ASPECT 2012
IRSE 100 YEARS INSTITUTION OF RAILWAY SIGNAL ENGINEERS

LEGACY TRAIN CONTROL SYSTEM STABILISATION

Dr. Reinhard Galler, Director, EQUIcon Software GmbH
Karl Strangaric, Senior Signal Systems Officer, Metro Trains Melbourne

SUMMARY

In 2003 the previous consortium franchisee Connex Melbourne in conjunction with the Department of Infrastructure embarked on a stabilisation program for the principal train control facility in Melbourne. The facility is commonly known as METROL. The Train Control System located at METROL provides a remote control facility for some signalling infrastructure which is the principal means of operation in the Melbourne inner metropolitan area, currently under the Metro Trains Melbourne franchise (MTM).

The need for the project came about due to a number of factors, some of these were:

1. *The need to accommodate extra operational load until a replacement train control system is rolled out.*
2. *Consequent pressure on the life expired existing system to accommodate network expansion.*
3. *Danger of compromising the integrity of the application Software, due to the utilisation of life expired media.*

The task of stabilisation required that we select an appropriate partner who had the specialist technical knowledge with proven performance in adaptation and integration in the Dec PDP-11 to PC environment. The partner we chose was a German company called EQUIcon Software. The success of the project has qualified their expertise in this field and proven them to be the correct partner of choice.

Basically we took a legacy system running on PDP-11 mini computers and ported it to a PC based platform utilizing a proprietary co-processor card (Osprey Co-processor) to run the application code (an Ericsson JZA 715 product). This runs inside the Co-processor completely, emulated cycle by cycle, and the PC host platform acts as the interface between the outside world and the application. The systems have performed exceptionally well for the last 6 years.

1 INTRODUCTION

The concept of retaining original software and also extending its life by adapting the hardware platform that it runs on had become a viable option as technological advances opened up new frontiers suitable for adaptation. In this particular train control application it has enabled the life extension of mature train control software, originally designed to run on a now life expired DEC PDP-11 platform to be migrated to a PC based platform. The original application software core is maintained 100% and remains unaltered running inside the new platform. This approach has allowed the re-use of the fully de-bugged Train Control application software as the core of the system. Consequently, the major part of the project was the 'hardware adaptation' that was required to integrate the new system into the existing fabric.

This approach allows 'old' software to be 're-cycled' in a very cost effective way. Additionally, systems can be potentially given new life, long term or even extended indefinitely.

2 THE MELBOURNE RAIL NETWORK

The Melbourne Metropolitan Rail Network covers the greater suburban area comprising of 16 electrified rail lines all converging on the Central Business District (CBD). The CBD is also serviced by underground rail tunnels that are looped around this central area. The network is primarily comprised of a range of single, double, triple and quadruple lines on which all services are delivered. The area under direct Train Describer Control is coloured black. See Fig 1.

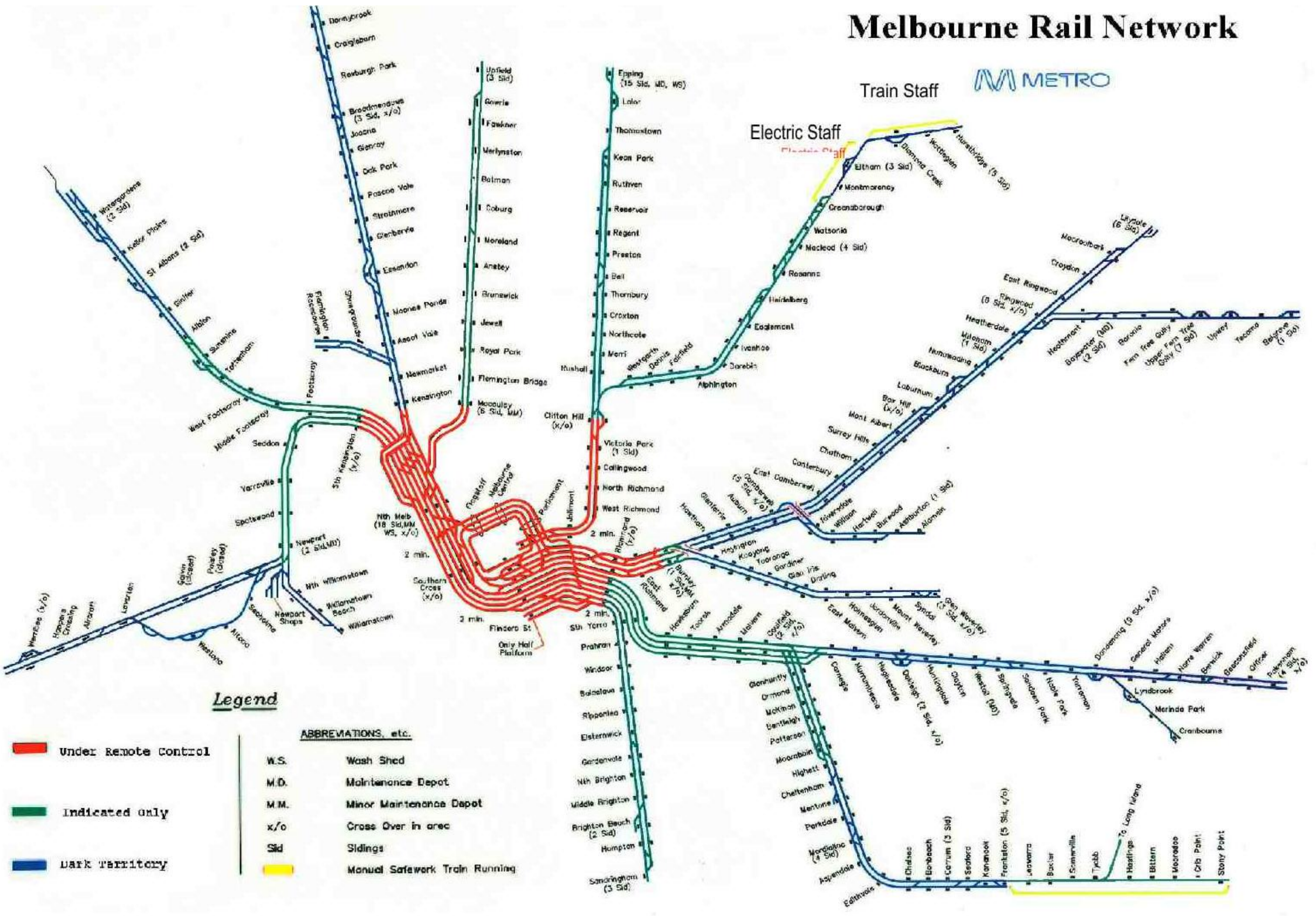


Figure 1: Melbourne Network Map

Service Dimensions

Table 1 quantifies the extent of the network and the dimensions of the services.

Description	Quantity
Number of metropolitan train stations	215
Number of stabling sites (one or more sites)	30
Number of scheduled services on a weekday	2904
Number of train sets running at any one time	160
Track kilometres within the metropolitan network	850
Signal Inter lockings in the metropolitan area	90
Number of Signals, Approximately	3200
Number of Points "	1500
Number of Track Circuits Approximately	4500
Number of Level Crossings	230
Estimated peak number of controls and indication events per minute for the network	2000
Train kilometres travelled on a week day	45,100

Table 1: Service Dimensions

During Melbourne's peak travel time- between 07:00 and 09:00 on any weekday morning a maximum density of approximately 400 train moves converge on the CBD's main station Flinders Street.

3 METROPOLITAN NETWORK CONTROL (METROL)

In the Melbourne suburban area we have a multi tier control system in place. See Fig 1.

This consists of:

1) The ***Inner area*** under central remote control and indication of all signal infrastructure. (colour Red)

2) The ***Outer area*** indications only with control being coordinated by the use of Voice Communication with intermediate Signal Boxes (colour Green)

3) ***Dark territories*** where there is no remote control or indications feeding back to Metrol and Voice Communication is the only means of control. (colour Blue)

3.1 Signal Control (TDS)

At the heart of the control system lies the Metrol Remote Control and Supervisory System, known colloquially as the Train Describer System (TDS). The Inner area is split geographically into three separate computer systems (A, B and C Systems) each running in a dual, master and hot standby configuration. Each has approximately 2700 field objects connected and can handle about 32 trains per minute moving around simultaneously. Traffic Plans are loaded daily into the TDS and are used to automatically route trains throughout the controlled area. Today system loading is at a maximum across all three systems. The Metrol TDS also lies at the heart of a web of interconnected, interdependent train control auxiliary functions such as PIDS (Passenger Information Display System) and OTMS (Train Management and Reporting System) etc. Fig 2.

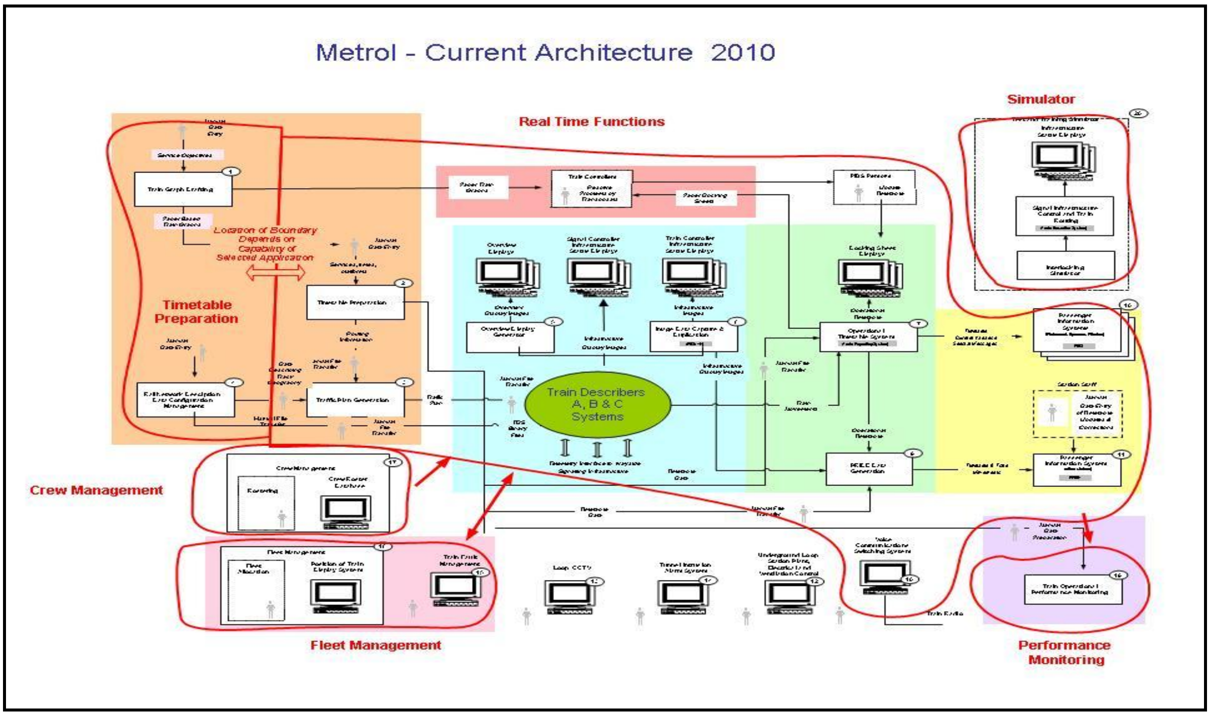

Metrol - Current Architecture 2010
Real Time Functions
Simulator
Timetable Preparation
Location of Boundary Depends on Capability of Selected Application
Train Describers A, B & C Systems
Crew Management
Fleet Management
Performance Monitoring

Figure 2: Metrol Architecture

The Simulator (to be seen in the upper right corner of Fig. 2) is comprised of a fourth completely equipped TDS system called the 'Model', with an interlocking simulator and communication interfaces, an arbitration system and a separate master and hot standby TDS computer. The existence of The Model was a key success factor for the later described Proof of Concept and Acceptance Testing stages of the whole project.

3.2 The Need to Upgrade

Due to network expansion and projected demand for additional services by an ever expanding population, in 2003 the franchisee of the Transport Business in conjunction with the Department of Infrastructure embarked on a large project to modernise the Train Control System (now referred to as TCMS) and simultaneously 'stabilise' the existing TDS.

To allow the existing TDS to remain on the life expired PDP-11 platform would mean exposure to unacceptable risk in terms of system integrity and operability.

The upgrade was carried out by migrating the whole system on to a modern PC platform utilising the proven co-processor called Osprey. This project had basically four objectives.

1. To migrate to a modern PC environment and thereby 'future proof' the aging hardware. This would also guarantee application software integrity. The software containing the core system was in danger of being irreversibly corrupted by the antiquated legacy media being used. These consisted of large open reel magnetic tape and large removable rotating hard disk drives.
2. To reduce the footprint of the entire system to enable the then planned Disaster Site to be built in the most cost efficient manner.
3. Guarantee more time for the existing TDS to enable a complete rollout of the new TCMS. This was a risk mitigation strategy.
4. To accommodate additional traffic that the planned new Time Tables would bring and allow limited expansion capability to absorb some extra field objects in order to accommodate limited network expansion.

4 THE PATH TO STABILISATION

4.1 Requirement Definition

The system requirements (deliverables) needed to be defined before we proceeded any further. This involved extensive discussion with our prospective partner to align each others expectations and focus on a common goal.

The difficult nature of the implementation became more obvious as we examined not only the requirements in the delivery path but equally importantly the immovable criteria. These are summarised as follows:

- The source code for the core software was locked down. There was no chance of any alteration to this code. This meant that the software must 'see' in the new environment what it 'saw' in the old environment. Consequently a hardware emulator capable of 'cycle by cycle' exact duplication of instruction code execution was the only practical path to follow. This was one major obstacle that needed to be cleared before we could move forward.
- The delivered system was required to have the same if not better performance than the existing system.
- There were to be no interruption to daily operations activity. The daily train running function was to be fully maintained and any project work in the way of extensions to the current geography involving the core TDS needed to be absorbed and therefore fully 'transparent' to all stakeholders.
- The TDS maintainers were required to be briefed, trained and 100% competent and comfortable moving between 'old' and 'new' hardware being rolled out side by side with the existing system that was being replaced. This proved to be the key to our success. All participants rose to the challenge and adopted the project as their own as we struggled to conduct 'Proof of Concept' testing on at times the live system while also trying to maintain 'transparency' to the operators.
- The 'tyranny of distance' was another challenge to overcome. Our partner, EQUIcon, have their operational base in Germany and being a small company, they did not have an Australian presence. The only time they were physically present was at key milestones like commissioning. All preparation and problem solving was done remotely via the telephone, Email, and a (secure) internet based remote management PC board enabling a remote maintenance feature for a virtual PDP-11. The project remains a testament to the efficiency of the new electronic communication age, and also to what can be achieved with the right protocols in place. We are very proud of this achievement.
- A key requirement was a 10 year life cycle for the stabilised system with an option to extend further if required.
- The nature of the emulation software environment and strict security protocol dictate that the machines are not to be integrated into the wider area networks within Metro. In effect the core TDS is required to be a closed system.

- The computer arbitration system required to be completely rebuilt due to end of life. This subsystem is the controlling gateway to selecting which machine is in master mode and is connected to the Colour Video Display Units (VDU) at the operator panels and is in the critical path for reliable operation. This 'project within the project' needed to be completed in parallel to Osprey deployment with final debugging being performed fully transparent to the operators of the system.
- The Time Table Generation System was also required to be migrated to a stable platform for it was also based on the life expired PDP-11 platform. The timing of this rollout was also essential due to the closely coupled nature of the system with the TDS*. Traffic Plan* updates are required to be uploaded regularly to the TDS and a seamless transfer is essential. Therefore the method of transfer chosen and the correct protocol engineered to allow the TDS to recognise the data as valid was of utmost importance. This system is not operating in real time mode (see also section 5) and therefore it was possible to replace the system by a software only based virtual PDP-11 which accelerated the compiling of a time table by a factor of about 60 compared to an original PDP-11!

4.2 Challenges to Success

Through experimentation we choose a well worn path to implementation. This involved scoping, rough design, detailed design, prototype and proof of concept before finally settling on an acceptable system configuration having taken into account all the above mentioned critical requirements. There were some notable challenges:

- One of the most important requirements for us was the selection of proven hardware to provide a solid platform backbone for the project. We selected only proven industrial brand name equipment with a solid history of performance and future supply guarantee. As the core requirement of a 10 year lifecycle was paramount, our supplier took extreme care in selecting every part of the PC platform and matching all selected components for reliable operation.
- The next challenge was the choice of the operating system for the host platform. We choose Windows XP because the Proprietary Co-processor card at the heart of the system was especially adapted to run under Windows XP. This operating system is not the best 'choice for real time applications' therefore special attention was required in the method of integration. Running the Windows operating system in cut down mode by disabling unnecessary apps and keeping it free of all 'gadgets' (Office software etc.) increased the system stability sufficiently. The system has now been running trouble free for over 6 years… it has been a good choice.
- Due to the core software limitation (no source code available) we were compelled to integrate some original PDP-11 computer cards into the final product and this resulted in a hybrid PC platform. There are two very distinct hardware technologies in use in each complete system. The PC environment which is the host platform and the DEC environment which is the 'legacy' system. The connection is made using a proprietary Unibus adapter that links the Osprey co-processor to the short Unibus. This Unibus which is totally contained inside the special PC housing allows us to retain some key legacy DEC PDP-11 cards that in turn deceives the software 'to think' it is still running in a full DEC system in terms of timing. A marriage of 'old' and 'new' was borne. This we have come to know as 'Rack Mount Osprey Chassis' (RMOC).

5 THE TECHNOLOGY OF VIRTUALISATION FOR LEGACY COMPUTER HARDWARE

The virtualisation of computer hardware is a well-used technology to save resources and to improve application portability. If a legacy hardware platform is virtualised then additionally a complete or partial independence from the life expired hardware is achieved. Of course, to virtualise old hardware a more complicated virtualisation kernel is necessary compared to for example VMWare®. The often used VMWare® virtualises an x86 platform using a host system which is itself an x86 platform with – in general - the same CPU instruction set and the same bus system and similar peripheral controllers. Nevertheless, the basic idea of virtualisation is to present a virtual resource set (the 'virtual machine') based on physical resources of the host machine as the software expects to run on the physical hardware. Therefore, the software, for example an operating system and the appropriate application software, will not 'detect the discrepancy' and thus run as expected on the physical hardware. For legacy hardware there is an inherent problem: the CPU instruction set, the bus system, the peripheral controllers and anything else can be different between the host and the virtual system. This is solved by a more complicated virtualisation kernel which has to translate between two worlds: the old legacy hardware and the up-to-date host platform but the basic idea as described above is still exactly the same. Frankly speaking, this kind of virtualisation is the original one but when it came to a certain economic importance in the late eighties of the last century it was called 'emulation'. The now well known definition of 'virtualisation' for the same technology was established about 15 years later.

The virtualisation kernel, often called 'hypervisor', might be done completely in software. Some successful implementations of that kind are available for old legacy hardware architectures: PDP-11, VAX-11, Alpha, HP3000 and others. Most of these platforms are much slower than the host systems they are running on now. The result is a huge potential speed-up of the application software. Even if legacy hardware is virtualised and other challenges like the instruction set and addressing scheme is addressed there is the additional problem to observe: Timing. Unfortunately, old software has often been programmed to take into account the timing conditions of the hardware platform. This problem is compounded by the interface to the process which is controlled by the 'old' application software. That interface must be visible for the application software as it was visible when the software was running on the legacy hardware. This can be achieved by 'virtual' interfaces. In general any 'old' legacy interface like 'serial line' or 'parallel interfaces', 'GPIB bus interface' and others can be virtualised by using a similar new PCI interface and a virtual driver which translate the new interface to the old one. For every serious legacy virtualisation implementation a set of reliable virtual device drivers is available. A good example is the driver for the old PDP-11 hard disks. The virtual interface uses a container file located at the host computers hard disk as the physical resource for the virtual old hard disk and emulates the old disk controller and disk drive. If the necessary process interface is not part of that commonly available driver set, the usual workaround is to connect the virtualisation engine to the customer-specific physical interface by means of a so called 'bus adapter' which generates the electrical and timing conditions of the original computer bus system.

This is the moment of truth for the legacy computer virtualisation. If all necessary components of the virtualisation environment are available and work with the original timing only then is a successful implementation of new hardware together with the legacy software for real time process control possible. The timing requirement is fulfilled by hybrid solutions comprising both hardware and software components. For a DEC PDP-11 platform, the Osprey co-processor offers all these properties and has proven its suitability in thousands of applications. *The formal equivalence of an Osprey virtual PDP-11 to a physical PDP-11 is usually proven by the original DEC architecture test software, for example the XXDP.*

Having the Osprey solution available makes it possible for a tailored computer system to replace the PDP-11 hardware of METROL's TDS (see Fig. 3 and 4). As shown, the old PDP-11 computer hardware is just removed and replaced by new PC hardware but the application specific field communication interfaces remain unchanged.

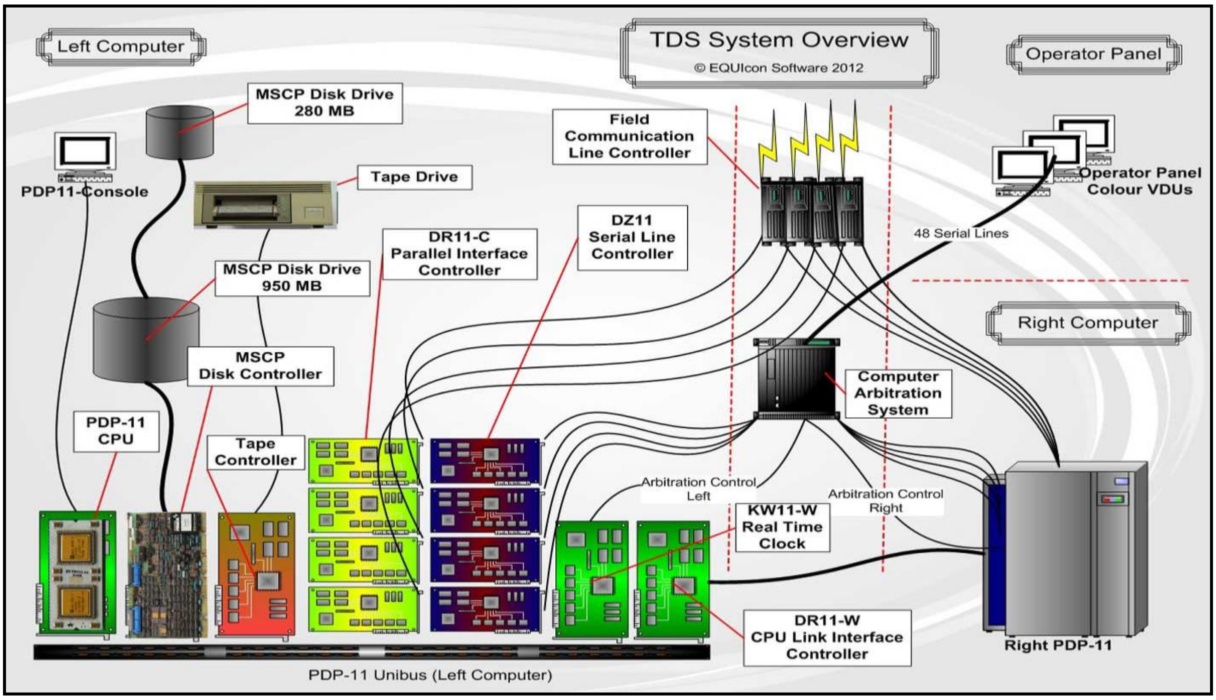


Figure 3: PDP-11 based TDS

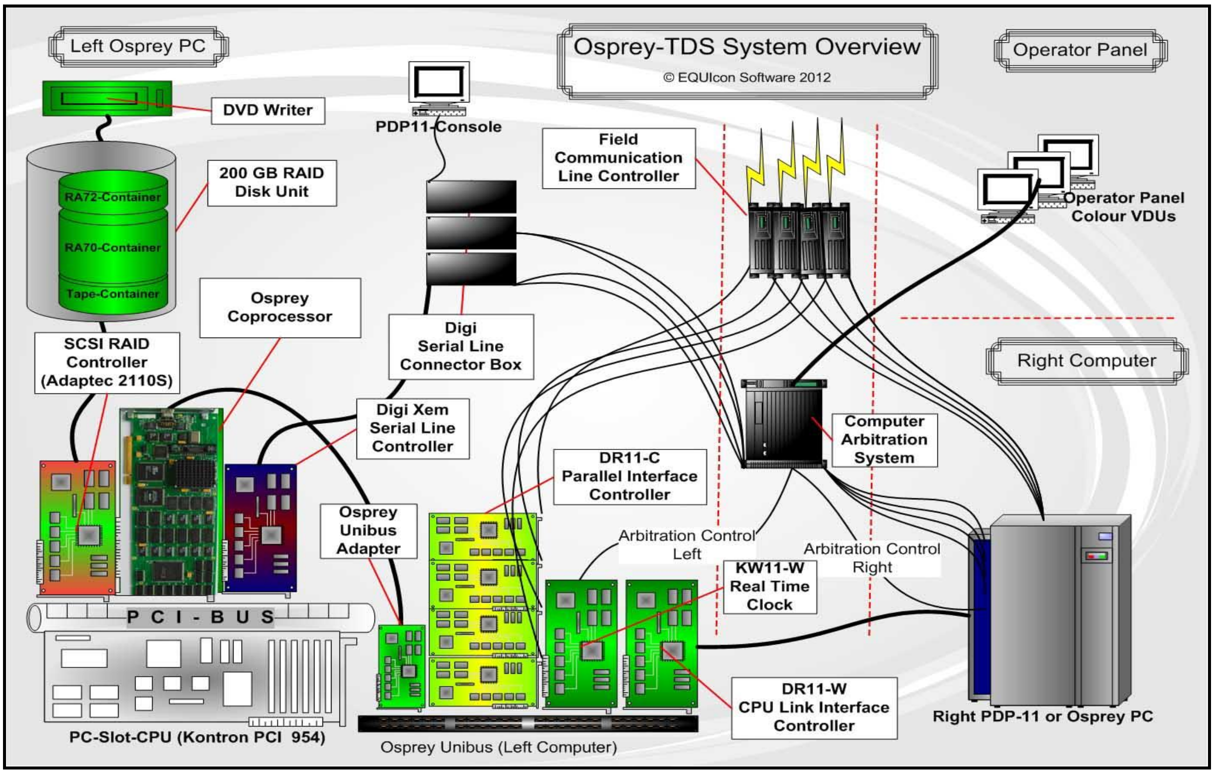

Left Osprey PC
Osprey-TDS System Overview
© EQUIcon Software 2012
Operator Panel
DVD Writer
PDP11-Console
Field Communication Line Controller
200 GB RAID Disk Unit
RA72-Container
RA70-Container
Tape-Container
Operator Panel Colour VDUs
Osprey Coprocessor
SCSI RAID Controller (Adaptec 2110S)
Digi Serial Line Connector Box
Right Computer
Digi Xem Serial Line Controller
Computer Arbitration System
DR11-C Parallel Interface Controller
Osprey Unibus Adapter
Arbitration Control Left
Arbitration Control Right
KW11-W Real Time Clock
P C I - B U S
DR11-W CPU Link Interface Controller
Right PDP-11 or Osprey PC
PC-Slot-CPU (Kontron PCI 954)
Osprey Unibus (Left Computer)

Figure 4: Osprey based TDS

The PC hardware, Osprey coprocessor, Osprey Unibus Adapter and the remaining Unibus interface cards may be built into a specialised cabinet, (RMOC) (Fig. 5). This is the base for the final design which will be described later.

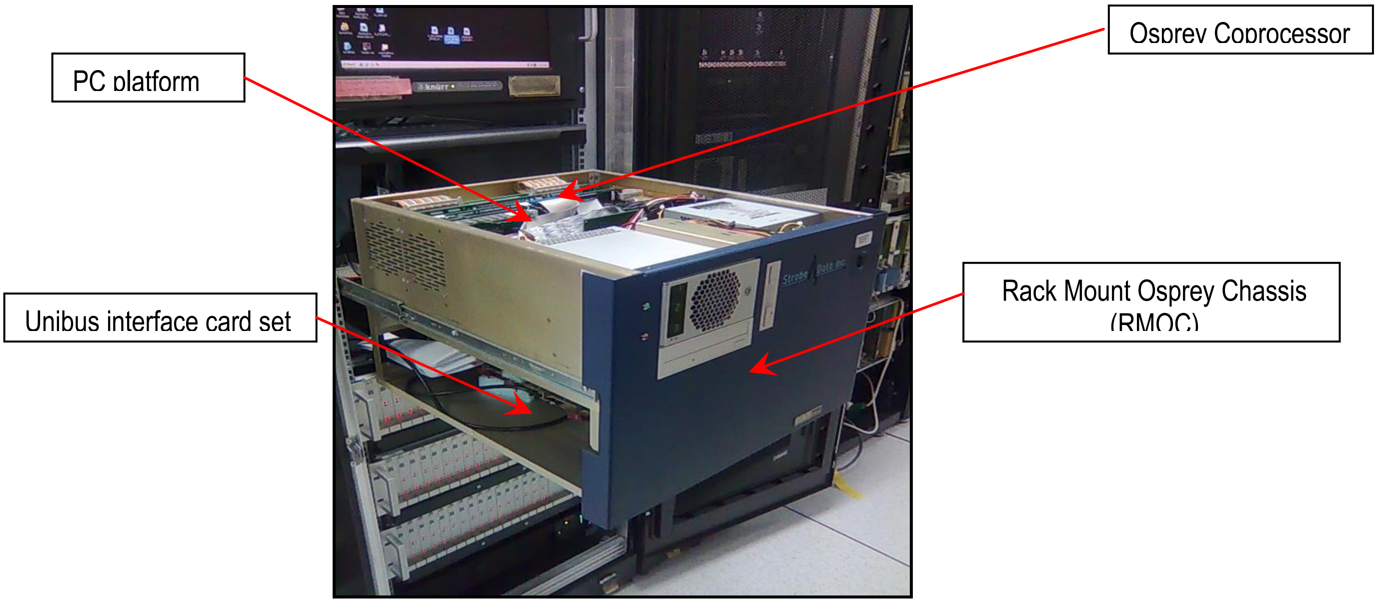


Figure 5: Osprey Chassis

6 THE DELIVERY

The Main Control Site reached practical completion on June 2006 approximately 2 years after the start of the project. The project delivery was on budget and on time. All the requirements were fully delivered and systems are performing in a stable manner. The systems not only delivered a totally viable replacement for the DEC PDP-11 hardware environment but gains were made across the board with respect to system performance, stability and reliability.

6.1 The Proof of Concept

Some research had already gone into the possibility of replacing the PDP-11 with an emulator back in the early 2000's. The purpose for this activity was to establish a fallback plan for the then replacement program called TMF (Train Management Facility). There was some high level discussion on the issue back then but due to funding restrictions the concept did not progress to a formal engineering process. Nevertheless these brief meetings had equipped us with the knowledge of the possibility of hardware emulation and the existence of an 'off the shelf emulator' called Osprey.

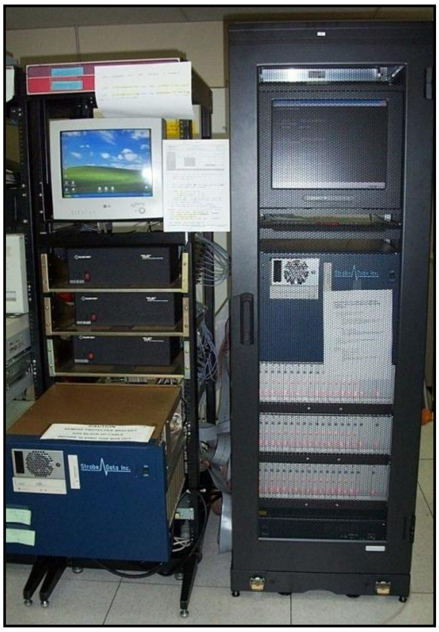


Figure 6: Prototype (left) and Final design (right)

We were now in a position to try and find a suitable partner with whom to develop the idea further with a view to implementation. Enquiries were made through the usual engineering networking circles and a potential candidate was identified. Our first contact with EQUIcon was by way of a telephone discussion and even at this early stage it became clear that EQUIcon through its historical association with PDP based computer systems was the appropriate choice for us.

Further discussions ensued and a sum of money was allocated for a 'Proof of Concept' to be developed. The initial check of function was performed by replacing the 'Model System' with a prototype Osprey configured PC. This experiment became the turning point in the decision to proceed for it performed exceptionally well with minimal customisation and it opened the door to possible spinoffs in the form of turbo boosting the co-processor cards. We examined two possible types of Osprey card, the ISA bus interface type and the PCI bus interface type. Both worked well but because of slight differences in timing we settled on the ISA bus interface and this has been the configuration for all the systems that we have installed.

Based on the above experiment we then proceeded to stage two of the project. This was to develop a 'prototype machine' and install it into a working live system for final debugging of the design. The prototype machine is shown next to the final design in Fig 6.

6.2 Acceptance Testing

Even if an Osprey PC is equivalent to a PDP-11 by architectural definition and verification (refer to chapter on 'The Virtualisation Technology') we thought it necessary that a thorough two stage acceptance testing was performed to double-check and evaluate the functionality of the final design based on the same procedure as used for a different model of the PDP-11 family. That means, all components of the computer system have been checked by the original DEC PDP-11 test program suite (XXDP). First, the PDP-11 architecture equivalence was confirmed (tests for CPU instructions, interrupt functionality etc.). Then, the peripheral interface tests verify that the physical and virtual interfaces (DR11-C, DR11-W, KW11-W, and DZ11) work properly. For the huge number of 48 serial lines an additional test step running in the host computer was implemented. The physical representation for the serial line controllers (DZ11) were chosen to be PCI DIGI Xem controllers and appropriate connector boxes supplemented by current loop converters. Their operational limit is 64kBaud, while DZ11 controllers are limited to 9.6 kBaud. Therefore, this interface cannot be fully tested by programs running in the virtual PDP-11 environment. The test has to start in the Windows environment to check the controllers, the current loop converters and connector cables to the highest possible speed (which in this instance was 64 kBaud) and then to proceed with the appropriate PDP-11 test program. The physical interfaces (DR11-C, DR11-W, KW11-W) are 'visible' only in the virtual PDP-11 environment and had to prove their functionality by the related XXDP test programs. Every machine was required to successfully run these tests for about a day before it was qualified for the next step.

The second step was cycling all computer systems through the 'Model System' where flawless running of the Ericsson JZA715 application was proven. Then the machines were certified for live system testing.

The project involved the replacement of the hardware platform and no changes were made to the application software and the telemetry system. In the telemetry system the card set that was used in the previous platform was retained, and the major connection to the standby machine was also retained. This arrangement meant that for the purpose of acceptance testing we could discount having to re-validate the data path and application databases. We could concentrate on validating the new computer platform and confirming that the application was running as well, if not better than as if it was running solely on the old machine platform.

6.3 The Final Design

Once we were satisfied with the design of the prototype in service we embarked on finalising the design and building the final product to go into service. The design needed to be on a mobile platform to facilitate maintenance and flexibility for speedy utilisation in a fault situation. To provide 100% burn in availability we decided to go with the 'whole of system' spares inventory. This ensures that all the spare sub assemblies would be in a fully assembled form thus providing complete systems that are constantly running and in a state of readiness to be rolled in as fully configured and validated system at a moments notice. This provides us with close to 100% confidence in all our spare machines.

Fig 7 and 8 below graphically illustrate the before and after state of one of the systems at Metrol.

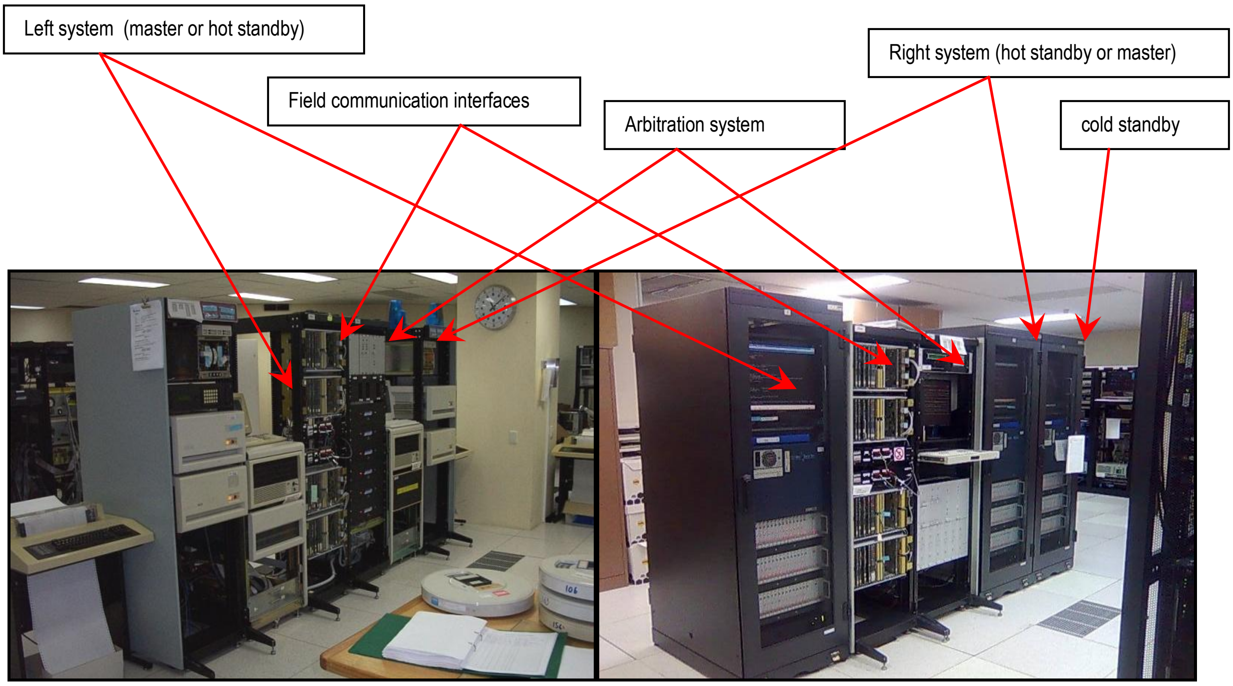

Left system (master or hot standby)
Field communication interfaces
Arbitration system
Right system (hot standby or master)
cold standby

Figure 7: Before

Figure 8: After-Final Installation

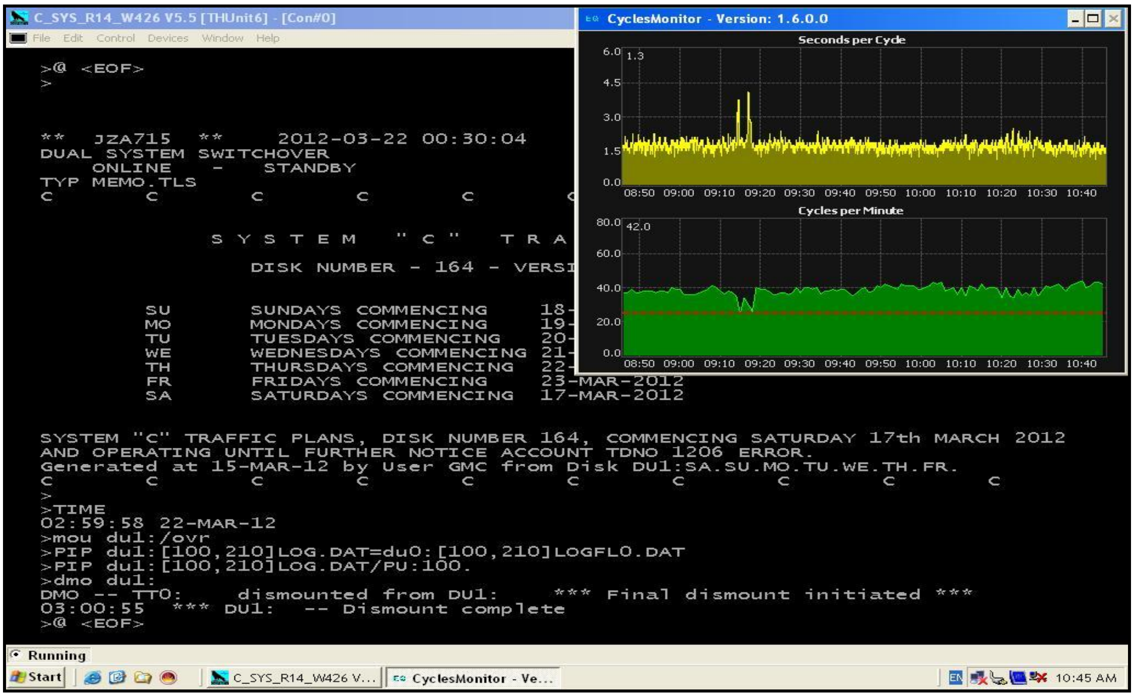
6.4 The Rollout to Production System

We decided to roll out the new machines as half systems. This meant that the old legacy PDP's were running half of each of the 3 systems as a fallback measure until we had built enough confidence in the new systems. This confidence building involved progressively increasing the load on the new systems by initially running after hours, then on the weekends and finally during the week through mourning and afternoon peak traffic. All load testing revealed some issues that were resolved progressively and as each modification was rolled out we gained more and more confidence until we were in a position to make the final leap and convert all of the remaining systems to the new environment. By installing Osprey systems we saved 30% of expensive computer room space and about 75% in electrical power consumption and air condition load, compared to the original PDP-11 site.

The Melbourne control centre is configured in such a way that if the primary control room is disabled due to any reason we are able to bring into service another site called the Disaster Recovery Site (DRS). This DRS site is configured as an alternate control room with a fully equipped equipment room housing a compliment of computers that can provide a fully operational facility. After rollout was finalised at the Primary site we proceeded to design and implement the DRS site with a full compliment of Osprey emulators to provide the computing resources needed. There are a total of eight Osprey emulators at the DRS. These were rolled out as a one step process, with the site reaching practical completion in June 2008.

7 THE NEED FOR MORE SPEED

Due to ongoing delays with the delivery of the TDS replacement system we were again approached by Metro Network Development to increase the Emulated System Capacity to accommodate another expansion to Timetabled Services. The experiences gained during the initial experimentation and research into migration to the PC platform has equipped us with the 'know how' to deliver further improvements and build on an already successful platform. The picture in Fig 9 shows a typical example of how the traffic density can impact on system performance. The sharp spike in the system cycle execution time is a direct result of a build up of traffic in a system that can, if left unchecked, bring down the system and exacerbate an already bad situation. This situation can compound delays in service delivery that become operationally impossible to recover from.



```
>@ <EOF>
>


**   JZA715   **     2012-03-22 00:30:04
DUAL SYSTEM SWITCHOVER
    ONLINE   -   STANDBY
TYP MEMO.TLS
C       C       C       C       C       C

                S Y S T E M   " C "   T R A

                   DISK NUMBER - 164 - VERSI

        SU         SUNDAYS COMMENCING    18-
        MO         MONDAYS COMMENCING    19-
        TU         TUESDAYS COMMENCING   20-
        WE         WEDNESDAYS COMMENCING 21-
        TH         THURSDAYS COMMENCING  22-
        FR         FRIDAYS COMMENCING    23-MAR-2012
        SA         SATURDAYS COMMENCING  17-MAR-2012

SYSTEM "C" TRAFFIC PLANS, DISK NUMBER 164, COMMENCING SATURDAY 17th MARCH 2012
AND OPERATING UNTIL FURTHER NOTICE ACCOUNT TDNO 1206 ERROR.
Generated at 15-MAR-12 by User GMC from Disk DU1:SA.SU.MO.TU.WE.TH.FR.
C       C       C       C       C       C       C       C       C       C
>
>TIME
02:59:58 22-MAR-12
>mou du1:/ovr
>PIP du1:[100,210]LOG.DAT=du0:[100,210]LOGFL0.DAT
>PIP du1:[100,210]LOG.DAT/PU:100.
>dmo du1:
DMO -- TT0:    dismounted from DU1:    *** Final dismount initiated ***
03:00:55  *** DU1:  -- Dismount complete
>@ <EOF>
```

Figure 9: A dip in performance

The solution to making the system more robust was complicated by the lack of source code. In practical terms, we were forced to examine other ways to 'streamline' the system and facilitate data flow from the field to the core application and out from the core to the operator panels. We concentrated on 5 areas for improvement:

1. Measurement Tools :

In order to measure improvements gained we needed an accurate tool that could compare 'before and after' performance parameters that we could use as a guide in determining 'real gain' as opposed to 'perceived gain'. The traditional measure was a measure of system cycle execution time. This was retained in order to maintain consistency and was enhanced to provide an instantaneous graphical view (Fig 9).

The traditional manual logging extraction and interpretation process was automated into a more readily accessible form allowing a quick and easy reference check to be made.

2. Osprey Performance Increase:

The speed and main memory of the Osprey co-processor were increased as far as the current hardware and application software conditions allowed without compromising system stability.

3. Interrupt Rate Decrease:

To free timeslots currently occupied by the colour monitor update process, changes to the peripheral interface program code were made. This reduced the number of interrupts the CPU required to process by about 50% without compromising system stability.

4. Peripheral Interface Software Re-engineering (Colour VDU):

A complete rewrite of this character handler package was required to overcome inherent flaws in the original program which became obvious at higher line speed. This allowed the line interface for each VDU to be increased by 100% thus further saving processing time.

5. VDU line Speed Increase:

Based on the success of items 2, 3 and 4 and by using the configuration features inherent in the Osprey co-processor the colour VDU line speeds were increased to 19200 baud, a 100% speed increase. This measure provided overall improved system responsiveness and enhanced stability.

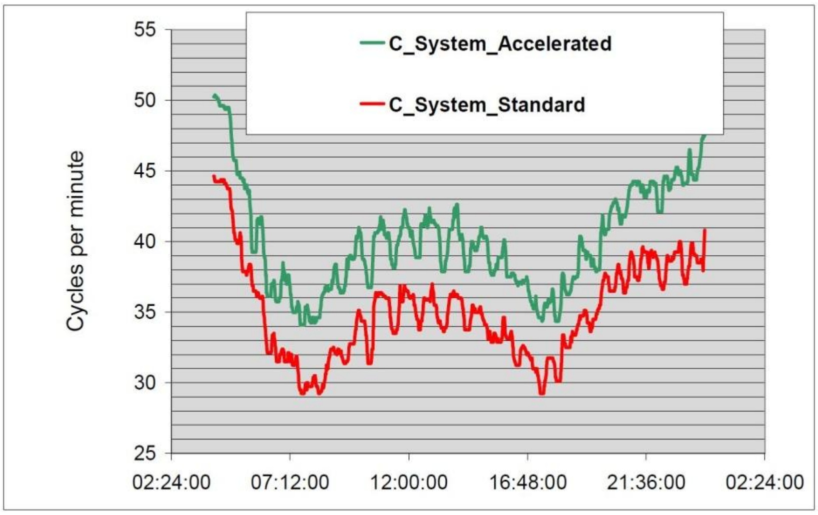


Figure 10: The result of the Speed increase program.

8 PROJECT OUTCOME

If the ‘proof of the pudding is in the eating’ then we are now feasting on the fruits of our labour.

The systems have performed exceptionally well. We can go back over 6 years from the date of commissioning and we have not had any hardware failures that have caused a total loss of system functionality. There are a total of 19 complete Systems running the Osprey platform and there have been no Co-processor failures in the last 6 years. The majority of failures have been the UPS systems. Due to the Master hot Standby configuration of each system we did not suffer any operational impact and we have now replaced the entire UPS installation with current state of art devices.

The other main failures have been the CMOS batteries on the PC motherboards. As expected these have a finite life and as soon as we detected a trend we replaced all of the batteries as a routine maintenance activity.

We have also lost one Hard Disk during this time.

We are now coming up to the half life maintenance renewal process and some of the key sub-systems like the hard disks will be replaced as a precaution to allow us to achieve the projected system life of 10 years.

9 CONCLUSION

This paper is seeking to demonstrate that there is a potential to breathe ‘new life’ into some legacy systems by extracting the most valuable part of these systems, namely the software core, and migrating this to a modern platform. Most projects that involve control systems of any type rely heavily on the Core Software for their success. One can argue the case that rewriting the program can add a magnitude of risk that in some instances should only be taken if the outcome is warranted. Closer examination of the project requirement may reveal that a complete re write is unnecessary and uneconomic when the full cost and risk posed by a re-write is taken into account.

Advances in hardware technology have opened up new frontiers in platform integration. Additionally a large drop in the price for replacement hardware and increasing costs associated with software development should force a re-evaluation of current best practice in project scoping.

10 ACKNOWLEDGMENTS

I would like to acknowledge the advice and valuable contribution made by the following people for the successful implementation of this project, Mr Paul Gartner, Mr Tom Sargant, Mr Jim Warwick, all the team at Metrol Signals and our supplier team at EQUIcon. I also thank the Senior Management Team of MetroTrainsMelbourne for their kind permission to present this paper.