Pervasively Distributed CyberInfrastructure (PDC) for Yottascale Data Ecosystems

Micah Beck (mbeck@utk.edu), Terry Moore (tmoore@icl.utk.edu) University of Tennessee

There is widespread consensus that the volume of digital data worldwide is increasing exponentially, with plausible projections putting the total at around 40ZB by 2020 (Figure 1). Given the correlated, explosive growth in prolific data generators, ranging from the Internet of Things (IoT) and intelligent infrastructure, to scientific, engineering, medical, military and industrial equipment of all kinds, the reality of a Yottascale (10^{24}) digital universe is just over the horizon. The challenge of creating a shared infrastructure that can provide the computing, storage/buffer, and networking resources to support data ecosystems at that scale is obviously daunting, to say the least.

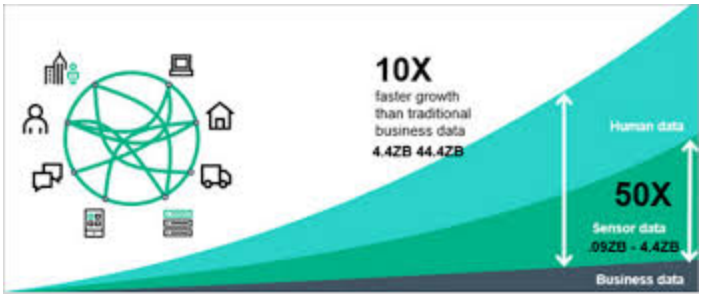

Figure 1: Growth of total digital data from 2010 to 2020. Source: InsideBIGDATA Guide to The Intelligent Use of Big Data on an Industrial Scale, insideBigData, LLC, 2017

What makes this problem especially "wicked" is not just the volume, velocity, variety, variability, and value of the data being produced, but also the fact that the data sources will be *everywhere*, and consequently system nodes that can store/buffer and process data must be everywhere too. The reasons that call for such a *pervasively distributed cyberinfrastructure* (PDC) are becoming familiar: 1) many important applications require low latency response; 2) absent data reduction, adequate and affordable bandwidth will often be lacking; 3) some applications will need high confidence security and/or privacy; and 4) some applications will require robust and highly survivable services [Satyanarayanan, "Emergence of Edge Computing," 2017]. Since the Internet is our best model of a shared PDC to date, one approach to this problem is to imagine a world in which all the Internet access points (e.g., mobile devices and Wifi hotspots, home/building/campus/city/regional level hubs and routers, etc.) are replaced by nodes that provide generic services for storage/buffering and processing as well.

Experience shows that designing a shared, general purpose distributed system with the requisite *deployment scalability* to realize this vision is a grand challenge problem. After all, this has been the explicit or implicit ambition of a succession of well known research and development efforts, including active networking, Grid computing, PlanetLab, and GENI. While these efforts have produced very valuable results, none have achieved the kind of viral growth that we expect a successful future-defining platform to exhibit. We can understand why the problem of designing a shared PDC for future data ecosystems is so formidable by considering four major design constraints:

Combine global interoperability with community specific security: "As a practical matter, real interoperation is achieved by the definition and use of effective spanning layers [Clark, "Interoperation, Open Interfaces, and Protocol Architecture," 1995]." If the PDC's global services layer (e.g., layer three in the Internet model) is also the global spanning layer (i.e.," the narrow waist of the hourglass"), then any use of the system must go through that interface, and therefore must be open to all system users. But the current Internet shows that this approach leaves the system wide open to all manner of malicious ingenuity by bad actors. If we want a shared PDC that provides global interoperability, can that constraint be reconciled with different levels of security/privacy for different communities?

***Manage tradeoffs between node autonomy and manageability*:** Logical centralization enables focused "command and control" of resources, but can suffer from bottlenecks, latencies and other *data logistics* issues due to the separation between the policy and processing planes. Physical distribution enables local control, but makes global optimization (e.g., for performance and reliability) and resource allocation

difficult. The design of current operating systems and networks conflate the placement of resources with their control. Can we design distributed systems in a way that provides appropriate performance and reliability in widely different environments?

***Create a topologically enabled interface to system resources for good data logistics*:** Effective and efficient management proximity/locality/logistics in a shared PDC demands 1) awareness of where the data and the resources to control and process it are, and 2) some control over when, where and in what form it goes next. The Internet's spanning layer hides topology from higher layers for good reasons, but in doing so it restricts their ability to perform data logistics optimizations. At what level must the spanning layer be placed to implement necessary tradeoffs between abstracting away from topology and heterogeneity serving multiple client communities?

Future-proof the design to maximize its sustainability: If a shared PDC is going to be sustainable, it will have to be able to absorb successive waves of innovation in the hardware substrate it runs on. The spanning layer of any *universal bearer platform* necessarily becomes a point of *design ossification,* inhibiting future innovation at that layer. [Anderson, et. al, Overcoming Barriers to Disruptive Innovation in Networking, 2005]. Future-proofing means placing it at the lowest layer at which practical implementation is possible. How low can we go?