

Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer: Irrelevant???]

Source: <http://coding.derkeiler.com/Archive/Cobol/comp.lang.cobol/2004-01/0660.html>

From: John S. Giltner, Jr. (giltjr_at_earthlink.net)

Date: 01/27/04

Date: Tue, 27 Jan 2004 04:34:39 GMT

Hello all. After Clark posted part of some comments to the bit.listserv.ibm-main I decided to check out comp.lang.cobol.

I also will "middle-post" and if this is a no-no here I do apologize, some of the news groups I follow allow it, in fact they encourage it on lost posts.

Joe Zitzelberger wrote:

> *I just had to drop in a few comments. Please forgive the middle-posting:*
<---- SNIP ---->

<---- END SNIP ---->

>

>>[*much snippage*]

>>

>>*Hum, I bet Pete was an application programmer and not a system
>>programmer. CICS only maintains the state of terminal that are in
>>session with it, just like any and all "distributed" server
>>applications. In the early days CICS did need to have pre-defined
>>tables entries for any terminal that was going to connect to it, but you
>>have been able to dynamically defined and delete this entries for over
>>20 years now. If distrubuted servers share the same single buffer for
>>all connections/sessions how does it keep track if information that is
>>unqiue to each connection/session.*

>

>

> *The conceit of this statement is amazing. Do you realize that all of
> the helpful applications that you, as a sysprog, use every day are
> written by applications programmers???*

I did not mean to offend anybody by guessing the Pete was an application programmer. My comment was to point out that most application programmers do not understand the inner working of CICS and that CICS

ang.cobol: Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer:

does not keep track of ALL terminals in a mainframe network. This is something that an application programmer would not normally know because they do not need to know this. I respect application programmers, they write code that I do not have the skills to write and it does make my life easier.

<-- SNIP -->

<-- END SNIP -->

>

>>>*PC networks don't work that way. The devices on the network are recognised by addressable hardware cards. There are a number of protocols (not least of which is the ubiquitous TCP/IP protocol that makes it possible for us to have this conversation) that simply transmit packets around the network. Network servers pass messages to each other until the required client is located. There is no requirement for terminal dependent buffers (DFHCOMM area or SPA) or maintenance of state (the system is stateless), and the whole thing is much more fail safe because if parts of the network are down or missing, packets get re-routed "automatically". (It was designed that way, as it was developed from a military system called ARPANET that was designed to enable the continuance of military command and control after parts of the network were destroyed by nuclear attack.)*

>>

>>*The comparison is of totally different things. One is a application server (CICS) the other is a networking protocol (TCP/IP). SNA does keep track of the status of every terminal that it owns. TCP/IP does not keep track of every IP address within it's subnet. However, a distributed server using TCP/IP as it networking protocol does keep track of each client that is connected to it by IP address. A web server has a "DFHCOMM area" for each HTTP connection. A single getting a single web page from a single client can create up to 5 HTTP connections to a Web Server, this means that for an equal number of clients, a Web server is keeping track of 5 times the connections.*

>

>

> *Here, you show that you do not understand subnets. TCP/IP keeps just fine track of all the addresses on a single host. A subnet is only a possible range -- using your implied criteria, does SNA keep track of all possible LU's defined by an 8-byte name space?*

I do fully understand subnets. SNA does not keep track of ALL possible LU's defined by an 8-byte name space. I did not state this. In SNA there is a concept of a physical node and logical nodes, and physical nodes do keep track of the status of all physical nodes directly connected to them, and all logical nodes defined on those physical nodes. In a IP network a host only keeps track of the IP address defined on that host. It does not keep track of other IP hosts, not at the IP level, you can write programs that keep track of other nodes, but this is not part the the IP architecture. Keeping track of other devices connected to you and devices (either physical or logical) that you own

Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer:

ang.cobol: Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer:

is part of SNA. This does cause a lot of trivial network traffic and is why SNA is a chatty protocol.

I guess I was trying to compare the statement made by Pete about CICS to a typical TCP/IP based C/S system, Web Server/Browser. He was attempting to state that CICS had to keep track of every terminal in the network and that CICS had buffers for the status of every terminal in the network. My point was that CICS does not keep track of every terminal in the network, it only keeps track of the terminals that are currently in session with CICS. He, in my opinion, seemed to indicate in a C/S application based on TCP/IP that there was nothing that kept track of the sessions and no buffers used. The point I was attempting to make was that it does not matter what environment you are in, if there is a session between a client and a server (be it CICS and a 3270 terminal, Web Server and a Browser, POP3 server and POP3 client, or anything else) the server must keep track of the session and must have buffers allocated to do this.

>
> *And an HTTP connection can create as many connections as the receiving client (and feeding server) will allow. Why only 5? Stateless protocols operate in stateless ways.*

Yes, HTTP as a defined protocol can create as many connection as allowed. However, most modern web browser limit the number of connections created to render a single web page to 5. The 1st connection is used to get the main html page and they use up to 4 other concurrent sessions to get the other files that may be reference in the HTML document. Many years back you had the option to change and the range was from 1 to 8, at least in the first version of Netscape that I used you had this option. You may still be able to this, but it is not obvious to me. If you deployed your application based on HTTP, or wrote your own Web Browser, you could start as many connections as you wish. In fact in Mozilla based browsers you can enable HTTP pipelining and this will use just a single HTTP connection to transfer all files referenced in the html document.

>
> *You seem to be trying to confuse the benefits of fixed circuit networking with virtual circuit networking. Or maybe just having an IE moment.*
>

I was, and it seems that I was unsuccessful, attempting clear up some of Petes comparison of CICS (me assuming SNA networking) to distributed (Pete stating TCP/IP networking). Which does bring me back to my original comments about Petes comments. Comparing CICS to TCP/IP is a bad comparison, they are not the same thing, one is an application server that uses networking protocols, the other is a networking protocol. It like comparing a car to a the road. And I did a bad job attempting to show that the two were different and a direct comparison could not be made.

Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer:

ang.cobol: Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer:

<--- SNIP --->

Re: [Fwd: Re: Mainframe not a good architecture for interactive was Re: What is the future of COBOL? Answer: