EPICS AND WANS: TRADEOFFS BETWEEN ISOLATION, SECURITY, ROBUSTNESS, AND TRANSPARENCY

J. Hill (LANL), K. Furukawa (KEK), S. Hunt (PSI), A. Johnson (ANL), R. Lange (BESSY), J. Sage (TJNAF), E. Williams (ORNL)

Abstract

The Experimental Physics and Industrial Control System (EPICS)[1] was originally designed for use in local area networks (LANs). Today, the system is routinely deployed into complex wide area networks (WANs) using specialized configuration options and proxy gateways. There are advantages to the current approach including robustness and control over isolation and security. However, some important features are missing including WAN transparent configuration, resource location monitoring, detection of name space collisions during installation, and wildcard queries into the resource attribute space. The paper discusses these issues in detail exploring the relative tradeoffs between different solutions and including our plans for future enhancements.

INTRODUCTION

An ideal wide area network (WAN) based control system would locate resources transparently, notify clients immediately when a server's state-of-health changes, isolate critical components of the control system from other parts of the system, transparently transfer loads on critical resources to less loaded systems, and automatically recover from hardware and software faults. The following paragraphs introduce some detailed requirements for the WAN aspects of a control system. The easier requirements are listed first transitioning into more difficult to implement requirements.

NAME RESOLUTION ESSENTIALS

In a generic control system clients need to determine the network address of the server for named resources. The following is a list of requirements on the name resolution subsystem for a WAN based and project generic control system.

a Clients will be informed of, and properly respond to, changes in the address of a resources throughout the lifespan of the client.

b Reasonable diagnostics require access to all process variables and their meta-data. Certain clients will need to resolve thousands of process variable names into network addresses during initialization. Performance and efficiency are important.

c Loss of the name resolution subsystem might disrupt the entire system. Robustness is important.

d It will be possible to integrate servers developed off-site without manually configuring a central authority (plug-and-play capabilities).

e The name resolution load for a large system will grow larger than what any single host can handle and will be distributed among multiple hosts as required. It is also desirable that this load balancing be transparent.

f Clients of the system will not need specialized configuration to find the location of a resource or to find the location of the name resolution subsystem.

g The name resolution subsystem will support wildcard queries into the process variable name space.

h The name resolution subsystem will detect name space collisions during installation.

i The name space will support hierarchical names, but the hierarchy shall not impose address boundary rules.

j A self repairing client side name resolution cache will reduce load on the system.

SERVER STATE-OF-HEALTH NOTIFICATION ESSENTIALS

A publish and subscribe system with notification upon process variable change of state, and potentially no server initiated message activity if the process variable does not change state, requires timely notification when the state-of-health of a server changes. Changes in the server's state-of-health might occur when the server is restarted, the process variable is moved to another server, the path through the network is temporarily down, or because of other hardware or software problems.

a Clients must be notified when a resource is unavailable so that they can enter a fail-safe state. Proper fail-safe confidence requires that clients must be continuously notified that a resource they are communicating with is operating properly, and that a path to it exists through the network.

b Clients must also be notified when a temporarily unavailable resource appears or reappears on the network so that they can immediately connect to it without loading the network with futile connection attempts.

ISOLATION ESSENTIALS

It must be possible to configure independent control system domains which are allowed to interact with each other only under strictly controlled circumstances. An isolation barrier between one control system domain and other control system domains enforces the isolation policy configured by control system integrators.

a An increasing number of clients outside of an isolation barrier will not result in more than one client equivalent load inside the isolation barrier.

b The isolation barrier will impose a security policy in addition to, and overriding, any policy implemented inside the isolation barrier.

c Other than the security policy that it enforces and communication delays, the isolation barrier will be transparent to clients outside of the barrier who must interface with process variables inside the barrier.
d An isolation barrier will not introduce a single point of failure for the control system.
e An isolation barrier will not become a bottleneck for the control system, and therefore will scale with increasing client load. Automatic load balancing capabilities are desirable.

CURRENT EPICS PRACTICE

Currently, clients of EPICS determine the network address of process variables by sending search datagram messages fully packed with process variable names to a list of server unicast[2] or broadcast addresses. The delay between search attempts P is based on the estimated client to server round trip time ε and the lowest unresolved process variable search attempt count η as follows.

$$P = 2^{\eta} \cdot \varepsilon$$

There is also a dynamic adjustment in the number of datagrams sent with each search attempt based on past success rates. After one hundred unsuccessful attempts for each process variable subsequent search attempts are abandoned.

Server state-of-health notification is communicated by a server beacon datagram sent to a list of client unicast[2] or broadcast addresses. Clients maintain a running average of the period between all server beacons received. A missing beacon from a server to which the client is connected results in the client manually verifying that virtual circuit with a loop back message. A substantial change in any server beacon results in a new server event in the client. This resets the search attempt count to no higher than six for each unresolved process variable typically resulting in a new initial search period of sixty four times ε.

EPICS BROADCASTING ISSUES

Broadcast messages are the network bandwidth efficient way to send a copy of an identical message to multiple hosts. IP broadcast addresses can be used to reach any server throughout the client's subnet and net-directed broadcasts[2] can be used to reach any server on a remote subnet as router configuration permits. Internet multicast addresses can be used to reach any server on the Internet that has registered interest in the specified multicast group id, independent of router configuration. EPICS could be easily modified to support multicasting. Broadcast based protocols are common. For example the internet protocols ARP, DHCP, the X window system's XDMCP can be configured to utilize hardware broadcasting for the purpose of locating resources. There are many preconceptions for and against hardware broadcasting and therefore the next few paragraphs are a discussion of the positive and negative aspects of the broadcast based protocols in EPICS.

The advantages of broadcast based server state-of-health beacons are efficient use of network bandwidth compared to periodic state-of-health heartbeats over a virtual circuit, and improved server communication loss event detect confidence compared to schemes using a centralized server state-of-health authority. The centralized scheme is oblivious to localized network failures and it also introduces additional links in the server loss detect logic introducing additional failure scenarios, and therefore decreased confidence.

A clear disadvantage of broadcast based server state-of-health beacons in large systems with multiple subnets is that proper configuration can become tedious. Proper client side detection of newly available servers requires that all clients see beacons from all available servers, and proper configuration ensuring this can be daunting for EPICS system managers, but this negative could be eliminated if EPICS used IP multicasting. Beacon period estimation errors induced by client load bursts, server load bursts, or aberrant network segments can result in false new server events (see below).

The advantages of broadcast based name resolution include no need to install and maintain a centralized name server, no single point of failure, and integration of autonomous offsite development without manually configuring a central name service authority. EPICS is capable of over 15k virtual channel connects per second on 2001 vintage PC hardware. This figure includes the datagram based name resolution phase, and also and the virtual circuit connect phase for each channel so in practice EPICS appears to meet requirement 1b.

The disadvantages of broadcast based name resolution include increased configuration complexity for off IP subnet clients that is especially problematic if routers are not configured to accept net-directed broadcasts[3], but this negative could be surmounted if EPICS were modified to support IP multicasting. Within long lifespan control systems the typical propensity is to end up with a modern workstation CPU, a modern LAN switch, and a legacy front end controller CPU. In this situation the name resolution efforts of the workstations tends to produce a more significant load on the front end controller CPU. Addition of a name resolution cache in the client would reduce traffic, but wouldn't resolve the primary problem: there is a tendency towards growing numbers of dispossessed process variable names in the client configurations of large systems. Exponential back off in the client's search rate tends to moderate this problem, but false new server events detected in the clients resulting from client load bursts, server load bursts, or aberrant network segments can be problematic. Of particular concern might be a self amplifying situation in large EPICS systems, where increased levels of name resolution related broadcasting activity resulted in server load bursts. The quiescent network and CPU loads resulting from this phenomenon tend to be quite low, but the peak loads have not been well characterized. Work is underway to allow these metrics to be archived over time. It is known that the software is designed to systematically

degrade when peak loads reach saturation. For example, in the server, the priority of the search message input thread is just below the beacon generating thread which is just below the threads servicing virtual circuits. This ensures that search message traffic does not disrupt regular beacons, but a subscription update load carried over the virtual circuits which saturates the CPU will disrupt the beacons and result in the clients detecting that the server is no longer responsive. While this is correct per-design behaviour it is also expected that any server in the system with an intermittently saturated CPU will produce intermittent beacons, and that could lead to false new server events in the clients.

LARGE EPICS INSTALLATIONS

There are a number of options available to large EPICS installations for managing the above issues. Several sites have used a server side plug-compatible name resolution interface to implement alternative name resolution services allowing clients to find their resources without issuing name resolution broadcasts. The name server used by the CEBAF control system at Jefferson National Laboratory is frequently used as a starting point for site specific name resolution services[3]. These solutions generally work well, but they introduce a single point of failure (violating requirement 1c) and do not communicate state-of-health information. An alternative solution with strong architectural merits employs the EPICS gateway to implement isolation barriers breaking up a large EPICS system into a loosely coupled group of lightly loaded subsystems, but this approach currently also introduces geographically limited single points of failure (violating requirement 1c). Use of EPICS gateways and alternative name services also reduce the configuration effort required for clients to communicate with off subnet servers.

POTENTIAL EPICS IMPROVEMENTS

The default EPICS system today only meets requirements 1a through 1e, 2a through 2b, and 3a through 3c. Some upgrades are required to meet additional requirements. Modifying the EPICS name resolution subsystem to support wild card queries (requirements 1g) and to immediately detect name space collisions (requirement 1h) will require a centralized name resolution authority. Some searching on the Internet indicates that the feature sets in DNS[4] and LDAP[5] are candidates for implementing requirements 1a though 1j. Some initial performance measurements for these systems indicate name resolution throughput between 10 and a 100 times slower than existing EPICS mechanisms (requirement 1b). This might not be problematic for many systems, but for EPICS which employs a particularly fine grained resource naming granularity the choice must be made carefully. Performance issues will be most likely to arise when clients connect large numbers of virtual channels and when initializing servers upload their resource name lists to the central name authority (requirement 1d). A potential solution might be to implement an alternative performance oriented client library for these systems allowing multiple name resolution requests to be sent per socket IO call, but this might be unnecessary considering that both LDAP and DNS have caching capabilities. The content synchronization extensions [6][7] recently proposed for LDAP are interesting because they may address requirement 2b. They would not however provide appropriate fail safe behaviour (requirement 2a) because proper fail safe behaviour and proper detection of potential hardware and software faults requires direct communication with the server. Redundancy and load distribution capabilities built into these systems probably do address requirements 1c and 1e.

Solutions for requirements 3d and 3e may also be coupled with our choices for the name resolution system. If the name resolution system informs the clients to switch to an alternative EPICS gateway (isolation barrier) whenever the primary gateway has failed, or is overloaded, then perhaps requirements 3d and 3e can be satisfied by running redundant gateways which operate in parallel. This approach might also be useful for implementing redundant EPICS input output controllers.

CONCLUSIONS

The paper provides a list of WAN requirements for a general purpose control systems, discusses the limitations present today related to installation of large EPICS system on WANs, and examines the components that are currently available for coping with these limitations. Our plans concerning new features including wildcard queries, resource name space collision detection during installation, easier configuration, more robust operation, and EPICS systems transparently available throughout the interconnected internet were also discussed.

REFERENCES

[1] http://epics.aps.anl.gov/
[2] “TCP/IP Illustrated, Volume 1”, W. Richard Stevens, Chapter 1.4, Addison Wesley
[3] J. Sage, “Using a Name Server to Enhance Control System Efficiency”, ICALEPCS’01, San Jose, November 2001.
[4] http://www.dns.net/dnsrd/
[5] http://www.openldap.org/
[6] http://www.openldap.org/pub/kurt/ldapsync.ppt
[7] http://www.ietf.org/internet-drafts/draft-zeilenga-ldup-sync-03.txt