January 08, 2018

Track Asynchronous Graph Traversal Completion using Time moving window Aggregators

Syed Abu Saleh
Hewlett Packard Enterprise

Arun Kumar Sivaraj
Hewlett Packard Enterprise

Track Asynchronous Graph Traversal Completion using Time moving window Aggregators

Abstract

In today's virtualized cloud infrastructure and advanced technology era, there are resources that are represented as an interconnected graph with root node and child nodes. These resources are growing exponentially on day-to-day basis and adding to the complexity of traversing and processing all the nodes in the graph. Major challenge involved is to traverse and process each child node asynchronously and mark the completion of all the nodes in the graph. Disclosed is an approach of tracking asynchronous traversal completion of a graph with indefinite nodes, using a time series moving window aggregator. This idea will address the graph traversal starting with a root node and process all its child nodes spanned by indefinite depth and breadth.

This disclosure is about an approach of tracking asynchronous traversal completion of a graph with indefinite nodes, using a time series moving window aggregator.

This idea will address the cases where in a graph has to be traversed starting with a root node and process all its child nodes spanned by indefinite depth and breadth.

In today's virtualized cloud infrastructure and advanced technology era, there are resources that are represented as an interconnected graph with root node and child nodes. These resources are growing exponentially on day-to-day basis and adding to the complexity of the graph. Major challenge involved is to traverse and process each child node asynchronously and mark the completion of processing of all the nodes in the graph. There are no definitive approaches available readily that try to address the traversal and processing completion of indefinite nodes graph asynchronously.

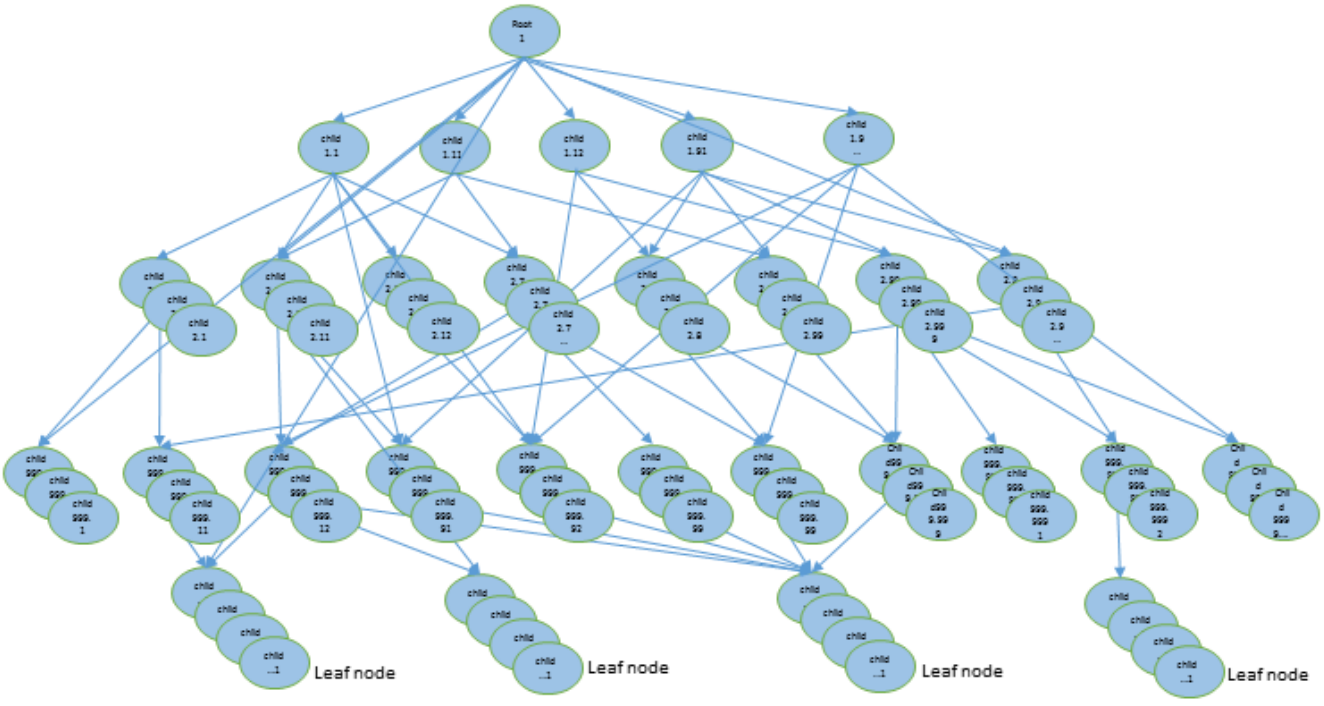

Fig #1: Indefinite Inter-connected Graph

The disclosed idea addresses this problem by using a specific time series moving window aggregator. The time window aggregator is instantiated with a user-defined configurable timeout value (For e.g. 120 seconds). Traversal process begins breadth wise from the root node asynchronously and starts traversing and processing child nodes.

After each child node traversal, the time window aggregator is invoked, which in turn resets the time-moving window aggregator. Further traversal in depth and processing will occur concurrently for each child node and will continue until leaf node is processed. Leaf node traversal makes a final invocation to time window aggregator. Similarly, all the other leaf nodes make a final invocation once they are processed.

Time window aggregator is no more invoked once all the leaf nodes are processed. Aggregator waits for the user-defined configured interval to time-out and marks the completion of the graph nodes asynchronous traversal process.

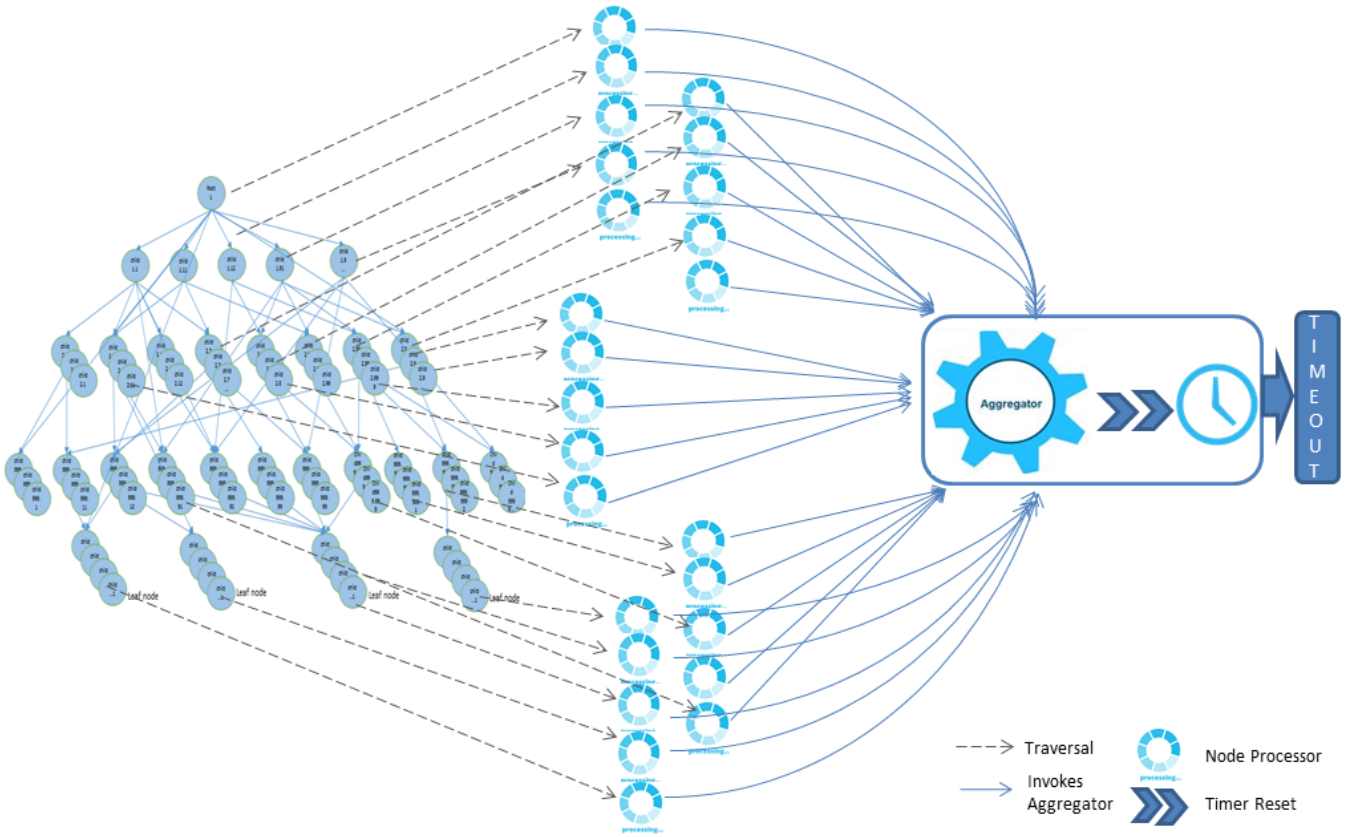

Fig #2: Diagrammatic Depiction of the Solution

Disclosed by Syed Abu Saleh, Arun Kumar Sivaraj, Hewlett Packard Enterprise