July 2023

SYSTEM FOR HEALTH EVALUATION INVOLVING DYNAMICALLY GROUPED ENTITIES

Mahesh Kumar Doraiswamy

Abhishek Sharma

Omanand Jha Vatsa

SYSTEM FOR HEALTH EVALUATION INVOLVING DYNAMICALLY GROUPED ENTITIES

AUTHORS:
Mahesh Kumar Doraiswamy
Abhishek Sharma
Omanand Jha Vatsa

ABSTRACT

In observability and monitoring solutions, metrics can be collected for entities and health status for the entities can be evaluated based on certain rules. Health evaluation and reporting typically occurs on the same entities on which such metrics are collected/ingested. However, there are instances in which this approach may not suffice, such as instances in which metrics may need to be derived for parent entities based on child entities. Presented herein are techniques through which virtual entities can be dynamically introduced in order to derive metrics, health status, etc. for a parent entity based on child entity metrics.

DETAILED DESCRIPTION

Observability and monitoring solutions typically involve collecting metrics on entities (e.g., services, service instances, etc.) and evaluating health status for the entities based on certain rules. The health evaluation and reporting typically occurs on the same entities on which the metrics are collected/ingested.

However, there are use cases where this approach may not suffice, such as:

- Evaluating the health of a service based on subset of service instances (e.g., service instances of version 2.0). For example, if the rate of errors per minute (EPM) for all service instances with version 2.0 is higher than specific threshold, declare the service as unhealthy. (apm:service_instance → apm:service); or
- If Kubernetes® pods allocatable across all nodes with a role as backend is less than a threshold, declare the node cluster as unhealthy. (node → node_cluster). Kubernetes® is a registered trademark of the Linux Foundation.

Thus, in some instances, as illustrated for the example use cases above, it may be advantageous to derive metrics for a parent entity based on one or more child entities of the parent. Such derivation of metrics should adapt to the dynamic changes in child entities (e.g., new entities created, entities getting deleted, attributes getting changed etc.) and understand how to derive the parent entity metrics (e.g., average, rate, sum etc.).

Presented herein are techniques through which virtual entities can be dynamically introduced in order to derive metrics, health status, etc. for a parent entity based on child entity metrics.

Consider an example scenario as illustrated via Figure 1, below, through which techniques of this proposal may be illustrated.

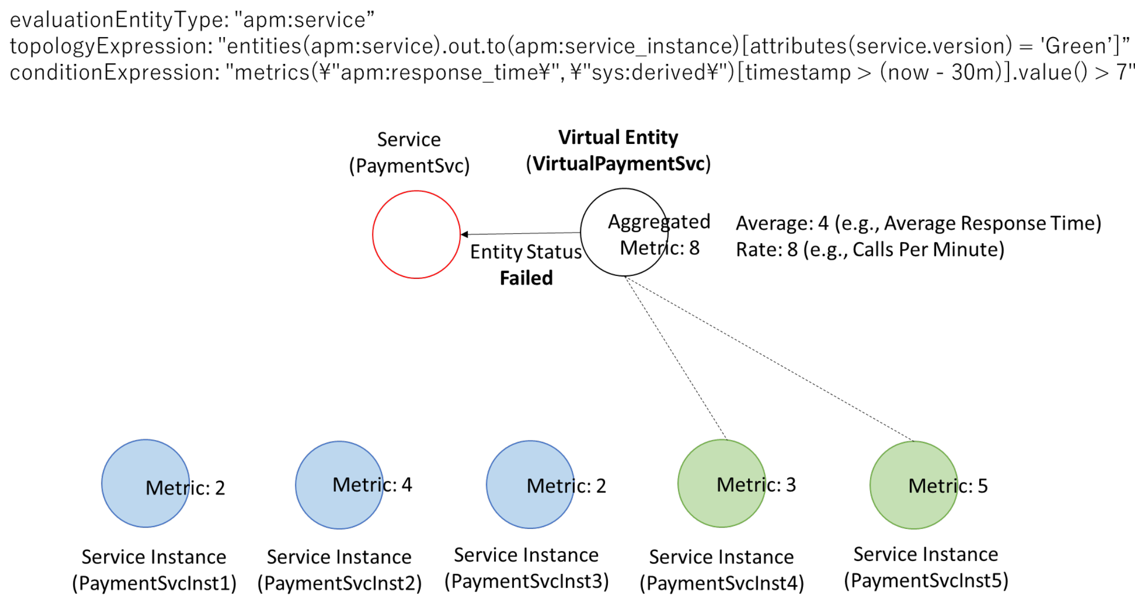

Figure 1: Example Parent/Child Scenario

For the example scenario as illustrated in Figure 1, the intent is to understand the health of a payment service when new service instance versions (as shown in Green) are introduced. For this scenario, a rule may be provided such that if the aggregated metric "errors_min" (error per minute) is greater than a defined threshold for all service instances of specific version (Green) then payment service (Parent) is considered to be unhealthy. The aggregation logic can depend on the type of metric (Average, Sum, Rate, etc.) and can be inferred from the metric definitions. In accordance with techniques of this proposal, the solution can adapt to changes in a number of child entities.

Figure 2, below, illustrates a flow diagram involving high-level steps through which the techniques of this proposal may be implemented.

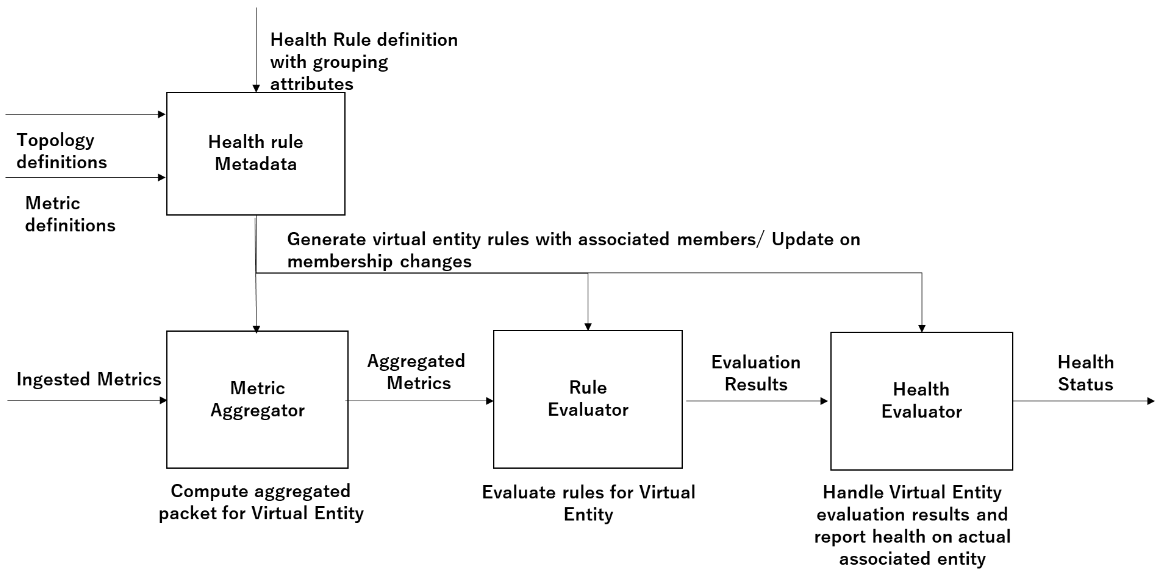

Figure 2: Example process flow

As illustrated in Figure 2, techniques of this proposal may leverage topology definitions (hierarchy of observed entities) and metric definitions (type of metrics). As shown above, on receiving a health rule definition with grouping attributes, a health rule metadata module can generate a virtual entity, associated child members, and rules. Next, a metric aggregator module/logic computes aggregated metrics for the virtual entity leveraging the metric definitions. Thereafter, a rule evaluator module evaluates the health of the virtual entity, and a health evaluator module handles virtual entity evaluation results and reports the health status of the actual associated parent entity.

Accordingly, techniques of this proposal may provide for dynamically introducing a virtual entity in order to derive metrics, health status, etc. for a parent entity based on child entity metrics.