

Posted to the Gilder forum - January 20, 2001

## Cao's Vision

Charlie:

I've been reading your posts re Avanex and Cao's proposed switchless solution. Forgetting the mechanics of it, it comes down to the following. The sender figures out which lambda goes to the receiver and sends the message down that lambda. Instead of putting an address on the packet, he just paints it the right color.

To have a truly peer to peer network with no switches (and no islands and no intelligence) every time you add a new user you need to add new connections. The number of additional connections per new user is equal to the number of old users. See the following table:

- If Cao can put 1,000 lambdas on a fiber and each user gets **one** fiber to his curb/home/office, then the network can have 45 users.
- If Cao can put 250,000 lambdas on a fiber and each user gets **one** fiber to the curb/home/office, then the network can have 707 users.
- If Cao can put 250,000 lambdas on a fiber and each user gets **ten** fibers to the curb/home/office, then the network can have 2,236 users.

Connections	Users
0	1
1	2
3	3
6	4
10	5
15	6
21	7
28	8
36	9
45	10
55	11
66	12
78	13
91	14
<b>990</b>	<b>45</b>
9,870	141
99,681	447
<b>249,571</b>	<b>707</b>
998,991	1,414
<b>2,498,730</b>	<b>2,236</b>

This means that we are going to need some intelligence somewhere. From the December GTR:

"Cao estimates that 250,000 lambdas can sustain a national network comprising four all optical islands -- with switches only between the islands -- that can supply 20 million users with gigabit per second connections."

I am going to assume that not all users need the capacity of a full fiber. For example, in my residential building there are 45 condos and probably 1 or two pairs of fibers should do us. An Ethernet LAN in our building would solve our end of the problem and, if all users could share in the same proportion, then the number of users in the above table would rise 40 to 50 fold with the same number of lambdas.

The above scenario is painted for peer-to-peer networking but I think most of the traffic will be client-server. In the February 2000 issue, GG talks about the latency of light problem and how Mirror Image and its 32 CAPs solve the problem. According to my table, 45 CAPs could be connected without switches using just 1,000 lambdas.

Now, suppose you connect each CAP with its users in a star configuration, and suppose that users are clustered at 45 per LAN, then you need less than 10,000 lambdas to connect all users to their respective CAPs (HUBs)

<b>Denny's calculations</b>	
Users	20,000,000
CAPs	45
Users per CAP	444,444
Users per destination	45
Lambdas needed	9,877

Cao might have arrived at his numbers with a similar method:

<b>Cao's calculations</b>	
Users	20,000,000
Islands	4
Users per island	5,000,000
Users per destination	20
Lambdas needed	250,000

Since the latency problem must be solved with caches, I think a more realistic scenario would include at least 30 caches/islands as follows:

Realistic scenario	
Users	20,000,000
CAPs / Islands	30
Users per CAP	666,667
Users per destination	20
Lambdas needed	33,333

To connect 30 CAPs/Islands you need 435 connections. Because of the volume of traffic, these must be multi fiber connections. Today's 80 lambdas per fiber is more than ample to build this core network without any switches. The solution for the edge is just a little further away.

I have the feeling that the future is just around the corner.

Charlie, what do you think of these numbers?

Denny

"Demand creates queues. Supply gets rid of them."

[Software Times](#)

---

Denny,

As a country boy who reads and appreciates your posts, I don't have any idea about what the hell you were saying.

Us non Nerds would appreciate a summation of who the information helps or hurts and how it affects the companies concerned as an investment. I respect your knowledge, but how can I interept it?

Thanks in advance for considering this suggestion

**Other than that I have no opinon.**

taLLboy

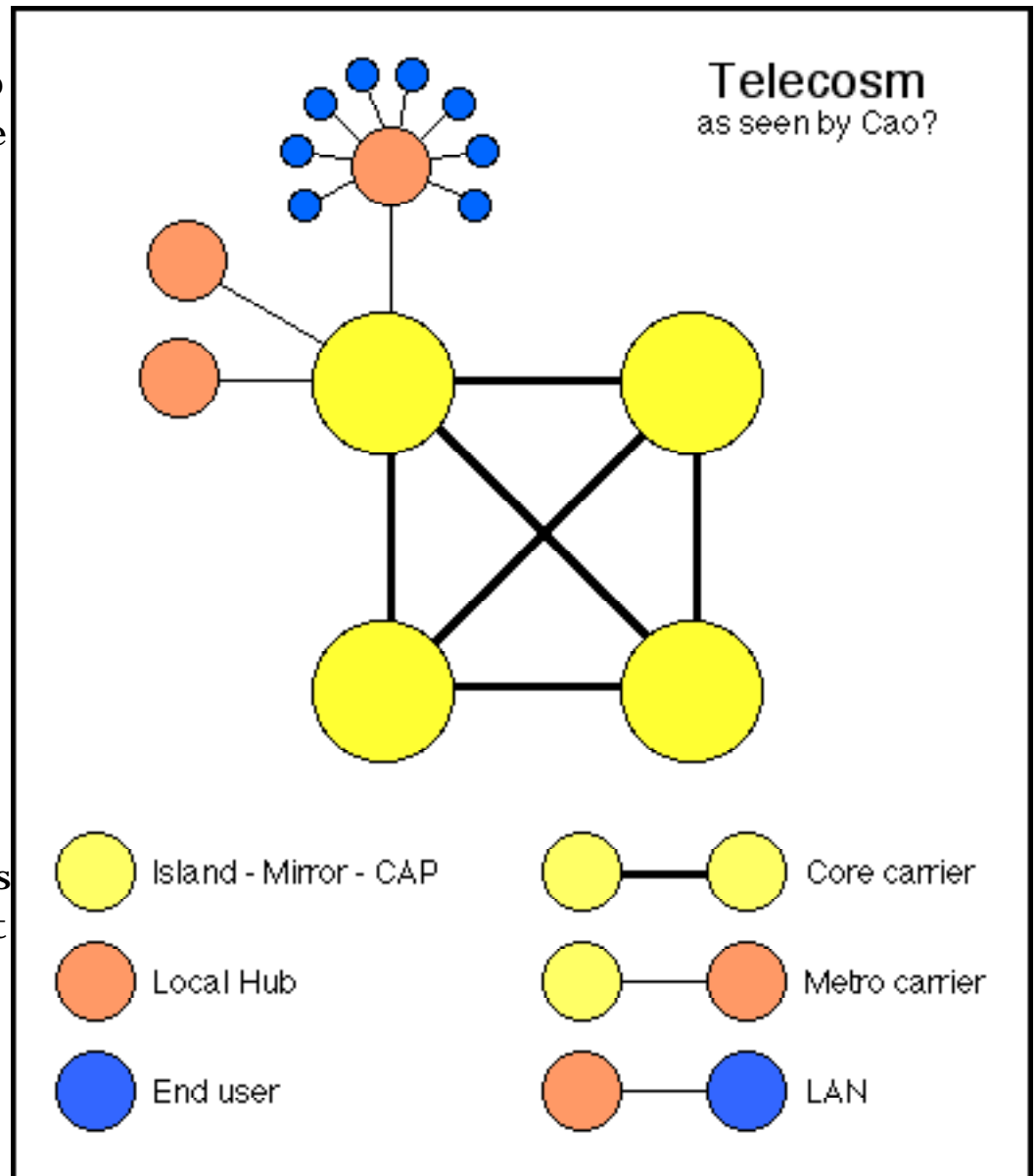
---

Tallboy:

I really can't answer your question. If it was that easy, I would be a gazillionaire and I would have bought out Forbes and the GTR or maybe a South Sea island complete with tulip garden. I believe that the structure of the Telecom, its topology, dictates everything else but since I'm not the designer, I'm trying to guess which will be the most probably outcome.

In its current state, the Telecom has two conceptual parts: core and edge but in practice it has three: core, metro and local loop. In my diagram, metro and local loop are joined, or if you will, the local loop is reduced to a local area network (LAN) or maybe better called campus wide network (CWN). This last link will be both wireline using Ethernet and wireless

and I don't know what network technology that might use. The companies involved in the last mile will be Cisco for the routers and Qualcomm for the wireless plus their respective value chains.



According to Cao, the metro carriers will use 250.000 lambda optical fiber to create a switch free network using Avanex PowerMuxes.

The core carriers can also build a switch free network but they don't need 250,000 lambdas. Probably the current 80/160 lambda technology is more suitable for their needs because the core network has few connections but very high bitrates.

The intelligence will reside in the circles and the circuits themselves, the lines in the diagram, will be dumb, using lambda based muxing and demuxing (on-ramp, off-ramp).

A point that I did not discuss in the original message is the correct place for the servers. In my opinion, the logical place for them is co-located on the same islands where Mirror Image has its CAPs. The owner-administrator of each server can be on the edge of the network like any other user (consider that the Software Times server is located in Seattle, WA and I manage it from Caracas via ftp and browser).

The above is a description of the client-server Telecosm but there will coexist with it a series of peer-to-peer connections. They could use separate private lines or if they could rent "virtual" networks" from the metro and core carriers. I'm not sure how that works but the gigabit Ethernet metro providers are offering that option.

At one time I asked about how the combination of packet switching and lambda transport might work. I think I can now answer my question. The end user, using his web browser or ftp, creates an Internet packet (IP) just as he does now and publishes it on the Ethernet LAN which is connected to the ISP via a local router. If the URL is not local, the router sends the packet to the ISP. The ISP color codes the packet and sends it to the local island. At the local island an intelligent (OEO) switch looks at the address (URL) of the packet and either returns it to the local metro carrier for delivery or sends it to the appropriate remote island. At the remote island, another intelligent (OEO) switch again looks at the URL, color codes the packet and delivers it to the local metro carrier for delivery to the end user. At the appropriate node the packet is demuxed and sent to the recipient's LAN where Ethernet makes the final delivery. In technical jargon, this is called IP packet "tunneling." The IP packet enters a lambda "tunnel" and when it exits at the other end, it continues on its merry way. This is a very good system because the old and the new can co-exist peacefully. It is not a case of

throwing out the old system to replace it with the new but a case where you can build incrementally.

To answer your question, the companies involved might be:

- End user = PC and cell phone = Dell, Compaq, IBM, Gateway, Apple, Motorola, Nokia, Ericsson, Kyocera, Palm, Intel, Arm Holdings, Qualcomm, RF Microdevices and many chip makers.
- LAN = Cisco.
- ISP= ILEC, CLEC.
- Metro = lambda switching = 360, L3, GX, MFNX, Williams, Avanex, New Focus, Corning, JDSU, Cisco.
- Islands = Internet hotels, caches and powerful OEO switches = Exodus, Mirror Image, EMC, Network Appliance, Juniper.
- Core network = GX, L3, 360, Corning, Avanex, JDSU, New Focus.

BTW: Stop being so nasty about Milken or did he milk you dry?

[Denny](#)

"Demand creates queues. Supply gets rid of them."

[Software Times](#)