UTC FOR ENABLING NEW APPLICATIONS AND SERVICES IN GRID NETWORKS

Mario Baldi* and Yoram Ofek**
* Politecnico di Torino, Torino, ITALY – mario.baldi@polito.it
** University of Trento, Trento, ITALY – ofek@dit.unitn.it

A central challenge in the GRID today is to develop applications and services, specifically, profitable services. Such development has to be supported by proper mapping from the logical/virtual application to the underlying combined/converged networking and computing physical infrastructure. Proper mapping should ensure: (1) well-defined resource allocation, such as: computing, storage, bandwidth – while guaranteeing (2) well-defined delay bounds. Proper use of UTC, ubiquitously and almost free of charge available from GPS and Galileo, enables proper (i.e., cost effective) allocation of computing and networking resources.

1. Introduction and Motivations

This work proposes that the use of global time (a.k.a. time-of-day or coordinated universal time – UTC) is essential in properly mapping GRID-based applications and services to the underying infrastructure. The objective of such mapping is to maximize the perceived quality of service. In order to achieve this objective it is essential to eliminate both switching bottlenecks — critical in the very high capacity network core — and link bottlenecks — at the low speed access (e.g., wireless and DSL). Global time obtained, for example, from GPS (global positioning system) or Galileo, is used in the context of this design to resolve both switch and link bottlenecks, and consequently, to effectively support streaming media applications and distributed computing.

The deployment of new high bandwidth applications has the potential to boost GRID network traffic over the optical backbone. However, since new applications will have to be widely available over a variety of "low capacity" access technologies, such as wireless and DSL. As mentioned, such heterogenous networking scenario is characterized by two main bottleneck problems:

1. *Switch bottleneck* created by emerging multi-media and GRID computing applications and services, and

2. *Link bottleneck* created by the bandwidth mismatch between access and backbone, e.g., between WDM and wireless.

UTC facilitates solutions to the switch and link bottleneck problems, specifically:

1. UTC-based *time-driven switching* (TDS) [1] can solve the switching bottleneck at the network backbone and

2. UTC-based *time-driven priority* (TDP) [2] can solve the (wireless) link bottleneck.

Both UTC-based TDS and TDP, mentioned above, eliminate link bottlenecks since it is possible to avoid congestion. Moreover, the design of TDS packet switches is highly scalable because it requires only speedup of 1, minimum buffers, and the lowest switching complexity. Thereby this design solves the switch bottleneck.

Many interactive applications, such as, telephony, videotelephony and virtual reality, require at the receiver *continuous playing* of the samples captured at the sender. Continuous playing requires a *constant delay* service to be provided by the application layer, i.e., where samples are acquired and played. By using UTC-based packet switching and forwarding it is possible to minimize and smooth of various delay components independent of the connection rate. Specifically, UTC-based packet switching and forwarding enable applications, such as, multimedia and distributed computation over a GRID network:

1. To synchronize the acquisition of samples at the sender (e.g., video capture card) and their continuous playing at the receiver (e.g., video display) with one another and with the TDP forwarding;

2. To guarantee a well-defined bound on the queuing delay, independent of the connection rate and the network load, also where there are link bottlenecks;

3. To enable the implementation of efficient packet switch architectures based on low complexity switching fabrics. This increases the scalability of switches and eliminates the electronic switching bottleneck.

2. Network Architecture and Deployment

According to various provisioning models, applications signal their Quality of Service (QoS) requirements to the network. If the network has enough resources to satisfy the request, they are reserved and packets transmitted by each application are handled in a way that QoS is guaranteed to their flow or connection. Most queuing algorithms used to implement such packet handling have to maintain status information for each flow and are recognized to be non scalable. Time-driven priority (TDP) forwarding <u>does not require per flow information</u> in intermediate nodes, while providing deterministic QoS to each connection. Thus, TDP enables the per-flow QoS for which the Integrated Services (IntServ) model was designed but has provisioning scalability comparable to the differentiated services (DiffServ) model.

In accordance to the traffic engineering and provisioning models currently deployed over the Internet, connections can be aggregated in the network core in order to improve scalability by increasing the granularity with which switches handle packet flows or connections. *Synchronous Virtual Pipes* (SVPs) can be set up over networks deploying global time in order to aggregate multiple connections, thereby relieving core nodes from participating in connection level signaling. An SVP can be regarded as a virtual leased line from the point of view of the service provided and as an MPLS LSP tunnel from the signaling point of view. In order to deterministically guarantee QoS to single connections, Access Bandwidth Brokers (ABBs) at the edges of an SVP handle signaling requests from the applications whose packet connections are to traverse the SVP, and determine the availability of resources within the SVP.

The Internet is based today on asynchronous packet switches, which do not feature TDP forwarding. Thus, especially in the initial deployment phase, TDP/TDS switches will coexist and interoperate with current asynchronous packet switches. shows a scenario, likely to be common in the early days of TDP deployment, in which end stations connected to asynchronous local area or access networks communicate through a TDP backbone. Synchronous *boundary nodes* control the access to SVPs set up on the synchronous backbone performing both policing and shaping of packets flows– i.e., they *synchronize* packet forwarding. TDP provides the minimum delay bound when deployed end-to-end, but it can be beneficial even when its use is confined to subnetworks. The node at the ingress of a TDP subnetwork, which shares the global time reference, eliminates the delay variation experienced by packets in the asynchronous network; then packets benefit of the controlled delay service provided by the synchronous subnetwork.

The delay jitter introduced by asynchronous segments can be completely canceled in the end-to-end communication if end stations use the common global time to attach time stamps to the packets they generate. The Real-time Transport Protocol (RTP) can be used to carry the time stamp. The source end system generates the time stamp according to UTC. Packets travel through the TDP/TDS network with controlled delay and no loss. When packets arrive to the destination they have experienced a variable delay in the asynchronous access subnetwork to which the destination is connected. By knowing the delay bound, say of *b* time units, on the asynchronous segments traversed by the packet, the end station can completely eliminate the jitter by determining the replay time as the time stamp value plus *b*.

3. UTC and Periodic Forwarding: Time-Driven Switching / Priority

All packet switches are synchronized with UTC. Time is divided into time periods that are called *time frame* (TF). The TF duration is derived from the UTC second received, for example, from a time distribution system such as GPS, Galileo, TWSTFT (Two-Way Satellite Time and Frequency Transfer). For example, by dividing the UTC second by 8,000, the duration of each time frame is T_f = 12.5 to 125 μs; however, the time frame duration can be set as needed.[1]

TFs are grouped into a *time cycle*; Figure 1 shows an example of a time cycle that contains 100 TFs, i.e., there are 80 time cycles in a UTC second. Time cycles are further organized in *super cycles*, each of which typically equals one

[1] UTC receivers from GPS are available from many vendors for a low price (for example, the price of a one PPS (pulse per second) *UTC* clock, with accuracy of 10-20 nanoseconds, is about $200). By combining UTC from GPS with local Rubidium or Cesium clocks it is possible to have a correct UTC (±1 μsecond) without an external time reference from GPS for days (with Rubidium clock) and months (with Cesium clock).

UTC second. This timing structure is useful to perform resource reservation in order to provide guaranteed services.

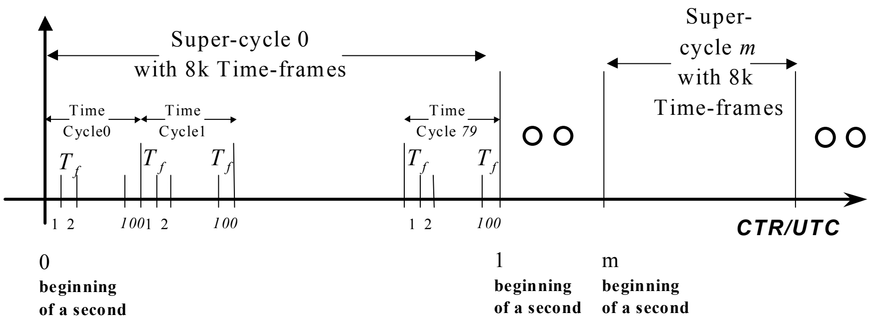

Figure 1: UTC as a global common time reference

Periodic forwarding indicates that the forwarding pattern repeats itself in every time cycle and in every super cycle. TDP guarantees that the end-to-end delay jitter is less than one TF and that reserved real-time traffic is transferred from the sender to one or more receivers with no loss due to congestion. The simple TDP operation is generalized by adding the following two conditions:

1. All packets that should be sent in TF *i* by a node are in its output port before the beginning of TF *i*, and
2. The delay between an output port of one node and the output port of the next down-stream node is a constant integer number of TFs.

Table 1: A comparison with other methods

Service capabilities / Communication methods		Circuit Switching or PSTN	Single Async. Priority w/no reservation	Multiple Asynchronous Priorities – DiffServ	Time-driven Priority / Switching
Data: mail, ftp, etc.		No	Yes	Yes	Yes
Interactive – on Global Scale	Phone	Yes	No	Not proven: depends on scheme	Yes
	Video-phone	Yes	No	Not proven: depends on scheme	Yes
Utilization vs. Loss: with high and low speed links		Full utilization and no loss	Low utilization or High loss	Utilization can be low, and loss can be high – requires overprovisioning	Full utilization, no loss, easy to schedule
Experience		100+ years	25+ years	New technology	New technology

The TDP/TDS forwarding is important because of the possibly long lasting software protocol processing in heterogeneous multi-protocol internetworking environments. In this case, a predefined, but fixed, number of TFs will be added in some intermediate switches with an increase in the end-to-end delay; the end-to-end delay jitter will remain constant. In Table 1 some of the unique properties of TDP is compared with four types of communication networks.

4. References

[1] M. Baldi, Y. Ofek, 2003, *Realizing Dynamic Optical Networking*, Optical Networks Magazine, Special Issue "Dynamic Optical Networking: around the Corner or Light Years Away?, Vol. 4, No. 5, pp. 100-111.

[2] M. Baldi and Y. Ofek, 2000, *End-to-end Delay Analysis of Videoconferencing over Packet Switched Networks*, IEEE/ACM Transactions on Networking.