A Guide to Automation in Freshservice

freshservice

Introduction

It's just another day at your Service Desk. Your team of trained tech experts is all set to provide customers with the best support ever, until they find some old tickets that need to be followed up on and some new tickets that need to be categorized and assigned to the right agent. Sound familiar?

An expert IT technician should not have to waste a part of his productive time on such mundane tasks when he has bigger things to worry about. In Freshservice, you can automate almost all of them by creating simple if-this-then-that rules.

Depending on the type of task you want to automate, you can choose one of the 3 automation tools available in the Admin console- The Dispatch'r, the Supervisor and the Observer. You can even carry out a bunch of updates with a single click using Scenario Automations.

Here are some use cases to help you get started.

Dispatch'r

What exactly it does:

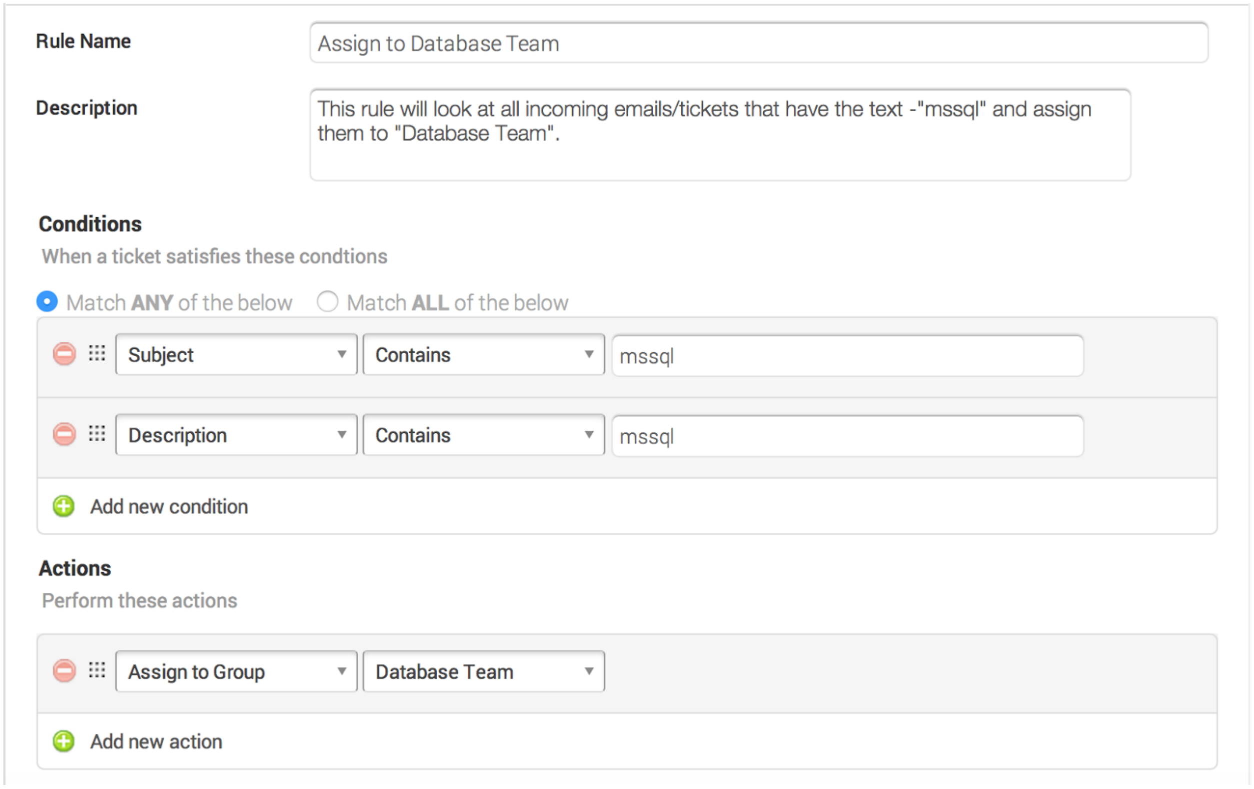
The Dispatch'r essentially helps you choose what needs to be done to a new ticket *right after it arrives*, based on its properties.

Especially helpful for automating tasks like:

Categorizing, prioritizing and assigning tickets.

Some examples:

- If the ticket contains the word *"mssql"* in its subject line or description, **assign** it to the Database team.



Rule Name	Assign to Database Team
Description	This rule will look at all incoming emails/tickets that have the text -"mssql" and assign them to "Database Team".

Conditions

When a ticket satisfies these condtions

[✓ Match **ANY** of the below] Match **ALL** of the below

Subject	Contains	mssql
Description	Contains	mssql

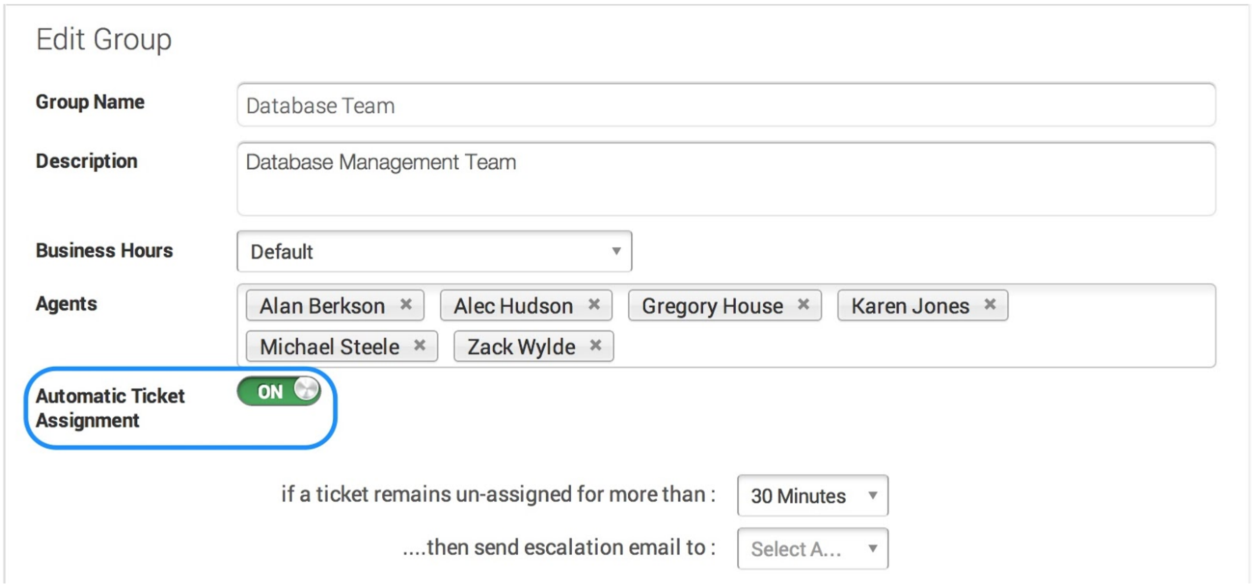
Actions

Perform these actions

Assign to Group	Database Team

Here's what the rule would look like in Freshservice

BONUS TIP: You can also have the ticket automatically assigned to the next available agent using **round-robin ticket assignment**. Go to *Admin → Groups* and enable "**Automatic Ticket Assignment**" for any of the existing groups.



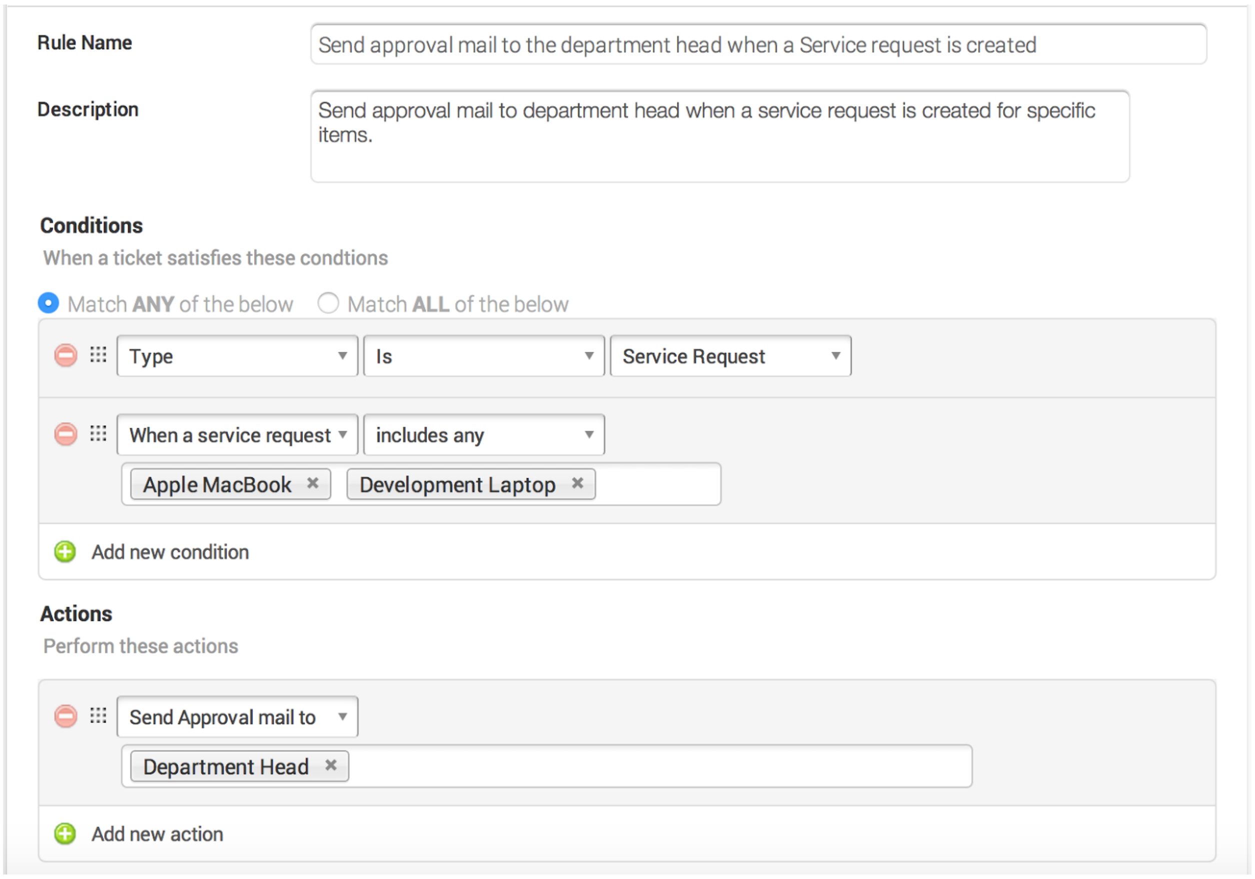
Edit Group

Group Name	Database Team
Description	Database Management Team
Business Hours	Default
Agents	Alan Berkson ×, Alec Hudson ×, Gregory House ×, Karen Jones ×, Michael Steele ×, Zack Wylde ×
Automatic Ticket Assignment	ON

if a ticket remains un-assigned for more than : 30 Minutes

....then send escalation email to : Select A...

- If a Service Request is created for specific items, **send an approval request** to the Department Head



Rule Name Send approval mail to the department head when a Service request is created

Description Send approval mail to department head when a service request is created for specific items.

Conditions

When a ticket satisfies these condtions

[✓ Match **ANY** of the below] Match **ALL** of the below

Type | Is | Service Request

When a service request | includes any | Apple MacBook, Development Laptop

Add new condition

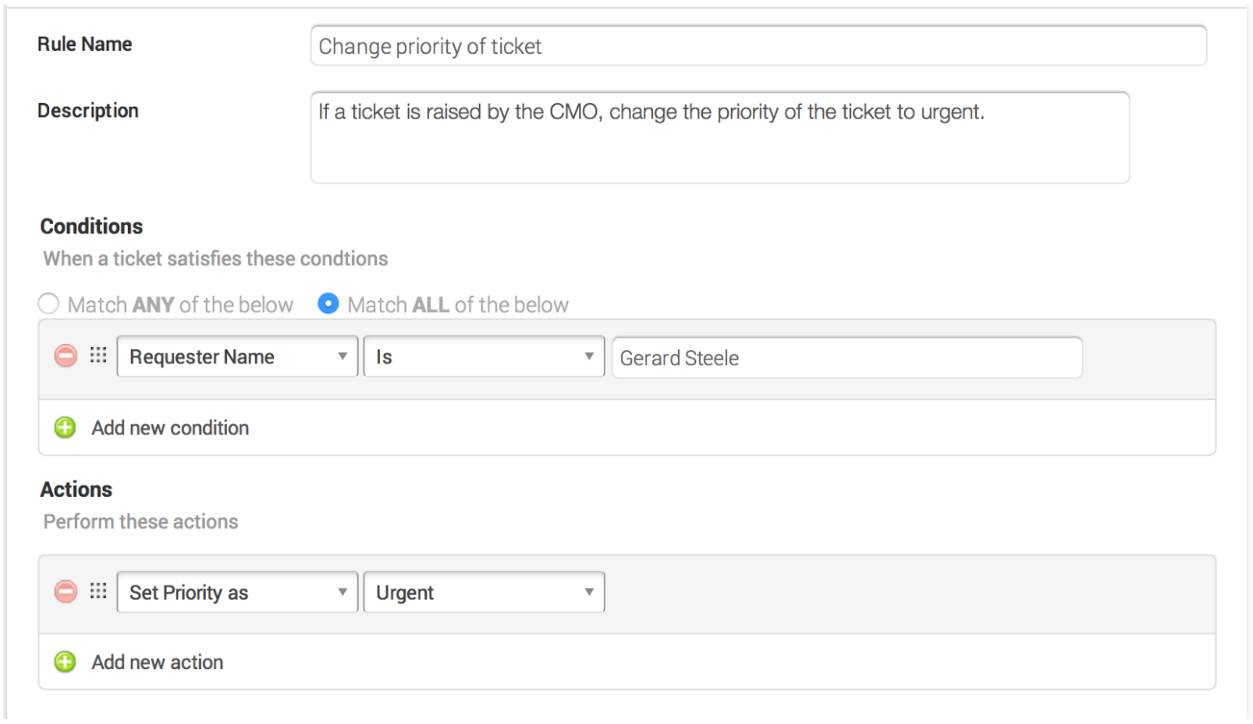
Actions

Perform these actions

Send Approval mail to | Department Head

Add new action

- If the ticket is raised by the CMO who travels frequently to attend marketing events, **change its priority** to *urgent*



Rule Name	Change priority of ticket
Description	If a ticket is raised by the CMO, change the priority of the ticket to urgent.

Conditions

When a ticket satisfies these condtions

○ Match **ANY** of the below [✓ ● Match **ALL** of the below]

Requester Name	Is	Gerard Steele

Add new condition

Actions

Perform these actions

Set Priority as	Urgent

Add new action

Supervisor

What exactly it does:

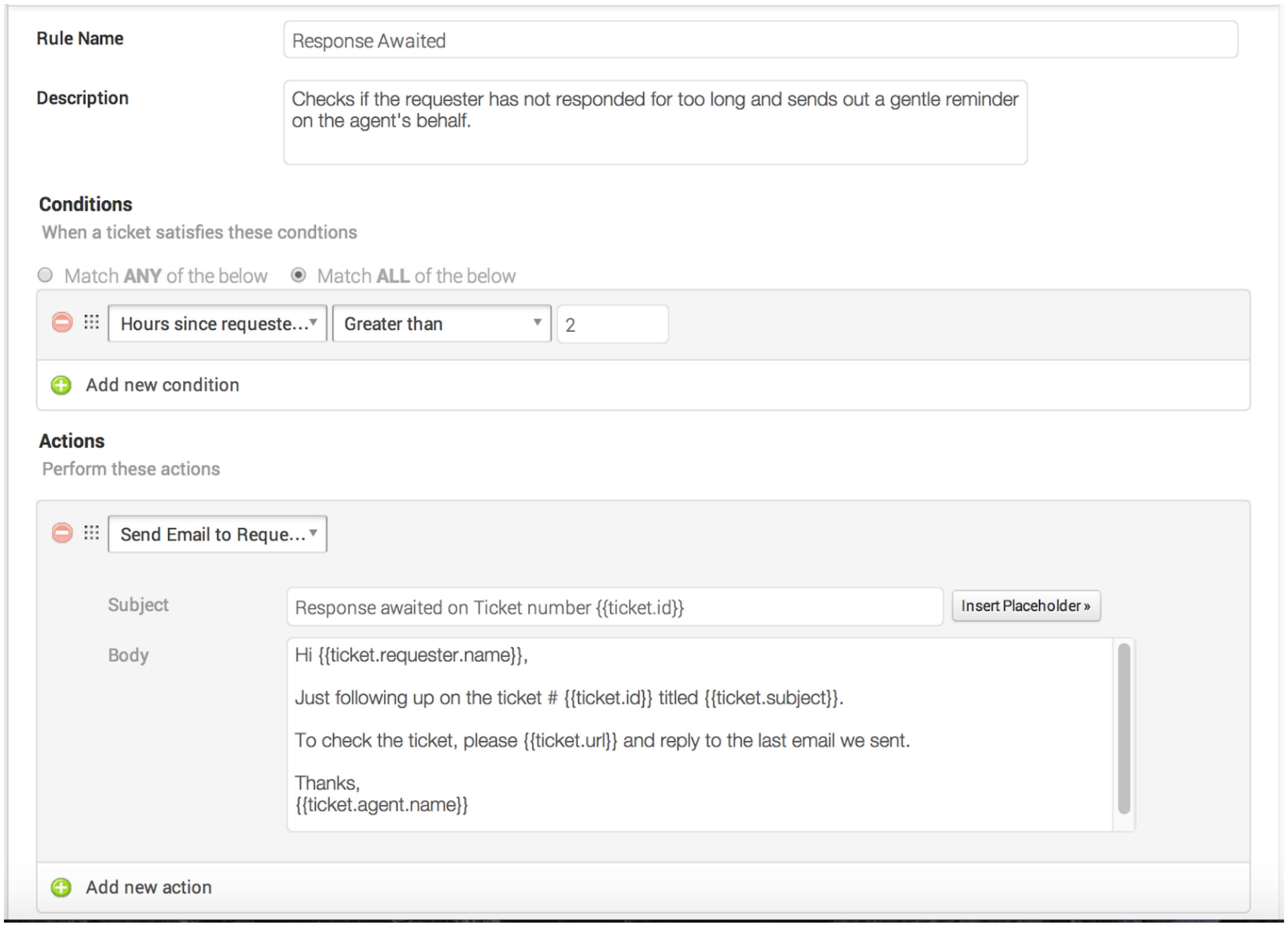
The Supervisor *runs every hour*, looks for loose ends in your Service Desk and ties them up.

Especially helpful for automating tasks like:

Following up on pending responses or overdue tickets and escalating tickets.

Some examples:

- **Send a reminder to customers** if they haven't responded to your email after 2 hours.



Rule Name	Response Awaited
Description	Checks if the requester has not responded for too long and sends out a gentle reminder on the agent's behalf.

Conditions
When a ticket satisfies these condtions

( ) Match **ANY** of the below [✓ Match **ALL** of the below]

Hours since requeste... | Greater than | 2

Add new condition

Actions
Perform these actions

Send Email to Reque...

Subject: Response awaited on Ticket number {{ticket.id}}

Body:

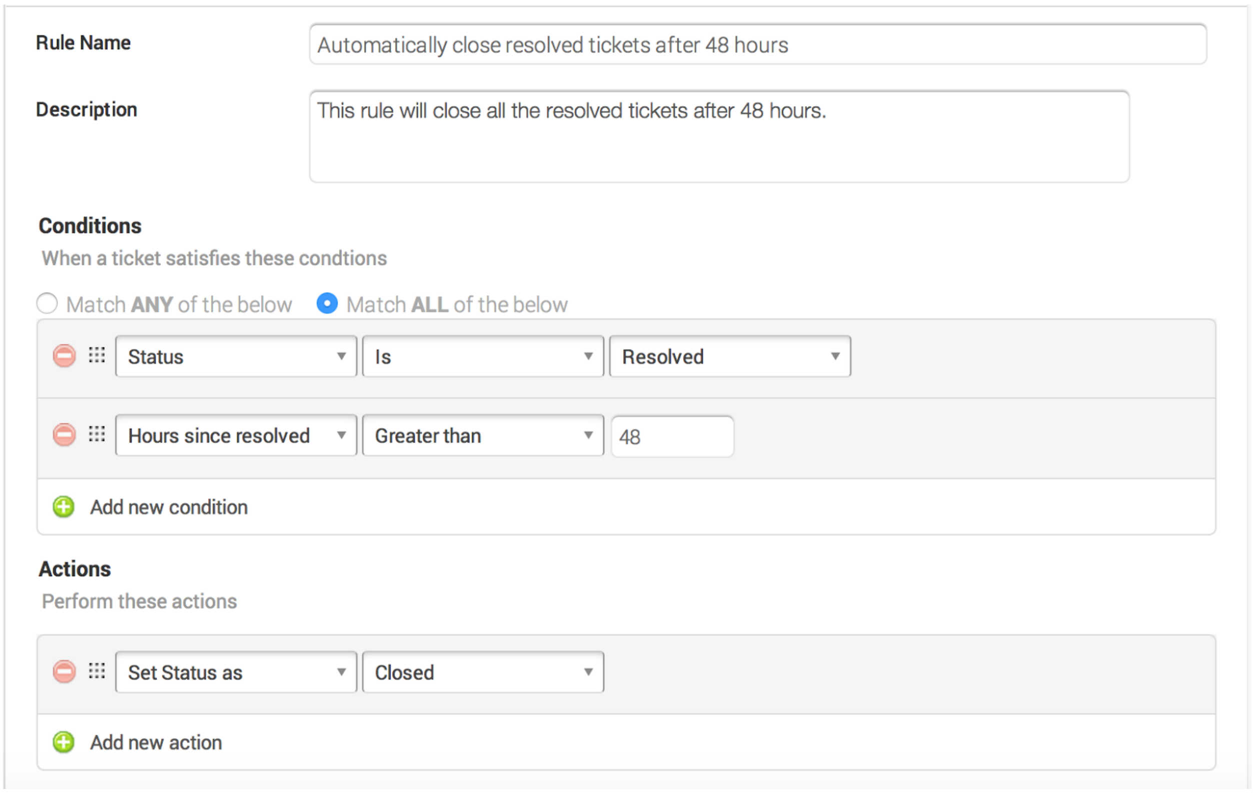
Hi {{ticket.requester.name}},

Just following up on the ticket # {{ticket.id}} titled {{ticket.subject}}.

To check the ticket, please {{ticket.url}} and reply to the last email we sent.

Thanks,
{{ticket.agent.name}}

- **Automatically close resolved tickets** after 48 hours, unless the requester responds to the ‘ticket resolved’ notification.



Rule Name	Automatically close resolved tickets after 48 hours
Description	This rule will close all the resolved tickets after 48 hours.

Conditions

When a ticket satisfies these condtions

○ Match **ANY** of the below ● Match **ALL** of the below

Status	Is	Resolved
Hours since resolved	Greater than	48

Add new condition

Actions

Perform these actions

Set Status as	Closed

Add new action

Observer

What exactly it does:

The Observer *keeps an eye out for events (triggers)* that call for certain actions. It then performs the actions without you needing to worry about it.

You can set Observer rules for Service Requests, Incidents, Problems, Changes and Releases.

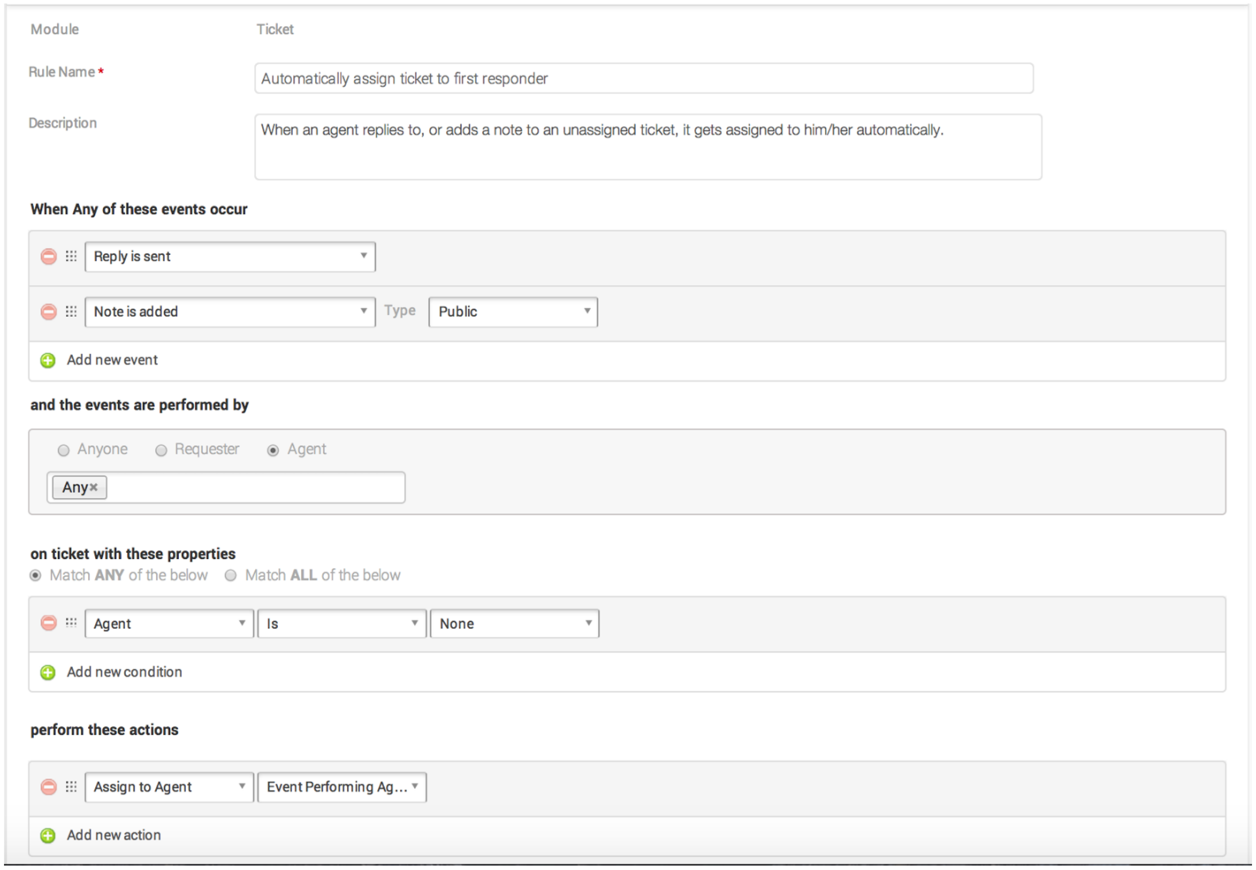
Especially helpful for automating tasks like:

Sending an alert to the CTO when there is an update in a product's module.

Some examples:

Ticket Rules:

- **Automatically assign tickets** to the first responder.



Module: Ticket

Rule Name*: Automatically assign ticket to first responder

Description: When an agent replies to, or adds a note to an unassigned ticket, it gets assigned to him/her automatically.

When Any of these events occur

- Reply is sent
- Note is added — Type: Public

Add new event

and the events are performed by

Anyone Requester [✓ Agent]

Any×

on ticket with these properties

[✓ Match ANY of the below] Match ALL of the below

Agent	Is	None

Add new condition

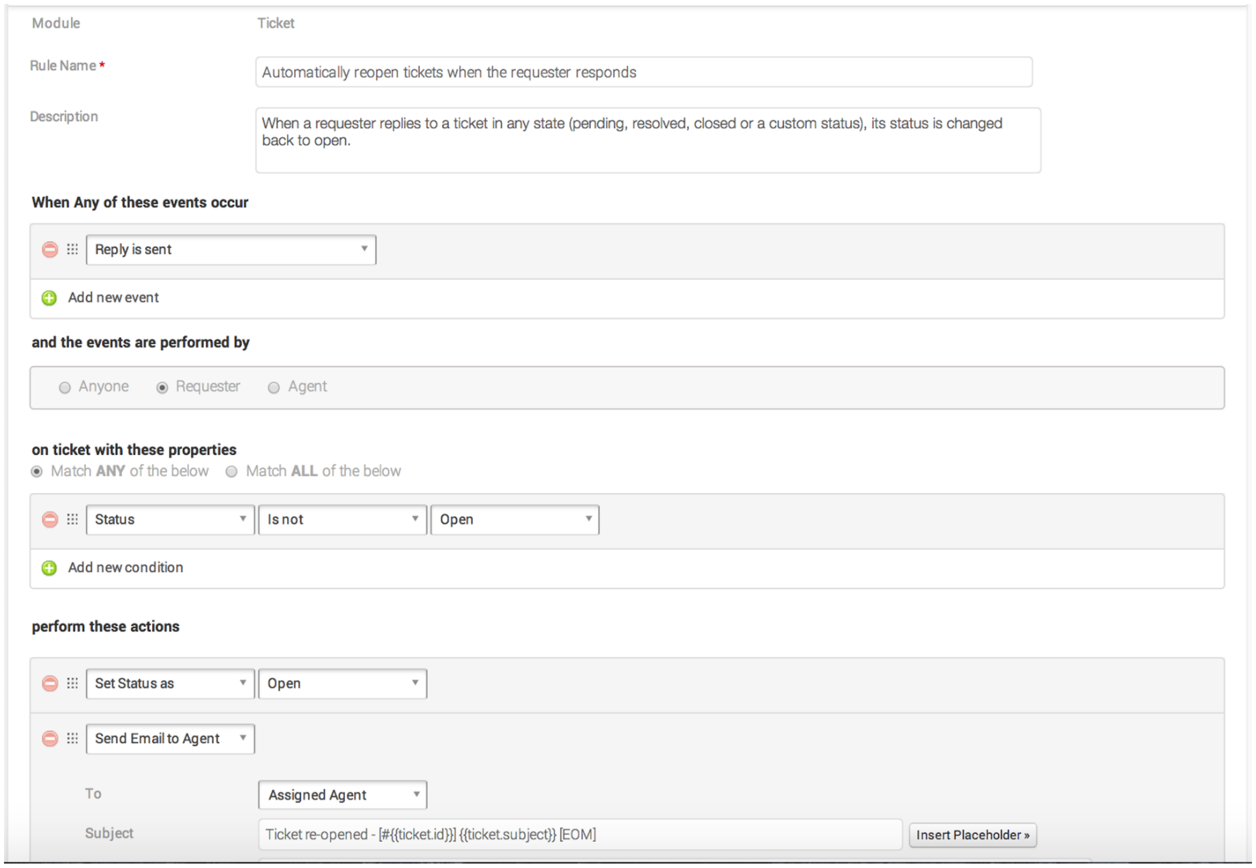
perform these actions

Assign to Agent	Event Performing Ag...

Add new action

freshservice

- **Automatically reopen tickets** when the requester responds



Module Ticket

Rule Name * Automatically reopen tickets when the requester responds

Description When a requester replies to a ticket in any state (pending, resolved, closed or a custom status), its status is changed back to open.

When Any of these events occur

Reply is sent

Add new event

and the events are performed by

Anyone [✓ Requester] Agent

on ticket with these properties

[✓ Match ANY of the below] Match ALL of the below

Status | Is not | Open

Add new condition

perform these actions

Set Status as | Open

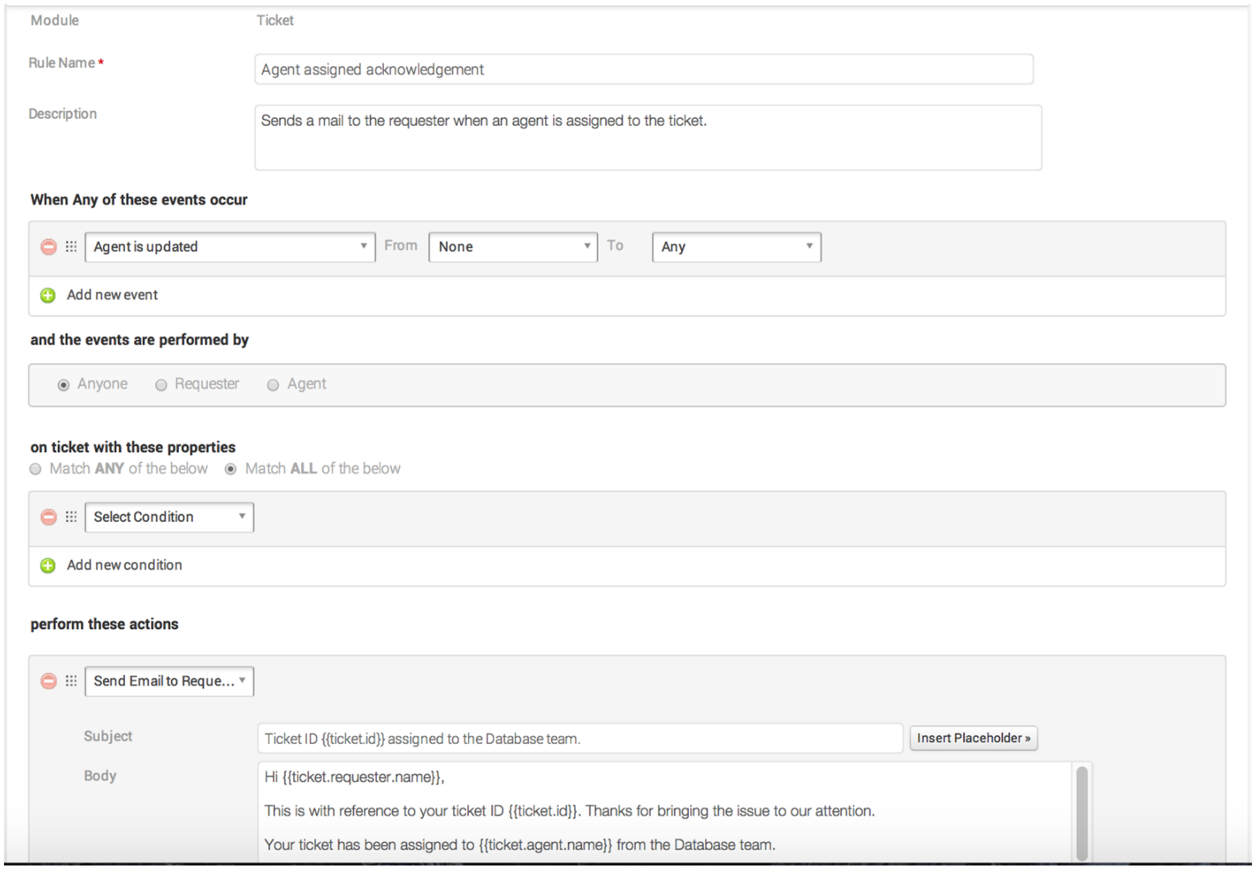
Send Email to Agent

To: Assigned Agent

Subject: Ticket re-opened - [#{{ticket.id}}] {{ticket.subject}} [EOM]

Insert Placeholder »

- **Notifying the requester** about the ticket being assigned to an agent.



Module Ticket

Rule Name * Agent assigned acknowledgement

Description Sends a mail to the requester when an agent is assigned to the ticket.

When Any of these events occur

Agent is updated From None To Any

Add new event

and the events are performed by

[✓ Anyone] Requester Agent

on ticket with these properties

Match **ANY** of the below [✓ Match **ALL** of the below]

Select Condition

Add new condition

perform these actions

Send Email to Reque...

Subject Ticket ID {{ticket.id}} assigned to the Database team.

Insert Placeholder »

Body

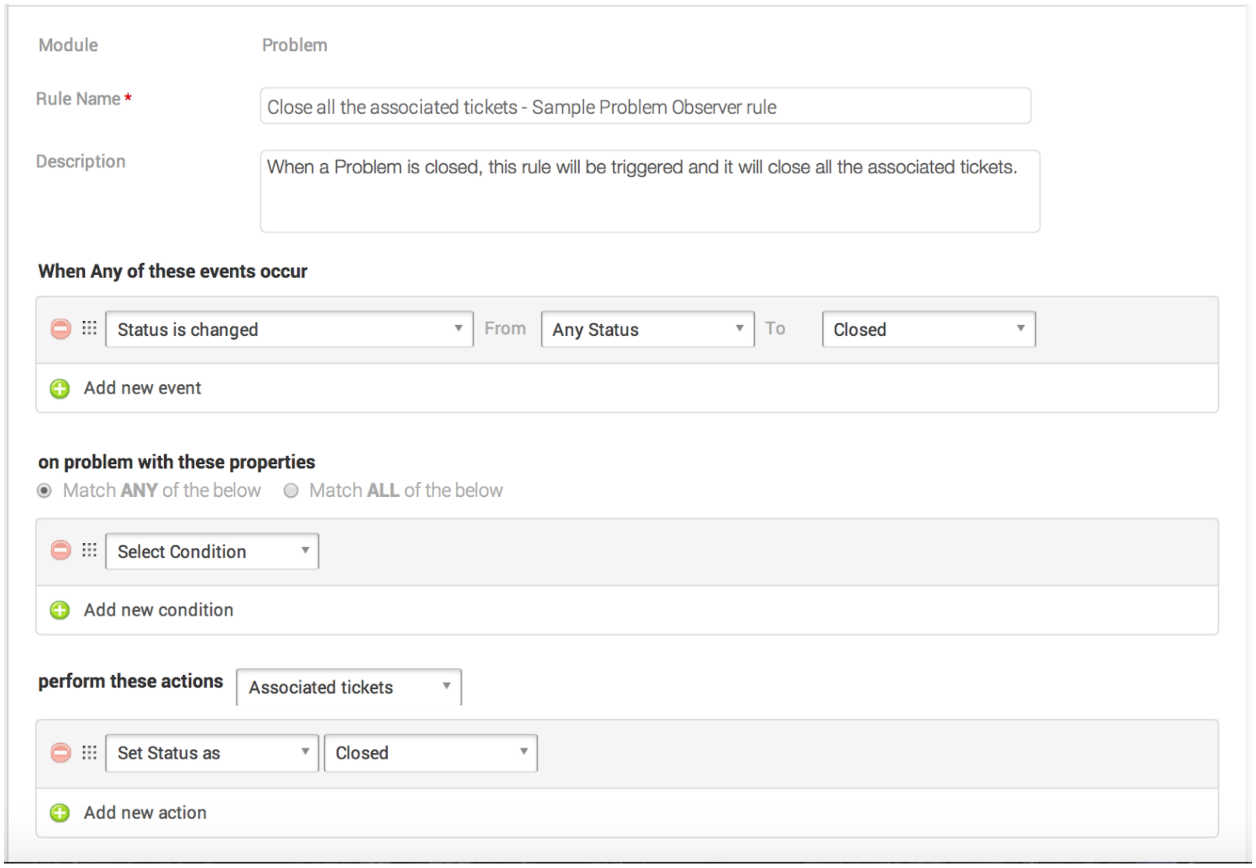
Hi {{ticket.requester.name}},

This is with reference to your ticket ID {{ticket.id}}. Thanks for bringing the issue to our attention.

Your ticket has been assigned to {{ticket.agent.name}} from the Database team.

Problem Rules:

- **Close all the associated incidents** when a problem is closed.



Module: Problem

Rule Name *: Close all the associated tickets - Sample Problem Observer rule

Description: When a Problem is closed, this rule will be triggered and it will close all the associated tickets.

When Any of these events occur

Status is changed From Any Status To Closed

on problem with these properties

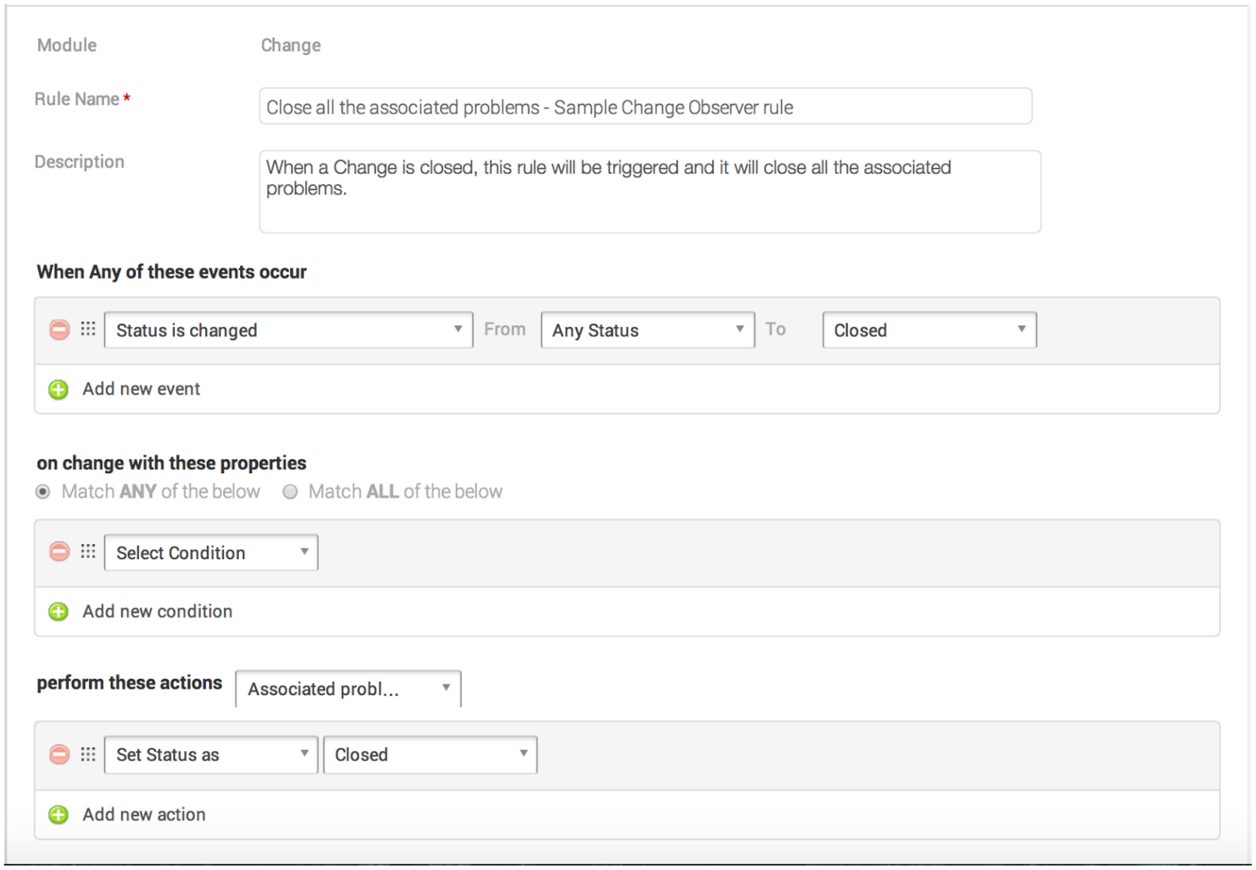
[✓ Match **ANY** of the below] Match **ALL** of the below

perform these actions Associated tickets

Set Status as Closed

Change Rules:

- **Close all the associated problems** when a change is closed.



Module	Change
Rule Name *	Close all the associated problems - Sample Change Observer rule
Description	When a Change is closed, this rule will be triggered and it will close all the associated problems.

When Any of these events occur

Status is changed From Any Status To Closed

Add new event

on change with these properties

[✓ Match **ANY** of the below] Match **ALL** of the below

Select Condition

Add new condition

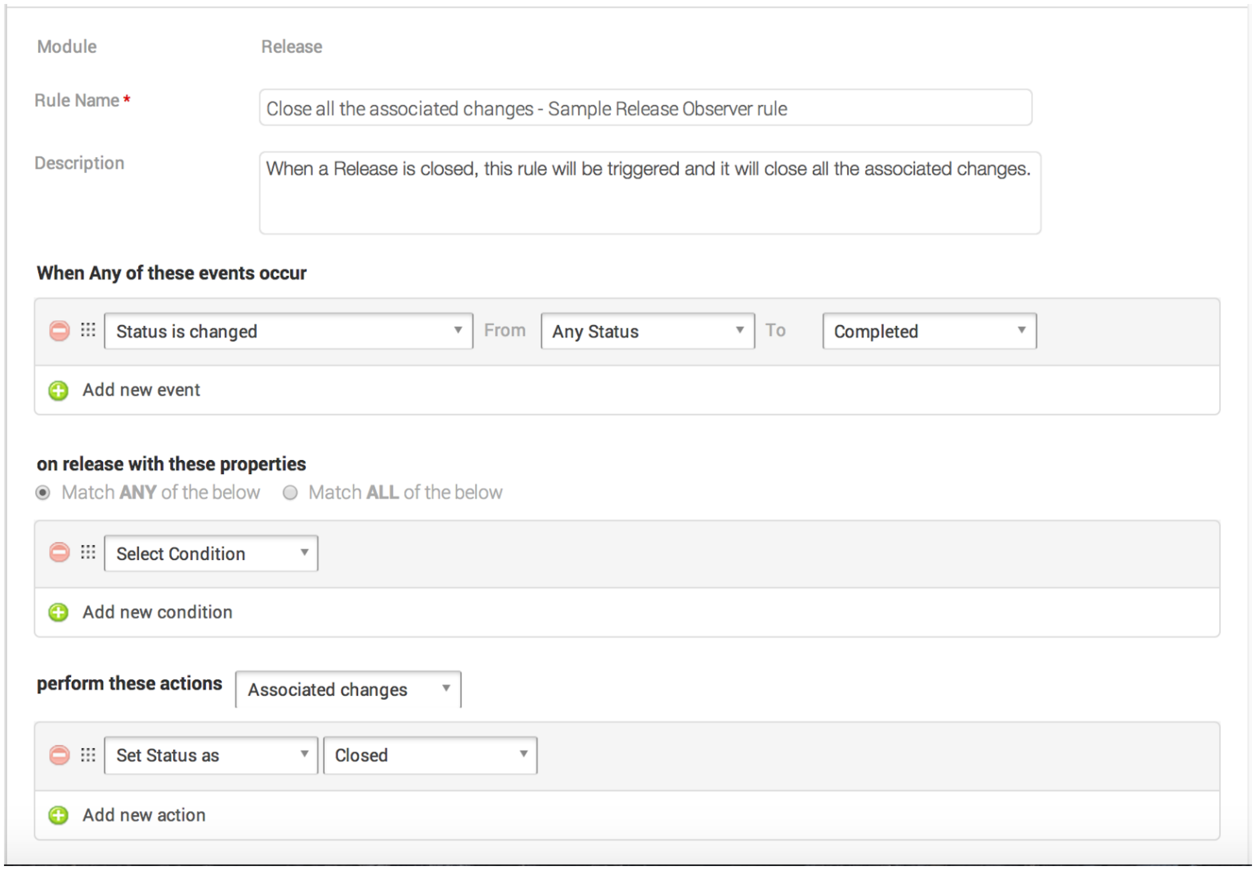
perform these actions Associated probl...

Set Status as Closed

Add new action

Release Rules:

- **Close all the associated changes** when a release is completed.



Module	Release
Rule Name *	Close all the associated changes - Sample Release Observer rule
Description	When a Release is closed, this rule will be triggered and it will close all the associated changes.

When Any of these events occur

Status is changed From Any Status To Completed

Add new event

on release with these properties

(● Match **ANY** of the below) ○ Match **ALL** of the below

Select Condition

Add new condition

perform these actions Associated changes

Set Status as Closed

Add new action

Scenario Automations

What exactly it does:

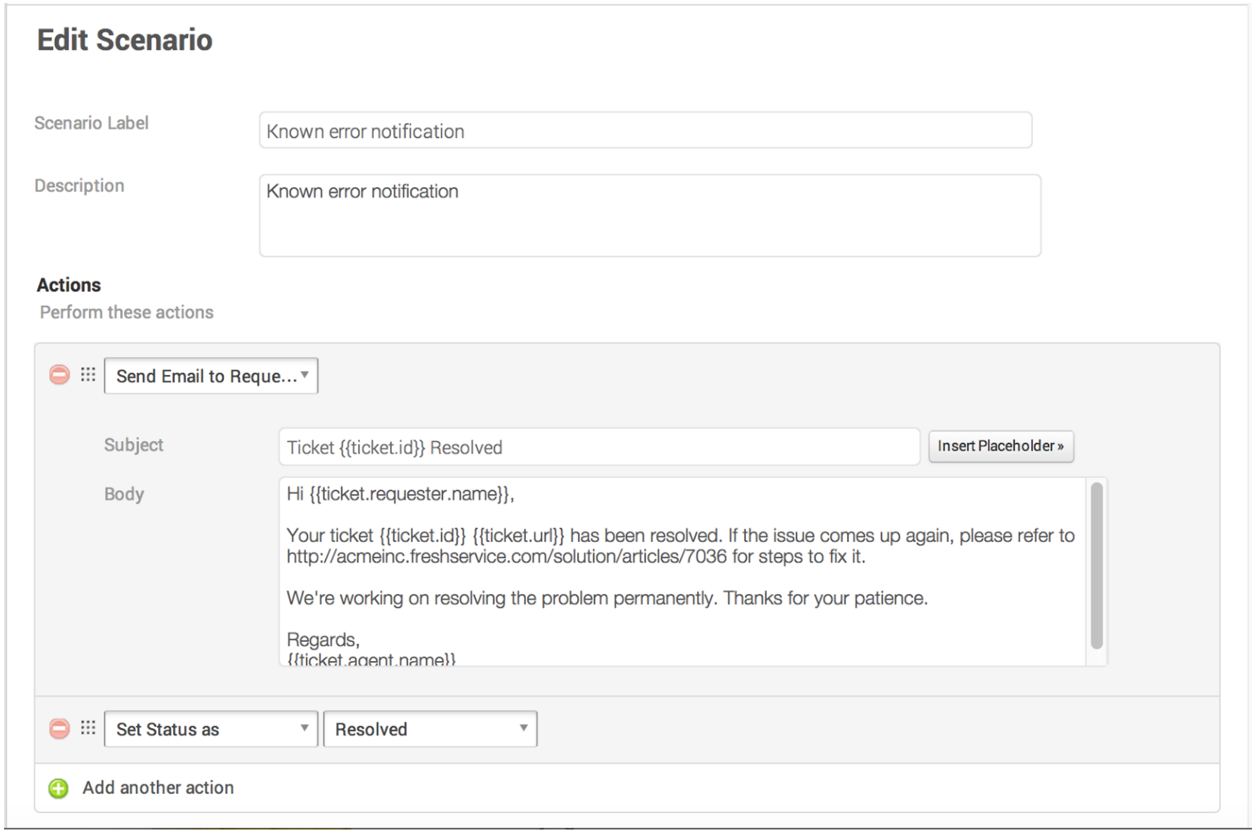
Scenario Automations let you carry out a bunch of pre-set updates to a ticket with a *single click*.

Especially helpful for automating tasks like:

Recurring scenarios where you have to carry out the same series of tasks every time.

Some examples:

- **Send Known Error Notification**: When you encounter a known error, you can set up a scenario automation that sets the ticket status as resolved and notifies the requester.



Edit Scenario

Scenario Label: Known error notification

Description: Known error notification

Actions
Perform these actions

Send Email to Reque...

Subject: Ticket {{ticket.id}} Resolved

Insert Placeholder »

Body:
Hi {{ticket.requester.name}},

Your ticket {{ticket.id}} {{ticket.url}} has been resolved. If the issue comes up again, please refer to http://acmeinc.freshservice.com/solution/articles/7036 for steps to fix it.

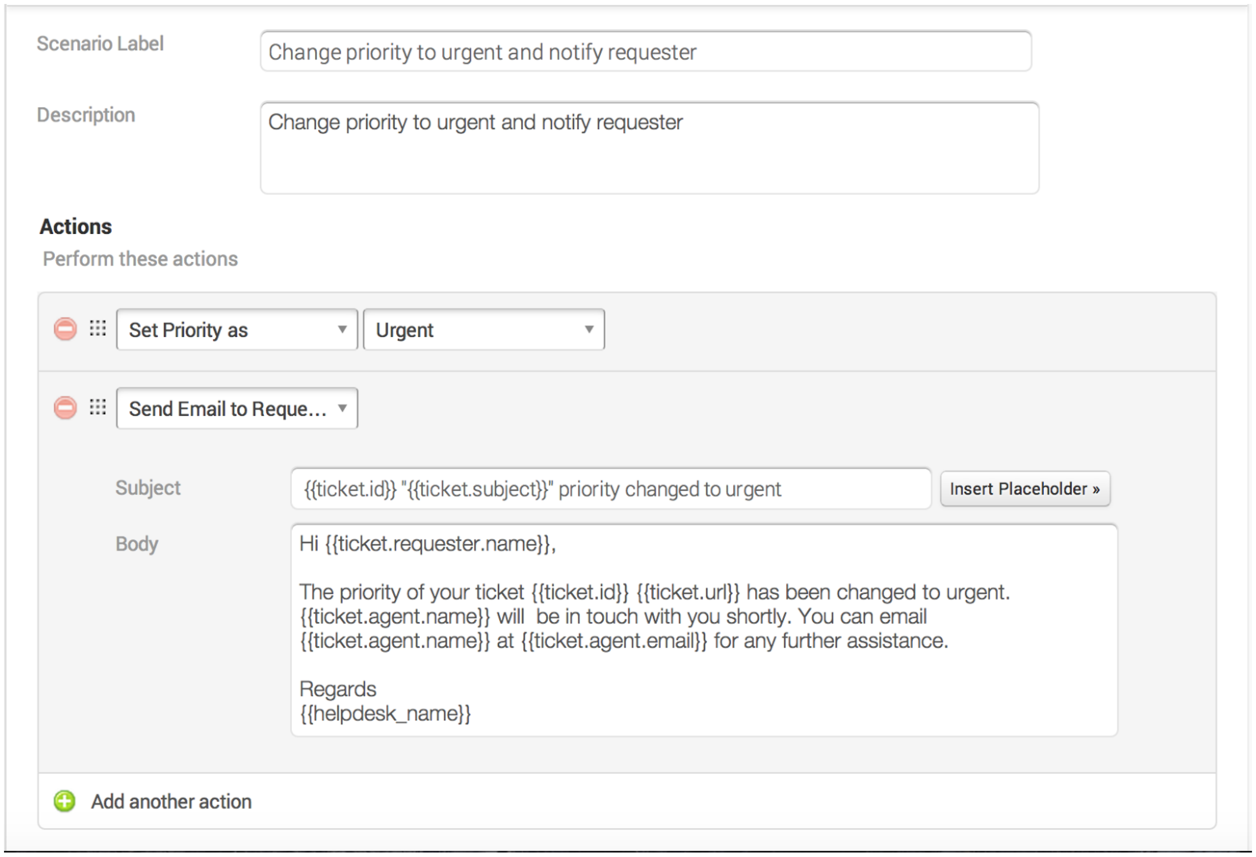
We're working on resolving the problem permanently. Thanks for your patience.

Regards,
{{ticket.agent.name}}

Set Status as: Resolved

Add another action

- **Change priority of ticket and notify requester**: In case a specific ticket needs to be resolved urgently, you can change its priority and send a notification to the requester about the change.



Scenario Label: Change priority to urgent and notify requester

Description: Change priority to urgent and notify requester

Actions
Perform these actions

Set Priority as: Urgent

Send Email to Reque…

Subject: {{ticket.id}} "{{ticket.subject}}" priority changed to urgent

Body:

Hi {{ticket.requester.name}},

The priority of your ticket {{ticket.id}} {{ticket.url}} has been changed to urgent. {{ticket.agent.name}} will be in touch with you shortly. You can email {{ticket.agent.name}} at {{ticket.agent.email}} for any further assistance.

Regards
{{helpdesk_name}}

We're just scratching the surface here. There's a ton of other tasks that you can automate using one (or a combination) of these rules.

Go on. Explore these features for yourself and leave all the boring work to them.

For other questions and clarifications, please contact support@freshservice.com.

freshservice