Health Rules

Understanding Health Rules

AppDynamics collects a wide range of metric information covering your entire application. Some of those metrics are key indicators of the overall state of the system. Being able to proactively track the status of those indicators gives you a window into the health of your system and can inform you when various important entities—nodes, business transactions, databases, etc.—may be in trouble.

What is a health rule?

<p>Health rules allow you to define acceptable values for key metrics associated with specific entities, and to monitor those essential metrics automatically. Should metric values exceed the ranges you have specified, the health rule is said to violate, and a health

rule violation event occurs and is surfaced in the controller user interface. The violation event can also be used to trigger a policy, which can initiate pre-defined actions to respond to the situation, from sending alerting emails to running remedial scripts. To understand how to create policies and actions, see Policies and Actions. The simplest way to create health rules is to use the basic health rule wizard. The wizard groups commonly used system entities and related metrics to ease the process of setting up your particular system's health rules. Should you need to create health rules that connect less commonly used entities and/or metrics, you can use one of the custom methods in the wizard, the Custom Health Rule Types or the Hybrid methods. For specifics on using the health rule wizard, see Configure Health Rules. <h1 id="HealthRules-UnderstandingtheHealthRuleWizard" >Understanding the Health Rule Wizard</h1> To understand how the health rule wizard works, you need to understand four basic concepts: How the wizard groups entities and metrics
How the wizard schedules health rule evaluations
How the wizard defines which entities are affected
How the wizard defines the metric conditions that are to be evaluated on those entities
<h2 id="HealthRules-HeathRuleTypes">Heath Rule Types</h2> To simplify creating health rules, the basic health rule wizard groups entities, like nodes or business transactions, with metrics that are commonly associated with those entities, into health rule types. Doing so allows the wizard to automatically show you relevant information during the health rule creation process. If your needs are not covered by these types, you can use one of the custom methods in the wizard, Custom Health Rule Types or the Hybrid methods. The health rule types are: Transaction Performance
Overall Application Performance: groups metrics related to load, response time, slow calls, stalls, with applications
Business Transaction Performance: groups metrics related to load, response time, slow calls, stalls, etc. with business transactions
Error Rates: groups metrics related to exceptions, return codes, and other errors with applications or tiers
Node Health
Node Health-Hardware, JVM, CLR: groups metrics like CPU and heap usage, disk I/O, etc. with nodes
Node Health-Transaction Performance: groups metrics related to load, response time, slow calls, stalls, etc. with nodes
Node Health-JMX: groups metrics related to connection pools, thread pools, etc with nodes
Databases & Remote Services: groups metrics related to response time, load, or errors with databases and other backends
End User Experience
Pages: groups metrics like DOM building time, JavaScript errors, etc. with the performance of application pages for the end user
Iframes: groups metrics like first byte time, requests per minute, etc.

with the performance of iframes for the end user
Ajax Requests: < /strong>groups metrics like Ajax callback execution time, errors per minute, etc. with the performance of Ajax requests for the end user
Information Points:< /strong> groups metrics like response time, load, or errors with information points< /li>
Service Endpoints: groups metrics like average response time, calls per minute, and errors per minute with service endpoints. New in 3.9.7
< /ul> If you select one of these health rule types, AppDynamics automatically presents you with a list of the commonly associated metrics, simplifying the health rule creation process by giving you a manageable number of relevant options. If the types do not cover the entities and/or metrics you wish to use, you can use one of the custom options:< /p>
Select the Custom health rule type, which allows you to create a rule based on any metric AppDynamics collects on any single entity that AppDynamics monitors
Use the Hybrid method, which allows you to create a rule based on any metric AppDynamics collects across multiple entities. If you wish to create health rules based on custom metrics that cover multiple entities, see Using the Hybrid Method.
<h2 id="HealthRules-HealthRuleSchedules">Health Rule Schedules< /h2> The metrics associated with a health rule are evaluated according to a schedule that you control. You can configure: when a health rule is in effect <a href="#HealthRules-HealthRuleEvaluationWindow">which data set should be used, based on time< /a>
what special rules should be in place during a violation event
<h3 id="HealthRules-HealthRuleEnabledSchedule">Health Rule Enabled Schedule</h3> By default, health rules are always enabled. But you can also configure your own schedules during which the rule is in effect.
Built-in schedules are: End of business hour< /li>
Weekday lunch
Weekday mornings
Weekdays
Weekends
< /ul> You can also create a new schedule based on UNIX cron expressions using your custom values. <h3 id="HealthRules-HealthRuleEvaluationWindow">Health Rule Evaluation Window</h3> Different kinds of metrics may provide better results using different sets of data. You can manage how much data AppDynamics uses when it evaluates a particular health rule by setting the data collection time period. The default value is 30 minutes. For metrics based on an average calculation, such as average response time, AppDynamics averages the response time over the evaluation window. A five minute window means that the last five minutes of data is used to evaluate if the health rule is in range. For metrics based on a sum calculation, such as number of calls, AppDynamics uses the total number of calls counted during the evaluation window. And so forth. Use values that work best with the kinds of metrics you are interested in. <h3 id="HealthRules-HealthRuleWaitTimeAfterViolation">Health Rule Wait Time After Violation</h3> The health rule wait time setting lets you control how often an action is taken while the conditions found to violate a health rule continue to violate that health rule. When the Controller first detects that a health rule's critical or warning condition has been violated, a health rule violation is opened with the corresponding status of Critical or Warning. An Open Critical or Open Warning event is generated to trigger any actions registered on policies that match the health rule and event. The Controller continues to evaluate the health rule every minute. If the Controller continues to detect the same health rule critical or warning condition violation, the health rule violation remains open with the same violation status. A corresponding Continues Critical or Continues Warning event may be generated to trigger any actions registered on policies that match the health rule and event. The health rule's "Wait Time after Violation" setting is used to throttle how often these Continues events are generated for continuing health rule violations. For example, every minute? Or every 60 minutes? The default is every 30 minutes. In the event that the

Controller is unable to evaluate the rule - for example, if a node simply stops reporting - the Evaluation Status of the health rule is marked as a grey question mark or "Unknown". The current violation event remains open until the "Wait Time after Violation" period has elapsed, at which point the violation event is closed and a new event is triggered, causing the Health status itself of the rule to display as "Unknown". Note that the values presented in the Health Rules Violations page for violating health rule conditions are only updated when a health rule violation event is triggered. To use Continues Critical and Continues Warning events, adjust the default Wait Time after Violation value to the desired frequency on a health rule. Then configure a policy matching that health rule with the "Health Rule Violation Continues - Warning" and/or "Health Rule Violation Continues - Critical" events selected in the "Health Rule Violation Events" section of the policy's settings. <h2 id="HealthRules-HealthRuleEntities">Health Rule Entities</h2> A health rule can evaluate metrics associated with an entire application or a very limited set of entities. For example, you can create business transaction performance health rules that evaluate certain metrics for all business transactions in the application or node health rules that cover all the nodes in the application or all the nodes in specified tiers. The default health rules are in this category. You can also create health rules that are very narrowly applied to a limited set of entities in the application, or even a single entity such as a node or a JMX object or an error. For example, you can create a JMX health rule that evaluates the initial pool size and number of active connections for specific connection pools in nodes that are share certain system properties. The health rule wizard lets you specify precisely which entities the health rule affects, enabling the creation of very specific health rules. For example, for a business transaction you can limit the tiers that the health rule applies to or specific business transactions by name or by names that match certain criteria. <img class="confluence-embedded-image" width="500" src="/download/attachments/24085922/HRule_BT_affects.png?version=1&modificationDate=1402681455000&api=v2" data-image-src="/download/attachments/24085922/HRule_BT_affects.png?version=1&modificationDate=1402681455000&api=v2" data-unresolved-comment-count="0" data-linked-resource-id="24249429" data-linked-resource-version="1" data-linked-resource-type="attachment" data-linked-resource-default-alias="HRule_BT_affects.png" data-base-url="https://docs.appdynamics.com" data-linked-resource-content-type="image/png" data-linked-resource-container-id="24085922" data-linked-resource-container-version="32"> For node health rules, you can specify that a health rule applies only to nodes that meet certain criteria: <img class="confluence-embedded-image" width="500" src="/download/attachments/24085922/matchnode.png?version=1&modificationDate=1402681454000&api=v2" data-image-src="/download/attachments/24085922/matchnode.png?version=1&modificationDate=1402681454000&api=v2" data-unresolved-comment-count="0" data-linked-resource-id="24249423" data-linked-resource-version="1" data-linked-resource-type="attachment" data-linked-resource-default-alias="matchnode.png" data-base-url="https://docs.appdynamics.com" data-linked-resource-content-type="image/png" data-linked-resource-container-id="24085922" data-linked-resource-container-version="32"> <h3 id="HealthRules-EntitiesAffectedbyaHealthRule">Entities Affected by a Health Rule</h3> If you are using the basic health rule wizard and health rule types, AppDynamics provides a list of commonly used sets of entities on which metrics can be evaluated. For an Overall Application Performance health rule type, the health rule applies the entire application, regardless of business transaction, tier, or node. For a Business Transaction Performance health rule type, you can apply the health rule to: All Business Transactions in the

application
All Business Transactions within tiers that you select
Individual Business Transactions that you select
Business Transactions with names that have patterns matching criteria that you specify (such as all Business Transactions with names that start with "INV") For an Error Rates health rule type, you can apply the health rule to: All Errors in the application
Specific error types that you select
Errors with the specified tiers
Errors with names that have patterns matching criteria that you specify For a Node Health – Transaction Performance or Node Health – Hardware, JVM, CLR health rule types, you can apply the health rule to: All tiers in the application
Individual tiers that you specify
All nodes in the application
Nodes types, such as Java nodes, PHP nodes, etc.
Nodes within specified tiers
Individual nodes that you specify
Nodes with names, meta-data, environment variables or JVM system environment properties with matching criteria that you specify For a Node Health – JMX health rule type, you select the JMX objects on which the health rule is evaluated and apply the health rule to: All nodes in the application
Nodes within tiers that you specify
Individual nodes that you specify
Nodes with names matching criteria that you specify For a Databases & Remote Services health rule type, you can apply the health rule to: All databases and remote services in the application
Individual databases and remote services that you specify
Databases and remote services with name matching criteria that you specify For End User Experience – Pages, iframes, and Ajax Requests health rule types, you can apply the health rule to: All such entities
Entities that you specify
Entities with names matching criteria that you specify For Information Points health rule types, you can apply the health rule to: All information points
Information points that you specify
Information points with names matching criteria that you specify Using the Custom health rule type limits you to a single entity hard-coded in the metrics themselves. If you want to use custom metrics but associate them with multiple entities, use the hybrid method. See Using the Hybrid Method. <h3 id="HealthRules-HealthRuleEvaluationScope">Health Rule Evaluation Scope</h3> The health rule evaluation scope defines how many nodes in the affected entities must violate the condition before the health rule is considered violated. Evaluation scope applies only to business transaction performance type health rules and node health health rules in which the affected entities are defined at the tier level. For example, you may have a critical condition in which the condition is unacceptable for any node, or you may want to trigger the violation only if the condition is true for 50% or more of the nodes in a tier. Options for this evaluation scope are: any node – If any node exceeds the threshold(s), the violation fires.
percentage of the nodes – If x% of the nodes exceed the threshold(s), the violation fires.
number of nodes – If x nodes exceed the threshold(s), the violation fires.
the tier average – Evaluation is performed on the tier average instead of the individual nodes.
<h2 id="HealthRules-HealthRuleConditions">Health Rule Conditions</h2> You define the acceptable range for a metric by establishing health rule conditions. A health rule condition defines what metric levels constitute a Warning status and what metric levels constitute a Critical status. A condition consists of a Boolean statement that compares the current state of a metric against one or more static or dynamic thresholds based on a selected baseline. If the condition is true, the health rule violates. The rules for evaluating a condition using multiple thresholds depend on configuration. Static thresholds are straightforward. For example, is a business transaction's average response

time greater than 200 ms? Dynamic thresholds are based on a percentage in relation to, or a standard deviation from, a baseline built on a rolled-up baseline trend pattern. For example, a daily trend baseline rolls up values for a particular hour of the day during the last thirty days, whereas a weekly trend baseline rolls up values for a particular hour of the day, for a particular day of the week, for the last 90 days. For more information about baselines, see Detect Anomalies Using Dynamic Baselines. You can define a threshold for a health rule based on a single metric value or on a mathematical expression built from multiple metric values. The following are typical conditions: IF the value of the Average Response Time is greater than the default baseline by 3 X the Baseline Standard Deviation . . . IF the count of the Errors Per Minute is greater than 1000 . . . IF the number of MB of Free Memory is less than 2 X the Default Baseline . . .
IF the value of Errors per Minute/Calls per Minute over the last 15 days > 0.2 . . . The last example combines two metrics in a single condition. You can use the expression builder in the health rules wizard to create conditions based on a complex expression comprising multiple interdependent metrics Often a condition consists of multiple statements that evaluate multiple metrics. A health rule is violated either when one of its condition evaluates to true or when all of its conditions evaluate to true, depending on how it is configured. You can correlate multiple metrics to focus the health rules for your environment. For example, a health rule that measures response time (average response time greater than some baseline value) makes more business sense if it is correlated with the application load (for example, 50 concurrent users or 10,000 calls per minute) on the system. You may not want to use the response time condition alone to trigger a policy that initiates a remedial action if the load is low, even if the response time threshold is reached. To configure this correlation, the first part of the condition would evaluate the actual performance measurement and the second part would ensure that the health rule is violated only when there is sufficient load. <img class="confluence-embedded-image" width="800" src="/download/attachments/24085922/conditiondefine.png?version=1&modificationDate=1402681454000&api=v2" data-image-src="/download/attachments/24085922/conditiondefine.png?version=1&modificationDate=1402681454000&api=v2" data-unresolved-comment-count="0" data-linked-resource-id="24249424" data-linked-resource-version="1" data-linked-resource-type="attachment" data-linked-resource-default-alias="conditiondefine.png" data-base-url="https://docs.appdynamics.com" data-linked-resource-content-type="image/png" data-linked-resource-container-id="24085922" data-linked-resource-container-version="32"> <h3 id="HealthRules-CriticalandWarningConditions">Critical and Warning Conditions</h3> Conditions are classified as either critical or warning conditions. Critical conditions are evaluated before warning conditions. If you have defined a critical condition and a warning condition in the same health rule, the warning condition is evaluated only if the critical condition is not true. The configuration procedures for critical and warning conditions are identical, but you configure these two types of conditions in separate panels. You can copy a critical condition configuration to a warning configuration and vice-versa and then adjust the metrics in the copy to differentiate them. For example, in the Critical Condition panel you can create a critical condition based on the rule: IF the Average Response Time is greater than 1000 Then from the Warning Condition panel, copy that condition and edit it to be: IF the Average Response Time is greater than 500 As performance changes, a health rule violation can be upgraded from warning to critical if performance deteriorates to the higher threshold or downgraded from critical to warning if performance improves to the warning threshold. <h1 id="HealthRules-DefaultHealthRules">Default Health Rules</h1> Out of the box,

AppDynamics provides a default set of health rules: <table class="confluenceTable"><tbody><tr><th class="confluenceTh"> Health Rule Name </th><th class="confluenceTh"> Health Rule Type </th></tr><tr><td class="confluenceTd"> Business Transaction response time is much higher than normal </td><td class="confluenceTd"> Business Transaction Performance </td></tr><tr><td class="confluenceTd"> Business Transaction error rate is much higher than normal </td><td class="confluenceTd"> Business Transaction Performance </td></tr><tr><td class="confluenceTd"> CPU utilization is too high </td><td class="confluenceTd"> Node Health – Hardware, JVM, CLR Performance </td></tr><tr><td class="confluenceTd"> Memory utilization is too high </td><td class="confluenceTd"> Node Health – Hardware, JVM, CLR Performance </td></tr><tr><td class="confluenceTd"> JVM Memory Heap is too high </td><td class="confluenceTd"> Node Health – Hardware, JVM, CLR Performance </td></tr><tr><td class="confluenceTd"> JVM Garbage Collection Time is too high </td><td class="confluenceTd"> Node Health – Hardware, JVM, CLR Performance </td></tr><tr><td class="confluenceTd"> CLR Garbage Collection Time is too high </td><td class="confluenceTd"> Node Health – Hardware, JVM, CLR Performance </td></tr></tbody></table> If any of these predefined health rules are violated, the affected items are marked in the UI as yellow-orange, if it is a Warning violation and red, if it is a Critical violation. In many cases the default health rules may be the only health rules that you need. If the conditions are not configured appropriately for your application, you can edit them. You can also disable the default health rules. <h1 id="HealthRules-PreparingtoSetUpHealthRules">Preparing to Set Up Health Rules</h1> AppDynamics recommends the following process to set up health rules for your application: Identify the key metrics on the key entities that you need to monitor for your application. See Suggested Metrics for Additional Health Rules for some common choices.
These metrics should be representative of the overall health of your application.
Click Alert & Respond -> Health Rules to examine the default health rules that are provided by AppDynamics.
Compare your list of metrics with the metrics configured in these rules.
If the default health rules cover all the key metrics you need, determine whether the pre-configured conditions are applicable to your environment. If necessary, modify the conditions for your needs.
You can also view the list of affected entities for each of the default health rules and modify the entities.
If the health rules do not cover all your needs or if you need very finely-applied health rules to cover specific use cases, create new health rules.
First, identify the type of the health rule that you want to create. See Health Rule Types.
Then decide which entities should be affected by the new rule. See Entities Affected by a Health Rule.
Then define the conditions to monitor.
Create schedules for health rules, if needed.
In some situations a health rules is more useful if it runs at a particular time. See <a

href="#HealthRules-HealthRuleSchedules">Health Rule Schedules.
If desired, configure policies and actions that should come into play when health rules are violated. See Policies and Actions.
<h1 id="HealthRules-HealthRuleManagement">Health Rule Management</h1> To view current health rules in an application, including the default health rules, and to access the health rule wizard, click Alert & Respond -> Health Rules. Current health rules are listed in the left panel. If you click one of these rules, a list appears in the right panel showing what entities this selected health rule affects and what the status of the latest evaluation is. You can also select the Evaluation Events tab to see a detailed list of evaluation events. In the left panel you can directly delete or duplicate a health rule. From here you can also access the health rule wizard to add a new rule or edit an existing one. To delete an existing health rule: Select the health rule in the left panel.
Click the minus icon. The health rule is removed. To duplicate an existing health rule: Select the health rule in the left panel.
Click the copy icon. The health rule is duplicated under the name you assign. To edit an existing health rule: Select the health rule in the left panel.
Click the pencil icon. The health rule wizard appears, with the rule's current values configured. You can modify these values in the wizard. To create a new health rule: Click the plus icon.
The health rule wizard appears with some default values configured. You define the health rule in the wizard. See Configure Health Rules for details on using the health rule wizard. <h1 id="HealthRules-add-rulesSuggestedMetricsforAdditionalHealthRules">Suggested Metrics for Additional Health Rules</h1> The following are suggested metrics you might want to monitor with health rules, based on what many customers have found useful. <h2 class="p1" id="HealthRules-AdditionalMetricsforBusinessTransactions">Additional Metrics for Business Transactions</h2>
Calls Per Minute
Slow Call Rate
Stalls <h2 class="p1" id="HealthRules-AdditionalMetricsforTiers">Additional Metrics for Tiers</h2>
Average Response Time
Calls Per Minute
Error Rate
Slow Call Rate
Stalls
<h2 class="p1" id="HealthRules-AdditionalMetricsforNodes">Additional Metrics for Nodes</h2> Some of these metrics might apply only to certain types of nodes, such as those running JVMs. Availability
CPU Utilization
Memory & Heap Utilization
GC Time Spent
Thread Pool – Utilization Rate
Thread Pool - Average Wait Time
Thread Pool – Queue Size
Connection Pool – Utilization Rate
Connection Pool – Wait Time to Acquire a Connection
Thread Contention
<h2 class="p1" id="HealthRules-AdditionalMetricsforBackends">Additional Metrics for Backends</h2>
Average Response Time
Calls Per Minute
Error Rate
<h1 id="HealthRules-LearnMore">Learn More</h1>
Notification Actions
<a href="/display/PRO39

/Configure+Health+Rules">Configure Health Rules
<a href="/display/PRO39
/Configure+Baselines">Configure Baselines
<a href="/display/PRO39/Events"
>Events
Policies
<a href="/display
/PRO39/Actions">Actions   </body> </html>