R*Time database generation for the Clinton Station PPC Replacement

John Carroll
Exelon Corporation
8/06/2007

Abstract

The R*Time off-line point database is arguably the most important part of the Plant Process Computer (PPC). It provides the information which all R*Time applications use to complete analysis, calculations, and alarming. If it is not accurate and complete, then you are faced with the garbage-in, garbage-out principle. Also, it is one PPC component that is normally, if not always, custom-made for each plant. Creating the database requires many decisions that can be difficult to make unless one has a good working knowledge of the R*Time system. This task also requires considerable knowledge of the existing legacy system being replaced.

At Exelon, we have chosen to generate the off-line point database in-house for the Clinton Station PPC Replacement project. This supports our desire to be as knowledgeable as possible about the new R*Time system and enables us to directly contribute our knowledge of the existing Honeywell system.

This paper describes our experience in building the R*Time off-line database. It starts with a discussion of our motivation for performing this work internally (instead of contracting that work to Scientech). It then provides an overview/background of the existing Honeywell PPC, examines some of the tools we have used to extract information from the existing system, explains how we generate the database tables from their Honeywell source/equivalent, and reviews some areas that require special handling. It concludes with some lessons learned from our experience.

Motivation

The major motivation for Exelon to generate the point database was that it is an excellent and practical exercise in learning and understanding the R*Time system. By having to examine the details of what to insert in each database field, it helps us to gain a better understanding of the internals of the R*Time system and how to best configure the points to have the functionality required by the station. Deciding on how to find and generate that information from the existing PPC helps make it clearer how the functionality of the existing PPC maps into the replacement PPC. It also spared Exelon the necessity of trying to put together an accurate and up-to-date text version of the existing point database (more on this later). It helps the team to have a high degree of confidence in the correctness of the field entries (or at least an understanding of how and why we got it wrong). It also forces the team to settle the numerous R*Time configuration details up-front and in a timely manner.

Some of the downsides are that it always takes much more time and resources than expected to generate the database. Given the ever-present shortage of personnel to work on the replacement project, this additional resource loading can be problematic. There is also a significant learning curve in understanding how the R*Time system will use the numerous tables and fields. Finally, the resulting database will still contain plenty of errors (a few of which are also in the current PPC database).

The Exelon team had a reasonable understanding of the R*Time system from the previous two replacement efforts at LaSalle and Dresden stations. This is actually the second database generation effort completed in-house as Exelon/the team also generated the database at Dresden. (Since Dresden is a dual-unit plant two databases were generated about a year apart. So in some ways the Clinton database could be thought of as the third database generation effort. However since the effort of generating the second unit's Dresden database was much like generating the first, I will refer to the Clinton database generation as the second attempt.)

For the central office PPC support group, the most difficult part of generating the replacement database was unfamiliarity with the Clinton Honeywell PPC. It is true that the Clinton Honeywell PPC has been partially supported since it became part of the Exelon Nuclear Midwest regional operating group in 2000. However this support has been very limited and direct knowledge of Clinton's Honeywell PPC is slight. This is very different than the other Honeywell PPC sites. These PPCs were, for the most part, configured by the PPC support group and not by the vendor (GE/Honeywell). This meant that we were very knowledgeable of these implementations. It also meant that for many of the PPC components tools and methods developed internally were used that often differed considerably from the vendor's approach. For example, the GE/Honeywell operator terminals (Aydins) were never part of the installed system. Instead graphic displays built around Ramtek video hardware (which were later upgraded to use a VME based graphics system) and request terminals based on serially interfaced ADDs terminals were the standard design.

Clinton Station is a single-unit plant that was originally owned by a utility (Illinois Power) with one nuclear plant. With limited reason for standards among sites, they used the vendor's configuration tools and equipment. The Central Office PPC Support Group did not have a role in building or configuring the Clinton Station Honeywell PPC. Thus part of our desire to generate the Clinton database was less to understand the R*Time system and more to understand the existing Clinton Station Honeywell PPC.

Our most accurate descriptions of the current Honeywell point database were the actual Honeywell system point builder generated binary point tables that existed in the Honeywell bulk memory devices. We already had a very useful scripting tool built on TCL (Tool Command Language) that we could us to extract and format information from the Honeywell's bulk memory. This Honeywell binary utility was originally developed to support Exelon's other Honeywell PPC sites.

One last reason for doing the conversion in-house was that it aided in ensuring that the Midwest R*Time systems would be as similar as possible. One of Exelon Nuclear's long-term goals is to achieve standardization using the R*Time system platform and to work toward configuring those systems as similarly as reasonably achievable.

Overview of current Clinton Station Honeywell PPC

At Clinton Station there are six Honeywell 4500 PPCs that are connected via a common core memory and share, in a fairly complicated manner, three bulk memory devices. The machines are divided into four groups: the data acquisition processors (DAP), the display control systems (DCS), the balance of plant system (BOP), and the nuclear steam system (NSS).

The two DAP machines are jointly connected to their point I/O through a system called TRU (Test and Reconfiguration Unit). Normally each DAP machine carries half of the point-scanning load but on command or when instructed by the TRU system, either machine can take on the entire load. (Note: both the BOP and NSS processors also have some point I/O attached to them.) The DAP points are notable in having high scan rates (considering the age of the equipment) of all analogs four times a second, and all digitals once a second.

The two DCS (display control systems) machines are attached to the ten operator display terminals in the control room and one display terminal in the EOF. Usually only one of the DCS machines drives all of the displays. The other DAP machine acts as its standby. This active/standby mode is also controlled through the TRU system. However, like the DAP systems, an operator can manually force which displays are driven by which DCS.

The BOP (balance of plant) machine has three display terminals. One is in the computer room and is used for display editing. Another display terminal is in the EOF. Finally, one of the control room display terminals can be switched between the DAP and the BOP systems. The BOP system also has an interface to the NSPS (Nuclear Safety Protection System).

The NSS (nuclear steam system) machine performs the alarming for the DAP machines. It also has a set of interfaces to two computer interface modules (CIM's) through which it collects control rod and lprm data. It also manages the interface to the core monitoring system (GE's 3D-Monicore).

Honeywell architecture

The Honeywell PPC reflects an era when PPC vendors built their own computer hardware from individual logic components. These are highly proprietary systems that differ greatly from the sort of computer architectures that are common today.

The Honeywell 4500s are 24 bit word machines. They have an extensive and useful instruction set with many bit oriented instructions that are well suited for manipulating the sort of bit-mapped words commonly used in PPCs. They have their own single and double precision floating point format but did most of their I/O acquisition and conversion using a scaled decimal system (similar to slide rules).

They have (in modern terms) small working memories with a backing store called bulk memory or large core store (LCS), which was their main storage mechanism. They support a removable hard drive disk pack, though its use was normally reserved for development since the hard drive technology of that time was not sufficiently reliable to support 24 by 7 operation.

The bulk memory is laid out in groups of 100 octal words (64 sets of 3 bytes), which makes the mapping to a 32 or 64-bit environment a challenge. [Typically when these bulk memories have been swapped out with plug-compatible replacements the vendor has chosen to devote 4 bytes of modern memory to every 3 bytes of Honeywell memory making the mapping a great deal easier.]

Clinton Station has a PC interface that is able to read the bulk memory information and write it to a binary file on a PC. This file can then be sent to our corporate office where a Honeywell binary utility (called bulks) which runs on a Sun Unix platform allows us to view, modify, and format the information. (This interface is also currently used to create point value and status text files that are read into the Clinton Station long-term data historian system (DNA). This text file will be used during our parallel run to get the Honeywell scanned data into the RTime system.)

Honeywell point types and tables

Clinton Station uses the Honeywell based SEER (Steam Electric Evaluation and Recording) utility point table builder that reads text input files to generate the numerous Honeywell point tables. However these text files were not always up-to-date or accessible and their format was not as uniform as expected. Therefore the decision was made to use the binary bulk image of the point tables whenever possible.

The Honeywell has five types of floating point value points:

- Scanned analog
- Composed – these are the results of simple arithmetical operations on other points
- Calculated – these points are calculated by applications and usually are a bit more complicated (thus requiring an application to determine their value)
- Transformed – averages and rate-of-change
- System constants

There are only two types of digital value points:

- Scanned digital
- Composed contacts (composed digital) – these are the results of simple logical operations on other points

Each machine with I/O has its own set of tables (DCS, NSS, and BOP) although not all of the machines have all of the point types. (For instance, only the BOP machine has analog transformed points.)

Tools

At the original Commonwealth Edison plants, the IT organization had a long tradition of building their own development environments. Historically, for the Honeywell, this meant using Prime mini-computers as the cross development platform. To do this, a cross assembler and a binary utility tool were written. The binary utility tool was used to perform validity checks of the Honeywell's bulk memory. The primary reason to perform the validity checks was to ensure that our code repository matched what was at the station. The bulk memory units were also prone to error and required magnetic tape backups and restores. This utility helped to ensure that the bulk memory image was correct.

The bulk utility tool was enhanced in the mid 90's when it was ported to the Sun/Unix environment. TCL (Tool Command Language) was added to gain the ability to write simple programs (called scripts) to read and write information from the Honeywell's bulk memory (usually a binary file image of the bulk memory since the utility was not used in the production environment at the station).

Other point information sources

Not all of the information needed for the new R*Time database was available from the Honeywell point tables. The most important of these was the new I/O addresses/tags for every scanned point. For this information a table built jointly by Exelon and Scientech that listed every point's new address was used.

During the conversion effort an attempt was also made to eliminate unnecessary points remaining from the Honeywell PPC. As part of its scanning system, every Honeywell analog cabinet had a pair of shorted and mV reference points. There is no counterpoint to these points in the RTP system and thus they needed to be removed. There were also some obsolete calculated points (mostly from the BOP and SPDS applications), a number of Honeywell performance indicator points (free time calculations), and some instruments that were being physically removed from the plant during the same outage as the PPC is being installed.

New scanned points are being added in the replacement system. Every I/O cabinet was given a RTD so that we could check its temperature (previously only thermocouple cabinets had these cabinet temperature points). Also a number of the TIP inputs that used to come into the Honeywell as processor interrupts were changed to be normal digital points.

Additionally, new point ids have been added for the control rod and lprms. Previously the control rod information was not kept in point ids; instead it filled a control rod array in memory. In addition to the rod's position, the RPIS (rod position indication system) also has a variety of rod status information. This information will now be saved in a separate set of points (before it was used in processing the position value to be stored but for the most part was not otherwise kept). In a similar vein, we are saving the lprm status information that is transferred along with its value.

New point ids to handle the many display format rotary switches that exist in the control room also have been added. On the Honeywell these were directly read by the display program using their hardware addresses and thus these inputs did not have point ids.

Method

To build the Clinton database a tcl script was created for each R*Time database table. The script extracted most of its information from the corresponding Honeywell table. Sometimes it would need to get some of the fields from manually entered data. Special purpose logic was added to each script to handle individual circumstances. It was worth doing this so the table could be regenerated as often as necessary without having to perform the work manually multiple times. The script is required to be run up to three times (once for the DCS bulk image, once for BOP, and once for NSS). C-shell scripts then take these text files and merge them with the manually entered info (sometimes this required minor processing to handle things like the internal point id). Finally the import facility of Microsoft Access was used to get the completed text file into the R*Time database.

Sample script fragment

```
proc sch_clinton { word1 word2 } {

  if {$word1 == 077777777} {
     return "Spare"
  } else {
     set l8 [expr ((($word2 >>  0) & 077) + 040)]
     set l7 [expr ((($word2 >>  6) & 077) + 040)]
```

```
 8)       set l6 [expr ((($word2 >> 12) & 077) + 040)]
 9)       set l5 [expr ((($word2 >> 18) & 077) + 040)]
10)       set l4 [expr ((($word1 >>  0) & 077) + 040)]
11)       set l3 [expr ((($word1 >>  6) & 077) + 040)]
12)       set l2 [expr ((($word1 >> 12) & 077) + 040)]
13)       set l1 [expr ((($word1 >> 18) & 077) + 040)]
14)
15)       return [format "%c%c%c%c%c%c%c%c" $l1 $l2 $l3 $l4
$l5 $l6 $l7 $l8]
16)   }
17)}
```

The Honeywell encodes 8 character point ids in 2 octal words (48 bits). To fit in all of the alphanumeric codes required (A-Z, 0-9, and -) into the 6 bits allotted for each character it subtracts off an ASCII space and stores the result. Thus to convert this packed point id back to a normal string, each character needs to be isolated and a space needs to be added back in.

Line 1: Name of command, takes two required arguments
Line 3: Special check for spare points which are indicated by having all bits set
Lines 6-13: Peels off each character, ANDs off other bits, and adds back the space (040 is 0x20 is ASCII space).
Line 15: Returns the point id string

Example:

The external point id table is located at the address 0205200 on the BOP bulk memory for the NSS calculated points. In the bulks utility example below first dumps out the first two words of the table. The sch_clinton command is then used to convert it. The bdump command dumps information from the binary file specified. "b_bop" is a binary file that holds an image of the BOP bulk memory. The boldfaced lines show my input commands.

```
command: bdump b_bop 02052000 2
02052000:  043252156 044252020
command: sch_clinton 043252156 044252020
C51ND500
```

Special Handling

Exelon did not create the new points needed by R*Time (workstation points, application stall points, etc.). Instead Scientech had the responsibility of creating these points and merging them into the final database.

When converting thermocouple points it is necessary to know the temperature at the screws where the thermocouple wire terminates. This temperature is known as the cold junction reference temperature. It is usually taken from a RTD that is mounted in the rear of the termination cabinet where the field wiring is landed. Ideally, the temperature within the cabinet is kept fairly steady and the temperature of the RTD and of the landing screws is equal. However when the cabinet doors are opened or to a lesser extent when the air conditioning unit cycles, there can be enough of a difference to introduce a noticeable error in the thermocouple points. To reduce this error the CJR point is both filtered and then feed into an average. The filtering removes the jitteriness seen in all scanned analog points. The average attempts to slow the RTD's response to be closer to cabinet metal temperature response. The settings used at Clinton were taken from the Honeywell (a one second digital filter and a fifteen second average).

In general, the RTP I/O cards are a good match for the Honeywell cards that they are replacing. One exception is a somewhat unique Honeywell analog input card that has an exceptionally long time constant of 3 seconds. For these points when their scan table entry is built their smoothing constant (AI_FILTER) field is set to 1/30 (0.033333). Since analog points are scanned at ten times per second (100Hz), this creates a similar amount of smoothing. (At Clinton most of these inputs are flow measurements that are inherently quite noisy.)

In the Honeywell system, when an input point is removed from scan and a value entered for it, it is still possible to see the converted engineering unit value of the actual input signal. In the R*Time system it is only possible to see the unconverted input signal. Thus to aid the instrument mechanics, who often wish to use the PPC when performing their work yet shield the PPC applications (in particular, the heat balance calculations) from seeing their manipulations, we have created two points for a limited number of inputs. One point is the normal point; the other is called the raw point (the normal point id with a _RAW attached to it). The raw version of the point is set to scan the same input channel and perform the same conversion equation as the normal point. When the IM's need to work on the instrument that feeds these input points, the operating staff can remove the normal point from scan and insert a value. This will prevent erroneous data from being used by applications (which look at the normal point). The IM's can still see both the input signal and its converted value by looking at the raw point. Currently we have set up raw points for the twenty-two heat balance input points.

Lessons Learned

It is always tempting to leave certain changes to be done manually after the point tables have been generated. However, I have found that I have ended up having to make those changes so many times that it is almost always worth spending the time up front to automate the process.

It is also somewhat inevitable that this will be a very iterative process yielding many versions of the database. In addition, the target system tends to be dynamic and not static (meaning that the existing PPC tends to continue to undergo changes even while we are trying to replace it). You will want to be sure that none of the interim changes are lost. Finally, refueling outages (which is when the replacement is going to take place) often feature many point changes. Preparing for those future changes is also a challenge.

Even though our Honeywell bulks utility tool was not originally envisioned as a general-purpose scripting tool, the addition of TCL to it has been very useful. It has proven to be a very nice way to generate the various point table text files. Its string handling is very powerful and the scripts are quite readable (unlike say Perl scripts which have the reputation of been practically indecipherable).

Summary

Generating the Clinton replacement PPC’s database has turned out to have been a good choice for Exelon. It has definitely broadened our understanding of the replacement R*Time system and helped us to understand its built-in flexibility. It has also given us an opportunity to become much more familiar with the existing Honeywell PPC. It will be a great help when troubleshooting during the site installation and also in the many post installation questions that inevitably arise. While this might not be the best path for all utilities making similar changes it has been quite worthwhile for us.