January 2020

RECONSTRUCTION OF ENTITY LIFETIME SUMMARIES FROM AUTOSUPPORT (ASUP) CONFIGURATION AND LOGS

Geetha Srikantan

RECONSTRUCTION OF ENTITY LIFETIME SUMMARIES FROM AUTOSUPPORT (ASUP) CONFIGURATION AND LOGS

AUTHORS:

Geetha Srikantan

ABSTRACT

Presented herein are techniques for rapid reconstruction of entity lifetime/ lifecycle summaries (lifetime of entities) from voluminous AutoSupport (ASUP) bundles.

DETAILED DESCRIPTION

Hyperconverged Infrastructure, such as Hyperflex, have several hardware and software components, all of which operate together to provide data platform services in virtualized / hypervisor environments. The components include, for example:

- Network elements, such as Fabric Interconnect, Switches, Routers, ACI, etc.;
- Servers, encompassing storage, compute, memory and network hardware;
- Virtual infrastructure, including hypervisor, virtual machines (VMs) and other virtualized devices; and
- Software components, including the Hyperflex software stack resident on the controller VM or the host.

Figure 1, below, illustrates a single node of a Hyperflex cluster, while Figure 2, below, illustrates a multi-node Hyperflex cluster.

Single Physical Node of Hyperflex Cluster and associated Controller VM
Vsphere VCenter
Management Tools – HxConnect User Interface; ReST Interface, CLI – Command Line Interface
vSphere VCenter
Eth0 – management interface
Eth1 – data interface
Eth2 – Replication interface
Controller VM
Management Servies
Control Services
Storage/filesystem Services
Data Protection Services
ESX host
Hypervisor services
VIB-based services
Datastores
Virtual Machine artifacts
HxConnect User Interface
ReST interface
CLI Command line interface

Figure 1

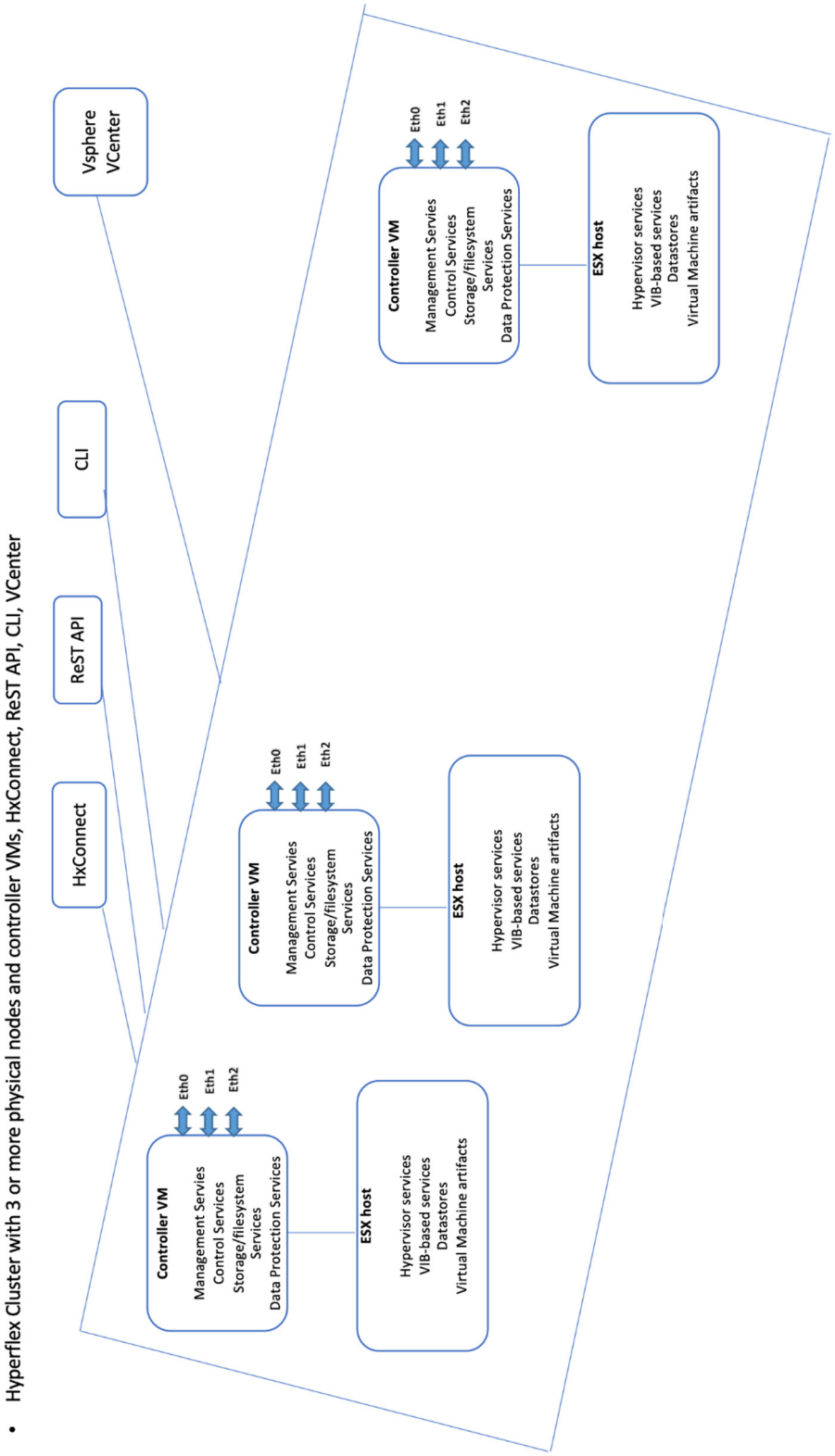
• Hyperflex Cluster with 3 or more physical nodes and controller VMs, HxConnect, ReST API, CLI, VCenter
HxConnect
ReST API
CLI
Vsphere VCenter
Controller VM
Management Servies
Control Services
Storage/filesystem Services
Data Protection Services
Eth0
Eth1
Eth2
ESX host
Hypervisor services
VIB-based services
Datastores
Virtual Machine artifacts

Figure 2

While triaging problems seen at customer deployments, an ASUP bundle is collected from the hosts and controller VMs and made available for debugging. Depending on the size of each deployment, the cluster may have 2, 3 or up to 32 or 64 physical nodes and an equal number of controller VMs. Each software component within each physical host and controller VM typically has some configuration data and some logs. Both of these need to be examined for any software components that may have been involved in a malfunction.

As can be seen by the numbers mentioned above, there is potential for huge proliferation of information in configuration and logs from all the software components on all the controller VMs and hosts of a single cluster. Furthermore, in Disaster Recovery deployments, there are 2 clusters participating in the activity, which could imply twice the number of configuration objects and logs that need to be examined and understood, to triage a problem. Figure 3, below, illustrates a pair of Hyperflex clusters that are paired for replication, with mapped datastores and protected VMs.

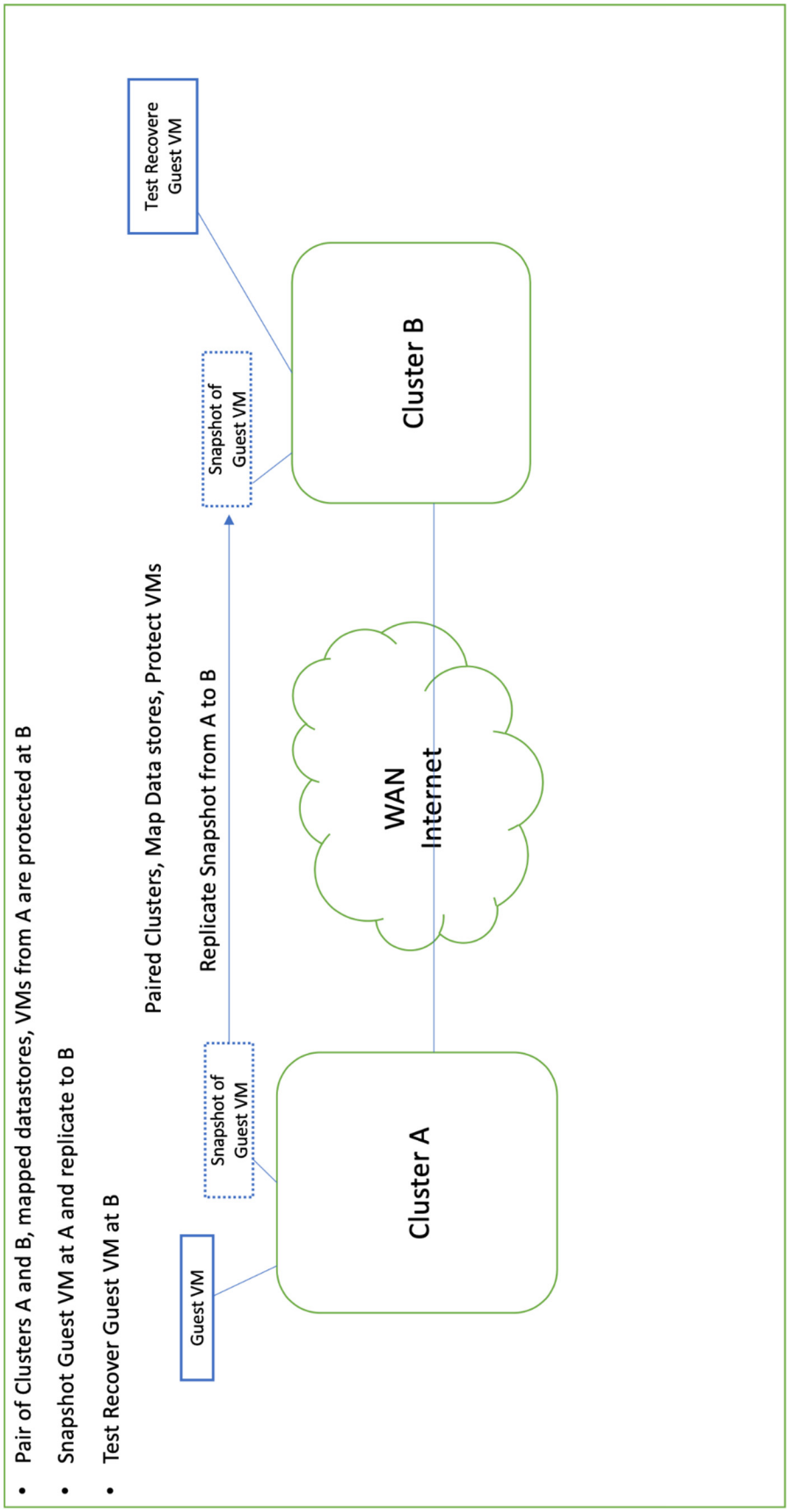
• Pair of Clusters A and B, mapped datastores, VMs from A are protected at B
• Snapshot Guest VM at A and replicate to B
• Test Recover Guest VM at B
Paired Clusters, Map Data stores, Protect VMs
Replicate Snapshot from A to B
Guest VM
Snapshot of Guest VM
Cluster A
WAN Internet
Cluster B
Snapshot of Guest VM
Test Recovere Guest VM

Figure 3

The time available to triage and rectify a malfunction is often short and would be greatly assisted by a meaningful lifecycle summary for each entity of interest. In the Disaster Recovery (DR) case, users are often interested in the lifecycle of protected VMs, including where does a VM reside (cluster and node), when was the VM protected, has replication been successful or broken over the lifecycle of the VM, were test-recoveries attempted and did they succeed, is the VM part of a group, etc. The users are also interested in general health of the clusters, resources (e.g., memory, compute, network, etc.) and whether resources were available and consumed as expected at the time or leading up to the time when a malfunction is observed.

The techniques presented herein build the lifetime of entities, such as protected VMs, cluster resources and other entities of interest, to enable rapid triage and resolution. The techniques presented herein involve examining the configuration and logs from a multitude of components to build the lifetime/story for entities of interest.

Currently the details available in ASUP bundles include:

- ASUP Details
  - **At each of Clusters, A and B**
    - CLI listing of protected VMs – identified by hypervisor uuid
    - Logs of several services from each controller VM & host, VCenter
  - Screenshots from HxConnect, if applicable
  - Symptoms of malfunction in plain text
  - Approximate time when malfunction was observed and/or reported
    - Time of reporting may be several days or weeks after the malfunction began
- Other User inputs – not always reported
  - Specific VMs noted with issues
    - Unavailable
    - RPO (Recovery Point Objective) Exceeded
    - Unable to
      - Recover
      - TestRecover
      - Failback
      - Migrate

There are no specific tools available for reconstructing the entity lifetimes from ASUP at this time. Instead, most developers use a combination of Unix tools (e.g., grep, awk, sed and such) to go through the logs. This approach is acceptable for simple deployments that have a small number of nodes per cluster and small number of protected VMs per DR-cluster-pair. However, this approach is not scalable for large deployments. In particular, recent releases have scaled to supporting over 1000 VMs for replication between VMs and these scalability limits would be increased in newer releases. Tracking the lifetimes of 1000s of entities from configuration and logs on 2 or more clusters can become tedious and error-prone.

The techniques presented herein outline a set of tools that can be put together to develop the story or lifetime of each entity of interest. Figure 4, below, illustrates a few example lifetimes that may be reconstructed in accordance with the techniques presented herein.


Time
Lifetime of VM1
Cluster A
Test Recover At B
Recover At B
Cluster B
Failback to A
Cluster A
Disaster at Cluster A
Lifetime of VM<n>
Cluster A
Maintenance
Disaster at Cluster A
Lifetime of VM1000
Cluster B
Snapshot failed at B
Replication failed
Snapshot succeeded at B
Recovrr at A
Cluster A
Disaster at Cluster A

Figure 4

In Disaster Recovery (DR) setups, there are typically at least 2 clusters and there may be a need to examine the configuration and inventory as well as logs on both clusters. For example, in the case of cluster resources (e.g., datastores), there may be a need to determine time of creation, size, usage trends, performance, space available, number of VMs residing on the dastastore, outages/errors, if any. In case of DR network resource, there may be a need to determine time of creation of the network, number of converged nodes, any nodes added later to the network, intra-cluster communication over the DR network, over time, and so on. In the case of cluster pair, there may be a need to determine time when paired, number of converged nodes on each cluster, datastores mapped in the pairing, changes in the pairing over time, replication pair network status over time, bandwidth usage over time and so on. In the case of Protected Virtual Machines, there may be a need to identify the time of protection, number of disks, configuration of VM, protection interval, success/failures in replication, whether the VM was ever test recovered, migrated or operated on; details of failures in snapshot, replication, recovery - including the time and frequency of failures, reason why and so on. In the case of snapshot and replication of virtual machines, there may be a need to gather details such as times when these operations worked fine and times when they did not. For other entities of interest, there may be a need to extract relevant information from configuration and logs.

The techniques presented herein provide a mechanism to summarize and present the above in an easy to consume representation, either visually or in textual format. Figure 5, below, outlines the steps in processing ASUP configuration and logs to reconstruct lifetimes of entities in accordance with the techniques presented herein.

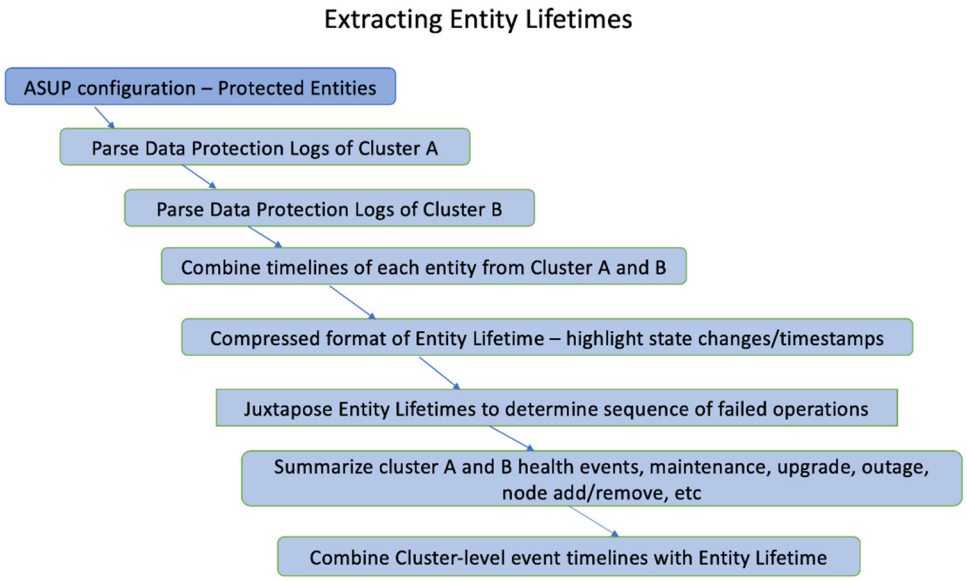

Figure 5

It is noted that DR setups tend to be long-lived (e.g., the clusters may have been paired for over a year), with and replications proceeding at, for example, 5-minute, 15-minute and hourly-intervals. Administrators may attempt TestRecovery on a regular basis, to validate availability of the workload VMs. Administrators may also use the Migrate or Recover capability for protected VMs at Cluster B, when, for example, cluster A has a maintenance window, and then Failback those VMs from B to A. This implies that the lifetime of each VM could encompass both clusters at different periods of time.

The volume of data that needs to be interpreted to reconstruct such lifetimes is quite large and grows proportionally to the size of the clusters and the number of protected VMs. As part of the design of the DR functionality, there are states in which a VM can be, and states it could transition to. This a-priori knowledge is to be leveraged in interpreting the ASUP data. Often, the ASUP bundle contains only a partial "picture" of both clusters, for long-lived setups, where the logs may have rotated. Relying on the state transitions per entity is very helpful in filling some of the gaps in information, in such cases.

Logs represent a semi-structured language developed by the builders of the system. There are now several Machine Learning/Natural Language Processing (ML/NL) techniques to extract meaning from the text. These techniques can be applied to the logs, to develop such a story or lifetime, in a generalized fashion. For instance, Long Short Term

Memory (LSTM) and newer Attention-based techniques can be applied to build a coherent story for items of interest, thereby removing dependence on developer inputs and possibly discovering newer patterns of usage and failure modes.

The techniques presented herein provide a number of benefits, including:

1. Automation of information extraction from ASUP configuration and logs to assist in triaging of issues;
    a) ASUP logs are voluminous for DR setups, particularly when the clusters have 4 or more nodes and over 500 protected VMs. Manually reviewing ASUP bundles is tedious and error-prone - automating these tasks would have several benefits.
2. Extracted Entity Lifetimes assist in honing in on the times when a state change happened with each protection entity and failures before/after the state change (as these are most useful points of information).
3. Reduced time to resolve customer issues;
4. Make it possible for Support staff who may not be familiar with the system to hone in on a few possible Root Causes of an issue, without having to manipulate any configuration or data on the clusters.