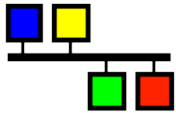


## *Issues in Accelerator Control*

---

Rok Sabjan

Based on presentation by Bob Dalesio



## *Outline*

---

- **Phases of a project**
- **Time stamping / event distribution**
- **Personnel and equipment protection**
- **Selecting I/O**

**Budget and Design – At this early phase of a project one should**

- identify all difficult requirements**
- determine which tools most closely meet these requirements**
- start to design and create those things that are not available**
- start to extend and improve tools in ways that will be useful**
- set up a relational database and tools to modify and use it**
- make/adopt a naming standard and start to name things**
- set up a relational database to keep standard names**
- start to identify people to use on the project**
- did I mention that now is the time to implement the database**
- select I/O and write and test the drivers – and hardware**

**Subsystem design, manufacture**

**Build test stands to support early integration of components**

**Support component test through archiving and automation**

**Identify all cross system requirements**

**Identify all external interfaces to each subsystem**

**Start to track components through the relational database**

**(If you forgot to do the RDB – too late)**

**Document requirements by subsystem**

**Document intended use / operation of subsystem components**

**Start to build end-to-end test stands**

**Make wiring lists**

**use RDB tools to build list of names that meet standard**

**Finish up any drivers/tools/special software**

## **Installation**

**all major development should be complete**

**your RDB is either a big help – or non-existent**

**major effort is engineering acceptance of components**

**operational requirements should start to be mentioned**

**archiving all parameters at this phase for vendor interactions**

**point to point tests**

**subsystem automation for burn in and calibration**

**development is aimed at commissioning tools and automation**

## **Commissioning**

**integrated tests of all subsystems**

**programs for conditioning needed.**

**avoid the critical path**

**keep archiving lots of data – even if no one asks for it**

**collect requirements for operational issues – like**

**operator screens**

**archive conditions**

**alarm limits**

**protections**

**make that database now and re-enter all your data**

## **Operation**

**Fix all problems immediately – no beam loss charged to you**

**Fix all of the things that were not done right from the last pages**

**- with less than half as much money**

**- with less than half as many people**

**- with the major contributors to your woes gone to another  
project to make someone else miserable**

**Replace the equipment that was chosen poorly**

**Learn the complexities to keep this beast operational**

**Start to deal with users and their issues**

## *Time Stamping and Event Synchronization*

---

How accurate of a time stamp is needed?

How fast can the process changes?

What analysis capabilities are required?

Time stamps are highly hardware dependent -

no hardware - bad synchronization

remote I/O from PLCs does not provide time stamps

Time stamps are also a function of you operating system

windows can have OS delays longer than 1 second

linux can have OS delays longer than 300 msecs

realtime OS have delays that are in the usec range

## *Time Stamping and Event Synchronization*

---

Time stamps are placed on data at the time a record is processed

In the database you can select to get a time stamp

- from the operating system which is loosely synchronized

- from an event queue where multiple time stamps are kept

- from device support – where the hardware provides its own timestamp

- from any other record - only input records have access to h/w timestamps

Allows data to be correlated in the channel access clients or archiver

Supports synchronized outputs – but this requires some pre-loading and verification

No good support exists for event flavor - yet (send data only from a given pulse id)

## *Personnel and Equipment Protection*

---

Personnel protection should never include a computer. PLCs were given some exception to this. New, multi-threaded PLCs should be viewed suspiciously.

Fast beam shutdown to mitigate beam induced damage requires a protection system outside of the control system. This is frequently a high speed carrier going from between all equipment locations where any input into it going low will drop the carrier and stop the beam in the low energy transport. The target time for shut off at SNS is 35 usec. The machine protection also latches all faults with an offset counter to identify an order of failure for fault analysis.

Most equipment protect itself. Vacuum gauges turn off or close valves when pressure is out of range. RF protects windows from arcs.

Slow process changes can be protected in software – for instance – a power supply may be turned off to protect it before it is hot – by detecting a loss of flow. Even across the network between two machines this is feasible. However, it must be configured to be fail-safe. An active heartbeat between computers is necessary that is set beating by and through the same software path needed to make up the interlock.

When implementing interlocks in hardware or software – make sure to disconnect everything that you depend on and make sure the system responds as you expect. I learned that a disconnected vacuum gauge reads  $10 \text{ e-}12$ . Unbelievably good!!!

## Selecting I/O - Factors

---

Precision

Data Rate

Response time required

Distribution of transducers

Density of the transducers

Ease of installation/replacement

Cost

Vendor support / experience

Externally Triggered

Reliability required

### **VME/VXI based**

High Precision >16 bits

MHz ADCs

1 KHz scan rates

Good for high density

Installation/wiring moderate

High Initial cost

Multiple vendors

Can be triggered externally

Multiple threaded - risky

### **PLC**

Moderate Precision 14 bits

KHz ADCs

10 Hz scan rates

Good for medium density

Installation/wiring easy

High Initial Cost

Choose one vendor

No external triggers

Not running other software

### **Remote Mux**

Low Precision

10 Hz ADCs

10 Hz scan rates

Good for low density

Installation/wiring hard

Low initial cost

Choose one vendor

No external triggers

No Intelligence

### **Diagnostics/RF/PS**

High Precision >16 bits

MHz ADCs

1 KHz scan rates

Good for high density

Installation/wiring moderate

High Initial cost

Multiple vendors

Can be triggered externally

High Compute load

### **Vacuum/Cooling**

Moderate Precision 14 bits

KHz ADCs

10 Hz scan rates

Good for medium density

Installation/wiring easy

High Initial Cost

Choose one vendor

No external triggers

Interlocks

### **Facility Control/Vacuum**

Low Precision

10 Hz ADCs

10 Hz scan rates

Good for low density

Installation/wiring hard

Low initial cost

Choose one vendor

No external triggers

No intelligence

## *Conclusion*

---

- **There are many significant decisions to make early in a project based on requirements that are rarely present**
- **The control system can be used to provide most of the significant integrating services.**
- **Make informed decisions early - or be one of the first ones to leave after commissioning.**