

Moogsoft Integration



Custom
Integrations

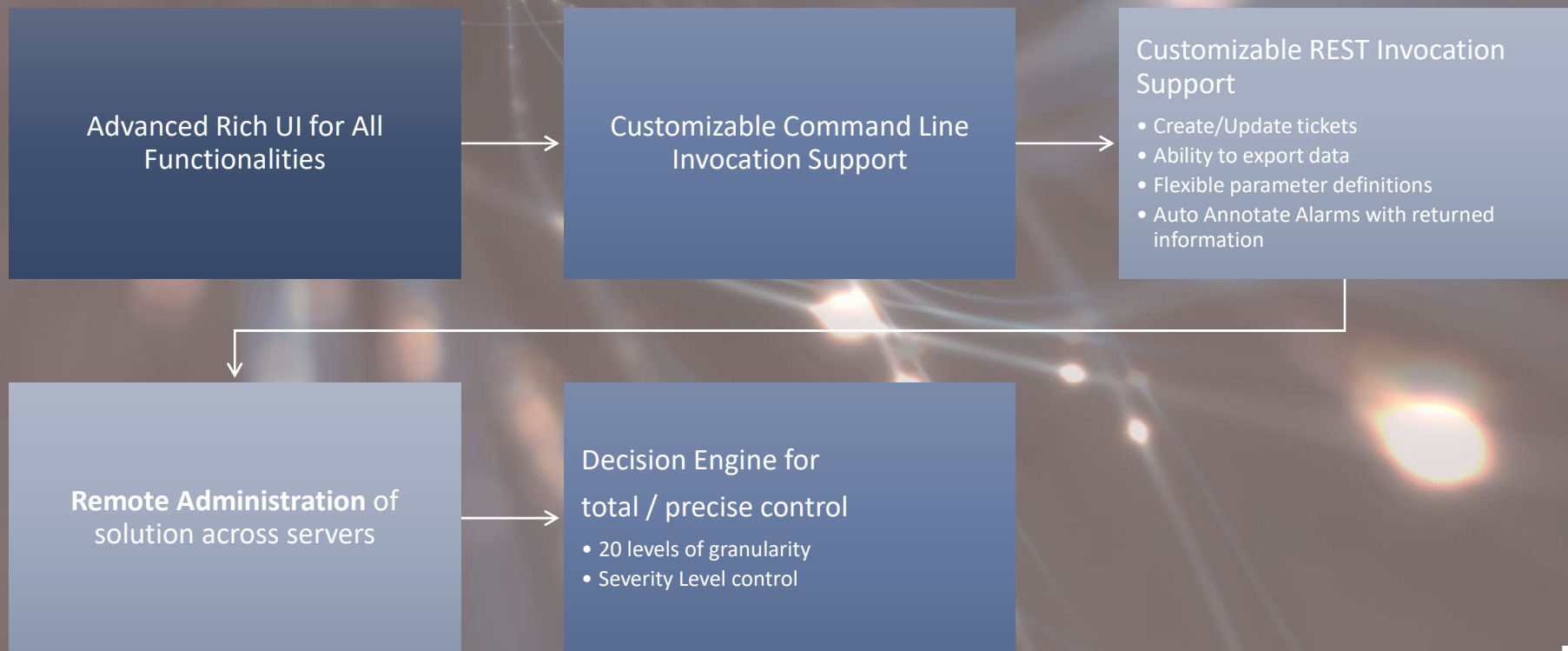
Why Moogsoft Integration from LightSpeed PM?

Integration of Moogsoft into Foglight without our custom integration services would require:

- Custom event driven rules
- Command invocation
- Extensive Groovy scripting for control over sent events and other features
- Currently no options to define Moogsoft fields without the integration



LightSpeed PM's Custom Moogsoft Integration Simplifies:



Moogsoft Integration Highlights

Pattern Editor: easily access the information related to the alarm

Use of Groovy scripts for power users

Persisted Queue so no events are lost

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 scripting with complete injected flow context

Ability to reject/delay ticket generation

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.



Moogsoft Integration Alarming Features

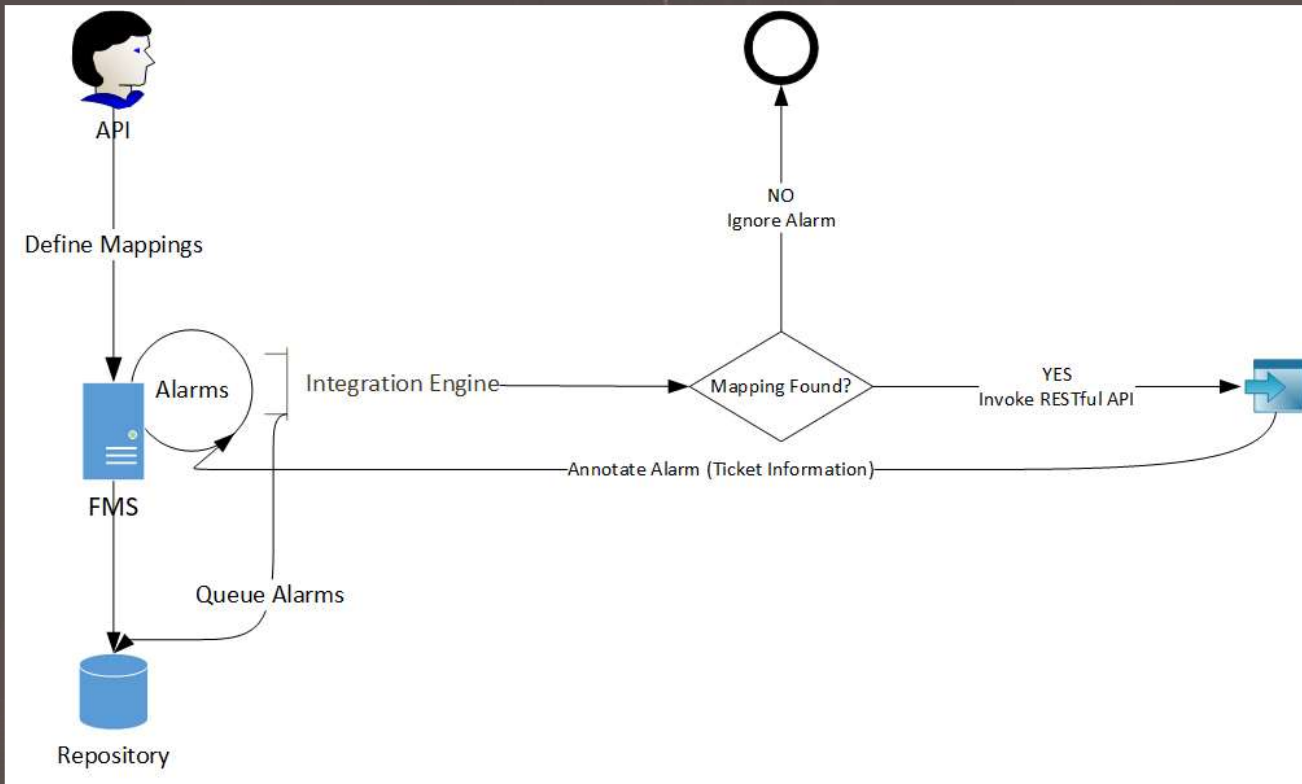
Auto Acknowledge Alarms

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 the transition has been cleared

Alarm Flood Prevention





Moogsoft Integration — RESTful API Flow



Moogsoft 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



Moogsoft Integration – Changing Settings

Navigation <

Expert View ▾

- ▼ Bookmarks
There are no bookmarks
- ▶ Homes
- ▶ Dashboards
- ▼ Foglight Services
 - Local FMS
 - Host Name
There Is No Data To Display

Integration Pack Registry Settings - Local FMS

Save | Undo | Filter

Name	Value	Scoping	
Moogsoft Integration			
Open Ticket			
PSO.MoogsoftIntegration.OpenTicket.NoEmptyValue	true		If true any tag that returns a null will be replaced by a space. If false null values
PSO.MoogsoftIntegration.OpenTicket.TimeZone	GMT		Defines the time zone to use when outputting date and time in the Command.
PSO.MoogsoftIntegration.OpenTicket.DateTimeFormat	dd/MM/yyyy:HH:mm:ss		Defines the format for the date and time to use when outputting date and time in
Queue			
Ignore			
PSO.MoogsoftIntegration.Queue.Ignore.Normal	false		Should cleared alarms be ignored by the queuing engine? Ignored alarms are not
PSO.MoogsoftIntegration.Queue.Ignore.Warning	false		Should Warning alarms be ignored by the queuing engine? Ignored alarms are no
PSO.MoogsoftIntegration.Queue.Ignore.Critical	false		Should Critical alarms be ignored by the queuing engine? Ignored alarms are not
PSO.MoogsoftIntegration.Queue.Ignore.Fatal	false		Should Fatal alarms be ignored by the queuing engine? Ignored alarms are not qu
PSO.MoogsoftIntegration.Queue.Ignore.Rules			A list of comma separated rule names that the queue should ignore. Alarms gener
Delay			
PSO.MoogsoftIntegration.Queue.Alarm.DelayPeriod.Warning	0		Time in minutes a warning alarm should be delayed in the queue before being for
PSO.MoogsoftIntegration.Queue.Alarm.DelayPeriod.Critical	0		Time in minutes a critical alarm should be delayed in the queue before being forw
PSO.MoogsoftIntegration.Queue.Alarm.DelayPeriod.Fatal	0		Time in minutes a fatal alarm should be delayed in the queue before being forward
PSO.MoogsoftIntegration.Queue.AlarmTransition.Enable	false		Should alarm transition be supported. If enabled the integration will update a tick
PSO.MoogsoftIntegration.Queue.Alarm.AcknowledgeUponSuccess	false		Should the alarm be acknowledged when the alarm has been successfully process
PSO.MoogsoftIntegration.Queue.Alarm.EnforceServiceFilter	false		When an alarm is being processed enforce the filters applied to the service. If true
PSO.MoogsoftIntegration.Queue.PausePeriod	0		Time in minutes no alarms should be forwarded to the target system. This is usef
REST			
PSO.MoogsoftIntegration.REST.Enable	true		Should the integration use the REST method (true)
PSO.MoogsoftIntegration.REST.URL	http://fms-593:8080		The URL for the RESTful API the integration should use in order to open a ticket.
PSO.MoogsoftIntegration.REST.Path	api/v1/script/runScript		The path following the URL.



Moogsoft Integration — Mappings Editor

Combinations

The screenshot shows the Moogsoft Mappings Editor interface. At the top, there is a toolbar with icons for Levels, Level 0, Credential Event, Add, Multi-Add, Save, Undo, Filter, Select All, Select None, Delete, Edit, and Duplicate. Below the toolbar is a table with columns for Granularity, Service, Host, Object, and Rule. A context menu is open over the table, showing options: Mapping Combinations and Mapping Combinations with Service.

Multiple Mappings

The screenshot shows the Moogsoft Mappings Editor interface. At the top, there is a toolbar with icons for Levels, Level 0, Credential Event, Add, Multi-Add, Save, Undo, Filter, Select All, and Select None. Below the toolbar is a table with columns for Granularity, Service, Host, Agent Type, and Rule. A context menu is open over the table, showing options: Mappings for Agents, Mappings for Agent Types, Mappings for Hosts, Mappings for Rules, Mappings for Services, Mappings for Topology Objects, and Mappings with Service.

Filters

The screenshot shows the Moogsoft Mappings Editor interface. At the top, there is a toolbar with icons for Levels, Level 0, Credential Event, Add, Multi-Add, Save, Undo, Filter, Select All, Select None, Delete, Edit, and Duplicate. Below the toolbar is a table with columns for Granularity, Service, Host, Agent Type, and Rule. A context menu is open over the table, showing options: Filter for a Specific Alarm, Filter for a Specific Alarm (On Service), and Only Show Mappings Valid on Current FMS.



Moogsoft Integration – Field Editor

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



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

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

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.
=[]	Equality
=[]?[]:[]	If equal else
![[[]]]!	Groovy Script
^U	Uppercase
^L	Lowercase
^T	Trim
^N	New Line Output
^R	Return

Moogsoft Integration — Pattern Editor



Moogsoft Integration — System Requirements



Minimum required
FMS version

5.9.3



Supported

MS SQL

2008

v 10.0.1600 or later

Database

Oracle

9i R2

Minimum

MySQL

5.1.45

Version

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

LightSpeed PM - Silver+ Quest Partner