SBN Online Systems Task Force Final Report

W. Badgett[1], E. Church[2], M. Convery[3], A. Fava[1,4], G. Horton-Smith[5], G. Karagiorgi[6], W. Ketchum[1], M. Kirby[1], J. Paley[1], G. Savage[1], D. Torretta[1], and S. Ventura[4]

[1]*Fermi National Accelerator Laboratory*
[2]*Pacific Northwest National Laboratory*
[3]*Stanford Linear Accelerator Center*
[4]*Istituto Nazionale di Fisica Nucleare, Padova*
[5]*Kansas State University*
[6]*University of Manchester*

May 2, 2016

Abstract

A Task Force, put together in Feb. 2016, has been intensively investigating status and perspectives of the DAQ architectures for the three detectors involved in the SBN Program, with the intent of pointing out commonalities in needs and timescales.
Findings, reported in this document, lead to the identification of many systems in which resources could be shared in designing a strategy, developing tools, and maintaining common components. The outcome is a strong recommendation to the MicroBooNE, ICARUS-T600, and SBND collaborations to start the activities as soon as possible in the form of common Working Groups, for which different priorities have been proposed.

1 Introduction

At the beginning of February 2016, upon suggestion of Peter Wilson, Coordinator of the SBN Program, and with the consensus of SBND, MicroBooNE and ICARUS-T600 collaborations, a Task Force was formed with the intent of providing recommendations on the path for developing online systems tools.
The mandate was to produce:

1. a set of recommendations on general approach for each of the components of online systems;
2. a description of areas where a common approach across multiple experiments would be possible;
3. suggestions of the time-lines for decisions to be made on DAQ and online systems architectures for each experiment that would influence work on common approaches;
4. an accounting of the working groups and available resources inside each experiment for working on online systems;
5. a proposed structure and plan for the coordination and communication among these working groups to aid in joint efforts.

Activities started officially with a kickoff meeting on Feb. 17$^{\text{th}}$ and proceeded with frequent meetings, informal interaction, and exchange of documentation and ideas, which allowed the Task Force to put together this closeout document in only two months.

The key question addressed by the Task Force was at which point along the whole data flow should the Data Acquisition (DAQ) systems of the 3 detectors interface with each other and be integrated into a shared platform.
Maximisation of the common parts would produce the great benefits of getting support from Fermilab, sharing efforts in installation, commissioning and maintenance (including shifting) as well as reducing systematics to be considered in data analysis.
At the same time, however, integration is not for free. Each detector should modify part of its Online/Nearline systems: in particular MicroBooNE has an already working DAQ and ICARUS-T600 could inherit the architecture adopted for its operations at LNGS, but SBND as well has already gone ahead in taking decisions on strategies for DAQ. Beyond that, new resources would need to be allocated to the development of tools smart enough to satisfy common needs while leaving room for customization.

With the intent of facilitating the process of identifying possible commonalities given the aggressive time scale of the start of data-taking by the new SBN experiments by 2018, the Task Force worked to address 4 major online systems:

- hardware of front-end and back-end electronics for all subsystems of the detectors (TPC, Photon Detection Systems and Cosmic Ray Tagger);
- triggering and synchronization, especially concerning distribution of trigger and clock signals;
- event building, incorporating data fragments from all detector subsystems;
- online and nearline tools for run control and data monitoring and data management.

Concerning the first system, the front-end and back-end electronics, the Task Force did not make specific recommendations or evaluate use of common components, but conducted a survey of the current plans and status of operations. For each of the remaining online systems, a preliminary investigation of requirements (distinguishing mandatory from desirable), available tools and infrastructures, time constraints for decisions to be taken and estimated resources has been carried out separately for the three detectors and used to feed the discussion on possible synergies.

On the basis of the collected information, reported from section 3 and onwards of this document, the Task Force has noticed not just the feasibility but also the convenience of common developments for some components of the online infrastructure. The recommendations of the task force on the development of working groups to further address common aspects of the online systems follow.

2 Recommendations

This Task Force proposes organizing common activities in several Working Groups (WG), each one relying on the participation of at least one representative from each participating collaboration together with experts of Fermilab staff. The Task Force has also given a prioritization of the main issues, taking into account both the deadlines and the present development stage. As a note, the priorities are not a determination of importance, rather they reflect the relative prioritization of working groups that would require and/or benefit from common collaboration across the experiments. In particular, high priority working groups have clearly-defined milestones that should be reached on a short time-scale, while low priority working groups may act more as a forum for sharing design developments, and do not necessarily have pressing deliverables needed in common by each experiment. The priority assigned may reflect in meetings of the WGs at recommended frequencies like weekly, bi-weekly, and monthly for high, medium and low priorities, respectively

The role of the MicroBooNE collaboration in participation with working groups is somewhat unique, given that the MicroBooNE detector is currently operating. MicroBooNE participation in each of the working groups is both welcome and encouraged, especially where the collaboration may envision an upgrade of current tools (for instance, in monitoring utilities, databases handling, and event building), but also in groups where MicroBooNE has developed particular expertise (for example, in slow controls and monitoring, and data management). Coordination on shared tools and development of common expertise could help mitigate any risk of future strains on MicroBooNE operations.

1. Triggering and clock distribution.
   The SBND and ICARUS collaborations should continue investigating independently the options for realizing their respective trigger systems. However, this WG could remain a forum for discussion on trigger system solutions, trigger signal handling, and keep open the possibility for common development.

   At the same time, the availability of a common infrastructure for clock distribution at the ns level of precision, strongly supported by Fermilab for both installation and maintenance, would be an asset. The White Rabbit option is, at present, the most appealing. The milestone for this working group should be proving the performance of the White Rabbit timing system and developing a plan for future development, commissioning, and long-term support at Fermilab by Autumn of 2016.

While updates to the MicroBooNE trigger and clock distribution system are unlikely, involvement of the MicroBooNE collaboration would be welcome.

It is opinion of the Task Force that this is a **high** priority topic.

Details on the triggering and timing systems can be found in Sections 4 and 5

2. Event building
   It is possible that all three detectors will adopt a software for data acquisition based on *artdaq* framework [3]: SBND plans to use *artdaq* as a framework for its DAQ; MicroBooNE's current DAQ software is modelled in a very similar manner but would need to introduce some modifications in what is presently used for data-taking; and, ICARUS is in the phase of validating the performance of *artdaq* compared to the DAQ used in LNGS run. However, given the peculiarities of the Front-End hardware and still in the absence of an agreement on a unique data format, development of event-building software currently may effectively proceed independently. This WG should remain a forum for discussion of possible common readout hardware components and opportunities for common event-building software (like light detection readout systems in SBND and ICARUS, and the CRT readout systems in all three experiments).

   A common effort should be spent at a later stage, once the basic functionalities will be already in place, for the development of additional features and most of all for operations and maintenance (e.g. in shifting). At this time, the working group should verify the compatibility with the other high-level DAQ tools (especially in terms of data format and message- and data-passing protocol), and remain a forum for discussion of event-building updates in *artdaq*.

   Hence, it is opinion of the Task Force that this is a **low** priority topic.

   Details on the event building software can be found in Section 6.

3. Databases
   This WG is not dedicated to a specific subsystem but to the development of tools of which many components could profit. Both ICARUS and MicroBooNE have proven the effectiveness, versatility and reliability of databases, suggesting their extensive usage, but have at the same time pointed out the need for a well-organized structure with some redundancy.

The databases we see being needed are (1) run configuration database, (2) slow controls archive, (3) data management/flow/monitoring database, (4) channel-map databases, and (5) calibration databases. The latter may be more an offline system, but it shares some structure with the other databases, and could be used in "nearline" data-quality monitoring.

The milestone of the working group is architecting a design for the structure of the above databases, clarifying the following details in particular: online/offline database access, server fail-over solutions, cached data access, intervals-of-validity, web monitoring, APIs in C++ and python that provide query/insert/update, etc, functionality. A detailed design should be prepared by Autumn of 2016 and the contribution of highly qualified FNAL personnel will be vital.

It is opinion of the Task Force that this is a **high** priority topic.

Databases are discussed in tandem with high-level online systems in Section 7

4. Run control, configuration and run history
   As noted in the Sections 7.1-7.2, the requirements for run control, configuration, and history are common across the three detectors, and so could clearly benefit from coordinated effort. However, much of the work on run control systems is dependent on the specifics of the event-building software (covered in WG 2), and the design and utilisation of the configuration database (covered in WG 3).

   This working group will be devoted mainly to the development of interfaces to the DAQ software and configuration database.

   It is opinion of the Task Force that this is a **medium** priority topic, but one which could largely be delayed until the event-building software framework and configuration database design is agreed upon. However, a general strategy for design and message-passing protocols should be in place by the end of calendar year 2016.

5. Slow controls and slow monitoring
   Given the uniqueness of the hardware, requirements for slow control are significantly different among the three detectors, but the exploitation of common tools like EPICS could be an asset and it is the recommendation of the Task Force that EPICS be explored for slow controls and monitoring.

The working group should go through the R&D necessary for the realization of the hardware components to the slow controls, and their interface with the EPICS database software before the end of 2016, in order to leave enough time for each collaboration to understand if, for instance, any modification of the existing hardware is needed for integration.

For this reason, it is opinion of the Task Force that this is a **high** priority topic.

While updates to the MicroBooNE slow controls and monitoring systems are unlikely, involvement of the MicroBooNE collaboration would be very welcome, especially given the expertise of the collaboration in using these tools.

Details of the slow control systems are discussed in Sectrion 7.3.

6. Online and nearline/data-quality monitor
   Each experiment agrees that monitoring of both detector operation features along with integrity and quality of recorded data is essential for achieving scientific goals. Additionally, the generation of automatic alarms and notification to experts is essential for maximizing duty-cycle of the detector and efficiency of data taking. Since all three detectors operating principle is that of LArTPCs, the requirements for DAQ and data monitoring systems are very similar to each other. Merging the strong points of the MicroBooNE and ICARUS solutions, with the support of a dedicated FNAL team, seems a promising approach in guaranteeing a working system while minimizing the required efforts.

   We define online monitoring as that which may be applied to every event, exploiting metadata and ancillary information (like trigger information) without requiring analysis of the waveforms, and the monitoring of the overall health of the data-taking software and hardware (buffer occupancies, data flow, errors detected, etc.). The working group should determine and develop an appropriate common framework for an online monitoring system, while leaving to the experiments the details of implementation.

   We define nearline monitoring as that which performs higher-level checks, including examining waveforms and performing some level of reconstruction/analysis. The Task Force encourages the adoption of a system could share a basic software framework for nearline monitoring (done in the online environment), and additional data-quality monitoring done in an offline environment. However, such a system can most effectively developed with a joint effort only if there is an agreement on a common data format. The working group should define such a format and a timescale on when a decision for this format should be agreed by each collaboration. Note, this data format need only be for the nearline/data-quality monitoring, and does not define the raw data format of the written events.

It is opinion of the Task Force that this is a **medium** priority topic.

Details of the monitoring systems are discussed in Sections 7.4.1-7.4.2.

7. Data management
The MicroBooNE collaboration has developed a machinery for handling data files which is operating with excellent reliability at a rate compatible with that of triggering. The architecture consists of the following elements: online/nearline computing nodes, PUBS software framework, custom python projects for task in the PUBS framework, a file catalog, the File Transfer Service, and dCache/Enstore permanent storage. The extension of this solution, already well fit in the Laboratory infrastructure, to SBND and ICARUS is envisaged. The need to architect performant online/nearline computing hardware is essential to ensuring the capability of online data management and so the expertise of Fermilab SCD should be utilized in the initial design. Fermilab should provide the support of a dedicated team (ideally the SLAM team), during SBND and ICARUS commissioning along with the foreseen 5 years of data taking in steady conditions with the BNB.

Regarding the software tools for online data management, the Task Force foresees a great advantage to having common solution both for support and analysis cross coordination. This is most important for the file catalog and permanent storage technology choices. For these, the Task Force strongly recommends the use of SAM , File Transfer Service, and dCache/Enstore across all three experiments and coordination of implementations as well. The utilization of PUBS has shown distinct advantages for MicroBooNE and the Task Force recommends that the advantages be evaluated within the perspective of development person-power support for the framework from within the experiments. This working group should define online data management tasks and conduct a thorough evaluation of PUBS to help the SBND and ICARUS collaborations develop a detailed online computing model, which is needed, at the latest, by early 2017.

It is opinion of the Task Force that this is a **high** priority topic.

Details of the data management can be found in Section 7.5.

In addition, the Task Force recommends a coordination board should take care of ensuring the synergy among the WGs and monitoring their advances in accordance with the timescale of the SBN Program. An addendum on additional considerations for common approaches outside the scope if this task force is included in Section 8.

3 Summary of Front-End DAQ Systems

In this section, we present a summary of the data readout hardware used or envisioned to be used in the three experiments. We note that while each experiment has its own TPC readout electronics, there is a potential for shared electronics in the light detection system for ICARUS and SBND, and in the CRT system for all three experiments.

3.1 MicroBooNE

A detailed description of the MicroBooNE detector can be found in [1]. An overview of the MicroBooNE TPC readout electronics is shown in Fig. 1. The signal from the wires on the TPC is shaped and amplified by "cold" electronics inside the cryostat, sent out through intermediate amplifiers located on the warm side of the cryostat feedthrough, and send into the front-end of the TPC readout boards. There, the signal is digitized at 16 MHz and then pipelined into an FPGA. There, the signal is further downsampled to 2 MHz, and is split into two streams: a continuous, zero-suppressed "supernova" stream, and a losslessly compressed (Huffman coded) "neutrino" stream. The supernova stream may be continuously read out, while the neutrino stream is read out only on the receipt of a trigger in the crate. Each readout board can handle 64 channels, and the nine readout crates contain up to 15 front-end boards each, and include a controller card for configuration and receipt of the trigger.

Triggered data is formatted in the way shown in Fig. 1. A clock signal is distributed to each crate, and a frame number corresponding to 1.6 ms windows of time since the beginning of data-taking is continuously incremented. Upon receipt of a trigger, the TPC readout electronics repackage the data from these frames, including in the data packet 1.6 ms of the waveform before the trigger and 3.2 ms after. This covers the entire drift period of MicroBooNE, and allows for the tagging of cosmic ray interactions that occur out of time with the trigger.

Each readout crate sends data to a dedicated DAQ server via a transmission module in the crate. Data is passed along the crate backplane via a token-passing mechanism, and the transmission module appends a header and trailer word before sending the data via 3 Gb/s duplex optical links to the DAQ server, called a "sub-event buffer" PC (SEB). The data is collected in the SEB by a custom PCIe card (one for the neutrino stream, and one for the supernova stream), which deposits the data to a dedicated DMA buffer. An online application collects the data from that buffer and begins the event-building processes.

The photon detection system (PDS) readout electronics operate in much the same way, with a few exceptions: full PDS waveforms are not recorded; there is no compression applied to PDS data; and four full 1.6 ms frames of data are recorded

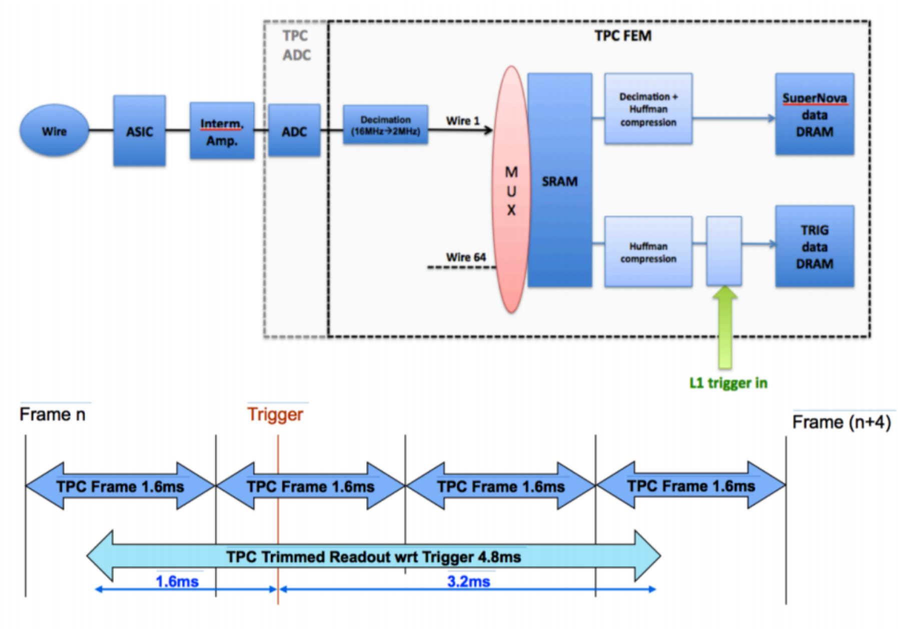

Figure 1: Schematic diagram of the MicroBooNE TPC electronics and waveform format for the triggered, "neutrino" stream of data.

with each trigger (the frame previous to when the trigger occurred, the frame in which the trigger occurred, and the two frames after the trigger occurred). The PMT signals are digitized at 64 MHz. The PMT readout electronics receive a copy of the BNB and NuMI beam signals (1$D and 1$F put in coincidence in the case of the BNB, and $A9 and $AE in coincidence in the case of NuMI, both properly timed-in), and these signals are used to determine an "unbiased" readout period for all PDS channels lasting approximately 23 μs, which covers the BNB (NuMI) beam spill period of 1.6 (10) μs. Light seen in individual PDS channels is also recorded (in smaller time window pieces) if it crosses a threshold value.

A CRT system is being installed in MicroBooNE, which will be identical to that used by SBND.

It is not envisioned that there will be further upgrades to the MicroBooNE readout hardware, and as such any shared components with the other SBN experiments, outside of the CRT system, would necessarily start after the front-end DAQ, at the event-building software stage.

3.2 ICARUS

The architecture of Front-End electronics for ICARUS-T600 detector is a continuous waveform recording based on analogue low noise “warm” amplifiers, 12 bit 2.5 MHz serial AD converters and programmable FPGAs that handle signal filtering, data storage in local memory buffers and readout.

All these functionalities are housed into one single board, CAEN A2795, serving 64 channels and plugged directly onto 2 of the 18 connectors of each detector feed-through, inside 1 of the 9 slots of a custom crate. This crate contains the fans for cooling down the electronics and distributes through the back-plane the power lines, both analogue and digital, provided by and external power supply, as well as a 1 wire serial bus, called TT-Link. The TT-Link carries a 10 MHz clock with modulated duty cycle from the board in the first slot, acting as a master, to all the others "slave" boards: it allows at the same time a 10 Mbit transmission of a set of commands and the synchronization of the 2.5 MHz ADC sampling clocks of the entire system.

Data are read out from the front panel of A2795 board through a bi-directional 1.25 Gbit/s optical link, whose physical layer is a multimode 62.5/125 μm fiber with maximum length of 200 m, equipped with a Duplex LC connector. At present the FPGA onboard is programmed to drive the optical link over a proprietary CONET-2 (Chainable Optical NETwork) protocol, handled at the other end by the CAEN PCI express board A3818 installed in a commercial PC, but this part of the readout is still open to alternative solutions. The CONET-2 protocol can stand, up to now, a 80 MB/s transfer rate with a maximum of 8 boards connected in daisy chain, which is compatible with the 47 MB/s data throughput foreseen for recording 1.6 full drift windows at each extraction of the BNB in case of an upgrade of the beam repetition rate to 15 Hz. Therefore, accounting for the 4 I/O channels each A3818 comes with, 14 PCs should be enough to equip the entire ICARUS-T600 detector.

A schematic representation of the overall architecture is reported in Fig.2.

ICARUS-T600 detector will be equipped as well with 360 PMTs immersed in liquid Argon to collect scintillation light signals, to be exploited both for triggering and reconstruction. Readout needs to be a waveform recording capable of providing a ns resolution on the determination of the interaction time, but detailed requirements for the electronics and evaluation of possible solutions are under discussion within the Collaboration at this very moment.

From the DAQ point of view it is clear that, performances and cost being equal, a CAEN board supporting the CONET-2 protocol is preferable in terms of compatibility with the TPC readout.

Similar considerations might be extended to the CRT system, where the status of the project is at the even earlier stage in which not even the number of channels

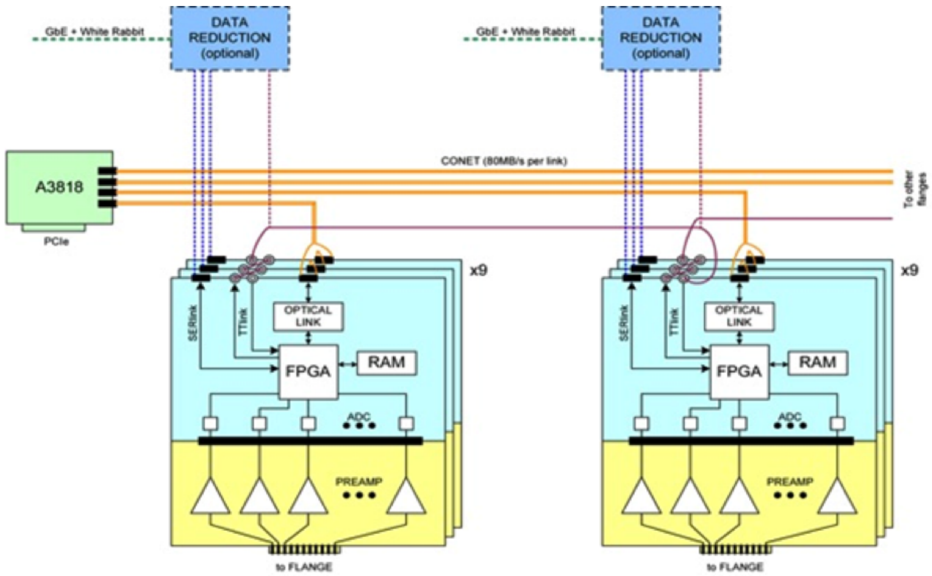

Figure 2: Schematics of ICARUS Front-End electronics.

to be readout has been defined yet.

3.3 SBND

We describe here the DAQ as it interfaces with a triggered readout. There will also presumably exist a non-triggered, "supernova" mode; we will follow the MicroBooNE model for that mode and don't discuss it further.

The cold electronics for the TPC for SBND is similar to MicroBooNE's, except in one crucial respect: the digitization is done on an ASIC on the SBND motherboard in the cold, rather than in a rack on the platform as for MicroBooNE. The in-cold digitization enables another difference. That is the heavy multiplexing of signals exiting the flanges in SBND, whereas for MicroBooNE each signal comes out on its own pair of differential pins. The SBND wire signal digitization runs at 2 MHz, as for MicroBooNE. The clock is still derived from the Nevis warm 32 MHz clock. Clock downsampling and fanout, however, are done on an FPGA in one dedicated warm box and sent to the on-flange BNL warm electronics, where the clock is distributed to the motherboard digital ASIC. The digitized signals are read out over 48 fibers per flange through the on-flange BNL-provided Warm Interface Boards (WIBs). There are a total of four flanges for the TPC readout. Thus, the warm SCSI cables that connect the flange to the Nevis electronics, as in MicroBooNE, are replaced with about half as many, far less cumbersome, fibers. This in-cold digitization also enables readout racks to be far further from the flanges.

Once the signal arrives at the Nevis crate the signals go straight to the Nevis warm electronics, where they are de-multiplexed and where huffman encoding may again take place, as in MicroBooNE. Configuration of the SBND motherboards is similar to MicroBooNE, where ASICs are configured from DAQ software. In SBND, instead of fanning that configuration out over UBS hubs, configurations are sent via ethernet to the individual WIBs, and the FPGA in the WIBs will configure both analog and digital ASICs.

The Photon Detection System (PDS) is new in SBND with respect to MicroBooNE. The PDS will consist of O(1000) PMTs and potentially many times that number of channels of SiPMs. We are designing to the PMTs, allowing accomodations for SiPMS as those decisions are made. Most important, unlike MicroBooNE, where the light and charge are readout under a common trigger and a sync'd clock, and a trigger forces common time readout of all crates, in SBND these systems are separate. Much of the PDS DAQ is under design, but we presume CAEN digitizers in a VME crate read out by three PCIe cards in a dedicated server. Digitization will be performed at at least 200 MHz, perhaps 1 GHz, unlike MicroBooNE's 64 MHz. Readout buffers are undefined, and trigger mechanisms are undefined, but we presume the broadcast to PCIe cards of trigger primitives under simple programmable conditions. We imagine that when certain thresholds on enough of the PMTs are reached pointers to buffers in the CAEN cards are held. The primitives are expected to be sent over a handful of lemos into an awaiting trigger board at the O(10) kHz range, comparable to the cosmics rate at SBND. The trigger board will arrive at a trigger decision (see section 4) based on these and other signals, and put a readout pulse on the white rabbit network which will be seen by the dedicated PDS server. A process on the PDS server that is listening for this pulse will command the VME crate via its PCIe cards to readout the buffers associated to the desired primitive.

In SBND there is also a Cosmic Ray Tagger (CRT) system. It operates on its own clock, continuously digitizing signals from the CRT bars. Seven servers are dedicated, one for each of the seven walls of light bars, to reading out its front end boards. This is done over daq NIC-like cards that sit in each machine running UDP. Each process on those servers determines coincidences among the x-y bars on each wall and sends subevents to the eighth CRT server, which runs a CRT sub event builder.

SBND will also have a laser system just like MicroBooNE's. It runs only in dedicated laser runs. This system is unique in that the laser will in fact produce the trigger, via the trigger board, rather than being subject to other inputs, like the arrival of PDS primitives or pulses indicating the arrival of beam spills.

Each of the above systems contributes an artdaq::fragment from its dedicated server, which is received and built by an event builder process running on a dedicated machine. In this way, triggered, assembled events are written to disk. Supernova mode fragments will remain on their own dedicated server until an after-the-fact process decides to assemble and write them.

The description of the triggering mechanics of SBND is left to section 4.

4 Trigger distribution and integration with DAQ

In this section, we describe the trigger systems in use or envisaged for the three experiments.

4.1 MicroBooNE trigger board

MicroBooNE benefits from both the photon detection readout system and the TPC readout system sharing the same clock and trigger distribution. A single master trigger board is therefore used to form a central trigger decision and distribute it to each of the nine TPC readout crates plus the light readout crate. This trigger board receives and mixes several trigger signals, including:

- BNB trigger
- NuMI trigger
- External (strobe) trigger
- Cosmic ray muon mini-tracker trigger
- PDS trigger (beam-spill-coincident)
- PDS trigger (out of beam-spill, i.e. cosmic trigger)
- LASER calibration trigger

Each trigger input can be masked on or off, and can be individually prescaled. During normal running, only the BNB, NuMI, EXT, and PDS trigger inputs are enabled, and OR'd. Alternatively, the BNB and NuMI triggers can be AND'd with the PDS beam-spill-coincident trigger. The latter is formed by the PDS readout system in fpga logic. In this logic, waveforms from specific groups of five (out of 32) photomultiplier tubes in MicroBooNE, having been discriminated, are checked against a multiplicity and summed pulse-height trigger condition which is active only during the beam spill. If such condition is met, the trigger marker is sent to the trigger board, along with a serial data word describing the trigger type.

The trigger board also receives and timestamps (to 64 MHz precision) the GPS 1pps signal. The board is also capable of generating its own calibration trigger as TTL output, which can be used to pulse any calibration system in a readout-synchronous way.

Trigger data is read out continually via a dedicated optical link and directly from the trigger board, containing status, trigger scalar counters, and latest GPS pps timestamp.

To allow additional flexibility and individually adjusting or mixing different timing signals, MicroBooNE utilizes a NIM bin containing additional logic. For example, in this NIM bin, early warning signals from BNB and NuMI are used to inhibit all other trigger inputs to the trigger board.

In MicroBooNE, the trigger board is used to form the first level trigger that does not use PMT information. A second-level trigger, applied in software on the event-builder, analyzes PMT information to determine if there is scintillation light in coincidence with the beam-spill period.

4.2 ICARUS option: NI crate

Trigger processing, integration with DAQ and distribution to the front-end was realised, for ICARUS-T600 operations at LNGS, in a commercial National Instrument PXI crate [2]. It handled the different trigger sources (Fig. 3): scintillation light collected by PMTs, timing synchronization with the CNGS extractions, charge signal collected on wires (GTO OR/MAJ) and test pulses for calibration. Furthermore, it was programmed with a multi-veto configuration in order to assign sequential orders of priority to the different trigger sources.
The system consisted of a Real Time (RT) controller (PXIe-8130) and two FPGA boards (PXI-7813R and PXI-7833). The RT controller implemented all the features that imply communication with external devices, such as the DAQ process or the reception of signals from beam extraction. Communication with the DAQ was implemented in handshake between the DAQ main process and the trigger manager. The RT controller also monitored the number of available buffers in the digital boards and prevented the generation of new triggers in case all the buffers are full. The FPGA boards implemented time critical processes, like the synchronization with the LNGS time, the opening of beam gate and the time stamp of each trigger. They also kept record of the trigger source and the trigger mask, monitored trigger rates from each source and controlled the overall system stability.

4.3 SBND option: possible trigger boards

SBND relies on the electronics of each detector subsystem reporting through its own dedicated server over a suitably fast network and to an event builder process running on another server. This situation is much as in MicroBooNE. Due to the new photon detection readout system (PDS) which runs separately from the TPC and CRT ones, one extra component required in SBND is a trigger "board"—effectively a box which can take high rates of digital or analog inputs, consisting of timing and potentially trigger primitive information, and make coincidence decisions. Upon such decisions, this trigger board must output both a trigger marker

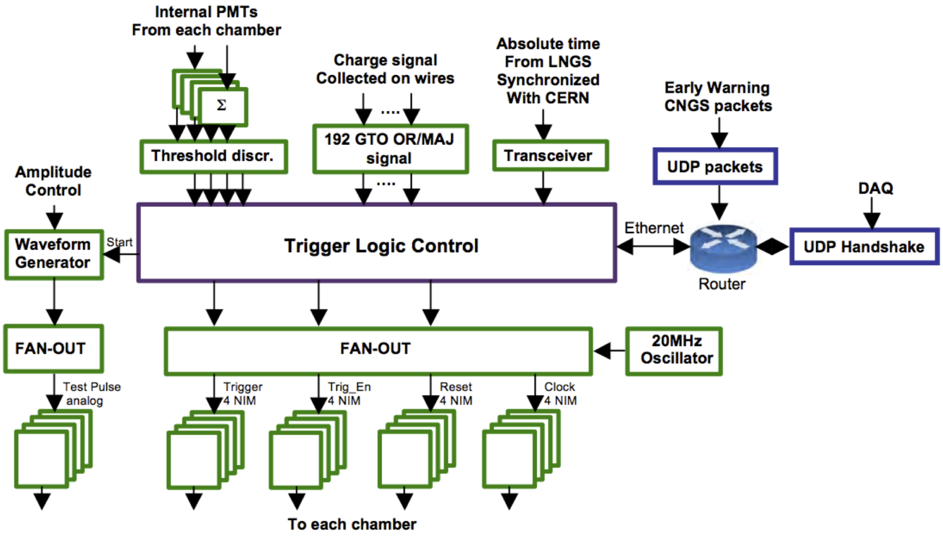

Figure 3: Schematic diagram of the working principle of ICARUS Trigger Manager during operations of T600 at LNGS.

and a trigger word packed in an artdaq::fragment. It must also distribute this trigger marker in the form of a pulse with known latency with respect to the inputs and known implemented deadtimes.

SBND is considering the use of a few possible trigger boards, namely the Penn Trigger Board, described below, the Fermilab Trigger Board, described in Sec. 4.4, and the Columbia/Nevis Trigger Board, which is similar to the one utilized by MicroBooNE, described in Sec. 4.1. The chosen board will make and disseminate the triggering decision. As of the date on this report, SBND is still in the process of developing the overall trigger and timing design, with a design review targeted for late May 2016.

Figure 4 illustrates a possible mechanism of triggering the needed subsystems, with the Penn Trigger Board (PTB) shown as an example, though other options are possible. The PTB receives the high-rate PDS primitives and the up-to-15-Hz FNAL BNB and NuMI beam signals and forms a few-Hz coincident trigger decision. If the decision is made to issue a trigger based on this information, a pulse is distributed on the WR network to arrive, for example, 0.5 msec later at the PDS server and the Nevis TPC readout system. Those systems then find the relevant data in their buffers, form artdaq::fragments from them, and ship them to the event builder.

We remind here that the SBND CRT does not require an external trigger and does not contribute to the trigger decision. CRT data fragments are requested by the event builder when a trigger is issued, and the appropriate CRT fragments are then added to the event in the artdaq::EventStore.

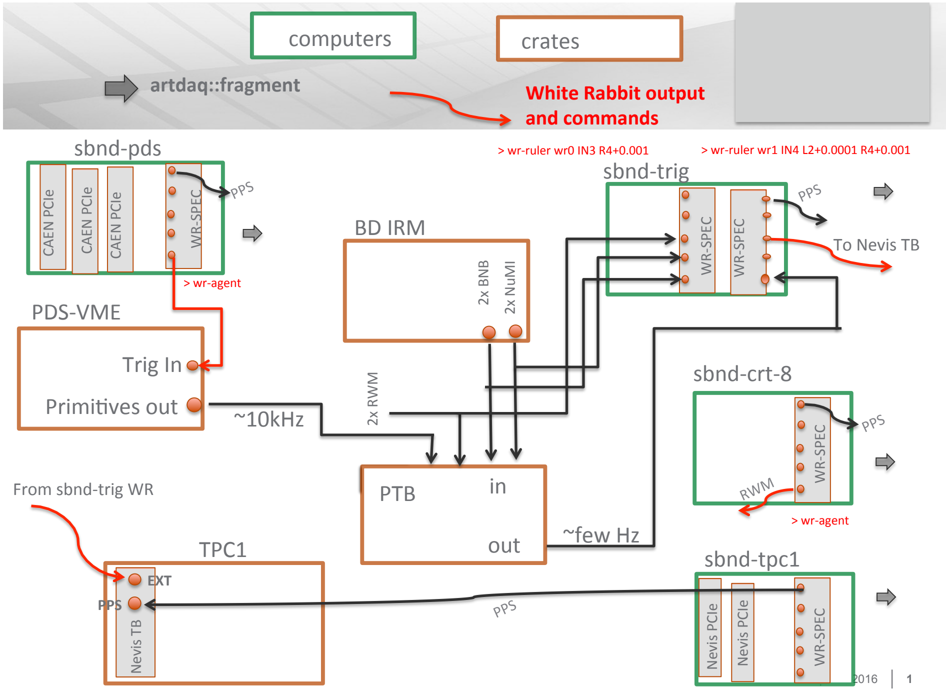

Figure 4: Schematic diagram of the SBND trigger and timing distribution. The "PTB" box could be replaced by the FNAL trigger board or some other solution. The "BD IRM" is the beams division internet rack monitor which provides the beam pulse information.

A picture of the PTB is shown in Figure 5. It has been used to run the 35-ton experiment's trigger with hundreds of high-rate, discriminated, individual PMT signals as inputs. In 35-ton, it communicates over a socket with a computer that runs an artdaq process. SBND would use it similarly without nearly so many LVDS or ECL inputs.

4.4 Fermilab option: custom design board

Members of the Real-Time Systems Engineering (RSE) department in the Fermilab Scientific Computing Division are currently developing a prototype electronics module that will provide a modern replacement for existing commercial NIM coincidence modules that are part of the Physics Research Equipment Pool (PREP) at the lab. This new module will be FPGA-based and will have user-selectable

Figure 5: Image of the Penn Trigger Board. The PTB holds a single board computer (MicroZed) with an FPGA (Xilinx Zynq) where most of the prompt trigger logic is implemented based on its many input signals. It interfaces via ethernet to a network where it finds its server and communicates with an artdaq process running there that assembles the trigger board's artdaq::fragment.

parameters, such as the threshold for the input signals and the number of signals that are required to form the coincidence, that are settable at run time using a web interface that will be provided as part of the complete package.

Based on discussions so far, it appears that the initial needs of the SBND trigger system could be met by a natural extension of this coincidence module. Additional features such as inhibiting the output signal based on "busy" signals from the data acquisition electronics, counting the number of input and output signals in a given time frame, and making those signals available to users and applications that need them have been discussed and can be added to the coincidence module design in order to provide a prototype trigger module.

The development of the prototype coincidence module has begun and some of the features needed for a trigger module have been incorporated in the design. We expect to have working firmware and web-based software for an initial implementation of this module during summer 2016. Our medium- to long-term plan is to incorporate modern, FPGA-based replacement modules, such as the ones described here, into the PREP pool and provide the necessary support for the modules on a long-term basis.

5 Clock and timing distribution

For the three SBN detectors, there can be multiple subsystems running asynchronously with each other (*e.g.* TPC, LDS, CRT, calibration systems). In order to be able to build events online, it is necessary to have some synchronization of timing among the subsystems. SBND has proposed a timing synchronization system that makes use of the White Rabbit network [?]. This network allows to disseminate, with a configurable latency, and with sub-nsec precision, the distribution of timing pulses from one server (DAQ node or subsystem) to another. Those pulses may be software decisions, or they may be actual TTL timing signals put onto the WR SPEC (White Rabbit Simple PCIe Card).

Section 4.3 described how a trigger decision pulse in SBND can be distributed on the WR network. Other distributed signals could include the pulse-per-second (PPS) from a GPS system and the RWM pulse, which gives the exact RF bucket of this spill of the 84 buckets which form the 1.6 μs BNB pulse. All subsystems can use a GPS PPS in order to sync their own clocks to a master clock, which itself is disciplined by a master GPS PPS.

The White Rabbit system has the potential to meet all needs of the clock distribution in SBND and ICARUS, as outlined in Table 6. Tests of the White Rabbit network for use by SBND are ongoing at a dedicated test-stand at PNNL, but further tests will be required to fully validate the timing distribution system. Given the significant effort for the establishment of additional test stands, development and proof-of-concept the timing system implementation, and establishment of expertise in the timing system at Fermilab to build and commission the production system, it would be sensible to coordinate effort between SBND and ICARUS.

		SBND	MICROBOONE	ICARUS
Requirements	**Mandatory**	Need to form a trigger from the IRM-provided beam signals and activity in the LDS. Trigger hardware/software needs to know the latency with which it redistributes signals to systems for readout. Some subsystems need 1nsec precision in order to match activity across subsystems.	GPS time sent to trigger board; trigger takes signals from BNB and NuMI, external calibration triggers (off-beam, laser, muon counters, etc.)	Trigger: 1) distribution of a "trigger stop" signal (freeze of front-end buffers) to ~100 TPC readout units, ~50 PMT readout boards and CRT units _ possible latiencies among the subsystems are tolerable if jitters are contained at the ns level; 2) possibility to open a beam gate relying on the beam spill extraction signal and look for a coincidence with a customisable logical processing of logical outputs both from PMT LVDS and TPC GTO; 3) handle back-pressure towards DAQ and multi-level veto in case of multiple front-end buffers. Clock: 1) the 3 subsystems (TPC, PMT and CRT) need share a stable clock at the ns level of precision (including also jitters) for assigning timestamps to the data fragments they send to event builders.
	Desired	Need GPS distributed to many detector subsystems + a trigger node with low (~1nsec) jitter across subsystems. Need 1nsec precision at CRT wrt the RWM in order to identify cosmics to beam event. LDS also needs precise timing wrt the trigger wrt which it forms and then the signal which tells it what data to read out. Trigger thus must fan out with known latency, with 1 nsec jitter.	configurable prescales for each trigger; timing fixes for TPC crate times vs. PMT	Clock: minimise the number of corrections to be taken into account afterwords to cross correlate TPC, PMT and CRT timestamps with each other (f.i. White Rabbit)
Integration	**Possible**	YES	NO	YES
	Dependencies	Clock: White Rabbit system commanding not too many developer FTE cycles. We plan to be able to develop on only the host code, and take the FPGA and driver code from Seven Solutions as frozen.		event builder for handling handshake, beam signal, sub-detectors back-ends in terms of compatibility with clock codification (f.i. WR)
Deadlines				Autumn 2016
Resources needed	**Integration**			1.5 FTE years
	No integration			2 FTE years + maintenance

Figure 6: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning trigger and clock distribution.

6 Event-building

After events are collected in DAQ servers from the front-end electronics, data from the detector must be combined to form a fully-built event, with each event corresponding to a drift period of the TPC (when triggered by a neutrino interaction), plus some buffer readout windows to aid in cosmic removal.

Requirements for the three detectors, focused on the TPC readout, are summarized in Table 1. In the end, the SBND and ICARUS experiments have similar total data volume requirements: around 650 MB/s maximum instantaneous rate of data-taking, and an average rate of a little less than 300 MB/s. These values are maximum requirements on DAQ performance for commissioning and early running: in reality, the total data rates and volume should be reduced substantially with the application of a light-dectrion-system+beam trigger.

	SBND	MicroBooNE	ICARUS
TPC channels (total)	11500	8256	53000
TPC Readout crates (total)	~ 10	9	~ 100
Number of TPC DAQ servers	~ 10	9	(under design/test)
TPC digitization	2 MHz, 12-bit	2 MHz, 12-bit	2.5 MHz, 12-bit
TPC drift(readout) period (ms)	1.3 (3.8)	2.3 (4.8)	1.0 (1.6)
TPC data size event (uncompressed, MB)	220	150	330
TPC data size event (compressed, MB)	45	35	40
Max instantaneous data rate (MB/s)	675	450	620
Max average data rate (MB/s)	270	180	250
Number of event-builder servers	2	1	(under design)

Table 1: Summary of the requirements from the three experiments for links/event-building.

A typical data-flow in a DAQ cluster is shown in Figure 7. Sub-event buffer DAQ servers (SEBs) receive data from detector elements, usually after some triggering of the readout electronics. The data is packaged by the SEBs and routed to event builders, where data fragments carrying the same event identification are combined together to form an completed event. Event builder nodes, with access to the full data, can perform additional triggering or filtering algorithms, and can perform further data compression if appropriate. Data is then routed to aggregator nodes, where the data is written to local disk and then managed by a data-management process to be sent to tape storage facilities. Data after the event building can also be routed to monitoring programs, which can do basic data quality checks and provide live displays of the data. The planned design and requirements of each of the three SBN experiments conform to this basic scheme. Thus, it would be possible to share a base DAQ event-building framework, like *artdaq*, the data-acquisition software framework being developed by FNAL SCD.

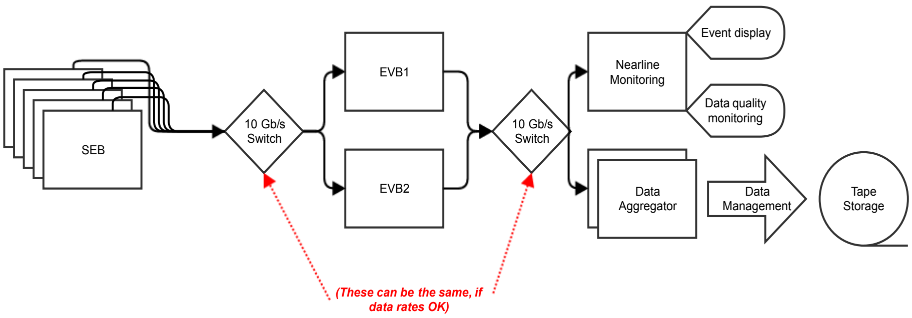

Figure 7: Typical data-driven flow of data in a DAQ cluster. Sub-event buffer (SEB) servers receive data from parts of the detector and send them to event-builder nodes. These nodes combine data across the detector, can perform simple filtering or compression algorithms, and send the data to aggregators which store the data, and to monitoring nodes.

One complication is ensuring that data fragments carry identifying information that can be used to build events. This is usually simple within a detector sub-system, as local time stamps or event numbers are often appended to detector data by readout electronics. However, when combining data across sub-systems, this is not guaranteed. Each experiment anticipates having a distributed GPS time that could be used to combine data from each detector component, but this GPS time may not exist in the data fragments themselves (needing to be calculated based on internal time stamps or correlation with different detector components).

Additionally, while each detector will have different readout electronics used in the TPC readout, the detectors may share similar electronics for the light-detection and cosmic-ray-tagger systems. Thus, it may be advantageous to design self-contained DAQ systems that can be used for those detector subsystems, and allow for an additional "global" event builder that can combine the data from each subsystem.

Both issues can be addressed by developing a global-event-building model, where individual sub-systems (TPC, PDS, CRT) have their own SEB and event building processes. After getting all data from a subsystem, it can be passed to a buffer node, where it may then stream to a global event builder either automatically or upon request. Requests may include lookup for particular data fragments: those that correspond to the data from other subsytems being built. Fully built events, after the global event builder, follow the same route as previously explained. This design would allow for addition of data from subsystems with some delay/time for determination of matching criteria, and would allow for re-use of DAQ software across experiments that use the same readout electronics. Subsystem event builders could also send data to monitoring programs and/or to a local disk for temporary storage.

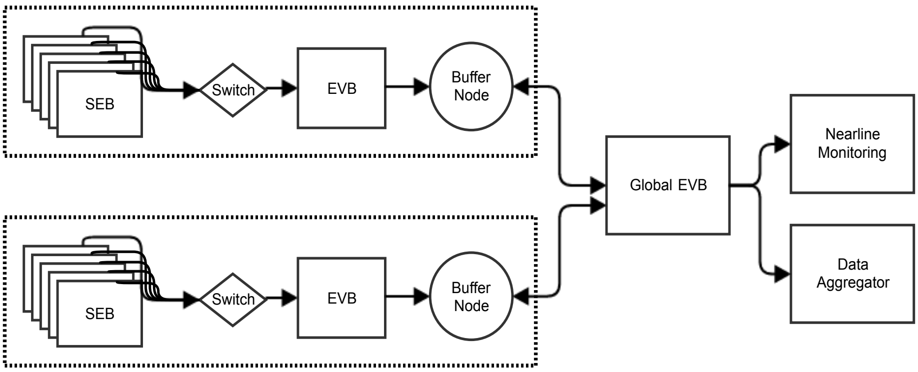

Figure 8: Idea for a global-event-building model. Individual detector subsystems have a full-fledged DAQ system designed (dashed lines), with buffer nodes in which complete events may be kept. These buffer nodes may stream data to a global event builder by default or upon a data request.

Given the similar nature of data flow and similar data volume requirements among the three detectors, it is sensible to pursue development of DAQ software within the same framework. This may allow for greater resources in developing new features, broaden DAQ expertise across the experiments, and allow for re-use of common components.

7 High-level DAQ and online systems

The "high-level" DAQ and online systems refer to the set of monitoring, configuration, and control systems that accompany the operating and validating of the detector data flow. Both MicroBooNE and ICARUS, with the experience of operating detectors, have more developed systems available to be used that share similar requirements and design decisions. It is expected that the development of upgraded tools for running as part of the SBN program could be done in collaboration, as both the mandatory and "desired" requirements for this system are shared.

7.1 Run Control

A collection of the mandatory requirements for a run control system that is shared among all three experiments is:

- automated starting, stopping, and restarting of normal data-taking runs;
- a central alarm console, with visual and audio alarms, and the ability to automatically email or page experts on alarms;
- offsite access for both offsite control rooms and expert monitoring/interaction;
- automated electronic log-book entries; and,
- accessible configuration options

In addition, there were a number of "desired" components that would be beneficial:

- a graphical user interface;
- an expert view for log files; and,
- a minimization of necessary shifter operations via automated workflows; and,
- a similar control interface for "test stand" and commissioning test environments as the final data-taking environment.

While both MicroBooNE and ICARUS have successfully operating run control/shifter interaction systems that meet their necessary requirements, each experiment could benefit from improvements and a re-design of some components, and a common interface across the SBN program would enable easier support for operations. In the design of an improved system, both MicroBooNE and ICARUS could offer their experiential expertise on major issues.

		SBND	MICROBOONE	ICARUS
Requirements	Mandatory	Need to config/run/stop move DAQ through fsm Must interact w runConfig db.	Automatic interface to FNAL elog; automatic starting/stopping/restarting of runs; visual and audible alarms; configurations and history per run stored in database; easy to upload/create new configurations; log files stored	Possibility to configure trigger, sampling and portions of detector; robustness in case of problems
	Desired	GUI or functional/intuitive script interface	GUI, visual status; expert view of log files; run sequencing	User-friendly graphical interface, possibly collecting all configurabile parameters in the same place and assigning to them clearly understandable name tags. Fast procedure to stop/start runs, minimising the total number of operations to be handles by shifters.
Integration	Possible	YES	YES	YES
	Dependencies	Fsm constraints	Interface to DAQ software (message passing system), and access to the DAQ machines	Event building software and trigger controller system + storage.
Deadlines		Spring 2017	None?	Before beginning 2017
Resources needed	Integration		3 FTE months	6 FTE months
	No integration	4 FTE months + maintenance	1 FTE year + maintenance	1 FTE year + maintenance

Figure 9: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning run control.

7.2 Configuration and run history

In addition to the run control, the configurations used for each run, which may cover both software and hardware settings, need to be specified, updated, and stored. History on these configurations should be maintained throughout the experiment. Typically, this has been handled in the experiments through use of a configuration database.

The basic requirements generated for the SBN experiments are:

- a database storing run configurations, linking each run with a specific configuration;
- ability to create/upload new configurations;
- ability to search configurations/runs both chronologically and by grouping based on configuration type;
- configuration classes/labels, like expert/non-expert, physics, calibration, etc.;
- mapping of configurations into user-friendly names or labels;
- fully integrated with the DAQ/run control (for instance, the translation of configuration database elements into appropriate formats for the DAQ software/run control to handle);

- a channel-mapping database to correlate readout channels with physics components (wires, PDS elements, etc.); and,
- access to all configuration/mapping databases in both online and offline environments.

There has been work from both FNAL Neutrino Division (ND) and Scientific Computing Division (SCD) on developing configuration management tools that meet these requirements. A common toolkit that was developed by the SBN collaborations with experts from Fermilab would ensure stable and reliable operations both online and offline and improve the quality of the tool.

		SBND	MICROBOONE	ICARUS
Requirements	Mandatory	Maintain/create/reproduce configurations for all needed runs : cold electronics settings, fcl-config, etc. Need django-like http-served access to this db from grid jobs.		Permanent storage of run conditions and performance (output of online monitoring), with the possibility to switch from chronological view to grouping similar configurations
	Desired	reasonably lean database with good interface and APIs		Possibility to automatically cross-check info with the ones stored in elogs, generating alarms in case of discrepancies
Integration	Possible	YES		YES
	Dependencies	artdaq fcl presumed, but any ascii txt file may be used		Online monitoring, logging of run control
Deadlines		1/1/2018		Late 2017
Resources needed	Integration			2 FTE months
	No integration	3 FTE months		3 FTE months + maintenance

Figure 10: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning configuration and run history.

7.3 Slow controls

The detector control and monitoring system (also called slow controls) provides real-time displays of the status of the detector systems and relevant surrounding conditions, records a history of all status for later browsing and analysis, provides alarms and warnings based on limits set for each reading, and allows control of certain devices such as power supplies.

The MicroBooNE experiment uses the Experimental Physics and Industrial Control System (EPICS) for the basic device I/O, network, and data structure layers of its control system, and Control System Studio (CSS) for the graphical interface, archiving engine, and alarm system, along with some custom software used for harvesting data from external systems. Direct access to the controls and graphical displays is not possible except by logging in to a Kerberos-secured MicroBooNE computer located in LArTF. A mirror of the SQL database containing archived data can be accessed from computers on the Fermilab network. Figure 11 shows a block diagram of the parts of the MicroBooNE slow control system.

The requirements of all three SBN experiments are similar, as shown in Table 12.

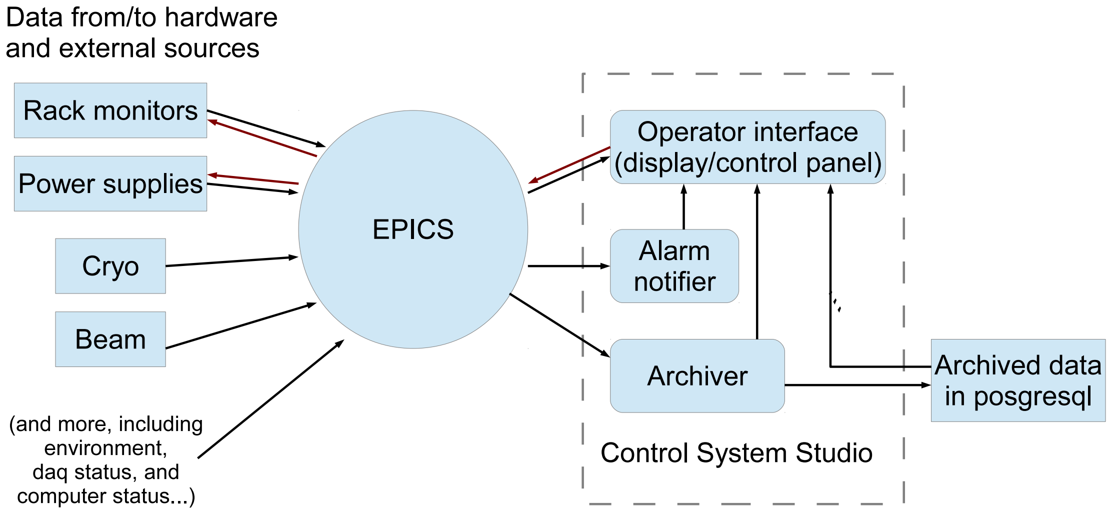

Figure 11: A block diagram of the MicroBooNE slow control system based on EPICS and Control System Studio.

		SBND	MICROBOONE	ICARUS
Requirements	Mandatory	Monitor/Archive/Alarm on detector/rack/environment hardware, and also harvesting of some DAQ quantities for display/monitoring. Need django-like http-served access to this db from grid jobs.	Display/archive/alarm on detector/rack/environment hardware and selected DAQ metrics, Cryo data, and IFBeamData. Control certain detector hardware, especially power supplies, with high security against external access and accidental control. Gui interface for display, control, plotting, alarm handling in control room. Offline access to archived data.	Record in real time: - temperatures of readout board (register accessible through optical fiber?), -TPC wire currents, -HV current and voltage, - functionalities of power supply for readout electronics with the possibility to notify alarms and take actions (such as power cut).
	Desired	for data integrity portion of this task, we like the model in which we connect up to shm segments with a process that makes histograms, to maintain orhogonality of tasks, but don't insist on it.		Integration into online monitoring system
Integration	Possible	YES	YES	YES
	Dependencies		EPICS, Control System Studio (or custom interface)	Layers carrying needed info
Deadlines		11/1/2017		Late 2016 for TPC wires power supply, Before beginning 2017 for the rest
Resources needed	Integration			6 FTE months
	No integration	6 FTE months		1 FTE year + maintenance

Figure 12: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning slow controls.

7.4 Monitoring

Monitoring of the data flow, detector health, and data quality is an essential function of the online systems. The experiments agree on the basic requirements for both online and nearline/offline data-quality monitoring:

- capability for automatic detection of errors during data-taking;
- a centralized alarm handler, giving the shifter one location to check for live detector problems;
- live updates of DAQ raw data metrics, monitoring each step of the data flow for both buffer occupancies and throughput rates;
- a nearline monitoring system that experiments write/incorporate higher-level data-quality checks that could include basic data reconstruction on a selection of collected data;
- a web-based interface for looking at higher-level monitoring histograms;
- an automated comparison of high-level monitoring histograms to a reference histogram;
- storage of recorded and processed information on database for offline access.

7.4.1 Online monitoring

Both ICARUS and MicroBooNE made extensive use of monitoring metrics to track the flow of data from the readout electronics through event-building, writing events to storage elements, and transfer of data across storage elements. Experience from both experiments shows it is feasible to put basic raw metrics on data flow into databases, and that those databases can give practically-live updates. Both experiments were also able to extract information from these databases to form alarms.

Because of the similarities in needs and previous strategy, if the same software framework is used for the DAQ software development in the SBN detectors, it would be natural to use the same online monitoring system. *artdaq* has the ability to log custom monitoring information into a variety of databases, including EPICS and ganglia. The ganglia monitoring software, in addition to allowing custom metrics, includes a number of basic cluster monitoring metrics that have proved valuable in MicroBooNE's experience (e.g., load on CPUs, disk usage, network traffic monitoring, and more). These quantities have proved valuable for improving the quality of the services provided by the Scientific Linux Architecture and Management (SLAM) team, as they can more easily be alerted to problems on the servers.

We recommend that the three experiments work, where possible, to incorporate a similar online monitoring framework, that includes the same system-level information to allow for easier system management.

		SBND	MICROBOONE	ICARUS
Requirements	Mandatory	Monitor: - DAQ perfomance, - data quality, - DAQ electronics	Live updates of DAQ raw data metrics; history stored in database; ability to alarm on values; easy creation of custom metrics/values to monitor; single user interface for all online DAQ pieces; automatic error handling (e.g. stop run)	Visualisation in real time of DAQ status, with the possibility to automatically generate alarms to be circulated by email and/or sms in case of problems and to take specific actions (f.i. disable trigger sources, restart run etc). Ex: detector status, subsystems included in data taking, portion of readout units effectively active, last collected event, beam status, trigger source, trigger rates, dead-time... All of this should be accessible without touching the raw data.
	Desired		shifter and expert specific views; auto-paging of experts;	Possibility to automatically generate entries into elog
Integration	Possible	YES	YES	YES
	Dependencies	Ganglia + artdaq? Actually no architecture has been strongly defined or approved yet.	DAQ software	Metadata to be provided by event building processes on easy-access platforms (databases?), run control and slow control
Deadlines		4/1/2017		Before beginning 2017
Resources needed	Integration			3 FTE months
	No integration	8 FTE months		6 FTE months + maintenance

Figure 13: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning online monitoring.

7.4.2 Nearline/data-quality monitoring

In addition to online monitoring to monitor the health of the DAQ system, there is a need for higher-level checks to monitor the quality of the data. For a high-data-rate experiment, this can be done by taking a subset of the collected data and running simplified data reconstruction algorithms over that data.

Because each of the experiments use the same basic components (a LArTPC, light detection systems, and a cosmic-ray-tagging system), many of these data quality checks could be very similar and high-level checks developed for one detector could be applicable to another. However, there will be experiment-specific checks that will need to be done. Essential to a maximally useful nearline monitoring system is the ability for collaborators on the experiment to be able to contribute new checks on the data, especially during periods of commissioning and during early running of the detector.

A shared, basic software framework for generating basic checks (with output histograms) using a subsample of data from the detector would serve to both allow many within an experiment to contribute data quality checks, and would allow multiple experiments to be able to generate their own data monitoring and potentially share monitoring algorithms. As an example, *artdaq* provides online monitoring facilities that are based on the *art* framework, where user-written modules can be run over fully-build events to produce monitoring histograms. It would be advantageous for the SBN experiments to work towards developing and using a common system like this as the basis of data quality monitoring. A precondition for this work is to agree on a data format for analyzing and displaying the detector data and results of monitored quantities.

Additionally, it is important that histograms created in data-monitoring be easily viewable and comparable to reference histograms. Again, given the shared needs among the experiments, it could be advantageous for the SBN experiments to work together to develop and use a common system for viewing data-quality histograms and doing automated comparisons against references.

		SBND	MICROBOONE	ICARUS
Requirements	Mandatory		Web-based interface accessible worldwide; historical logging of monitoring histograms; comparisons to reference histograms; ability to prescale data samples to monitor; ability to monitor PMT and TPC separately;	Real time monitoring of: - DAQ performance (data integrity and their consistency with metadata), - readout electronics performance (signal to noise, gain, dead channels, etc), - LAr purity, - trigger efficiency, with the possibility to generate alarms and take automatic actions
	Desired		automatic alarms/expert notification on problems; easy development/addition of new monitoring histograms; automated quality checks	GUI
Integration	Possible		DIFFICULT	YES
	Dependencies		DAQ software, data format, offline software,	Data format and algorithms for evaluating data quality; event builder or whatever source of data/metadata to be used in DQM
Deadlines			3-4 months	before beginning 2017
Resources needed	Integration		?	6 FTE months
	No integration		9 FTE months	18 FTE months + maintenance

Figure 14: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning nearline and data quality monitoring.

7.5 Online Data Management

For the purposes of the Task Force, Online Data Management scope is defined as the processing and tasks associated with data files once those files have been written to disk by the experiment DAQ. This can includes tasks such as transferring files to permanent storage, cataloging the data files, assuring data integrity, and portions of data quality monitoring. Once these task have been completed, the data should be available for offline processing and in non-volatile storage with redundancy. Robustness of the Online Data Management system is considered critical since failures can potentially lead to DAQ downtime or data loss. In order to achieve greater robustness, it is the recommendation of the Task Force that the Online Data Management tasks should be limited to essential processes only since CPU load and hardware limitations can create bottlenecks and impact operations. As well, the specifications of online/nearline computing hardware ensure that no bandwidth limitations are created that could impact detector operations. Ideally, the detector readout will only be limited by FrondEnd DAQ rates.

7.5.1 MicroBooNE PUBS System

The MicroBooNE Online Data Management is built around the PUBS (PostgreSQL UBoone Software) framework and database engine. The PUBS software framework is designed to perform a sequence of tasks for every DAQ file declared to the framework. The PUBS framework allows for the tuning of the amount of parallel processing, refresh rate for each task, and the order of operation within queued files. Using what is essentially a Directed Acyclic Graph of task written in python, files processed by the PUBS framework are passed through python projects along a dependency tree. Within each project, a file may have multiple states depending upon logic built into projects. The state and dependency information for each file in a project is permanently kept within the project tables in the PostgreSQL database server. The record keeping within this PostgreSQL database has been found to be important for diagnosing data integrity issues and allows for quick evaluation of file status and processing success or failure. The interface of a monitoring GUI to the PUBS database has been an important feature of shifter responsibility. The Task Force recommends that the experiments consider the benefits of utilizing PUBS within the perspective of limited support from Fermilab SCD for the software.

7.5.2 MicroBooNE Online Tasks

MicroBooNE performs a series of six tasks for each file produced by the DAQ system with projects run on two online nodes. In the case of MicroBooNE, a PUBS daemon is run on each of the event builder node and the nearline processing node launching appropriate python projects when needed and connecting with the PSQL database server run on online database node. Initially, the checksum for each binary files is recorded for future reference when transferring the file from one storage element to another storage element. The checksum is stored in the PUBS database as part of this first project status information. Next, the metadata for the file is generated for use when cataloging the file in permanent storage. After registering the file within the file catalog, the file is transferred off of the online system to dCache storage element. The checksum from the dCache storage element is compared with the original checksum, and if valid, the file is removed from the online system. These are the basic tasks performed with the PUBS framework, and once they are complete, each DAQ file has been removed from the online system and has been passed to data storage. This is seen as the minimal set of tasks needed for MicroBooNE to properly ensure that data is stored and cataloged for later analysis offline. The Task Force recommends that each experiment develop a similar minimal set of tasks needed for online data management in consultation with the experience of MicroBooNE and ICARUS.

While these tasks are easily enumerated, the actual CPU load and disk I/O associated with each task can be considerable. For examples, during unbiased commissioning readout of the MicroBooNE detector, DAQ write rates were in excess of 200 MB/s and exceeded the capacity of the RAID array used for online staging of data files. This limitation had a considerable impact on operations and the availability of data for offline analysis. Based on the information in Table 1 for SBND, even with 2 event builder machines the MicroBooNE implementation would not meet requirements. The Task Force recommends that the experiments draw upon the experience of MicroBooNE and the expertise within Fermilab Scientific Computing Division (SCD) to design nearline computing and networking to meet the requirements for Online Data Management tasks. While having similarly configured nearline computing nodes is not a requirement, a common design is highly recommended by the Task Force as this would allow for shared support from the Fermilab Scientific Linux & Architecture Management team.

7.5.3 MicroBooNE Offline Data Storage

The MicroBooNE Offline Data Storage is based upon three components: File Transfer Service (FTS), Sequential Access via Metadata (SAM), and dCache/Enstore storage elements. The File Transfer Service (FTS) is a service developed and supported by Fermilab SCD that will transfer files based upon selection criteria from a dropbox to a storage element. The service is extremely robust with no data loss after transferring more than 3 PB of MicroBooNE data. The MicroBooNE FTS has experiment-developed rules for multiple file formats with varied final storage locations from multiple dropboxes. As well as local storage, FTS has the ability to transfer files to storage elements located anywhere in the world to include CERN, universities, and other US national labs. FTS can be used in conjunction with SAM or as a standalone service, but MicroBooNE has chosen to utilize FTS with

SAM. MicroBooNE has configured a dropbox to be located on Fermilab's dCache Public Scratch space in order to stage files for transfer to permanent storage. The limitation of dCache Public Storage is that this volume is shared by numerous experiments at Fermilab and so files have a limited lifetime. The oldest file lifetime within dCache Public Scratch at the writing of this document was 43 days and so FTS transfers must be performed within this time period. Fortunately, FTS transfers at MicroBooNE have never exceeded a wait of more than three days. The Task Force recommends the utilization of FTS for file transfer of data files to permanent storage.

Serial Access via Metadata (SAM) is a file catalog and retrieval service developed and maintain by Fermilab SCD that is used by MicroBooNE track and deliver all of the experiments production data. SAM metadata can be used to select files from different data streams, software versions, and low-level quantities such as run number or creation date. SAM supports using only the native metadata within the experiment database server or can access external databases to incorporate additional information. MicroBooNE takes full advantage of the SAM service, and has designed the data access model for offline production around catalog and transfer tools provided by SAM. The SAM file catalog is currently used by all neutrino experiments with great success. The Task Force recommends utilizing SAM for all three experiments as this will provide the ability to seamlessly access correlated datasets from the different detectors.

Once files are registered within the SAM catalog and transferred to the FTS dropbox, data files are then put into permanent storage within the Fermilab dCache/Enstore system. Enstore is the magnetic-tape, storage system maintained by Fermilab SCD with redundancy and separate tape libraries located in different buildings on campus. Enstore is accessed through a dCache hard-drive, staging system that is maintained by Fermilab SCD. This combined dCache/Enstore system is at the backbone of all data storage for all of the Fermilab neutrino experiments, Fermilab CMS Tier 1, and the Tevatron experiments as well. While other storage technologies would be possible, the cost to an SBN experiment to implement and maintain something other than Enstore makes this option impractical. The Task Force recommends that all experiments utilize Enstore and Fermilab redundant facilities.

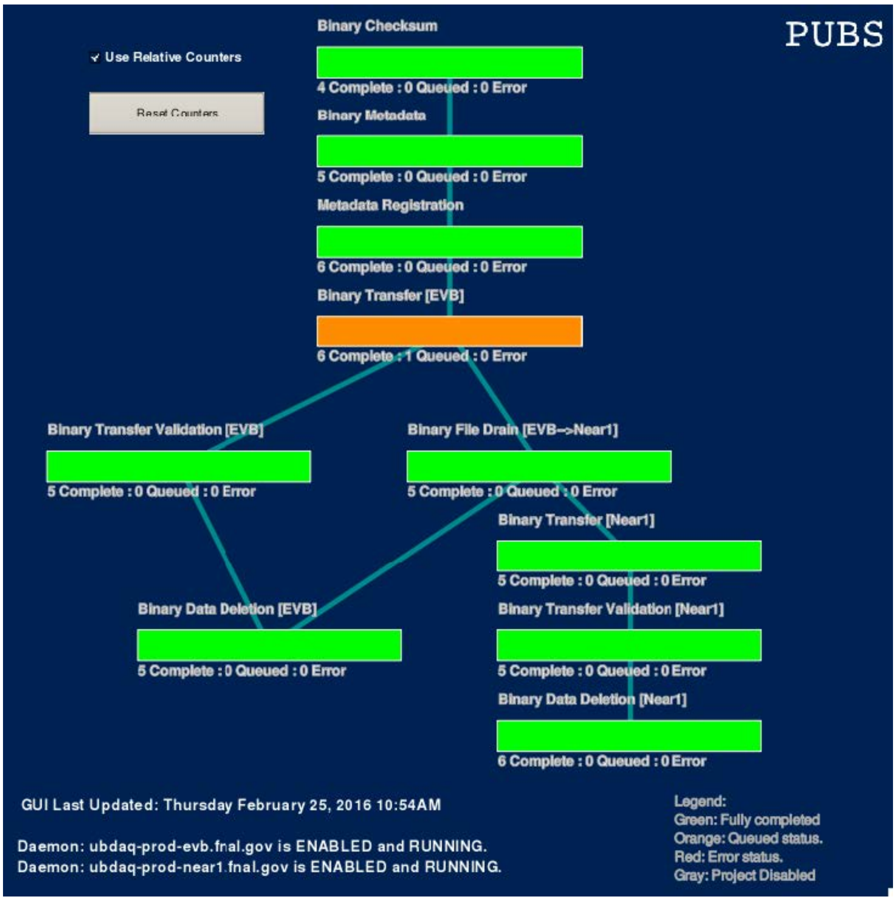

Figure 15: Online monitor of the PUBS workflows for MicroBooNE. The diagram shows the tasks that are performed to transfer the data files from the online system to the File Transfer Service dropbox. As well, tasks generate metadata for file cataloging and checksum values for assuring the final files validity.

		SBND	MICROBOONE	ICARUS
Requirements	Mandatory	Move 1GB/sec artdaq raw data files out to SAM/Enstore, convert to artroot, beamdaq-merging, split out to filestreams, reco processes running on each. SAM db service is adequate.	move 400 MB/s compressed raw binary files to Enstore; register all files with SAM; validate transfer to FTS dropbox; files available for offline processing within 12 hours; monitoring of file status and alarms	Permanent storage of raw data, keeping at least 2 copies of the same file from the most upstream possible in data processing chain, live updates of data location and processing they have gone through
	Desired	1 GB/sec raw flow -- database state machine, GUI interface, alarmed monitoring, shifter monitoring of status too	convert to artroot at 100 MB/s; beamdaq merging; split trigger streams; reconstruction processing at 100 MB/s; calibration, alignment, special runs	Tools for having easy access to data for non-expert users (f.i. without having to mount filesystems through ssh tunnel)
Integration	Possible	YES	YES	YES
	Dependencies	SAM, dCache, presumably, but would not mind seeing a more multi-site distributed storage/computing model supported here.	SAM, dCache, Enstore, FTS	FNAL infrastructure
Deadlines		1/1/2018	NOW	Late 2017
Resources needed	Integration			6 FTE months
	No integration	8 FTE months	3 FTE months	6 FTE months + maintenance

Figure 16: Requirements and deadlines for SBND, MicroBooNE and ICARUS concerning data management.

8 Addendum on additional activities

The Task Force has considered and discussed also a few collateral issues beyond its original scope but strictly related to it.

For instance, at some point each Collaboration will be needing a test stand dedicated to DAQ where to try and put together the systems developed by the several WGs. It seams perfectly reasonable having one collective infrastructure, not least for optimizing logistics and resources committed to the support of the common tools and possibly for sharing part of the hardware (such as timing distribution and maybe PMT readout electronics). Even though it might be premature to set up these facilities right now, the Task Force strongly recommends taking into account these findings and identifying as soon as possible a proper location.

The Task Force is also convinced that the SBN Program should profit of Fermilab computing support beyond the specific tasks codified into the working group activities: namely in System Administrators, networking and every aspects related to equipment procurement, installation, configuration and maintenance.

Another important topic is the distribution of the beam extraction signals and the interface with other systems in the Accelerator Division. As an example, it should be verified that fibers from the Resistive Wall Monitors (RWM) to all the experimental halls already exist, and that the electronics presently used to send the RWM signals to MicroBooNE is enough to serve SBND and ICARUS detectors as well. It is opinion of the Task Force that it should be clarified if this is under the responsibility of the Installation and Integration team, otherwise another Working Group should be appointed. In general, the understanding of the beam signals and their handling should be shared across the collaborations.

Finally, the Task Force notes with pleasure that there have already been discussions on commonalities between the online systems development for the SBN program and what will be needed for ProtoDUNE and DUNE. In these discussions, common points have already been identified, and we hope the structure envisioned here for the SBN program to collaborate on common systems may also serve as a model for further collaboration and development across collaborations. We hope that strong communcation between the SBN and DUNE programs continues, and that we work to benefit from shared resources where feasible.

References

[1] MicroBooNE Collaboration, "The MicroBooNE Technical Design Report," http://www-microboone.fnal.gov/publications/TDRCD3.pdf, 2012.

[2] M. Antonello *et al.* [ICARUS Collaboration], "The trigger system of the ICARUS experiment for the CNGS beam," JINST **9**, P08003 (2014) doi:10.1088/1748-0221/9/08/P08003 [arXiv:1405.7591 [physics.ins-det]].

[3] K. Biery, C. Green, J. Kowalkowski, M. Paterno and R. Rechenmacher, "artdaq: An Event-Building, Filtering, and Processing Framework," IEEE Trans. Nucl. Sci. **60**, 3764 (2013). doi:10.1109/TNS.2013.2251660