AUGMENTING SERIAL STREAMING TELEMETRY WITH INET DATA DELIVERY

Carl Reinwald
Senior Principal Engineer
CSC®, Santa Maria, California

ABSTRACT

Incorporating network-based telemetry components into a flight test article creates new types of network-based data flows between a test article and a telemetry ground station. The emerging integrated Network Enhanced Telemetry (iNET) Standard defines new, network-based data delivery protocols which can produce various network data flows. Augmenting existing Serial Streaming Telemetry (SST) data flows with these network-based data flows is crucial to enhancing current flight test capabilities. This paper briefly introduces the network protocols referenced in the iNET Standard and then identifies the various data flows generated by network-based components which comply with the iNET Standard. Several combinations of SST and TmNS data flows are presented and the enhanced telemetry capabilities provided by each combination are identified. Identifying time intervals of unused telemetry network bandwidth explicitly for reallocation to other test articles is also addressed.

KEY WORDS

iNET, TmNS, SST, PCM, data flow, data transport, RC Delivery, LTC Delivery

INTRODUCTION

For over 50 years, the IRIG 106 Standard has defined a simplex Serial Streaming Telemetry (SST) standard for telemetry communication links. Over the past few years, the integrated Network Enhanced Telemetry (iNET) project has been developing a standard for a bidirectional, packet-based network communication link that will augment the existing SST link. By establishing a Telemetry Network System (TmNS) between a test article and a ground station, the ground station can control and possibly reconfigure devices within the test article. A ground station may request device status and receive data from the test article on the TmNS (rather than the SST link). Figure 1 presents an overview of a TmNS.

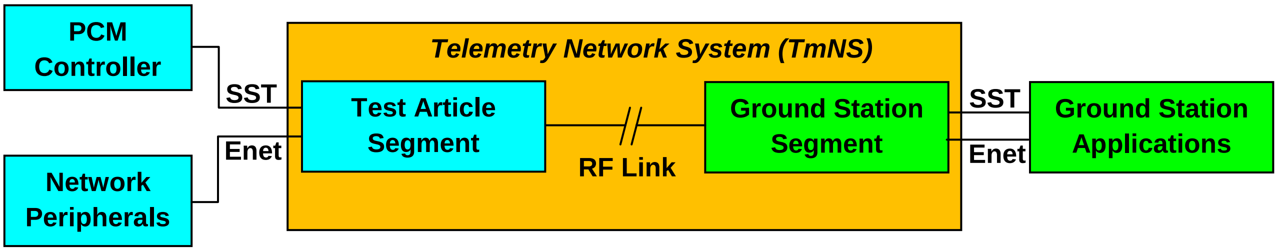


Figure 1. Telemetry Network System (TmNS) Overview.

The practical ramifications of establishing a TmNS between a test article and a ground station are numerous including the potential for lossless telemetry, efficient spectrum utilization, and remote control of test article acquisition equipment. This paper focuses on the new, network-based data flows made available by implementing a TmNS. Before identifying these network data flows, this paper presents a brief overview of the network protocols specified by the iNET Standard. Several types of test-article-to-ground-station TmNS data flows are presented along with some of the new capabilities that a TmNS makes available to the flight test community.

NETWORK DATA TRANSPORT PROTOCOLS

The iNET Standard defines a TmNS Data Message as a “container” for encapsulating data (measurement data, bus data, etc.). To propagate these messages across a packet-switched network, the TmNS uses the Internet Protocol (IP) Suite. The iNET Standard specifies two data transport protocols for data delivery on the network: Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). For network delivery, each TmNS Data Message is encapsulated in either one or more TCP segments or one UDP datagram; each TCP segment or UDP datagram is then encapsulated into an IP packet (see Figure 2).

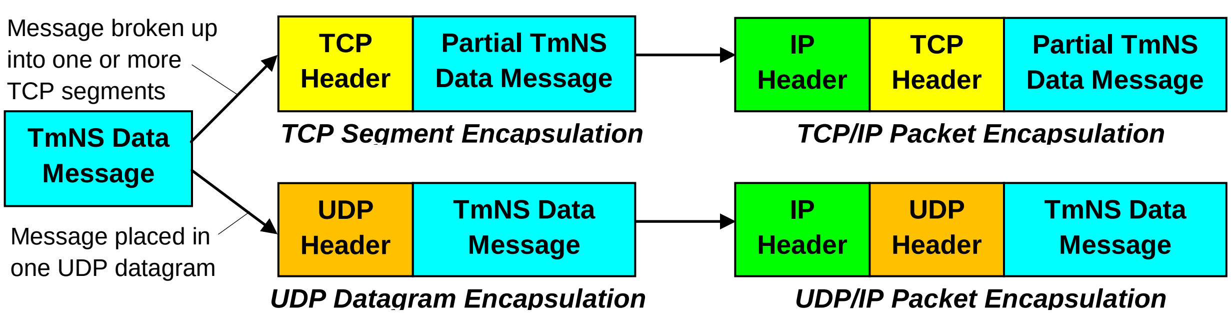


Figure 2. TmNS Data Message Data Transport Encapsulation.

TCP provides a reliable, connection-oriented protocol for data transport on a packet-switched network. TCP ensures proper ordered delivery of data and automatically detects and retransmits lost packets. The iNET Standard defines *Reliability Critical (RC) Delivery Protocol* as TmNS Data Messages transported via TCP. TCP provides a relatively “robust” protocol that requires network overhead to perform reliable data delivery. However, TCP’s mechanism for the retransmission of lost packets introduces a potential for large and unknown data latencies. If a packet is lost, TCP retransmits the lost packet resulting in a packet data latency that is nondeterministic and is affected by the network link integrity.

Since the majority of telemetry applications require low latency, deterministic data transport, the iNET Standard also defines *Latency / Throughput Critical (LTC) Delivery Protocol* as TmNS Data Messages transported via UDP. UDP provides a lightweight protocol as compared to TCP resulting in a simple, connectionless transmission model that does not guarantee data delivery or data ordering. Transporting TmNS Data Messages via LTC data delivery provides low latency data transport, but data delivery is not guaranteed. The iNET Standard also specifies the use of UDP/IP multicast which supports multiple data consumers of the same datagram. A data producer writes a single datagram to a multicast address and port. Multiple data consumers can subscribe to that multicast address and port to receive the same datagram. This lightweight protocol coupled with multicast support makes UDP more attractive for latency sensitive data delivery than the more “robust” connection-oriented TCP.

In summary, LTC data delivery is implemented via UDP/IP (nominally UDP/IP multicast) which provides relatively low data latency, data can be sent to multiple consumers efficiently, but data delivery is not guaranteed. RC data delivery is implemented via TCP/IP which provides reliable data delivery but data latency is nondeterministic and network overhead is greater than LTC data delivery.

TmNS DATA FLOWS

In the TmNS vernacular, an End Node is a data producer or consumer. A typical data producing End Node would be a Data Acquisition Unit (DAU); one type of data consuming End Node would be a TmNS Data Recorder (records TmNS Data Messages). A ground station is also considered a data consuming End Node and can receive TmNS Data Messages from other End Nodes. When a data consuming End Node receives data from a data producing End Node, a TmNS data flow is established. Although there are only two data transport protocols (LTC and RC), several types of TmNS data flows can be identified. There are three types of LTC data flows:

- LTC-AQ – Acquisition TmNS Data Messages that originate directly from an acquisition source (ex: DAU)
- LTC-NAQ – Non-Acquisition TmNS Data Messages which do not originate directly from an acquisition source (ex: simulator generated or playback from a recorder)
- LTC-REL – Relayed TmNS Data Messages, an RC data flow transformed into an LTC data flow for purposes of relaying messages

There are two types of RC data flows:

- RC-DR – Data Retrieval, TmNS Data Messages retrieved from a TmNS recording unit
- RC-NRT – Near Real-Time TmNS Data Messages, an LTC data flow transformed into an RC data flow for reliable data transport

Figure 3 illustrates the paths of each of these TmNS data flows.

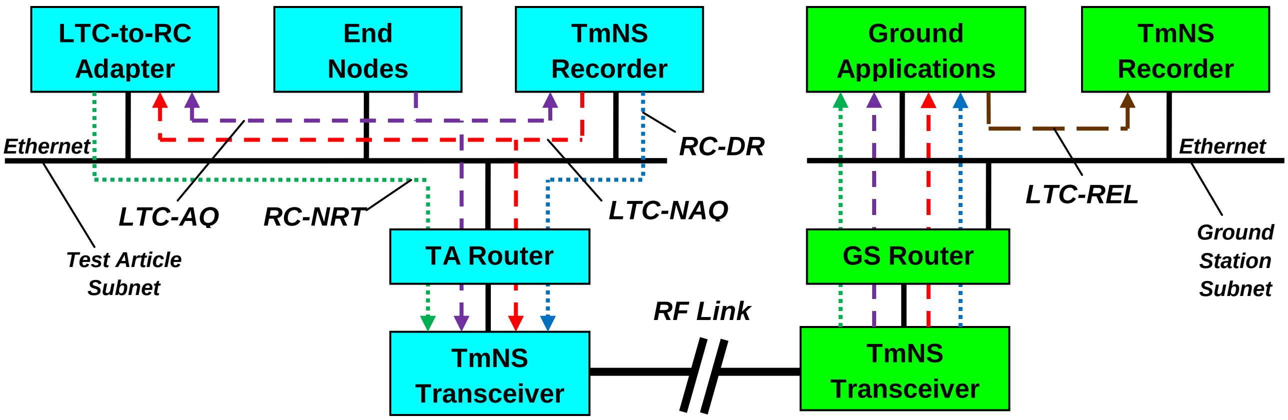


Figure 3. TmNS Data Flows.

End Nodes multicast TmNS Data Messages onto the Test Article Subnet. Any End Node can receive these messages by subscribing to the corresponding multicast address and port. An LTC-AQ data flow is established when one End Node multicasts data to a particular multicast address and port and another End Node subscribes to the same multicast address and port. When a ground application subscribes to a particular multicast address and port, this routing information is added to both routers to enable the corresponding LTC-AQ data flow from the Test Article Subnet to the Ground Station Subnet (and ultimately to the ground application). If no ground application subscribes to a particular multicast address and port, the Test Article Router will not propagate the corresponding data through the router. In other words, an LTC-AQ data flow through the TmNS Transceiver is not established unless at least one ground application subscribes to the corresponding multicast address and port. This multicast publish and subscribe mechanism prevents unnecessary transmission of test article data to the ground station.

A TmNS Recorder on the test article can receive a particular LTC-AQ data flow by subscribing to a multicast address and port, and then subsequently record that LTC-AQ data flow. When a ground application requires data from the test article recorder, the ground application establishes an RC session with the recorder and an RC-DR data flow is established from the recorder to the ground. This TmNS-enabled capability to retrieve recorded data from a test article recorder during a test is a fundamental enhancement to the current unidirectional SST.

As mentioned in the Network Data Transport Protocols section, LTC-AQ data flows provide low latency data delivery but do not guarantee data delivery and data loss may occur. Since RC data flows do guarantee data delivery, a user now has the option to accept a non-deterministic amount of latency for a specific data set in order to guarantee receipt of that data. The RC-NRT ("Near Real Time") data flow provides this functionality; RC-NRT is the result of transforming an LTC data flow (nominally LTC-AQ) into an RC data flow. An LTC-to-RC Adapter subscribes to multicast addresses to receive TmNS Data Messages via an LTC-AQ data flow. The received data is immediately written to an existing RC session resulting in a TmNS Data Message flow using TCP that guarantees data delivery. (There are operational limitations of TCP that reduces the term "guarantees" to "guarantees under specific operating conditions".) Under nominal conditions, TCP will retransmit lost data packets resulting in a lossless data transfer. RC-NRT data flows provide one critical part of a mechanism for achieving lossless telemetry between a test article and a ground station over an imperfect RF link.

LTC-NAQ data flows are identical to LTC-AQ data flows except the data source is not directly from an acquisition source. One example of an LTC-NAQ data flow source is a simulator generating an LTC data flow to exercise the system. Another possible LTC-NAQ data flow source is a TmNS Recorder that has the capability to playback recorded TmNS Data Messages as an LTC data flow (nominally TmNS Recorders playback data as RC data flows). Both of these LTC data flows are different than the "normal" LTC-AQ data flows and have been categorized together as the LTC-NAQ data flows.

The LTC-REL data flow is simply the inverse of an RC-NRT data flow. An application (typically a ground application) establishes an RC session with an End Node on the test article. Since RC sessions are unicast (point-to-point), the only End Node that can receive the RC-DR data flow is the initiating ground application. But what if another application requires the same RC-DR data flow? For example, a ground recorder needs to record all received RC-DR data flows. To satisfy the ground recorder, the ground application receiving the RC-DR data flow can immediately multicast the same data. The ground recorder can then subscribe to the corresponding multicast address and port and an LTC-REL data flow is established between the ground application and the ground recorder.

The three principle TmNS data flows that appear across an RF link during a test are

- LTC-AQ – Data that originates directly from an acquisition source (ex: DAU)
- RC-DR – Data Retrieval, data retrieved from a TmNS recording unit (ex: recorder)
- RC-NRT – Near Real-Time data, usually originating from an LTC-AQ data flow

The other data flows (LTC-NAQ and LTC-REL) will not be addressed further in this paper.

AUGMENTING SST WITH TmNS DATA FLOWS

Adding the aforementioned TmNS data flows to an existing SST data flow creates new flight test capabilities currently unavailable with SST only implementations. This section explores various ways in which TmNS data flows might be added to a baseline test article configuration and then describes the resulting capabilities.

The baseline test article configuration, shown in Figure 4, consists of DAUs and a recorder being connected via a bus to a PCM controller. The PCM controller constructs PCM frames which are sent to an SST transmitter. The PCM controller may also propagate the PCM frames back to the recorder or a different PCM frame format may be sent to the recorder. For example, the recorder-bound PCM frames might contain all test parameters whereas the SST-bound PCM frames contain only a subset of the test parameters.

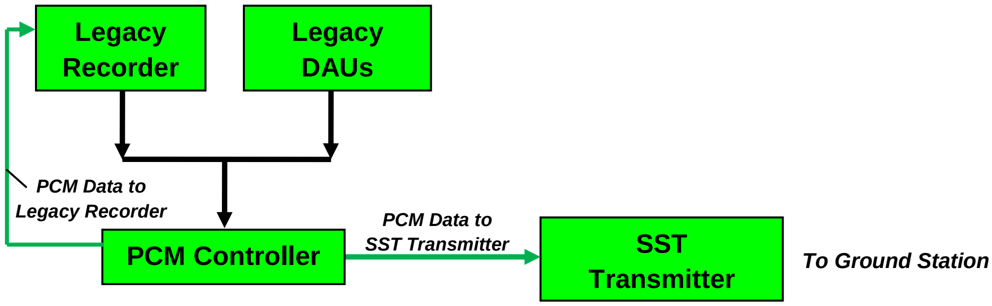


Figure 4. Baseline Test Article Configuration – SST Data Flow Only.

The first capability to add to the baseline configuration is support for lossless SST data (see Figure 5). When the SST RF link drops out, SST data is lost. Adding the capability to retrieve lost SST data from an onboard TmNS recorder provides one mechanism for establishing lossless SST data. Along with the test article TmNS transceiver and router, two End Nodes are required to perform this function:

- SST–TmNS Gateway: receives SST-bound PCM frames from the PCM controller, encapsulates the PCM data into TmNS Data Messages, and writes the messages onto the network
- TmNS Recorder: receives and records TmNS Data Messages and supports retrieval of selected TmNS Data Messages

PCM data destined for the SST transmitter is also routed to the SST-TmNS Gateway. The Gateway inserts the PCM data into TmNS Data Messages which are then published to a specific multicast address and port. The TmNS Recorder subscribes to the same multicast address and port and a subsequent LTC-AQ data flow is established on the test article with PCM data contained inside the TmNS Data Messages (LTC-AQ, PCM Data). To retrieve missing SST data for an RF link dropout period, a Ground Station application establishes an RC session with the TmNS Recorder and retrieves the desired TmNS Data Messages (RC-DR, PCM Data) for that time interval. By receiving the majority of SST data via the SST link and the lost SST data via the RC-DR mechanism, a TmNS-equipped test article and ground station can provide lossless SST data to the flight test community with efficient use of the available RF link bandwidth.

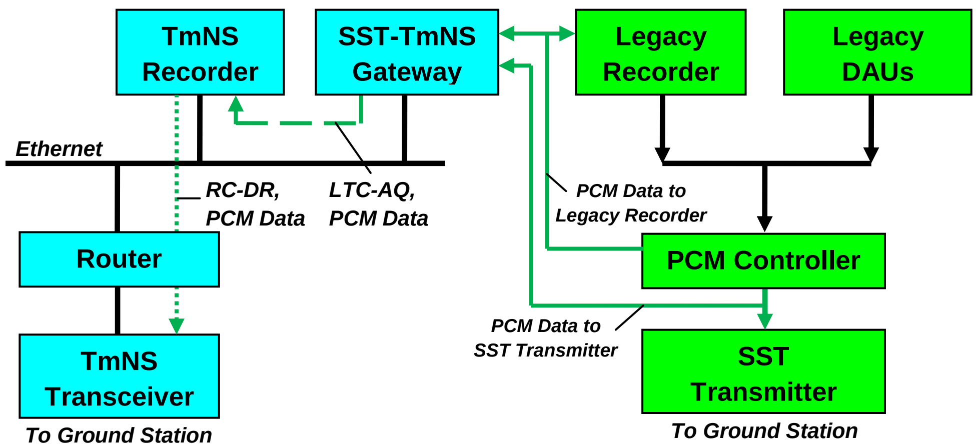


Figure 5. Add TmNS RC-DR (Data Retrieval) PCM Data Flows.

Although providing lossless SST data is a considerable enhancement over traditional simplex SST, incorporating native TmNS End Nodes adds even more capabilities (see Figure 6). Native TmNS Data producing End Nodes (ex: DAUs) publish TmNS Data Messages to one or more multicast addresses and ports. When the TmNS Recorder subscribes to a particular multicast address and port, an LTC-AQ data flow is established with DAU data contained inside the TmNS Data Messages (LTC-AQ, DAU Data). If a ground station application subscribes to that multicast address and port, then an LTC-AQ data flow with DAU data will also exist between the End Node and the Ground Station.

For any End Node data recorded by the TmNS Recorder, a ground station application can initiate an RC session with the TmNS Recorder to retrieve the End Node data (RC-DR, DAU data). Retrieved data can either be lost TmNS Data Messages (received via LTC-AQ) or data that were never transported to the ground (retrieval of non-transmitted data). This capability to retrieve any recorded test article data at any time, regardless of whether the data was initially telemetered is another substantial enhancement to SST.

Measurement data from End Nodes that generate DAU data can also be inserted into PCM frames for transport via SST. The SST-TmNS Gateway becomes a bidirectional gateway where high priority TmNS data could be sent back to the PCM controller for incorporation into PCM frames (see the red line data flows in Figure 6).

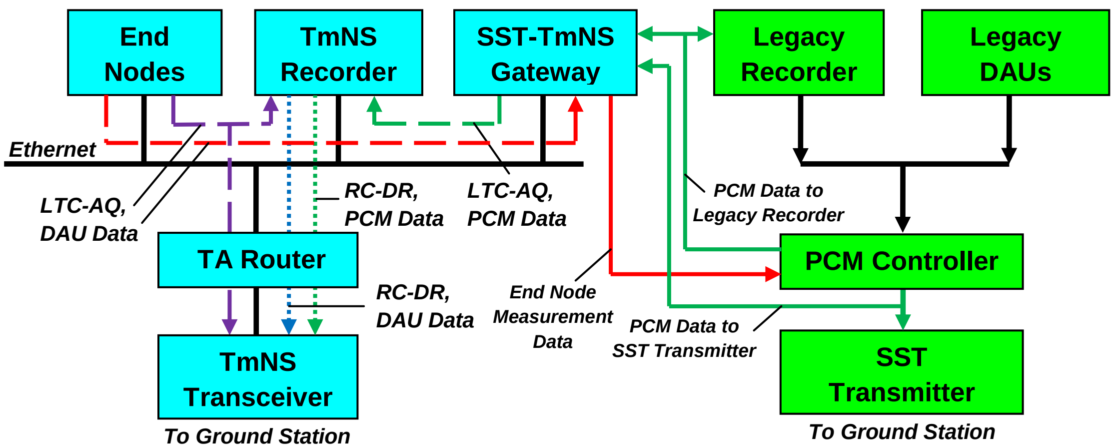


Figure 6. Add TmNS LTC-AQ and RC-DR DAU Data Flows.

The addition of an LTC-to-RC Adapter enhances the TmNS by providing an RC-NRT data flow (as described previously, see Figure 7). A ground station application initiates an RC session with the LTC-to-RC Adapter for a specific set of data. The LTC-to-RC Adapter would subscribe to one or more multicast addresses and ports to receive the requested LTC-AQ data flows. The Adapter would then immediately write each TmNS Data Message to the ground station application via the RC connection (this is an RC-NRT data flow).

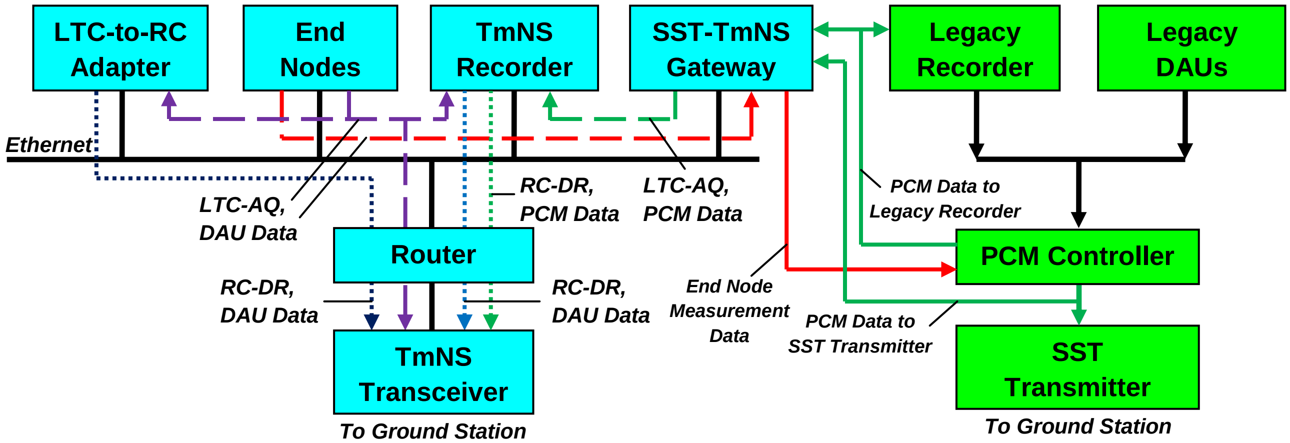


Figure 7. Add LTC-to-RC Adapter and RC-NRT Data Flows.

Under what situation would a ground station application prefer an RC-NRT data flow to an LTC-AQ data flow? Since the LTC data delivery protocol does not guarantee data delivery, obtaining lossless telemetry using LTC data delivery would also require executing RC data retrieval to retrieve TmNS Data Messages that were lost in transport. The combination of LTC-AQ with RC-DR provides one form of lossless telemetry of End Node data. Another approach is to change the LTC transport to an RC transport for selected End Node data. Since RC data delivery provides guaranteed data delivery, a user may be willing to accept a non-deterministic amount of latency for a specific data set in order to guarantee data delivery without needing to perform an RC data retrieval (RC-DR). A premise being made regarding the need for an LTC-to-RC Adapter is that typical data producing End Nodes will not have the sophistication to perform RC session management. Therefore the majority of End Nodes will only generate LTC-AQ data flows and will not support RC sessions. However, a TmNS Recorder would be an ideal place to incorporate an LTC-to-RC adapter.

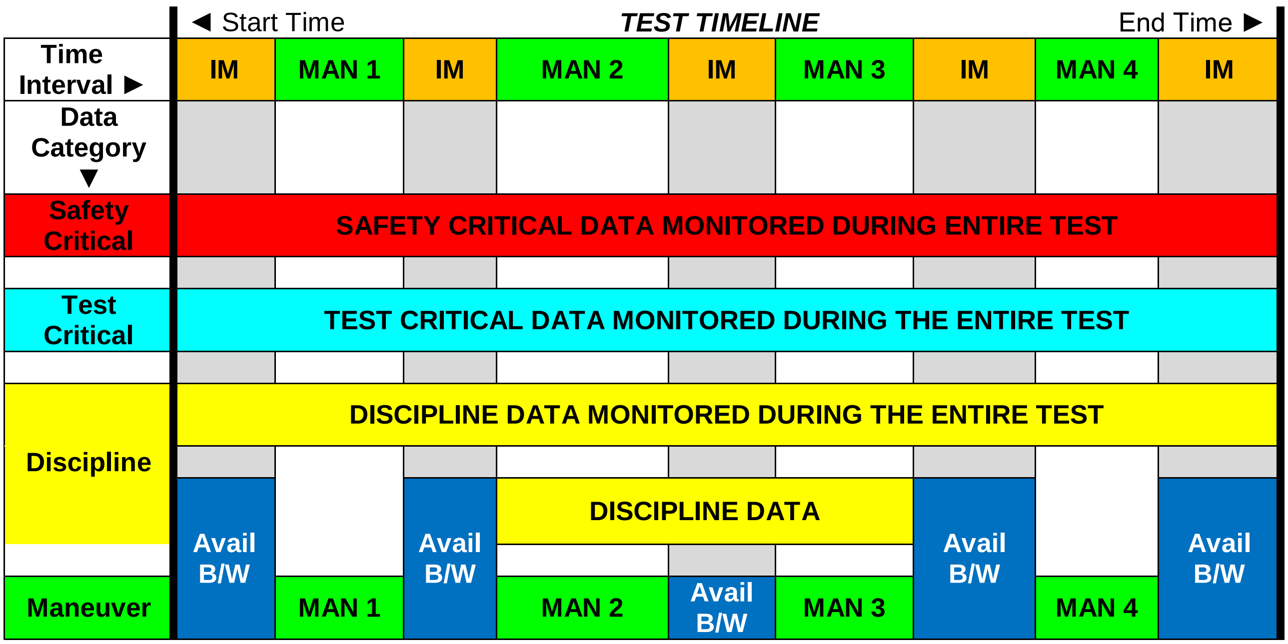
To summarize: for minimal latency but potentially lossy data, use SST data transport for PCM data sources and use LTC-AQ data transport for TmNS data sources. For lossless SST data, use SST combined with RC data retrieval (RC-DR) to retrieve lost SST data from the TmNS Recorder. For lossless TmNS data, use LTQ-AQ combined with RC data retrieval (RC-DR) to retrieve lost TmNS data from the TmNS Recorder or incorporate an LTC-to-RC Adapter to generate an RC-NRT data flow.

CATEGORIZING DATA FOR DATA TRANSPORT

Augmenting SST with TmNS data flows provides significant enhancements to current flight test capabilities. Categorizing the type of data for transport and then identifying the appropriate data transport mechanism (SST or TmNS) for each data category provides another step towards optimizing telemetry spectrum usage. This paper groups all telemetered data into one of the four following data categories (how telemetered data parameters are categorized and which parameters qualify for TmNS data transport are range and test specific):

- ***Safety Critical Data*** is monitored during the entire test and must arrive at the ground station with minimal latency. Examples: altitude and airspeed.
- ***Test Critical Data*** is also monitored during the entire test and must also arrive at the ground station with minimal latency. These parameters are not safety critical but are still deemed critical to the test being conducted. Example: flight conditions.
- ***Discipline Data*** falls into two sub-categories: parameters monitored during the entire test and parameters monitored only during specific time intervals. Example: propulsion.
- **Maneuver Data** is monitored only during the specific maneuver interval.

Figure 8 presents a typical test timeline for each data category. Data that is monitored during the entire test provides no opportunity for telemetry spectrum reallocation (since all data must be continuously transmitted). However, data that is monitored only during specific time intervals (some discipline data and maneuver data) provide windows of opportunity for telemetry spectrum reallocation (see the "Avail B/W" time intervals in Figure 8).



IM = Inter-maneuver time interval; **MAN n** = Maneuver time interval 'n'
Avail B/W = Time intervals where data is not being transmitted during the current test.

Figure 8. Test Timeline and Data Category Definitions.

Transmitting data only when that data is needed for monitoring is unfamiliar to the current flight test community because SST forces all data categories to be merged into PCM frames for SST data transport. A TmNS does not require the merging of data categories for transport. Consequently, some telemetry spectrum could be shared between multiple test articles on an as-needed basis for data that does not have to be continuously transmitted (ex: some discipline data and maneuver data time intervals as shown in the 'Avail B/W' time intervals in Figure 8). Figure 9 shows several possible combinations of SST data transport and TmNS data transport for the four data categories.

DATA CATAGORY	DATA TRANSPORT OPTIONS					
	SST Only	Combined SST and TmNS				TmNS Only
Safety Critical	SST	SST + RC-DR	SST + RC-DR	SST + RC-DR	SST + TmNS[1]	TmNS[1]
Test Critical	SST	SST + RC-DR	SST + RC-DR	SST + TmNS[1]	TmNS[1]	TmNS[1]
Discipline	SST	SST + RC-DR	SST + TmNS[1]	SST + TmNS[1]	TmNS[1]	TmNS[1]
Maneuver	SST	SST + RC-DR	SST + TmNS[1]	SST + TmNS[1]	TmNS[1]	TmNS[1]

[1] TmNS means RC-DR with LTC-AQ and possibly RC-NRT.

Figure 9. Possible Combinations of SST and TmNS Data Flows per Data Category.

For the column in Figure 9 where all four data categories use SST + RC-DR, all data would still be transported via SST but RC data retrieval (RC-DR) could be used to retrieve data lost during transport. As flight data transport is moved away from SST and towards TmNS, more bandwidth becomes available for telemetry spectrum reallocation.

CONCLUSION

Augmenting existing Serial Streaming Telemetry (SST) data flows with TmNS data flows provides many enhancements to current flight test capabilities. The emerging iNET Standard includes traditional TCP/IP and UDP/IP data transport protocols that result in several different TmNS data flows between a test article and a ground station. Several combinations of SST and TmNS data flows have been presented, each with their corresponding enhancements to traditional telemetry capabilities. Categorizing test parameters into four data categories provides a framework to identify unused telemetry network bandwidth explicitly for reallocation to other test articles. The TmNS architecture embraces network technology to offer many new enhancements for flight testing including lossless telemetry and dynamic spectrum allocation.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the significant contributions of all iNET Ground System Working Group members to this publication.

REFERENCES

1. Cookson, William, "Test Capabilities Requirements Document for integrated Network Enhanced Telemetry (iNET) Aeronautical Environment Implementation", Revision 1.1, Department of Defense, June 2008.