IWODSS 2010
Montreal, 17-18 May, 2010

External Reference Entity Method For Link Failure Detection in Highly Available Systems

Vilho Raatikka and Antoni Wolski

IBM Finland, P.O. Box 265,, 00101 Helsinki, Finland
{first_name.last_name}@fi.ibm.com

Abstract. One of the biggest challenges in high-availability (HA) system is to prevent a situation called split brain, whereby there are two active nodes in the system, which both allow for updating the data. Split brain situation may occur when a node failure cannot be distinguished from a communication link failure. External Reference Entity Method can be used to determine whether a communication link between the active and standby components of a HA system has failed or not. External Reference Entity (ERE) is a passive network component having the sole responsibility of responding to simple connectivity check requests like those of the ping protocol. With the ERE Method implemented in the active and standby nodes of an HA system, no third (active) processing node is needed to detect active-standby link failures.

1 Introduction

In highly available systems, the most common configuration is that of active-standby, called also hot spare, or hot standby [1] (HSB), or a 2N model [5]. In the HSB configuration, there are two peer computer nodes: active and standby. The active node is the one where the software service unit is running in the active HA state. At the standby node, the service unit is running in the standby HA state whereby it is being continuously updated with the latest service state information replicated from the active unit. Should the active unit or the active node fail, the failover is performed whereby the standby unit becomes the active one.

A distinction should be made between a unit, or node, failure and the HSB communication link failure between the two. The HSB link might be implemented in a way making it a single point of failure. If the HSB link failure it not recognized (and interpreted as a node or unit failure), the result might be a *split-brain situation* whereby there are two active nodes, in the system. That might endanger data integrity, especially if the service offered is a database service.

Traditionally, an HSB communication link failure in a two-node active-standby configuration is detected by using failure detection software residing in a third node. If there is an occurrence of a communication link loss at either of the active/standby nodes, the third node may use its own heartbeat connections to the active/standby nodes to detect the failure. If both nodes respond to heartbeat messages affirmatively, then the possibility of either active or standby node failure is excluded, and the conclusion is reached that a link failure between the active and standby node has

occurred. A third node approach can be also used as a part of a distributed HA framework functionality.

A method requiring a third node for link failure detection is costly because of the added cost of the third node. Another alternative method [2] of detecting link failures assumes using dual (or multiple) physical links between the active and standby nodes. In that solution, correlated failures (within a narrow time window) of both (or all) links is interpreted as a node failure. Here, an additional cost is born, in the form of parallel physical links.

The motivation behind this work was to find a cost-efficient approach to link failure detection, especially in small systems where no HA framework is used. The proposed ERE Method does not require any additional system component. It is assumed that a device suitable for ERE function already exists in the system. Such a device can be, for example, a network switch that is capable of responding to ping requests (most are). Another possibility is to use any other computer node available in the system, for that purpose. Despite the used device, ERE doesn't require any additional software instance to be installed in the device.

The ERE Method allows users to deploy a fully functional HA software in a minimal configuration of just two nodes: active and standby. The method has been implemented and tested in real production environments.

2 Description of the ERE Method

2.1 Architecture

The configuration of a system utilizing the ERE method is shown in Fig.1.

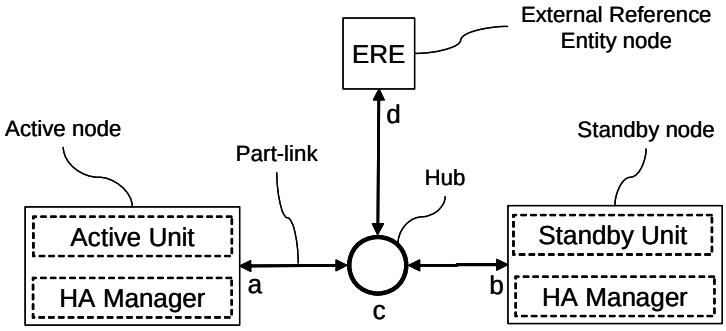

Fig. 1. Configuration of an HA system utilizing the ERE Method.

Solid-line boxes represent computer nodes or other network-addressable components. The circle element represents a physical connectivity component like a network hub (repeater hub) or a network switch. Each of the components has a single physical network access point (a, b, c, d). The active and standby nodes are

interconnected by way of two *part-links* (a←→c and c←→b) that constitute a logical *HSB link* (a←→b) having the purpose of performing real-time service state replication between the active and standby service units. The ERE node is connected to both the active and standby nodes by way of existing network components like repeater hubs or network switches. The configuration in Fig.1 is a generalization in the sense that the ERE node can be collapsed with a switch, making the link d←→c nonexistent. ERE can be implemented in such a way that it becomes a single point of failure. That, however, can be avoided by adding redundancy in a form of two ERE devices.

HA Manager is a software component responsible for both failure detection, and failover mediation. The algorithms of both the failure detection and failover mediation are presented below.

2.2 Algorithms

HA Manager failure detection algorithm

Once the active and standby units lose their connections to the peer units, the corresponding HA Manager instances are made aware of that fact. Subsequently, the network failure resolution processing is enacted in both nodes. The algorithm is illustrated in Fig. 2. It is started by the HA Manager instances in both nodes by trying to reach ERE (a→d, b→d) with ping requests.

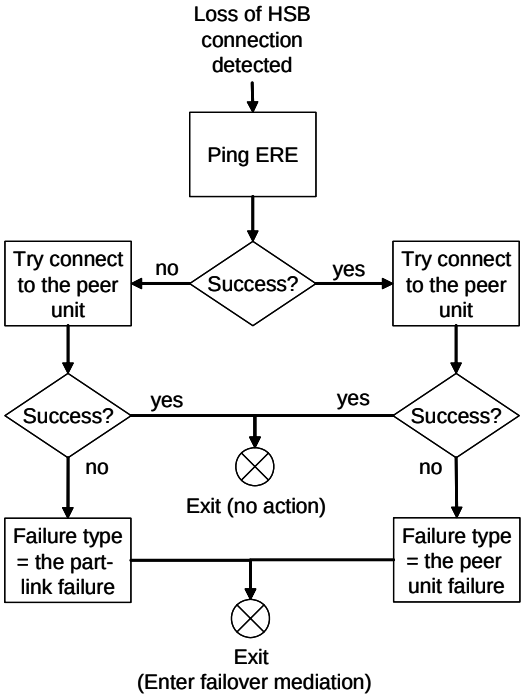

Fig. 2. Flowchart of the link failure detection algorithm.

If ERE is responsive, the peer unit is considered either failed or suffering from other failure in the node. If ERE is not responsive, then it is concluded that the part-link is broken. A broken part-link means that neither the HSB link can be established nor the service can be offered over the network, from that node. However, at the peer node, the part-link may be operational deeming it possible to offer at least the service over the network. If both part-links are concluded to be broken, no service can be offered.

After the ping attempts, HA Manager instances reassure that their perception about the failure is correct by trying to re-establish connection with the other node (a➔b, b➔a). If the connection can now be established, then–whatever the cause of the failure was–it is assumed to has been intermittent, and no further actions are taken. Otherwise, the failover mediation algorithm is started.

HA Manager failover mediation algorithm

After the failure type has been resolved, the failover mediation algorithm is enacted. If the node suffers from the part-link failure, that indicates that that unit cannot be reached from the network and thus cannot be used. Consequently, the unit's HA state is changed to standby. That is done to prevent the situation of two active units that are isolated from each other. The node for which the ERE is responsive (the part-link is functional) continues as the active node. If that node is a standby node, a failover is initiated by HA Manager, if necessary. The failover mediation algorithm is shown in Fig. 3.

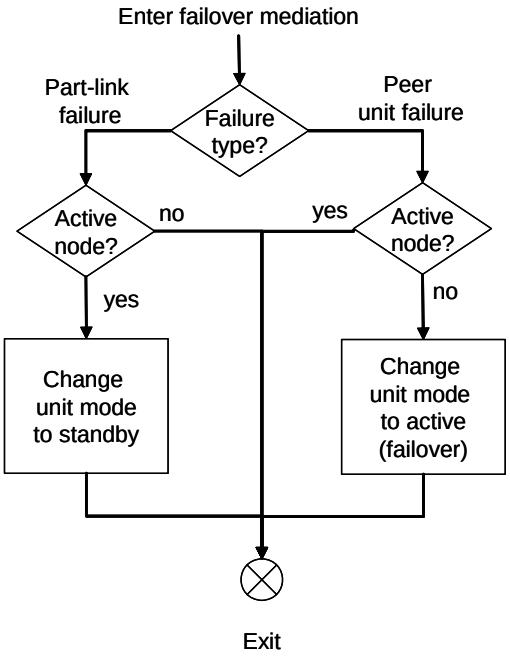

Fig. 3. Flowchart of the failover mediation algorithm.

2.3 Discussion

In the configuration discussed, only the hub is a single point of failure. A hub failure invalidates the whole HSB link and the system cannot operate. If a hub fails in a configuration shown in Fig. 1, not only the HSB link becomes broken but the failure detection algorithm (Fig. 2) executed in both nodes declares both part-links failed and, consequently, the failover mediation algorithm fails to select an active unit.

A remedy here would be to introduce redundant hubs both connected to an ERE device, as shown in Figure 4. Here, if one hub fails, still one of the part-links can be declared operational and the corresponding unit selected as an active one.

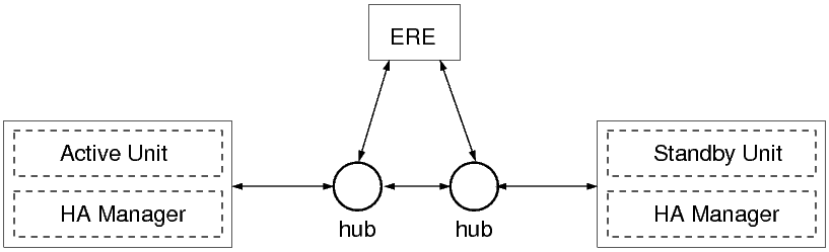

Fig. 4. Avoid single point of failure by using single physical part links, and redundant hubs.

Another possibility is to add a redundant hub and physical part-links, as shown in Figure 5. Here, if one hub fails, the other one can operate. Additionally, the configuration is more failure resilient than any of the previous ones because of a redundant HSB link. A single part-link failure will not even cause a failover because at least one end-to-end HSB link will be operational.

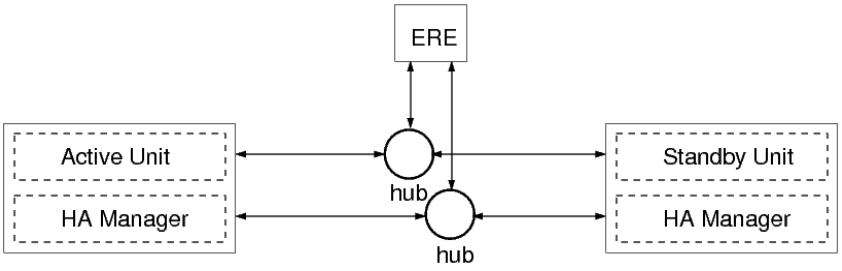

Fig. 5. Avoid single point of failure by using redundant physical part links, and redundant ERE devices

A single ERE node is not a single point of failure because ERE is not involved in the execution of the service. If the ERE node fails, that does not directly affect the execution of the service. The assumption is that the ERE node is restarted or replaced as fast as possible. If there happens a concurrent failure, i.e. the active node fails during the time ERE is inoperable, the HA Manager will not be able to execute the failover until ERE is operational again. Thus, the system, as pictured above, can

handle only single failures (be it in ERE, in the part-links or in the service executing nodes). If higher failure resiliency is needed, more component redundancy can be applied.

2.4 Implementation and experience

The ERE Method has been implemented in the IBM solidDB HA database product [3, 4, 6], starting from version 6.1 released in June 2008. It has ben received well in the field: most new solidDB HA applications utilize the ERE method although it is not enabled by default.

3 Conclusions

The External Reference Entity Method can be used to reduce the number of computer components if a high availability system utilizing the hot-standby configuration. By using the method, hot-standby link failures can be detected without the need for a third active computing node. With the ERE method, only a passive network component is needed to respond to ping-like inquires. The corresponding failure detection algorithm is implemented in the HA Manager that is a software component deployed at both the active and standby nodes. HA Manager is also responsible for executing failovers.

References

[1] Gray, J. and Reuter, A.: Transaction Processing Systems, Concepts and Techniques. Morgan Kaufmann Publishers, 1992, ISBN 1-55860-190-2.

[2] Carlson, W.C.: Failure Detection in a Symmetric System. Prior Art Database (ip.com), IP.COM number IPCOM000114622D, March 29, 2005.

[3] Wolski, A. and Raatikka, V.: Performance Measurement and Tuning of Hot-Standby Databases. Third International Service Availability Symposium (ISAS 2006), May 15-16, 2006, Helsinki, Finland.

[4] IBM solidDB High Availability User Guide, Version 6.5, IBM Corporation, November 2009, available at http://www-01.ibm.com/software/data/soliddb/.

[5] SA Forum Application Interface Specification, Availability Management Framework (SAI-AIS-AMF-B.04.01), Release 6.1, February 19, 2010, available at http://www.saforum.org.

[6] Wolski, A. and Hofhauser, B.: A Self-Managing High-Availability Database: Industrial Case Study. Proc. Workshop on Self-Managing Database Systems (SMDB2005), April 8-9, 2005, Tokyo, Japan.