

What Does the Customer Really Want?

Geoff Huston
September 2000

Spare a sympathetic thought for your local water company. These folk have to condition every drop of water they supply to the highest possible health standard around. After all there is the chance, however remote, that someone may actually drink what comes out of the tap. So the water company spends considerable effort to make sure that the water you use to water the garden or wash the car is drinkable. And they can only charge fractions of cent per litre for this top quality product, because most of the water they provide is splashed liberally on the garden or on your car. Yet we are prepared to pay more than a dollar a litre for drinking water from the local store. The problem with the water company is that in providing a uniform product to a very diverse range of potential uses they have to provision the quality of the product to the most stringent use, that of drinking, yet they can only price the product to match the highest volume use, that of garden water.

The carriage industry is in a similar bind when then construct data circuits for customers' networks. The data circuit is constructed to the most stringent set of demands. To support the most basic set of attached devices the data circuit clocks the bit rate of the data with a phase-locked clock that does not drift. The circuit itself is operated with zero jitter, 99.999% availability and a bit error rate that is less than 1 bit in 10 billion or greater. Tough stuff, and readily capable of supporting any form of data transfer from unframed real time bit stream transfer through to framed packet transmission. So what do most of the carriage industry's customers put upon these precisely engineered high quality real time data circuits? Internet Protocol packets. Or, to be more precise, asynchronous IP packets.

The problem is that when used for IP transmission such high quality circuits are often a case of providing the customer with expensive attributes that the customer's application neither wants nor values, and ultimately the customer does not want to pay for such attributes.

With the use of IP as a close to universal substrate for data networking there has been a shift in the relative roles of the circuit network and the packets that are carried across it. Data circuits are provisioned as real-time synchronous bit streams. IP Packets are asynchronous, and IP applications must assume that there will be some level of imposed jitter. IP packets may use network clocking, or as with Ethernet, packets may have their own clocking preamble prefixed to the packet, creating, in effect, a self-clocking packet. Data circuits are engineered to operate with exceptionally low bit error rates. IP packet protocols operate with the assumption that the network can and will discard packets. Data circuits assume a bit tunnel across the network, connecting one point to another. IP networks assume a many-to-many connectivity domain, where each packet contains the address of its particular destination. Data circuits enforce a strict resource use regime, where each circuit can operate no faster than its clocked rate. IP networks create a variable resource sharing regime, where individual applications adapt their data rates to the prevailing conditions within the network. Data circuits provide a single service model, that of a constant rate point-to-point bit stream, whereas packet networks can operate with a variety of payloads simultaneously, allowing a number of service models to coexist within the same packet network.

What we are seeing is a shift to place increasing levels of utility as encoding in the packet frame and the packet header, and fewer and fewer assumptions being made about the basic characteristics required from the network itself. This is often described as "smart packets, dumb network".

So what is the likely fate of the precisely engineered point-to-point data circuit? In a world of packet data it may well be that we just don't need it any more. Instead of circuits expect to see Virtual Private Networks, and instead of precise latency, clock rates and bit error levels, expect to see Service Level Agreements which describe the general characteristics of the packet switching network without describing the exact fate of every packet.

The carriage industry is slowly moving out of an architectural model where data services were a minor adjunct to the major business of carrying voice. Voice is a demanding task master, and voice circuits do require strict network control over the characteristics of each voice circuit. Data circuits were originally modelled as a minor variation on the basic model of voice circuits. Data is now big business in its own right, and certainly big enough for carriers to design networks that are designed solely for data. Such data only networks are different, in that there is no value in carrying a network clock, no value in carrying point-to-point circuit state and no value in imposing a fixed resource segmentation scheme. Data networks, in their simplest form, are built to switch packets as quickly and as cheaply as possible. It may well be that convergence in the carriage world is no longer the objective. Instead of thinking that a single network platform can be all things to all people, we may see the IP data systems head off into a carriage environment suited precisely to the Internet protocol suite, and see real time applications continue to use stringently engineered carriage systems that provide the necessary service rigor.

A change for the worse? Not really. Its just a case of giving each carriage customer precisely what they need and precisely what they value, and no more.
