

Service Dynamics

Impact and Event Management
Installation and Administration

Zenoss, Inc.

www.zenoss.com

Zenoss Service Dynamics Impact and Event Management Installation and Administration

Copyright © 2012 Zenoss, Inc. All rights reserved. Redistribution or duplication of any portion of this document is prohibited without the express written consent of Zenoss, Inc.

Zenoss and the Zenoss logo are trademarks or registered trademarks of Zenoss, Inc. in the United States and other countries. All other trademarks, logos, and service marks are the property of Zenoss or other third parties. Use of these marks is prohibited without the express written consent of Zenoss, Inc. or the third-party owner.

Part Number: 41-112012-4.2-v02

1. About Impact and Event Management	1
1.1. Introduction	1
1.1.1. About Services and Service Elements	1
1.1.2. About Service Policies	1
2. Installing and Configuring Impact	2
2.1. Backing Up Impact	2
3. Working with Impact and Event Management	3
3.1. Using the Interface	3
3.1.1. All Services	3
3.1.2. Dynamic Service Views	4
3.1.2.1. Overview	4
3.1.2.2. Impact Events	4
3.1.2.3. Impact View	5
3.2. Working with Dynamic Services	8
3.2.1. Create a Dynamic Service	8
3.2.2. Edit Service Details	8
3.2.3. Clone a Service	9
3.2.4. Organize Dynamic Services	9
3.2.5. Add Services to Another Dynamic Service	9
3.2.6. Add Elements to a Dynamic Service	9
3.2.7. Delete Elements from a Dynamic Service	10
3.2.8. Define Service Policies	10
3.2.8.1. Custom State Providers	12
3.2.8.2. Suppressing Service Events	12
3.2.9. Define a Logical Node	12
3.2.10. Service Impact Notifications	14
3.2.11. Adding Context Information to Notifications	14
3.3. Example Service Scenario: Redundant Components	15

Chapter 1. About Impact and Event Management

1.1. Introduction

The impact and event management feature of Resource Manager lets you define high-level services for which you want some level of assurance. Through service visualization, Resource Manager allows you to quickly determine the state of a service, service elements, and current dynamic network and infrastructure resources.

1.1.1. About Services and Service Elements

A *service* is a collection of *elements* (devices, components, logical nodes, and other services) to which you apply potentially complex rules, allowing it to evaluate its own status and change state based on that evaluation. The state of a service is determined by the state of the elements it comprises.

Logical relationships among service elements can be:

- **Physical** - Systems, infrastructure, and network devices.
- **Logical** - Aspects of a service that must be measured or evaluated.
- **Virtual** - Software components (for example, processes).
- **Reference** - Elements represented by other domain managers.

Service element relationships are *fixed* and *dynamic*. Fixed relationships identify definitions, such as a Web application belonging to a specific service. Dynamic relationships are managed by the model.

1.1.2. About Service Policies

Service policies are a collection of *state triggers*, which are conditions under which the state of a service or service component changes. Policies are applied to a service (or any item in a service) and evaluate the state of, and relationships among, elements by:

- Providing impact roll-up so that services affected by low-level device failures are known
- Assisting in root cause identification so that a multi-level-deep failure in the infrastructure can be seen from the service
- Providing multiple availability and performance states. (For example, if a single Web server in a load-balanced group is down, the service may be degraded.)
- Providing event storm filtering, roll-up, and windowing functions

You can apply *contextual policies* and *global policies* to determine a service's availability or performance state. With the default service policy, an element's state is equal to the worst state of its impacting elements. You can change how its state is determined by adding contextual policies (which apply to the element only within a given service) or global policies (which apply to the element in all services). If a global policy is defined, it will override the default policy; if a contextual policy is defined, it will override the global policy.

A policy comprises triggers, which translate the state of impacting elements into a state for the impacted element. All triggers are evaluated, and the worst resulting state takes precedence. You can set a policy that says that an element is considered "down" only if three of its impacting elements are also "down."

Chapter 2. Installing and Configuring Impact

Impact requires:

- Adobe® Flash® Player Version 10.2.0 or higher
- These ZenPacks:
 - Impact (ZenPacks.zenoss.Impact)
 - Dynamic Service View (ZenPacks.zenoss.DynamicView)
 - Advanced Search (ZenPacks.zenoss.AdvancedSearch)

The Dynamic Service View and Advanced Search ZenPacks are installed when you install Resource Manager.

Install the Impact ZenPack from the command line:

1. As the zenoss user, enter:

```
zenpack --install Filename
```

where *Filename* is the Impact ZenPack file name.

2. Restart the system with the command:

```
zenoss restart
```

Note

On distributed collector installations, the updated configuration must be pushed to each collector.

When you restart the Resource Manager master, three newly installed Impact daemons are started:

Daemon	Description
zenimpactserver	Required for Impact functionality. Maintains Impact relationships and propagates state through the Impact model.
zenimpactgraph	Translates Resource Manager model changes into Impact model changes, and sends them to <code>zenimpactserver</code> .
zenimpactstate	Translates Resource Manager events into states, and sends them to <code>zenimpactserver</code> .

Table 2.1. Daemons

2.1. Backing Up Impact

Do not back up Impact via the file system unless you first confirm that the database has been cleanly shut down. Any `cron` jobs with `rsync` should take that into account, and cleanly shut down the application before taking a file system snapshot.

The Impact server must have localhost access to port 6362 for `zenbackup` to succeed.

For more information about `zenbackup`, refer to the section titled "Backup and Recovery" in *Zenoss Service Dynamics Resource Management Administration* guide.

Chapter 3. Working with Impact and Event Management

3.1. Using the Interface

Resource Manager offers multiple views that provide detailed information about dynamic services and service elements. Read the following sections for information about how to navigate the interface and use the data provided by:

- All Services
- Dynamic service views

3.1.1. All Services

This view provides a complete picture of the state of all services. The top portion of the view presents data graphically for:

- **Availability Health** - Shows the current availability of all services. The system categorizes available services as unknown, up, at risk, degraded, or down.
- **Performance Health** - Shows the current performance of all services. The system categorizes service performance as unknown, acceptable, degraded, or unacceptable.

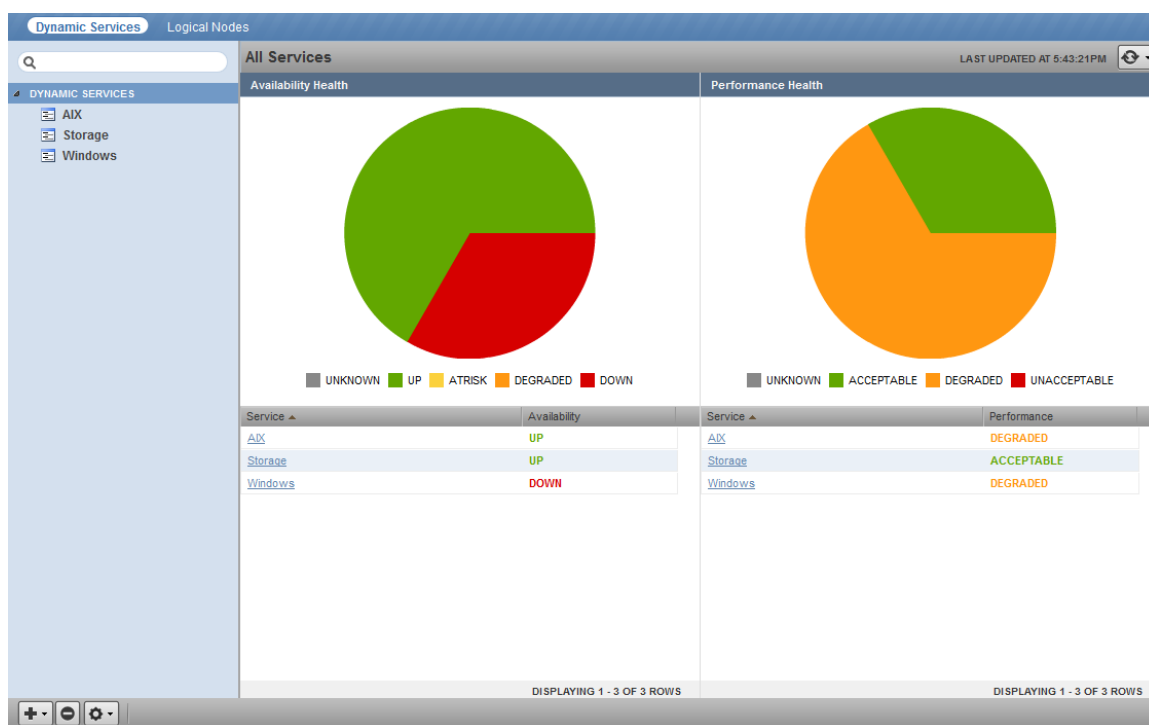


Figure 3.1. All Services

Hover over each portion of a graph to show the number of services included in the status group and the percentage of services it represents.

The lower portion of the view lists each service and its availability status. Click a service in the list to go to the Impact view for that service.

You can refresh the view manually or specify that it refresh automatically. To manually refresh the view, click **Refresh**. You can manually refresh at any time, even if you have an automatic refresh increment specified.

To configure automatic refresh, select one of the time increments from the Refresh list. By default, automatic refresh is enabled and set to refresh each minute.

3.1.2. Dynamic Service Views

You can work with individual dynamic services through several views:

- Overview
- Impact Events
- Impact View

3.1.2.1. Overview

To view the service Overview, select a service in the Dynamic Services list, in the left panel.

The service Overview lists each element that directly impacts the service. The Health column shows the current availability and performance status of each element.

Icon	Name	Health	Type
	/ (test-aix53-1.zenoss.loc)	A P	FileSystem
	/home (test-aix53-1.zenoss.loc)	A P	FileSystem
	/admin (test-aix61.zenoss.loc)	A P	FileSystem
	/tmp (test-aix53-2.zenoss.loc)	A P	FileSystem
	/opt (test-aix53-1.zenoss.loc)	A P	FileSystem
	/usr (test-aix53-2.zenoss.loc)	A P	FileSystem
	/opt (test-aix61.zenoss.loc)	A P	FileSystem

Figure 3.2. Dynamic Service Overview

Click an element name to view its overview and component information.

3.1.2.2. Impact Events

To access this view, select Impact Events from the list of options located on any of the dynamic service views.

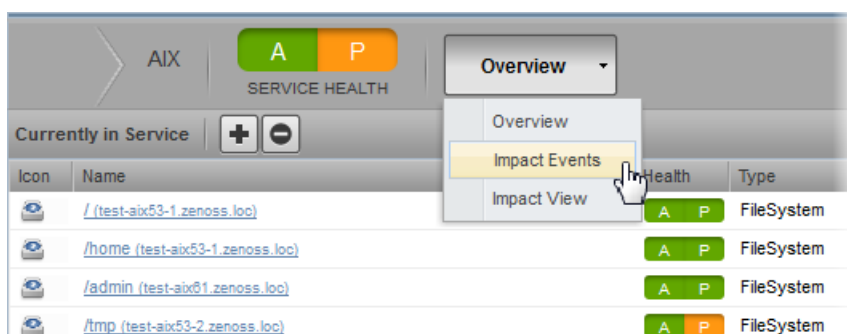


Figure 3.3. Dynamic Service View Selection




The Impact Events view lets you view and work with the events that impact the status of a service.

Status	Severity	Service	Event Class	Summary	First Seen	Last Seen	Count
DEGRADED	Warning	AIX	/Service/State/Performance	Service AIX is DEGRADED.	2011-07-21 12:10:1	2011-07-21 12:44:4	77

Confidence	Status	Severity	Resource	Component	Event Class	Summary	First Seen	Last Seen	Count
50%	Warning	Warning	test-aix61.zenoss.loc	/usr	/Perf/Filesyst	disk space threshold: 90.7% used (154.3MB free)	2011-07-20 17:16:3	2011-07-21 12:44:4	1171
			Impact chain: /usr -> AIX						
50%	Warning	Warning	test-aix53-2.zenoss.loc	/tmp	/Perf/Filesyst	disk space threshold: 100.0% used (4.0KB free)	2011-07-20 17:16:0	2011-07-21 12:44:3	1170
			Impact chain: /tmp -> AIX						

Figure 3.4. Dynamic Service Impact Events View

The top portion of the view provides summary information about events impacting the service. Select a row to display impacting event information in the lower portion of the view. From there, you can select one or more events, and then:

- Acknowledge events - Click  to acknowledge events.
- Close events - Click  to close events. The events are moved to the event archive according to a configured event archive interval.
- Return events to new status (revoke acknowledged status) - Click  to return acknowledged events to new status.

You can view events moved to the event archive related to the impact events. Select the Show historical events option.

3.1.2.3. Impact View

To access this view, select Impact View from the list of options located on any of the dynamic service views. The Impact View graphically represents the connections between the elements in a service.

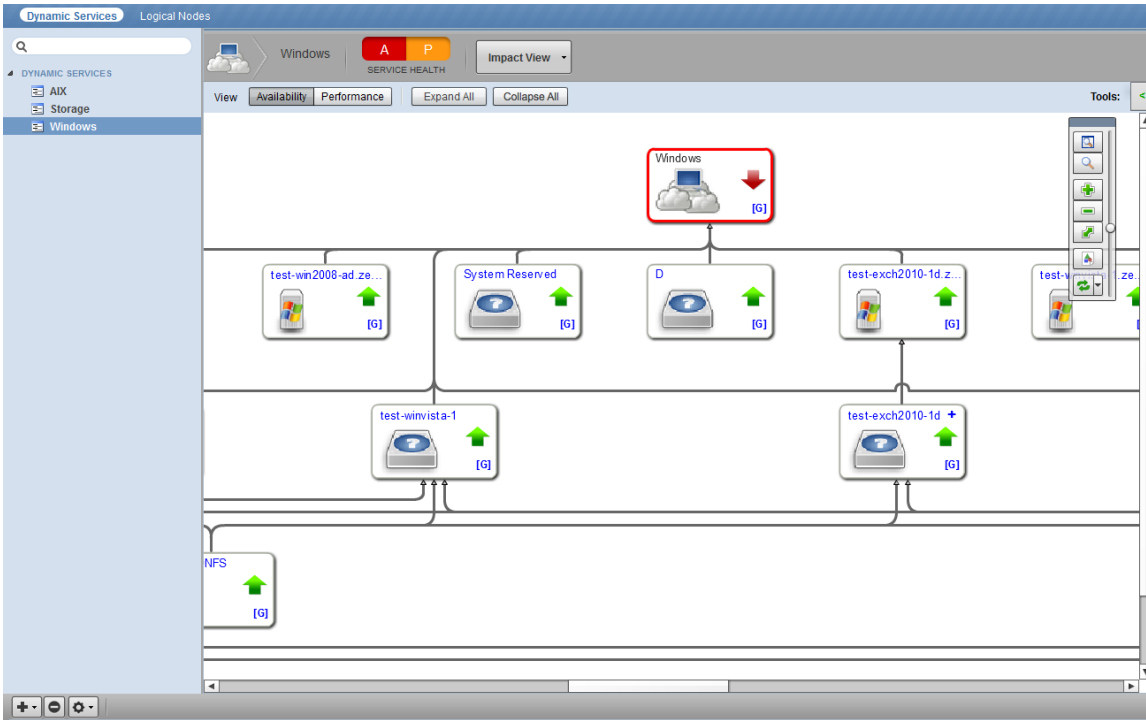


Figure 3.5. Dynamic Service Impact View

Click **Performance** or **Availability** to switch between availability and performance information in the view. By default, the Availability view displays.

Click **Expand All** to expand and display all service elements. Click **Collapse All** to collapse the view to show each element's root level and its immediate children.

Tools allow you to manipulate the view and choose the information you see. To access the Tools area, click



at the top right of the view.

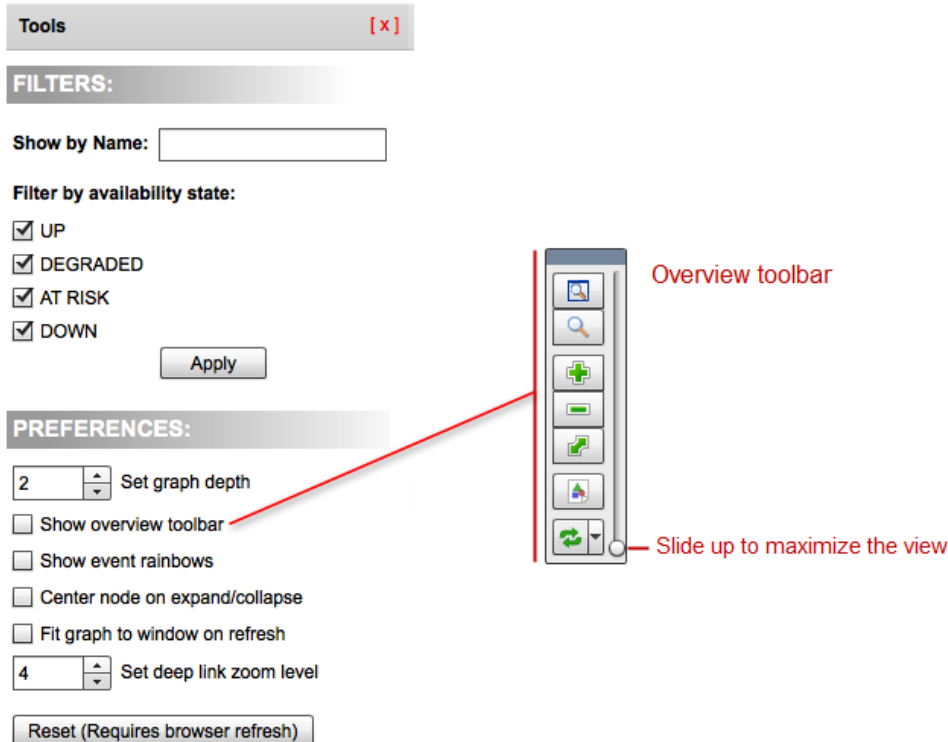



Figure 3.6. Impact View Tools






You can additionally make selections to customize your view from the overview toolbar. Select Show overview toolbar in the Tools area.

Make one or more selections from the Tools area to customize view data. Click **Apply** to apply your choices.

- **Show by Name** - Enter case-sensitive text to limit the elements that appear in the view.
- **Filter by availability state** - Make or remove selections to limit the elements that appear in the view based on their availability or performance state.
- **Preferences** - Select one or more preferences:
 - **Set graph depth** - Use the controls or enter a number to increase or decrease the number of levels in the hierarchy that appear in the view.
 - **Show overview toolbar** - Select to show a toolbar that allows you to further manipulate the view.
 - **Show event rainbows** - Select to show "event rainbows" on each element in the view. Turning this on may slow view rendering speed.
 - **Center node on expand/collapse** - Select to cause the currently collapsing or expanding node to center in the view.
 - **Fit graph to window on refresh** - Fit the graph to the current window upon refresh.
 - **Set deep link zoom level** - Specify to set the initial zoom level to render when linking to a node from outside the graph.
 - **Reset** - Return preferences to default selections on next browser refresh.

Use the overview toolbar to:

-  Toggle the hierarchy overview, a miniature representation of the service hierarchy. Drag the focus area to change the portion of the hierarchy that appears in the view.

-  Toggle the magnifier, which allows you to look more closely at a portion of the view.
-  Zoom in and out of the view
-  Fit the hierarchy to the window
-  Save an image of the view. By default, the image is named `impactviewcomponent.png`.
-  Refresh the view and set refresh intervals

3.2. Working with Dynamic Services

The following sections provide information and procedures to help you define and work with dynamic services and service elements.


- Create, edit, clone, and organize services
- Add elements to, and delete elements from, a service
- Define service policies
- Define logical nodes (create, add, delete)

3.2.1. Create a Dynamic Service

To create a dynamic service:

1. From the Resource Manager interface navigation bar, select **Services**.

The Dynamic Services page appears.

2. Click  (Add), and then select **Add Dynamic Service**.

The Add Dynamic Service dialog appears.

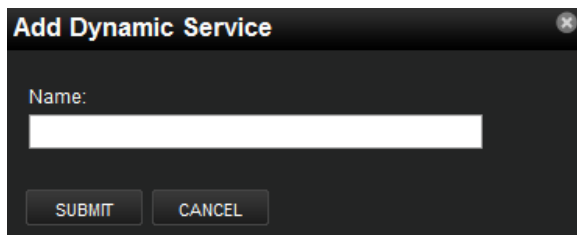



Figure 3.7. Add Dynamic Service

3. Type a name for the new dynamic service, and then click **Submit**.

The newly created service appears in the list of Dynamic Services, in the left panel. By default, the service Overview appears.

3.2.2. Edit Service Details


After creating a service, you can edit its name and add a description. To edit service details, select **View and Edit Details** from  (Action menu).

Note

Each time you edit a service--details, events, or elements--a *service event* is created.

3.2.3. Clone a Service

Cloning allows you to copy an existing service that you can then modify to quickly create a new service. To clone a service:

1. Select a service in the list of Dynamic Services.
2. Select Clone Service from  (Action menu).

The Clone Service dialog appears.

3. Enter a new name for the service, and then click **Submit**.

3.2.4. Organize Dynamic Services

You can create a structure to organize the services you create. To do this:

1. From the Resource Manager interface navigation bar, select Services.

The Dynamic Services page appears.

2. Click  (Add), and then select Add Dynamic Service Organizer.

The Add Dynamic Service Organizer dialog appears.

3. Enter a name for the organizer, and then click **Submit**.

The newly created organizer appears in the list of Dynamic Services, in the left panel.

4. Drag one or more services to the organizer folder to include them in that organizer.

3.2.5. Add Services to Another Dynamic Service

You can impact a dynamic service by adding one or more services to it. Drag a service in the dynamic services hierarchy to the service you want to contain it.

3.2.6. Add Elements to a Dynamic Service


After creating a service, you must add the service elements that define it. To add elements to a service:

1. From the Resource Manager interface navigation bar, select Services.

The Dynamic Services page appears.

2. Select a service in the list of dynamic services, in the left panel.

The service Overview appears. It shows the elements currently included in the service.

3. Click  (Add) at the top of the view.

The Add to Service dialog appears.

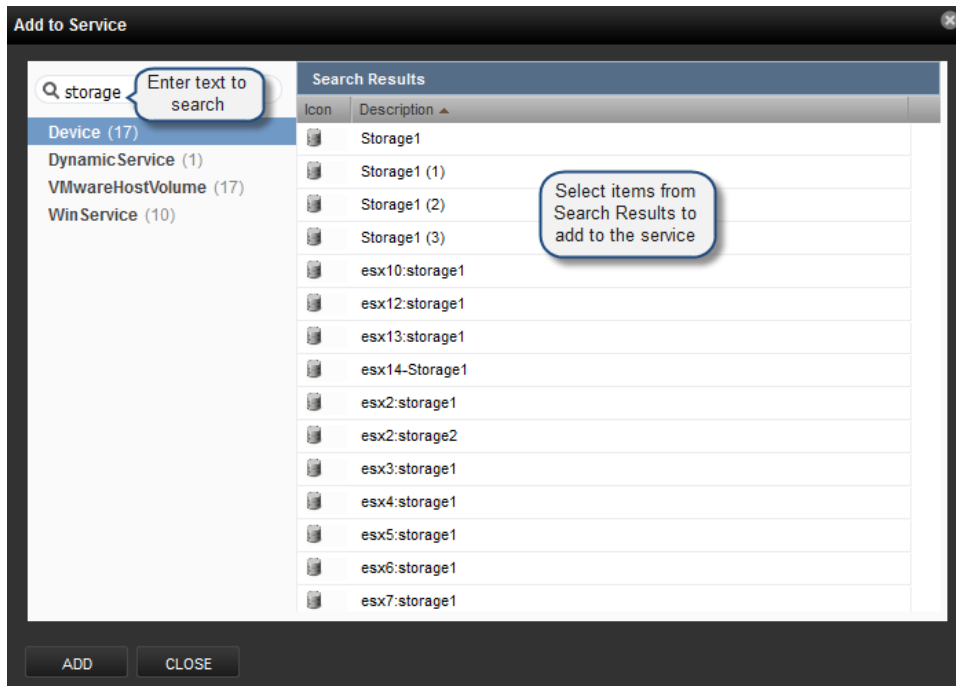


Figure 3.8. Add to Service

4. Enter text to search for elements to include in the service.

Search results appear to the right of the search area.

5. Select one or more items in Search Results, and then click **Add**.

The selected items are added to the service and are listed in the service Overview.

6. Enter a new search term to add further items to the service, or click **Close** to close the Add to Service dialog.

3.2.7. Delete Elements from a Dynamic Service

To remove an element from a service:

1. Select one or more service element rows in the Overview.
2. Click Delete.
3. Click **OK** to confirm.

The selected element or elements are removed from the service.

3.2.8. Define Service Policies

You set one or more policies on elements in a service to control the service's availability or performance state. Policies can be:

- **Contextual** - Applies to an element only in a given service. If defined, a contextual policy overrides a global policy.
- **Global** - Applies to an element in all services. If defined, a global policy overrides the default policy.

To set a policy on a service:

1. From the dynamic service Impact View, select a node.
2. Click the **[G]** icon at the bottom right of the node.

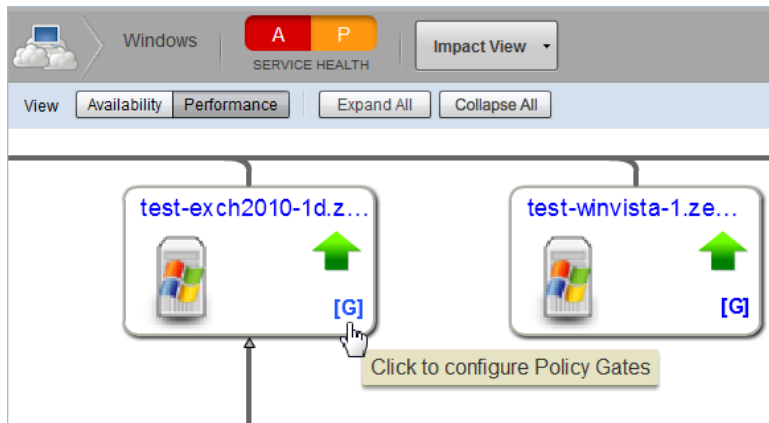


Figure 3.9. Configure Policy

The Impact Policies dialog appears.

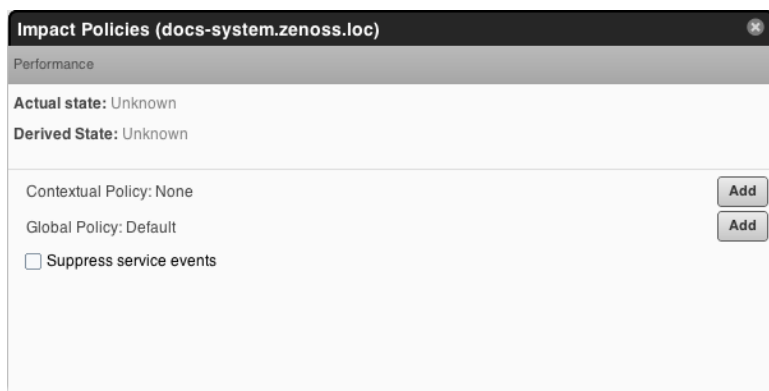


Figure 3.10. Impact Policies

- To set a contextual or global policy, click **Add**.

For both contextual and global policy types, the dialog re-displays with a series of selections that let you define the policy. Click **Add** to begin, and then enter or select values:

- **My state will be** - Select the state of this service if the trigger applies. Selections are DOWN, DEGRADED, ATRISK, or UP.
- **If** - Set the conditions that will trigger a state change. To specify a percentage rather than an absolute value, select the % option (check box).
- **Of type** - Select the entity to which the trigger will apply.
- **Are** - Select the state of the impacting elements (DOWN, DEGRADED, ATRISK, or UP).

- Click **Save Changes**.

Click **Back** in the dialog to return to the performance overview information.

Example

For example, if you want to set your state to be DEGRADED if two or fewer percent of any entities are UP, you would make these selections:

- My state will be: DEGRADED
- If: <= 2% (select the option box to specify percent)
- Of type: Any
- Are: UP

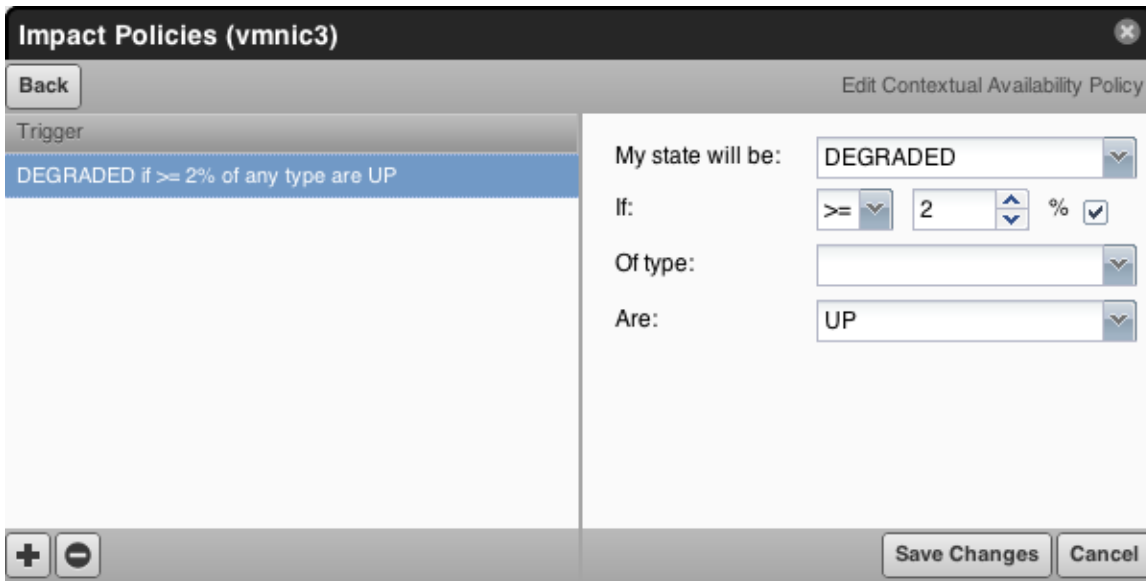


Figure 3.11. Example: Contextual Policy

3.2.8.1. Custom State Providers

The states of elements are determined by events coming in to Resource Manager, and are evaluated by the `zenimpactstate` daemon. In some cases, however, you might choose to define a custom state provider. Given specific events on a device or component, events matching that event class will cause the element to be in a specific state.

3.2.8.2. Suppressing Service Events

If you have defined services that are used solely for grouping, and you do not need information about the status of these events, select the Suppress service events option when defining a policy. State calculation and propagation are unchanged, but Resource Manager does not send any events into Resource Manager.

3.2.9. Define a Logical Node

Logical nodes represent groupings of events in the Impact graph for nodes that do not exist in Resource Manager. Logical nodes comprise an event filter and a simple state provider. The state provider maps events in a given event class to states based on event severity (for example, Critical > DOWN or Error > DEGRADED).

For example, you might create a Web transaction node so that certain `/Status/Web` events can affect the state of the graph.

Name:

Description:

Criteria:
 of the following rules:

Device	equals	<input type="text"/>	<input type="button" value="+"/>	<input type="button" value="-"/>	<input type="button" value="⋮"/>
Event class	starts with	/Status/Web	<input type="button" value="+"/>	<input type="button" value="-"/>	<input type="button" value="⋮"/>

Availability State:

Events for this node in this event class:

will result in these availability states:

Critical:	<input type="text" value="DOWN"/>
Error:	<input type="text" value="DEGRADED"/>
Warning:	<input type="text" value="ATRISK"/>
Info:	<input type="text"/>
Debug:	<input type="text" value="UP"/>
Clear:	<input type="text" value="UP"/>

Performance State:

Events for this node in this event class:

will result in these performance states:

Critical:	<input type="text" value="UNACCEPTABLE"/>
Error:	<input type="text" value="DEGRADED"/>
Warning:	<input type="text" value="ACCEPTABLE"/>
Info:	<input type="text"/>
Debug:	<input type="text" value="ACCEPTABLE"/>
Clear:	<input type="text" value="ACCEPTABLE"/>

Figure 3.12. Web Transaction Node

Logical nodes can be added only to services; they cannot affect other elements deeper in the graph.

You can configure logical nodes through the interface. Each event that matches the event filter triggers evaluation of the state provided on the logical node.

When defining a node, you can specify any event subclass it contains by ending with a trailing slash. For example:

- /Status - Includes only the /Status class
- /Status/Web - Includes only the /Status/Web class
- /Status/ - Includes any subclass of /Status, including Status/Web

Follow these steps to define a logical node:

1. From the Resource Manager interface navigation bar, select Logical Nodes.

The Logical Nodes page appears.

2. Click (Add), and then select Add Logical Node.

The Add Logical Node dialog appears.

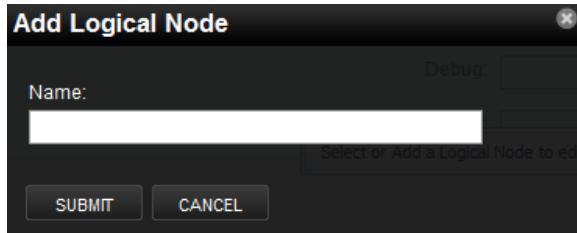


Figure 3.13. Add Logical Node

3. Type a name for the new logical node, and then click **Submit**.

The newly created logical node appears in the left of logical nodes, in the left panel.

4. Enter information or make selections to define the logical node:
 - **Description** - Optionally enter a description of the node.
 - **Criteria** - Set up the rules that define the node. By default, a single rule appears. Click **Add** to define further rules.
 - **Availability State** - Enter an event class, and then select an availability state for each event type.
 - **Performance State** - Enter an event class, and then select a performance state for each event type.
5. Click **Save**.

3.2.10. Service Impact Notifications

You can set up notifications for service impact events in the same way as other standard Resource Manager events. If you want to receive service impact events specifically, you can add an "Event Class Start with / Service/State" rule to a new trigger. (Alternatively, you can enable the "Example Service Trigger" and use it in a notification.)

You can use standard notifications actions (such as command, email, page, and SNMP trap) with service impact events. For these events there is also an SNMP Trap w/Impact notification action that differs from the standard SNMP Trap action. SNMP Trap sends a ZENOSS-MIB::zenGenTrap; SNMP Trap w/Impact sends a ZENOSS-IMPACT-MIB::zenImpactEvent. zenImpactEvent includes more service and root cause information. You can find more details in ZENOSS-IMPACT-MIB, in `$ZENHOME/ZenPacks/ZenPacks.zenoss.Impact*/ZenPacks/zenoss/Impact/share/mibs/ZENOSS-IMPACT-MIB.txt`.

3.2.11. Adding Context Information to Notifications

In addition to the context information available to notifications, Resource Manager events populate a variable (`esa`) which has one variable (`causes`), which is a list. Each list entry is an event, with these attributes:

- `evt`
- `clearEvt`
- `eventSummary`
- `clearEventSummary`
- `urls`

Another entry, `impactChain`, contains the chain of resources that Resource Manager uses to determine that this event caused the service problem. The causes are listed in order of probability of root cause, from highest to least.

To iterate over the list items in `causes` using TALES, you can use the `tal:repeat` clause. For example:

```
<tal:block tal:repeat="item esa/causes">
Impact Chain: ${item/impactChain}
Device: ${item/evt/device}
Component: ${item/evt/component}
Severity: ${item/evt/severity}
```

```

Time: ${item/evt/lastTime}
Message:
${item/evt/message}
<a tal:attributes="href item/urls/eventUrl">Event Detail</a>
<a tal:attributes="href item/urls/ackUrl">Acknowledge</a>
<a tal:attributes="href item/urls/closeUrl">Close</a>
<a tal:attributes="href item/urls/eventsUrl">Device Events</a>
</tal:block>

```

To use solely the most probable cause, use `tal:define` and a Python expression:

```

<tal:block tal:define="topcause python:esa['causes'][0]">
Impact Chain: ${topcause/impactChain}
Device: ${topcause/evt/device}
Component: ${topcause/evt/component}
Severity: ${topcause/evt/severity}
Time: ${topcause/evt/lastTime}
Message:
${topcause/evt/message}
</tal:block>

```

For more information about notifications, see the section titled "Working with Notifications" in *Resource Manager Administration*.

3.3. Example Service Scenario: Redundant Components

This example will show how to construct a service model that represents a service provided by a single application, as represented by the following network diagram.

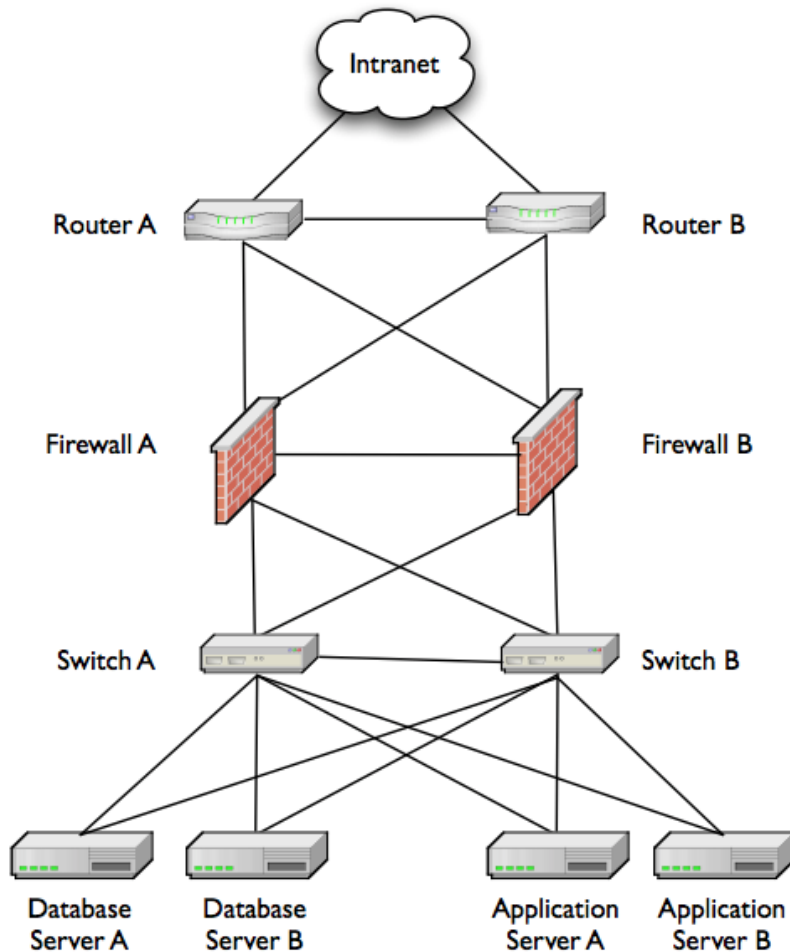


Figure 3.14. Single Application Service

You might initially consider constructing a service model that mirrors the Layer 2 topology; however, this likely will not produce effective service assurance.

Instead, you should consider the services provided, and what is required for them to be considered in a "good" state. In this case, the service you want to assure is a load-balanced CRM application, running on Application Servers A and B, and using a database cluster running on Database Servers A and B.

Three conditions must apply to this service for it to be considered "up":

- At least one instance of the application must be running
- At least one database server must be running
- There must be at least one complete path that provides connectivity to the application server

Step 1: Create a Service and Sub-Services

First, create a two-level CRM service and three sub-services. When created, they will appear in the Impact View as shown in the following figure.

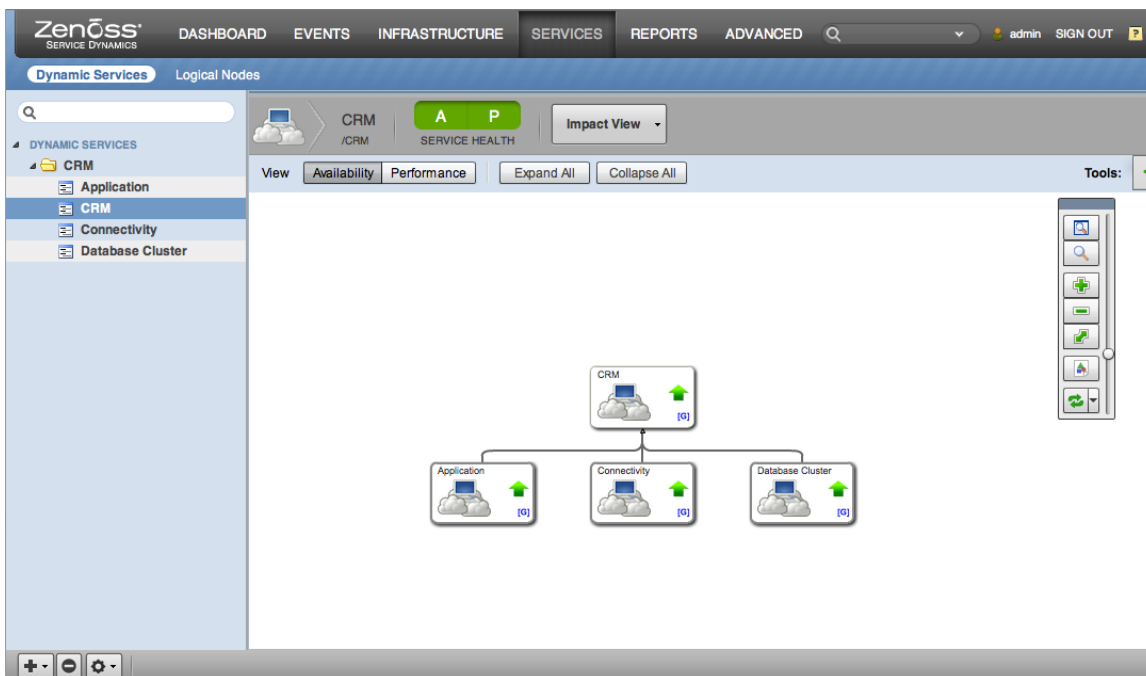


Figure 3.15. Dynamic Service: CRM

Step 2: Create the Application Service

Create the Application service. The critical component of each instance of the application is the process running on each application server. Add those to the Application service, as shown in the following figure.

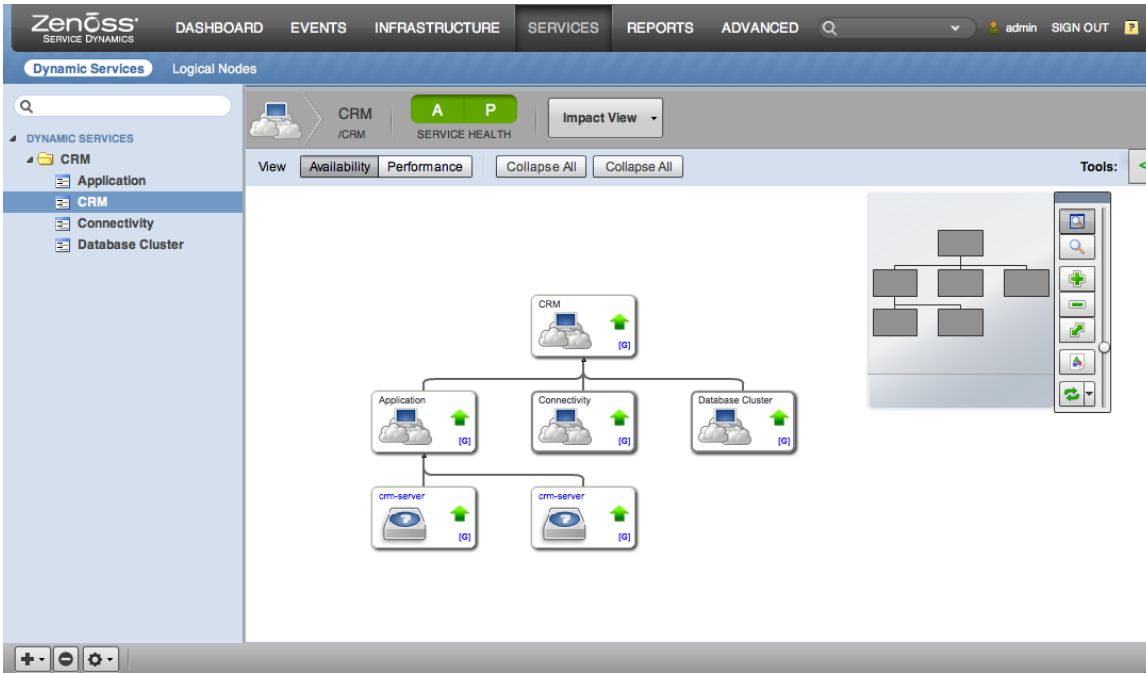


Figure 3.16. Dynamic Service: Application

Step 3: Set Global Policy on the Application Service

The Application service should include a global policy describing its availability. It should be considered down only if all of the CRM server processes are down, and at risk if one or more of them is down.

The following figures illustrate selections to make when setting up the global policy for the Application service.

The screenshot shows a dialog box titled 'Impact Policies (Application)'. It contains a 'Back' button and a section for 'Edit Global Availability Policy'. The configuration is as follows:

Target State	Type
DOWN	Percentage
ATRISK	Threshold

Configuration fields:

- Target State: DOWN
- Type: Percentage
- Number/Percent Of Dependents: 100
- Dependents' State: DOWN

Buttons at the bottom include '+', '-', 'Save Changes', and 'Cancel'.

Figure 3.17. Application Service Global Policy: Target State DOWN

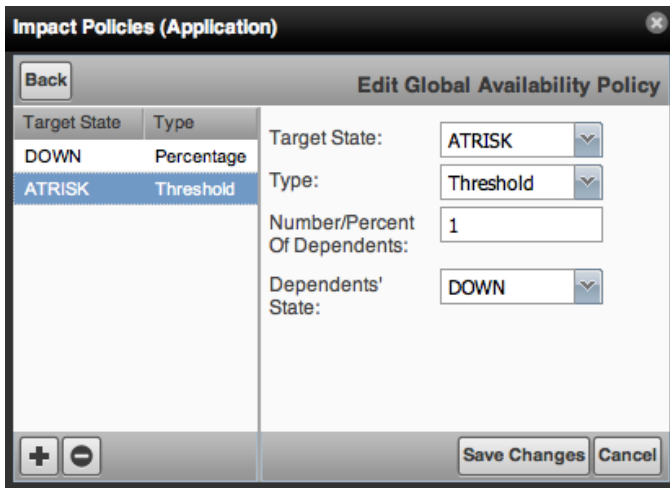


Figure 3.18. Application Service Global Policy: Target State ATRISK

Step 4: Create the Database Cluster Service

Set up the Database Cluster service similarly to the Application service. Add the database server processes to the newly created service, and set the same global policy. The resulting impact graph appears in the following figure.

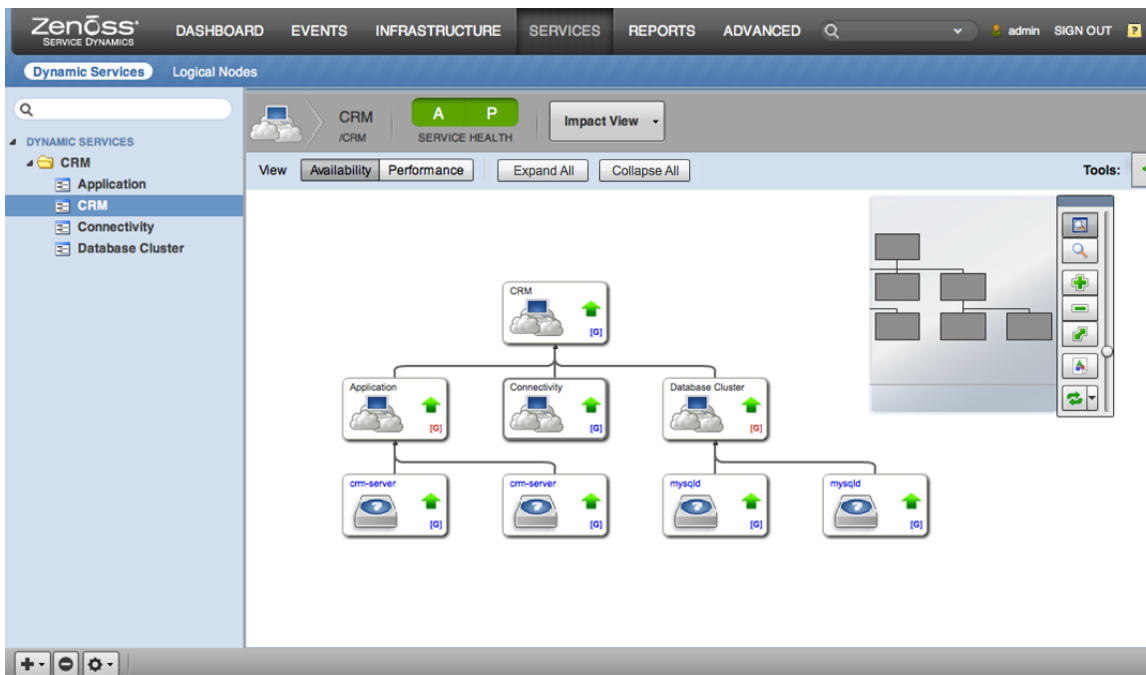


Figure 3.19. Dynamic Service: Database Cluster

Step 5: Create the Connectivity Service

Given the high level of redundancy, this service requires a much deeper service graph. The basic strategy is to wrap the interfaces that compose a link between two components into a service that represents the link itself, and then wrap redundant links in a service representing connectivity between layers. This results in three services that comprise the Connectivity service:

- Router-Firewall Connectivity
- Firewall-Switch Connectivity
- Switch-Server Connectivity

The process is nearly the same for each layer. Beginning with the Router-Firewall Connectivity service, create four services that represent the links containing the interfaces, each with no policy (meaning that, if either end-point is down, the link is down).

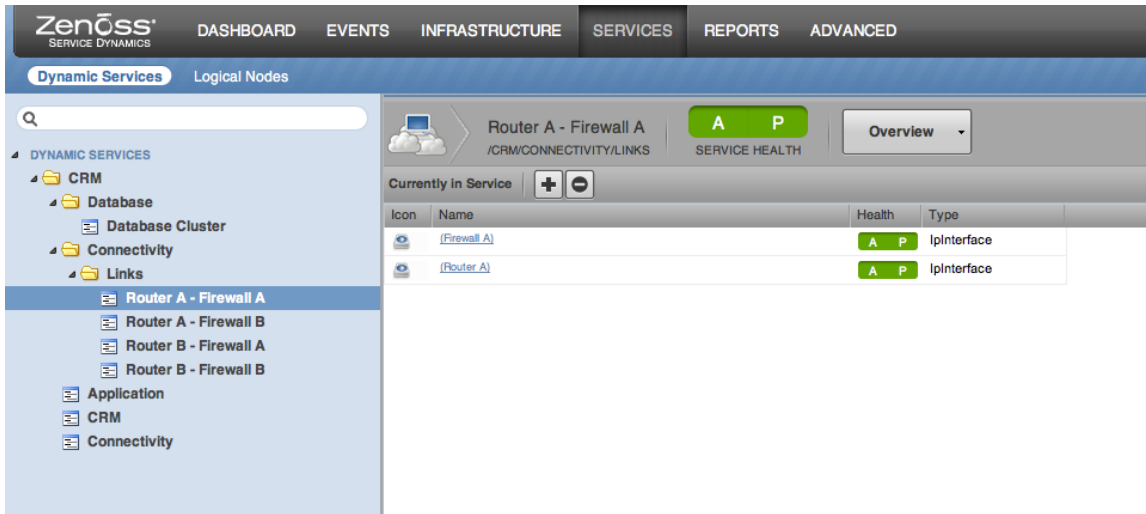


Figure 3.20. Dynamic Service: Router-Firewall Connectivity

Next, create services to handle the redundancy of the links from each router to the firewall layer, and put each pair of links into one. Create a global policy indicating that the pair is down only if both links are down, and at risk if one is down.

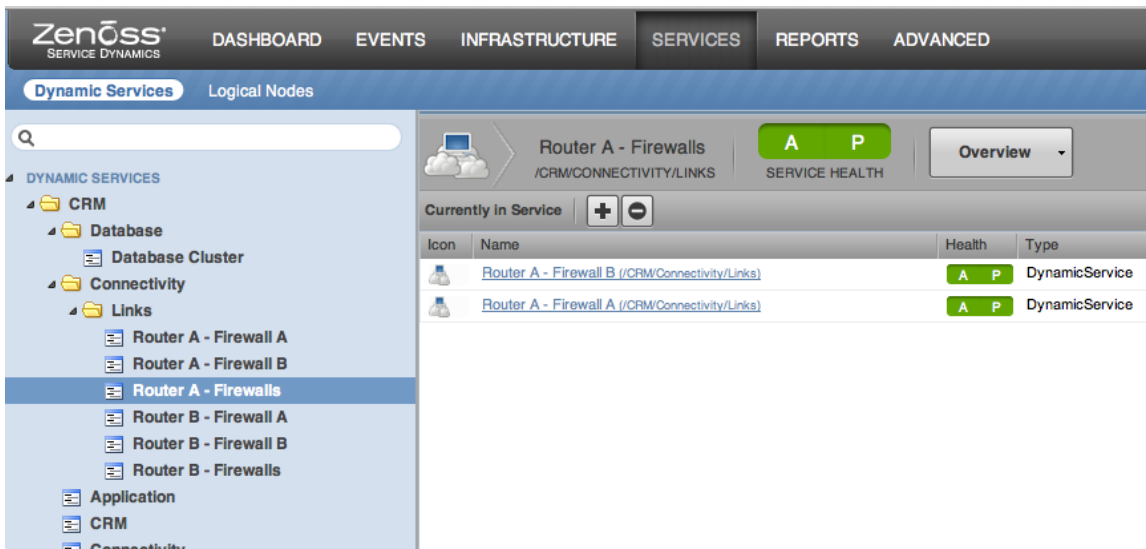


Figure 3.21. Connectivity Links

Finally, create a service that represents overall connectivity between the router layer and the firewall layer. Put the two Router-Firewall services into that service.

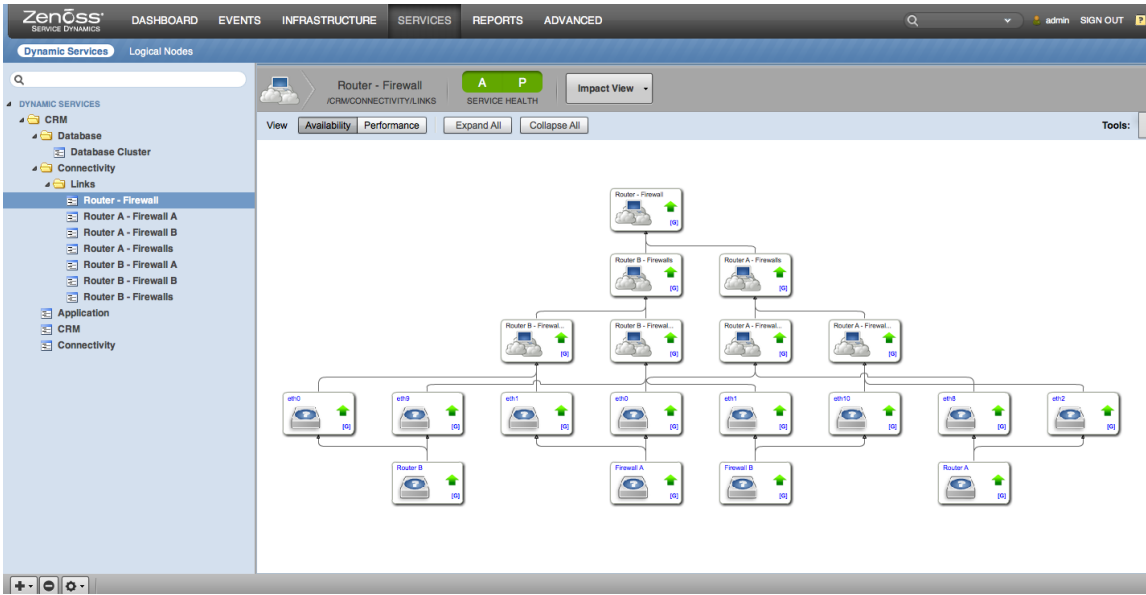


Figure 3.22. Router-Firewall Connectivity Service

Note

The device for each interface is automatically included in the Impact graph, since if the device pings down, the interface also is considered down.

Follow the steps used to create the Router-Firewall Connectivity service to create the Firewall-Switch Connectivity service.

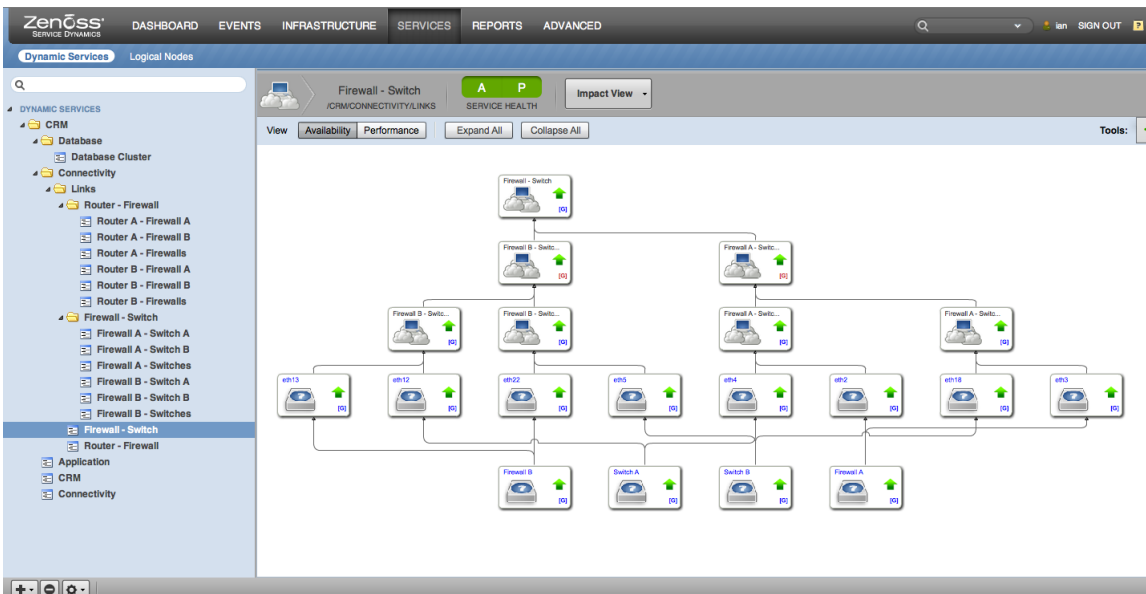


Figure 3.23. Firewall-Switch Connectivity Service

Finally, apply the same idea to create the Switch-Server Connectivity service. This service is more complex due to the number of servers on the Server layer. Because the purpose of the servers in this service is known, this service should include another layer of services (Database Servers and Application Servers), so that you can be informed if, for example, connectivity to the database servers is lost. Since connectivity to at least one database server and one application server is necessary, this is essential. The resulting graph is one layer deeper (on the Database Server side).

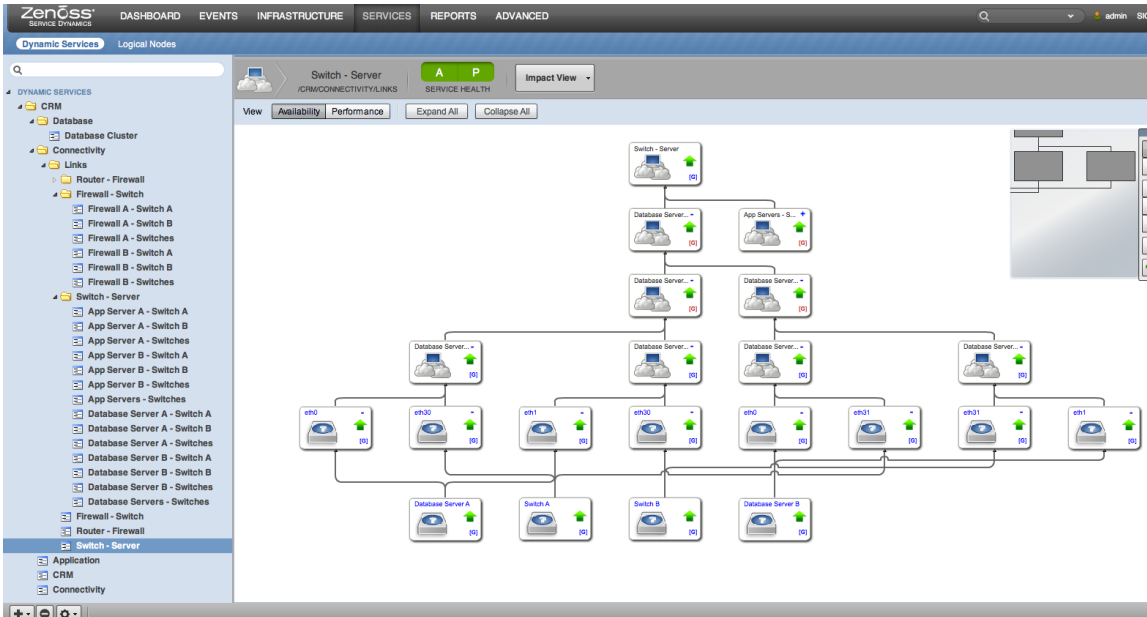


Figure 3.24. Switch-Server Connectivity Service

Step 6: Validate the Service

After the CRM service is defined and all redundancies are accounted for, you can validate the service.

Start by sending an event indicating that one of the MySQL processes is down. As shown in the following figure, the top-level CRM service is merely at risk, because at least one member of the database cluster is functioning properly.



Figure 3.25. Database Cluster: At Risk

Now, send an event indicating that an entire switch is down. As shown in the following figure, connectivity is still merely at risk, because of the level of redundancy.

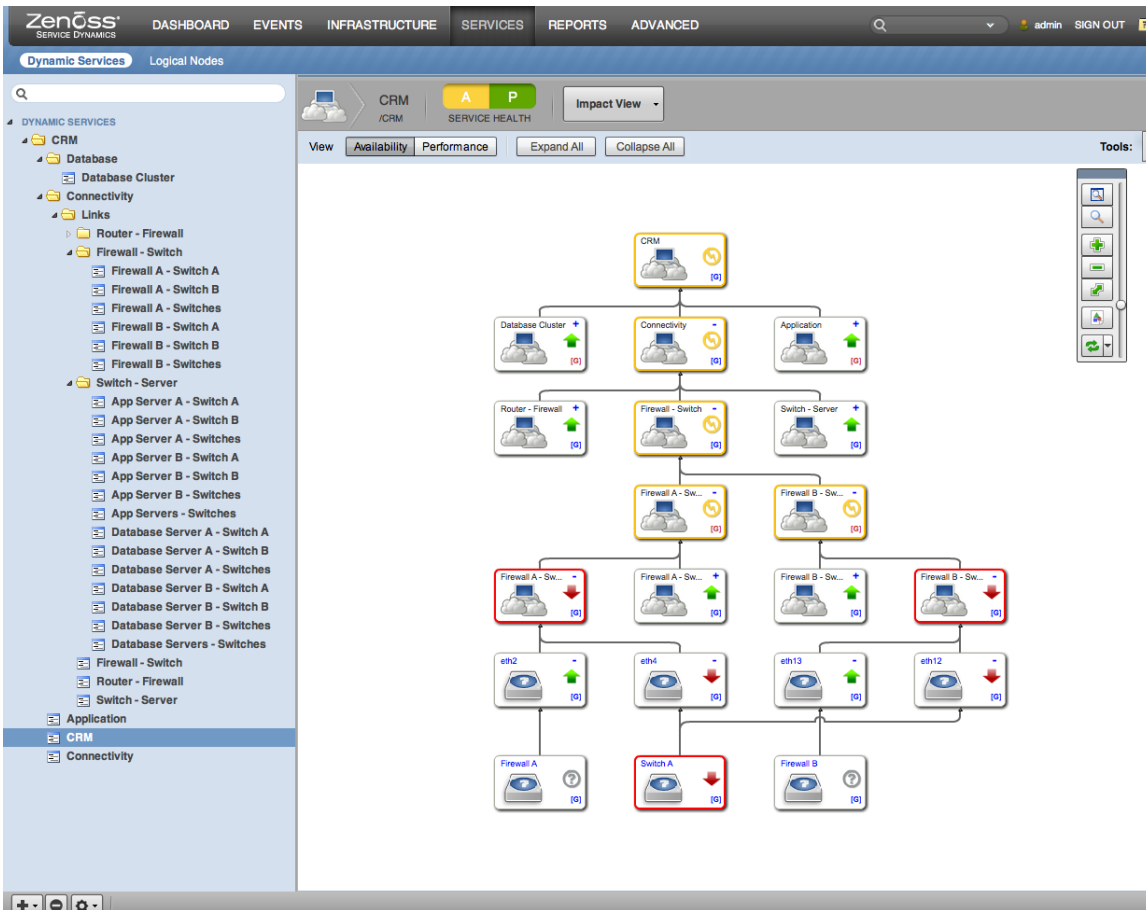


Figure 3.26. Database Cluster: At Risk

Send an event that takes down a firewall and a router. As shown in the following figure, connectivity still is only at risk, because there is a full path from the intranet to the servers.

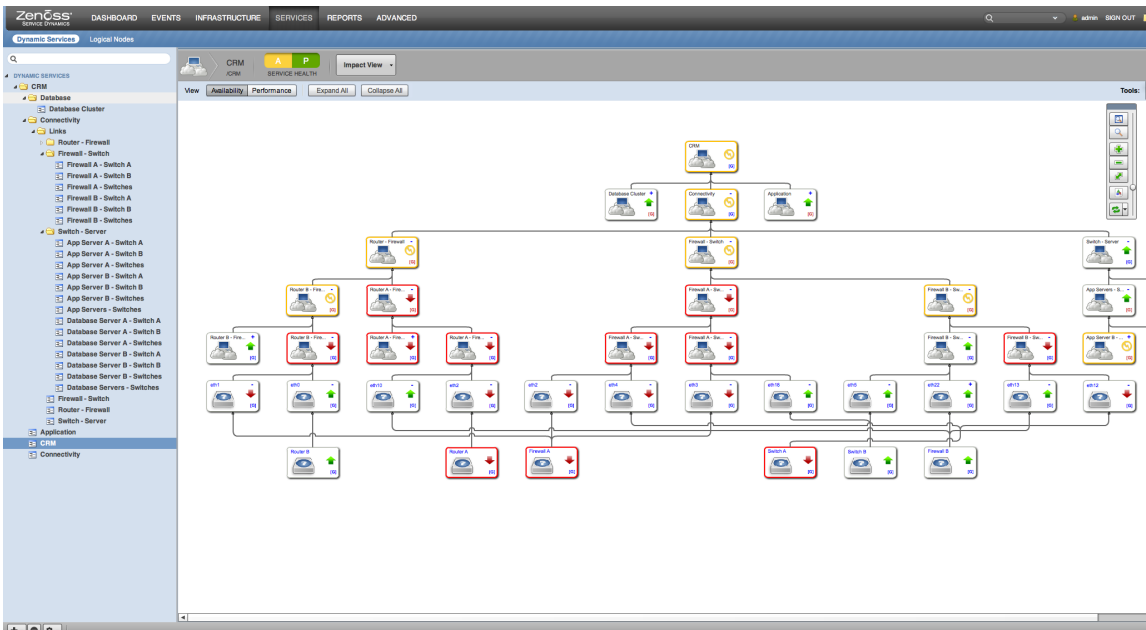


Figure 3.27. Database Cluster: At Risk

Finally, send an event that takes down the other firewall. This time, the CRM service is down, because connectivity to the servers from the intranet has been lost.

Working with Impact and Event Management

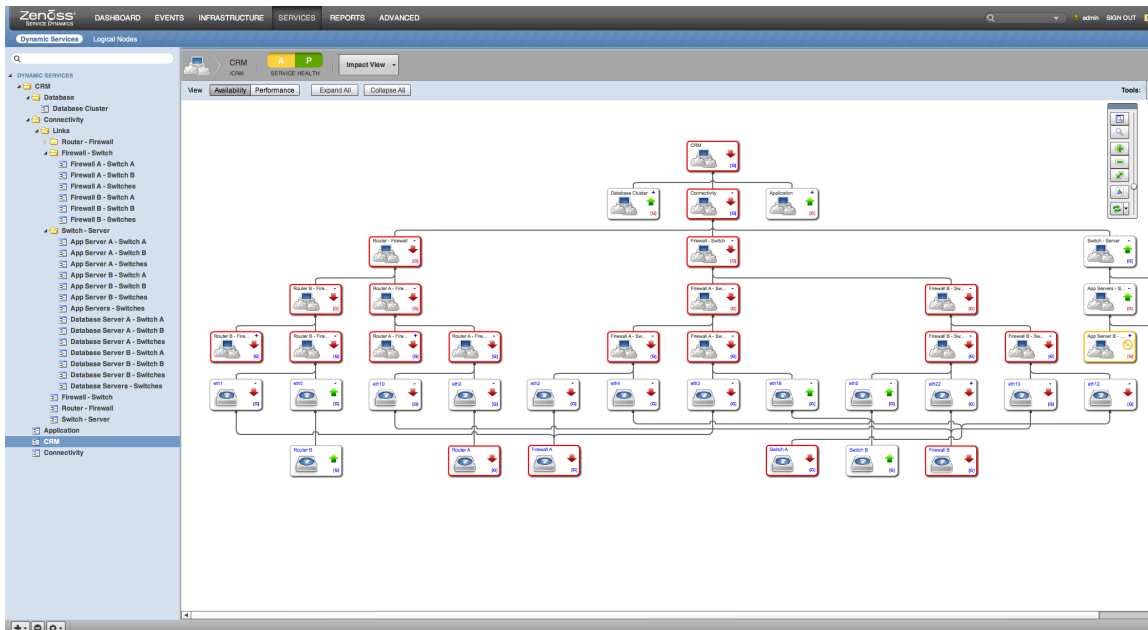


Figure 3.28. Database Cluster: Down