**<Note to Editor: Add partial content from Mark Hantel’s presentation in** [**http://ieee802.org/1/files/public/docs2019/60802-Hantel-SampleSelection-0319-v00.pdf**](http://ieee802.org/1/files/public/docs2019/60802-Hantel-SampleSelection-0319-v00.pdf)**, as modified by the comments made during the presentation>**

**4.1.2 Control Loop Applications**

Applications need to have a common understanding of time among each other or with the network if they need to synchronize and sequence events. This is used for things such as synchronizing multiple axes of motion, for timestamping various events. The working clock enables a common understanding of time. If an application doesn’t need to use a common understanding of time, it may be executing based on a local timescale.

The working clock synchronizes a TSN network using IEEE Std 802.1AS. While bridges propagate the working clock, they will only synchronize transmission according to the working clock if Enhancements for Scheduled Traffic (IEEE Std 802.1Q-2018) is enabled.

While many traffic types are implemented on an industrial automation system, only a few traffic types require minimized latency and known delivery windows. Examples are isochronous real-time traffic and cyclic real-time traffic.

Isochronous real-time traffic is commonly used for motion control and can have application cycle times lower than 50µs. Isochronous traffic expects the application cycle to be aligned with the working clock. Cyclic traffic is typically used for reading sensors and activating output signals at 1 ms and above.

**<Note to Editor: End of adopted Hantel Contribution>**

**<Note to Editor: Add simplified Application Types Table as agreed in Vancouver>**

Table 5 – Application Requirements

| Level | Isochronous Application | | Non-Isochronous Application | | |
| --- | --- | --- | --- | --- | --- |
| **Application** | Synchronized to network access | | Synchronized to local timescale | | |
| **Network access** | Synchronized to working clock | | | | Synchronized to local timescale |
| **Network/Bridges** | Synchronized to working clock | Free running | Synchronized to working clock | Free running | Free running |

**<Note to Editor: End of added table>**

**<Note to Editor: Add partial content from Mark Hantel’s presentation in** [**http://ieee802.org/1/files/public/docs2019/60802-Hantel-SampleSelection-0319-v00.pdf**](http://ieee802.org/1/files/public/docs2019/60802-Hantel-SampleSelection-0319-v00.pdf)**, as modified by the comments made during the presentation>**

**4.1.3 Mechanisms that can be used to meet Control Loop Latency Requirements**

Meeting latency requirements on a network can be accomplished in multiple ways. Some of those methods are:

1. Define and test all possible application combinations
2. Overprovisioning the network
3. Providing scheduled time slots for each application to transmit on the network
4. Preempting lower priority traffic
5. Providing scheduled time slots for certain traffic classes
6. Synchronizing network access
7. Simulating the traffic load of specific implementations ahead of time
8. Using Enhancements for Cyclic Queuing and Forwarding

Preemption is defined in 802.1Q-2018 and 802.3-2018.

Reserving time on the network for certain traffic types can be done through specific implementations of Enhancements for Scheduled Traffic (802.1Q-2018). A network cycle needs to be defined for this to work. The network cycle is a multiple period of the applications that are traversing the network. Once that cycle time is defined, portions of that cycle time can either be allocated to streams or classes of streams.

When a single application is implemented on a network, that application can sequence the communications of the controller, sensors and actuators. When multiple applications share the same network, those applications need to validate that the network will always meet their latency requirements. This can be accomplished by synchronizing Network Access. This is a method for coordinating transmission of all the traffic that shares a Network to validate all application requirements can be met and increasing utilization.

Creating a digital model of network traffic ahead of time will guarantee that a solution will work and can allow network access to be synchronized.

**<Note to Editor: End of adopted Hantel Contribution>**