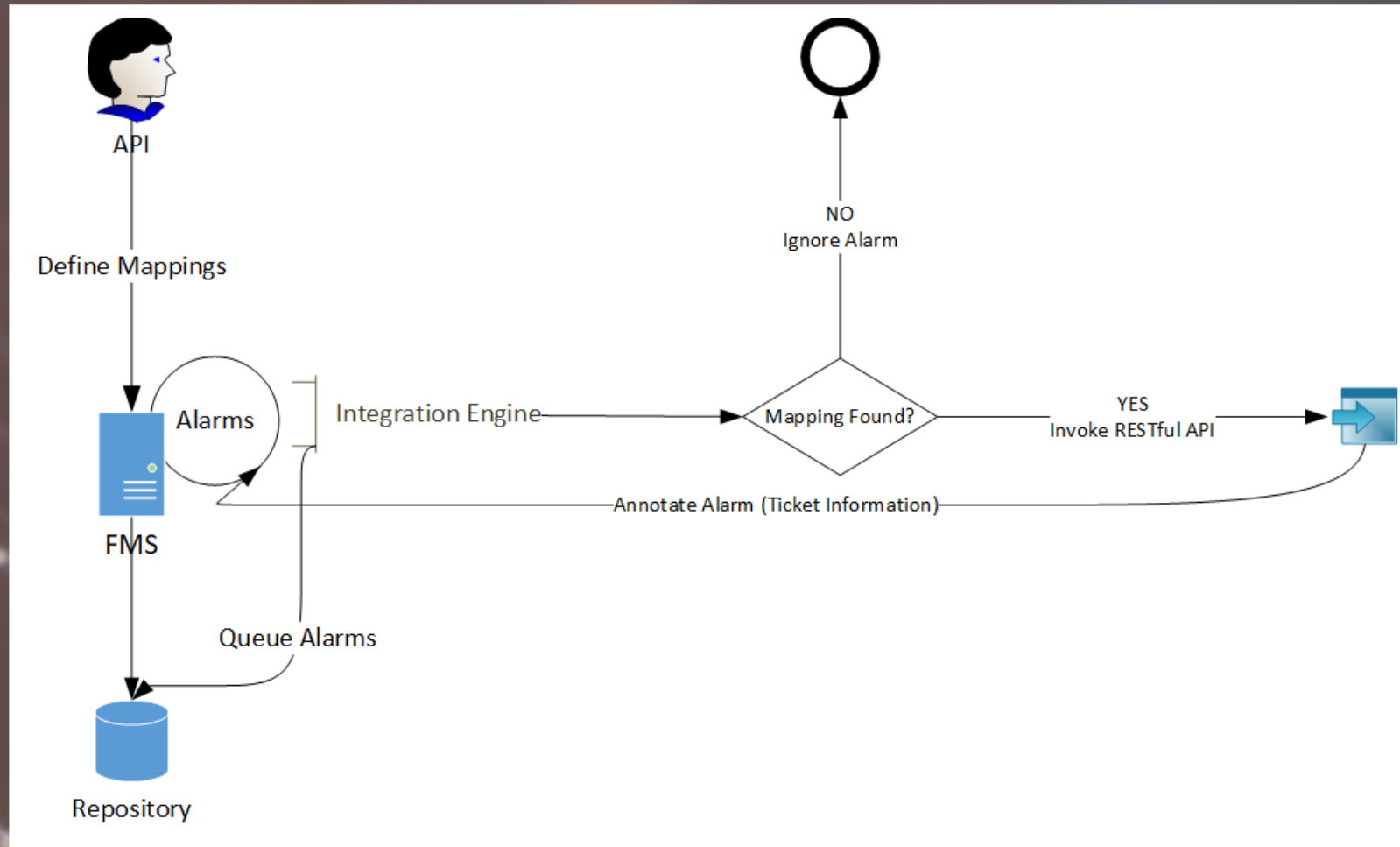
All transition information
is accessible from the
Pattern Editor

ServiceNow (Advanced) Integration Command Line Flow



ServiceNow (Advanced) Integration Web Service Flow





ServiceNow (Advanced) Integration — RESTful API Flow

ServiceNow 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

ServiceNow (Advanced) Integration Changing Settings

Integration Pack Registry Settings - Local FMS

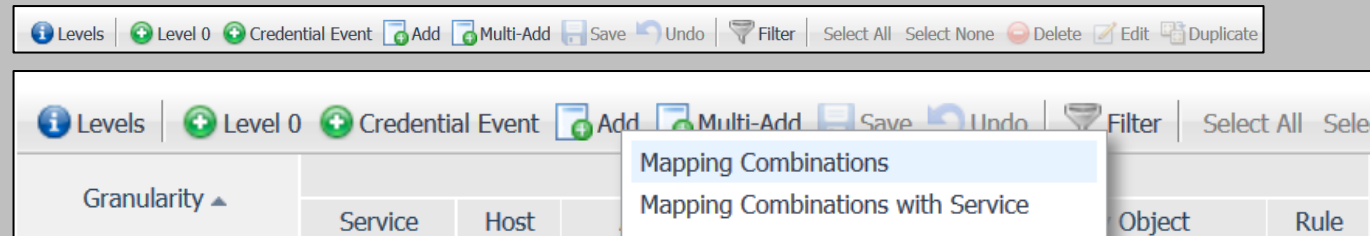
Navigation: Expert View, Bookmarks, Dashboards, Foglight Servers, Local FMS, Host Name

Name	Value	Scoping	Description
ServiceNow Integration			
Open Ticket			
PSO.ServiceNowIntegration.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.ServiceNowIntegration.OpenTicket.TimeZone	GMT		Defines the time zone to use when outputting date and time in the Command.
PSO.ServiceNowIntegration.OpenTicket.DateTimeFormat	dd/MM/yyyyHH:mm:ss		Defines the format for the date and time to use when outputting date and time in the Command.
Queue			
Ignore			
PSO.ServiceNowIntegration.Queue.Ignore.Normal	true		Should cleared alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load on the system and engine.
PSO.ServiceNowIntegration.Queue.Ignore.Warning	false		Should Warning alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load on the system and engine.
PSO.ServiceNowIntegration.Queue.Ignore.Critical	false		Should Critical alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load on the system and engine.
PSO.ServiceNowIntegration.Queue.Ignore.Fatal	false		Should Fatal alarms be ignored by the queuing engine? Ignored alarms are not queued thus reducing the overall load on the system and engine.
PSO.ServiceNowIntegration.Queue.Ignore.Rules			A list of comma separated rule names that the queue should ignore. Alarms generated by ignored rules are not queued thus reducing the overall load on the system and engine. This should not be used to provide a huge
Delay			
PSO.ServiceNowIntegration.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 still active (not cleared) after the delay period expires.
PSO.ServiceNowIntegration.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 still active (not cleared) after the delay period expires.
PSO.ServiceNowIntegration.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 still active (not cleared) after the delay period expires.
PSO.ServiceNowIntegration.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 goes from severity to severity and will also only clear the event when all events have been cleared.
PSO.ServiceNowIntegration.Queue.Alarm.AcknowledgeUponSuccess	false		Should the alarm be acknowledged when the alarm has been successfully processed.
PSO.ServiceNowIntegration.Queue.Alarm.EnforceServiceFilter	false		When an alarm is being processed enforce the filters applied to the service. If true a service is not considered to be a hit (mapping wise) if the alarm is excluded by the filters defined on the service in question.
PSO.ServiceNowIntegration.Queue.PausePeriod	0		Time in minutes no alarms should be forwarded to the target system. This is useful when the target system is unavailable due to maintenance in which case alarms are queued but not processed until the pause period expires.
REST			
PSO.ServiceNowIntegration.REST.Enable	false		Should the integration use the REST method (true)
PSO.ServiceNowIntegration.REST.URL			The URL for the RESTful API the integration should use in order to open a ticket.
PSO.ServiceNowIntegration.REST.Path			The path following the URL.
PSO.ServiceNowIntegration.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 from one severity to another.
PSO.ServiceNowIntegration.REST.Update.Method	POST		Which method to use when doing an update: POST or PUT
PSO.ServiceNowIntegration.REST.ResponseProperty			The name of the property to retrieve the ServiceNow ticket ID after opening a ticket.
PSO.ServiceNowIntegration.REST.OtherResponseProperties			A comma separated list of properties to retrieve from the REST response. These properties are secondary to the main property used to retrieve the Ticket ID.
PSO.ServiceNowIntegration.REST.TrustAllSSLCerts	false		If true all SSL certificates will be trusted. Should only be used in Development environments where certificates are not always kept up to date.
Authentication			
PSO.ServiceNowIntegration.REST.AuthenticationMode	HTTP_HEADER		HTTP_HEADER, BASE64_ENCODING or URL_ENCODE authentication when not tokens are needed, or BASE64_ENCODING_AND_TOKEN and URL_ENCODE_AND_TOKEN where a token is used after a valid login or NONE
PSO.ServiceNowIntegration.REST.Username			The username to gain access to the RESTful API the integration should use in order to open a ticket.
PSO.ServiceNowIntegration.REST.Password	*****		The password to gain access to the RESTful API the integration should use in order to open a ticket.
PSO.ServiceNowIntegration.REST.Authentication.URLEncodeFormat	grant_type=password&client_id=ENTER...		The format in which to send the login information. property=(username) or (password). It is possible to add other properties to send where the values are used as is. property=value. Use only if Authentication mode is URL_ENCODE_AND_TOKEN
PSO.ServiceNowIntegration.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 Authentication mode is BASE64_ENCODING_AND_TOKEN or URL_ENCODE_AND_TOKEN
PSO.ServiceNowIntegration.REST.Authentication.Token.LoginPath	oauth_token.do		The path following the URL to use in order to login and retrieve the token needed for subsequent calls. Use only if Authentication mode is BASE64_ENCODING_AND_TOKEN or URL_ENCODE_AND_TOKEN
PSO.ServiceNowIntegration.REST.Authentication.Token.Body.ContentType			Set only if the login process requires fields to be sent or if the content type must simply be specified.
PSO.ServiceNowIntegration.REST.Authentication.Token.Body.Content			If content type is set to JSON we can define a comma separated list of the fields to send. Format: key=value where value can be a String by surrounding the value with double quotes. Integer by simply providing a number.
PSO.ServiceNowIntegration.REST.Authentication.Token.ResponseProperty	response.access_token		The path.name of the property that is used to return the token after a successful login. Use only if Authentication mode is BASE64_ENCODING_AND_TOKEN or URL_ENCODE_AND_TOKEN
PSO.ServiceNowIntegration.REST.Authentication.Token.HeaderFormat	Authorization=Bearer (token)		The format that is to be used when sending the token. <Header_Property>=(token). Use only if Authentication mode is BASE64_ENCODING_AND_TOKEN or URL_ENCODE_AND_TOKEN
PSO.ServiceNowIntegration.REST.Authentication.Token.Expiry	15		Time in minutes the token is valid for. Use only if Authentication mode is BASE64_ENCODING_AND_TOKEN or URL_ENCODE_AND_TOKEN
Content			
PSO.ServiceNowIntegration.REST.ContentType	JSON		Should the information be sent using JSON or XML.
PSO.ServiceNowIntegration.REST.XMLContent			The XML pattern to use when sending content as XML.
PSO.ServiceNowIntegration.REST.Update.XMLContent			The XML pattern to use when sending content as XML and an alarm is cleared or if Alarm Transition is enabled when an alarm transitions from one severity to another.

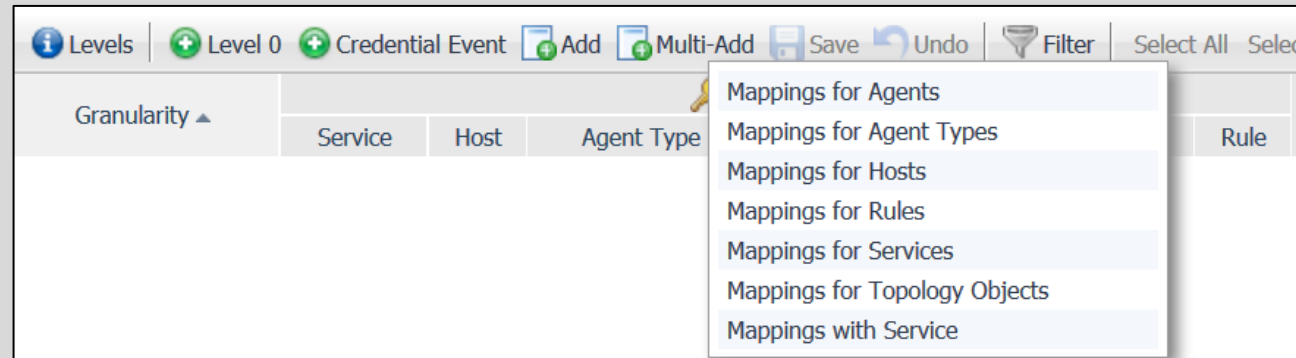


ServiceNow (Advanced) Integration Mappings Editor

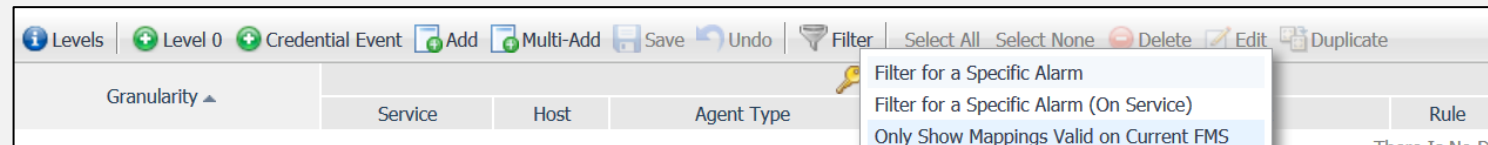
Combinations



Multiple Mappings



Filtering



ServiceNow (Advanced) Integration — Field Editor

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

UpdateUpdate SelectivelyResetClearCancel

ServiceNow (Advanced) 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

Foglight comparison to ServiceNow (Advanced)

Function	OOTB	SNOW Integration Advanced
Create a ticket from an alert	X	X
Uses REST API	X	X
Customizable Web Service/REST invocation support		X
Customizable Command Line invocation support		X
Clear/Acknowledge Alarms	Polled 15 minutes	Automatic/Realtime through REST
Close ticket when last alarm is cleared	X	X
Auto Annotate Alarms with returned information (Ticket number or other information)		X
Ability to reject/delay ticket generation		X
User overrides – Customer can define ticket impact/urgency		X
Assign a ticket to a username/id	Acquired from SNOW	Not required – assignment based on values in mapped fields
Pattern editor to easily access many of the information related to the alarm. (No groovy required) <ul style="list-style-type: none"> • Access to dozens of fields to easily provide values to parameters • A dozen common operators available to easily avoid any coding 		X



ServiceNow (Advanced) Comparison (continued)

Function	OOTB	SNOW Integration Advanced
Ability to send multiple additional custom fields – Pattern editor for user friendly/easy configuration	x	X
Use of groovy scripts for power users, including complete injected flow context		X
Decision engine for precision control <ul style="list-style-type: none"> 20 Levels of granularity Severity Level control 		X
Log History for failed attempts		X
Pause or Disable Ticket Creation	X	X
Manageable Mapping -Up to 20 different Levels from generic, Rule Level, Service Level down to a specific Topology Object (Map once and apply to all FMS's)		X
Remote/Centralized Administration from one FMS (for multi-FMS environments)		X
Persisted Queue so no events are lost <ul style="list-style-type: none"> Guaranteed delivery Keep history for rejected, failed and successful operations for total control Pause or delay control Consumer Thread settings/control as to manage concurrency and volume. 		X
Alarm Transition Support <ul style="list-style-type: none"> 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 the transition has been cleared All transition information is accessible from the Pattern Editor 		X



ServiceNow (Advanced) 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

