

How to manage life cycle of vessel critical equipment

Focus on obsolescence management of control & monitoring systems

Mohammed Tounsi
PhD, Product Lines Manager
Exail, 24 rue Jan Palach, 44220 Couëron - FRANCE

Philippe Novelli
Captain Armament corps (Ret.), General Manager

Abstract – this document deals with life cycle management of Exail equipment installed onboard navy submarines/vessels. An introduction to Virtual Application Platform (VAP) used in Exail OPSYS system and developed by MicroEJ [1] allows to understand benefit of using VAP concept in application software design and in future hardware obsolescence management. Demonstration of obsolescence management cost reduction is done on a real autopilot calculator using VAP concept. Analysis of recent upgrades done by Exail on three important control and monitoring systems shows the interest of integrating VAP concept from beginning of system design in order to maximize the obsolescence management cost reduction factor.

Keywords—component; OPSYS, Virtual Application Platform, Hardware Obsolescence management; Java Virtual Machine, obsolescence management cost reduction factor

Content

1. Introduction	1
2. Obsolescence management: a real case	1
Introduction	1
Application to submarine steering console autopilot calculator	2
3. Example of modernization based on OPSYS VAP concept	5
4. Conclusion	7
5. Annex I. introduction to virtual application platform concepts by MicroEJ (authors: Fred Rivard, Alexandre Leprieult)	7
Introduction	7
MicroEJ industrial process	8
Hard real time and Java	9
Conclusion	10
6. Benefit of virtual application platforms: savings greater than 5% of the overall French defense budget (author: Fred Rivard)	10
7. References	12

1. Introduction

As submarine/vessels development process follows a long time cycle process, several years stand between Equipment design/manufacture end date and submarine/vessel operational exploitation start date. Life cycle requirement for Navy systems is usually up to 30 years.

In the 80's, to respect 30 years life cycle, military systems required one modernisation upgrade after 15 years of exploitation. In the nineties and early 2000s, modernisation upgrade requirement increase to four (every 7.5 years). The growing use of COTS system or devices in military equipment is increasing number of modernisation upgrade (every five years) [2].

Obsolescence treatment of numerical control and monitoring systems takes a large part of modernisation upgrade cost. This part is increasing in the future systems as new submarine/vessels equipment design integrates more numerical control and monitoring systems than older design.

Recent equipment integrates complex control and monitoring system based on electronic cards and software technologies. Equipment qualification follows tests and evaluations rigorous process to validate hardware and software design before Equipment on-board commissioning.

The life cycles of the hardware/software product technologies used in such equipment can be quite short, particularly for those addressing evolving markets like electronic CPU, COTS PLC and software markets. This situation leads to reduce equipment life cycle and compels to treat obsolescence in order to extend the equipment life cycle.

The cost of obsolescence treatment increases strongly in case of development of new hardware and software due to all constraints in the tests and evaluation (T&E) process.

The concept of ECA to avoid this situation consists of introducing, at the beginning of equipment design, tools and methods allowing the reduction of hardware and software obsolescence impacts on equipment life cycle.

One of the keys of this concept is to separate the hardware device from the software application layer by using a Virtual Application Platform concept (VAP). An introduction to VAP concept and MicroEJ development process is given in annex 1.

This separation between software and hardware simplifies application software development. Developer is focused only on the application functions design, without worrying about hardware target on which its application will run. This separation simplifies also hardware obsolescence treatment and it minimizes the T&E process. This guarantees the same functionality of the application software in case of using new hardware.

This method allows cost reduction of equipment obsolescence treatment and increases its life cycle.

Section II summarizes results of study made by Exail and MicroEJ to validate the use of VAP for portability of application software on a real steering control system for submarines in case of hardware obsolescence. This enables cost reduction analysis related to treatment of obsolescence of individual system like steering control system.

In section III, modernisation experiences of different systems based on Exail concept are presented and a focus is made on obsolescence treatment cost reduction using VAP principle.

A micro analysis of obsolescence cost reduction impact on individual equipment or system may have low relevance, but macro analysis on overall defense budget cost become interesting.

In annex II, results of macro analysis made by MicroEJ for French Ministry of Defense to estimate expected gains using Virtual Application Platforms (VAP) in French defense programs are presented.

2. Obsolescence management: a real case

INTRODUCTION

In 2012, French DGA (Direction Générale de l'Armement) requested to Exail and MicroEJ to demonstrate through a realistic case the expected gains using VAP technology to manage CPU hardware obsolescence of embedded computers in military and marine systems.

The first objective of the study was to demonstrate the possibility to reuse the same application software when the hardware/processor is obsolete in order to avoid a huge workload to rewrite all from scratch (specification)

The second objective of the study was to measure obsolescence treatment cost reduction (in terms of development and validation time gains) with two different approaches:

- Approach N°1: design of new VAP for new hardware. In this case existing application software (code and data) is reused without any modification.
- Approach N°2: redesign of new application software (rewrite software from specification) compatible with new hardware

APPLICATION TO SUBMARINE STEERING CONSOLE AUTOPILOT CALCULATOR

• Introduction to the design of the OPSYS calculator

Since 1998, Exail has designed and manufactured Steering and Diving Control Consoles (SNDC) for submarines. Exail products are installed onboard several submarine classes and used by several Navies, particularly by the French, Indian, Malaysian, Chilean and ROK Navies.



Figure 1: OPSYS real-time DATU

The SNDC control architecture is based on a digital control and monitoring system called OPSYS [3]. The OPSYS system is also used by Exail for the Integrated Platform Management System design and machinery control system [4].

OPSYS real-time calculator (aka Data Acquisition and Treatment Unit - DATU, Fig.1) is designed in a modular 3U Europe format unit based on COTS boards (CPU board, Ethernet and serial link communication boards) and on proprietary boards (acquisition board, SMIO boards and input/output modules).

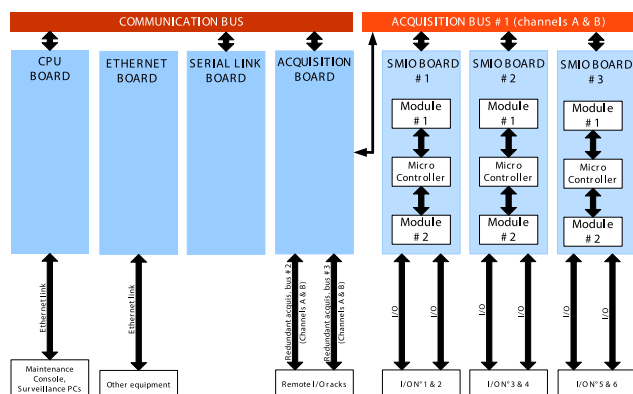


Figure 2: Functional block diagram of a basic DATU

Usually, software applications, including complex calculation algorithms and automation logics, are implemented in the CPU board. The CPU board manages also all communication links (Ethernet and serial links) and also Input / Output (I/O) acquisition links. The Support Mother for Input Output (SMIO) board is used for signals I/O management (Read / write functions). I/O Modules #1, #2 are purely electronic boards (no software) dedicated to direct I/O analog signal acquisition. Module types correspond to usual automation I/O modules (DI, DO, AO, AI, etc...)

The design of the OPSYS system integrates from the beginning the constraints of COTS boards life cycle reduction and future obsolescence management. The design of DATUs (hardware and software) respects a separation between the hardware and software layers not only for CPU COTS boards but also for SMIO boards to anticipate microcontroller obsolescence. Fig. 3 shows the layer separation.

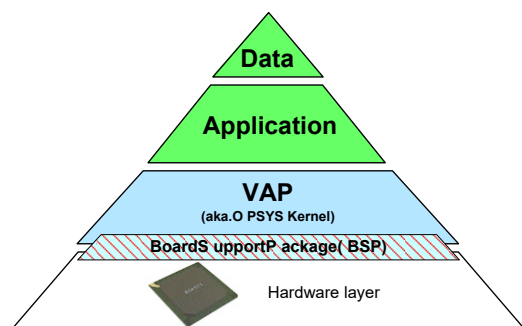


Figure 3: Hardware & Software separation

The integration of VAP concept in the CPU board allows to keep the application software unchanged (including complex multi-control loop algorithms) in case of CPU hardware obsolescence. The integration of VAP concept in SMIO boards allows to keep the signal treatment unchanged (like filtering, accuracy, signal synchronization, etc...) in case of microcontroller hardware obsolescence.

• Demonstration subject

The chosen calculator for the study is the OPSYS calculator integrated inside SNDC to manage autopilot function of submarine. SNDC Autopilot function is a software application block implemented on OPSYS CPU board.

As a reminder, the SNDC autopilot function on Fig. 4 controls the motion of the submarine to reach a desired attitude (depth, course, etc...), by acting on the diving planes and on the rudder angle position. Basically, as shown in figure [1], the SNDC autopilot integrates complex multi-control loop algorithms in order to generate plane angle settings according to operator orders (attitude setting order), actual attitude data (navigation sensors) and actual planes angles (RPT sensors).

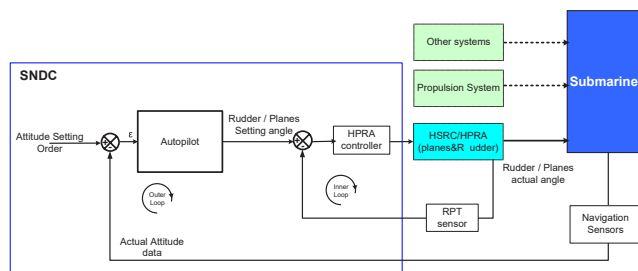


Figure 4: SNDC autopilot block diagram

The choice of autopilot software for this demonstration is driven by the fact that autopilot function is the most critical function of SNDC and validation tests of submarine autopilot requires a strict protocol trial at the sea. In navy submarines (and particularly for nuclear SSBN French submarines), any change linked to autopilot function (hardware or software) can't be installed on board SNDC without giving all certifications that there is no risk for submarine safety.

Demonstration, in case of autopilot CPU hardware obsolescence, that identical autopilot software application (algorithm and coefficients) will be reused on new hardware target, allows to obtain compatibility certification and also to reduce duration of sea trial validation.

• Demonstration test bench description

The demonstration test bench in Fig. 5 is composed of:

1. Calculator 1: DATU same as in the real SNDC fitted with CPU PowerPC (500 Mhz 512Mo RAM and Flash memories). This calculator is used as reference one.
2. Calculator 2: CPU board fitted with microcontroller from ATMEL AVR32 (72Mhz 64Ko ram and 512Ko flash memory). This calculator will be used to implement approach N°1.
3. Calculator 3: Desktop PC under windows XP (2Ghz, 2Go ram and 500Go HDD). This calculator will be used to implement approach N°2. As Windows XP is not real time O.S. an external synchronisation signal (managed by calculator 1) is used to synchronize calculator 3 with calculator 1.
4. 2/3 signals analog comparator: Electronic board used to compare planes & Rudder angle order (8) calculated by each calculator (analog signal) in order to check signals identity. In case of discrepancy between three signals (> defined threshold) an alarm is set to indicate signal number. In case of signal identity, this board selects one of the three signals as output.
5. Planes & rudder simulator: representative simulator of planes & rudder system, which allow to calculate actual planes and Rudder angles (9) according to angles orders issued by calculator (8).
6. Submarine simulator: representative hydrodynamic submarine model, which allows calculation of attitude data (7) (depth, course, trim, speed, etc...) according to actual planes and Rudder angles (9) and propulsion orders.

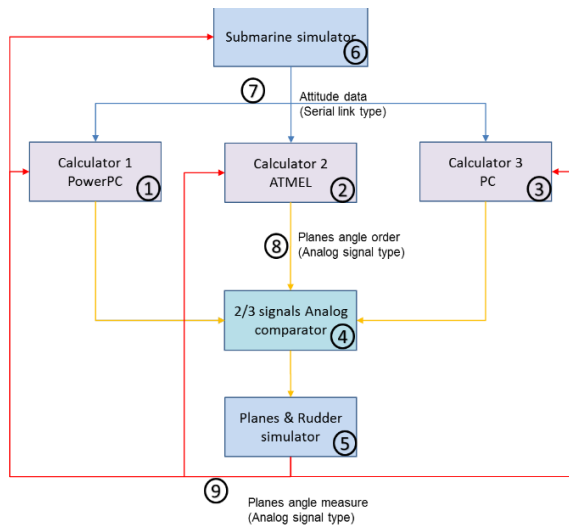


Figure 5: Demonstration bench test block diagram

A demonstration test bench integrates the same simulator units used in real SNDC tests & validation process in order to stimulate autopilot inputs and to compare algorithm dynamic behavior for each calculator. As the three calculators have different processors and Input/output module technologies, comparison is done on analogue output representing autopilot output setting order. With this method, all phenomena linked to digital analogue converter accuracy, filtering, etc... are integrated in the comparison.

Validation test plan used for this demonstration is an extract from real SNDC Autopilot validation plan. Tests objectives are:

- Autopilot dynamic behaviour conformity check for Calculator 2 and 3 with reference one.
- Autopilot algorithm Safety management (defect detection, downgraded modes management, etc...)

• Demonstration results

As expected, both autopilot application software implemented on calculator 2 and 3 generates the same planes & rudder setting angles with an acceptable accuracy ($< 0.1^\circ$) for submarine control.

Autopilot algorithm Safety management of both calculators is identical to reference calculator N°1.

Approach N°1, confirms the fact that same application software can run on different hardware using VAP concept.

Regarding gains of using VAP concept, following workload indicators was measured:

Indicators	Approach N°1 Workload in men/month	Approach N°2 Workload in men/month
VAP design & testing	3	Not Applicable
Application software design	Not Applicable	6
Hardware/software integration	0,5	2
Final T&E process	0,5	2
Onboard T&E process estimation*	0,25	1
Sea acceptance tests*	0,25	1
Total	4,5	12

* Onboard T&E process and sea acceptance tests were not done during demonstration. The estimation of required workload is done upon Exail experience in SNDC onboard T&E process and sea acceptance tests.

Table 1: Workload summary for autopilot CPU obsolescence management

If we define the obsolescence management cost reduction factor as:

$$\text{Cost Reduction Factor} = \frac{\text{Cost of system redesign}}{\text{Cost of obsolesence treatment}}$$

The results of this demonstration shows that using VAP concept in the original design allows to reduce by approximately a factor of 3 the cost of electronic obsolescence management of Autopilot calculator CPU.

Following table shows extrapolation of previous result to SNDC equipment which usually contains 4 calculators using the same VAP (autopilot calculator, monitoring calculator, diving planes calculator and rudder calculator).

Indicators	Approach N°1 Workload in men/month	Approach N°2 Workload in men/month
VAP design & testing	3	Not Applicable
SNDC Application software design	Not Applicable	6x4
Hardware/software integration	0,5x4	2x4
Final T&E process	0,5x4	2x4
Onboard T&E process estimation*	0,25x4	1x4
Sea acceptance tests*	0,25x4	1x4
Total	9	48

*Onboard T&E process and sea acceptance tests were not done during demonstration. The estimation of required workload is done upon Exail experience in SNDC onboard T&E process and sea acceptance tests.

Table 2: workload summary for all SNDC calculators CPU obsolescence management

CPU electronic obsolescence management cost reduction factor increases to more than 5 for all SNDC calculators.

As a conclusion of this study, recommendations have been made in order to maximize obsolescence management cost reduction in complex control and monitoring systems using big quantity of calculators:

1. Use of a modular design calculator (separation between CPU board, communication board and I/O boards).
2. Use of a calculator with same CPU technology.
3. Implementation of VAP concept on each calculator. VAP should be identical for all calculators. That's mean that VAP should propose the largest services and API.
4. Application software design for each calculator should respect VAP services and API.
5. Avoid of use of COTS PLC to implement application software due to the fact that application software will be strongly linked to the PLC hardware. Ascendant compatibility is not always managed by a COTS PLC manufacturer in case of hardware obsolescence.

As shown in section (II-B-1), OPSYS calculator, developed by Exail, respects all these recommendations. That's why Exail is able to propose obsolescence management for old OPSYS systems installed on several vessels with a high obsolescence management cost reduction factor.

3. Example of modernization based on OPSYS VAP concept

Since 1995, the OPSYS system has been installed on board many vessels to manage such different installations as steering & diving system, Propulsion control system, Integrated Control System and IPMS. 50 OPSYS units have been installed by 17 navy and civil users.

With the use of software / hardware separation principle, Exail manages four major obsolescences linked to COTS CPU processor end manufacture and several minor obsolescences linked to electronic I/O modules. OPSYS Obsolescence management was conducted by taking into account OPSYS installed base compatibility:

- Reuse of existing application software and data with new CPU target
- Electrical and mechanical interfaces compatibility of DATU and RIO racks.

OPSYS ascendant compatibility, allows modernization of existing systems without any change in installation mechanical cabinets or electrical modification. Keeping application software and external interfaces original, allows the benefit of reducing the upgrade period on board.

With this method, Exail operates modernization upgrades of OPSYS systems installed on board different vessel types in the 2000s. Three of important control and monitoring system upgraded by Exail are presented in this section.



Class	Amethyst	LEEUVIN.	Charles DE GAULLE
Units	4	2 HMAS LEEUVIN and HMAS MELVILLE	1
Owner	ENSCO (formerly Pride Int.)	Royal Australian Navy	French Navy
Commissioning	2001	2000	2001
System	Vessel Management System: • Power management system • Propulsion system • Ballast, auxiliaries, HVAC, ...	Integrated Control System: • Autopilot • Diesel normal/emergency generation • Power management system • Propulsion system • Ballast, auxiliaries, HVAC, ...	Power management system: • Power plant monitoring • Generator Synchro-coupling
Nbr I/O	7,500	3,000	4,000
Nbr of MFC*	4	5	N/A
Nbr of DATU	9x2 (Redundant)	6	14
Nbr of RIO unit	14	7	6
Topology	Redundant star network	Redundant star network	N/A
Certification	Lloyd's	Lloyd's	French Navy Standard
Upgrade year	2010	2014	2017
Upgrade lead time	9 months	6 months	12 months
Upgrade period	3 weeks per rig.	1 weeks per ship.	3 weeks
Obsolescence management cost reduction factor	9	7	20

*MFC: Multi-Function Console

Table 3: example of modernisation

For the three projects, the customer requested modernization of existing control & monitoring systems initially considered a complete replacement of the existing system by PLC COTS systems. Exail pointed out the inconvenience and risks linked to complete such replacement of old systems.

Exail highlighted the advantage of using VAP in OPSYS system and capability to reuse application software without any modification.

Proof was done to customer of benefits using VAP for requested modernization and also for system life cycle extension.

One of the advantages of using the VAP concept is the reduction of modernization lead time (six up to twelve months depending of system size). Upgrade period (duration of onboard modernization works) is also reduced (one week up to three weeks depending of system size).

As shown in table 3, obsolescence management cost reduction factor was calculated for the three project based on estimation of cost of new design solutions.

As mentioned in the previous section, the obsolescence management cost reduction factor increases according to number of software applications and calculators using same VAP. This result is confirmed on the three upgrade experiences done by Exail.

We note that the highest factor (20) is obtained for the nuclear aircraft carrier, due to the high cost of redesign of a new system. As the vessel is under French nuclear regulation, a very large workload is required for new designs (design rules, certification, documentations, T&E process, quality management, etc...). In this case reuse of application software without any modifications have allowed us to lighten applicable rules.

4. Conclusion

In this paper we demonstrate the use of the VAP concept in application software design in order to anticipate hardware obsolescence.

Based on Exail long term support management concept and refit experiences, recommendations have been made in order to maximize obsolescence management cost reduction in complex control and monitoring systems using big quantity of calculators.

Previous modernization experiences done by Exail are presented and benefits of using the OPSYS VAP concept in obsolescence management cost reduction are confirmed.

Reuse of application software in the case of hardware obsolescence management allows life cycle extension of equipment and systems.

5. Annex I. introduction to virtual application platform concepts by MicroEJ

(authors: Fred Rivard, Alexandre Leprieult)

INTRODUCTION

Java technology is based on two different items

- A modern programming language: Java
- And a logical processor called Java Virtual Machine (JVM)

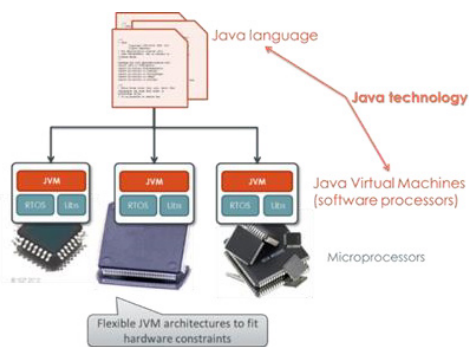


Figure 6: Java technology overview

The JVM is a logical (software) processor with a 32-bit instruction set (203 instructions). The Java language generates a bytecode and this bytecode is executed by the JVM. Portability of the Java bytecode is insured by the fact that its specification never changed.

VAP definition: Virtualization Application Platform, is composed of all the specific software elements managing the hardware and all the software elements that have a service to offer to the above application. VAP hides the complexity of the hardware and also of the drivers (C code to manage hardware at Chip level), stacks, Board Software Package BSP (All drivers needed to manage the hardware of a board), Real Time Operating System (RTOS) and allows for the developer to focus on the added value, the application.

VAP should also include existing C code that a customer wants to reuse.

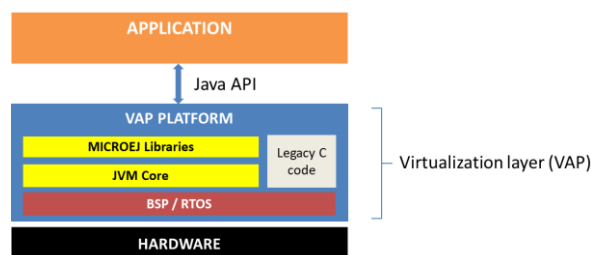


Figure 7: Virtual Application Platform overview

The Java technology has successfully been used and deployed on servers, desktops, smartphones/tablets (through Google Android), and highly secure smart cards.

But in embedded systems existing Java technology implementations was suffering of some bottlenecks and problems like:

- Big size (in terms of memory needs: footprint)
- Low performance
- Needs of huge processors using a lot of power consumption
- The time to create a new JVM or a new VAP for a new hardware
- No Hard Real Time

The achievements of MicroEJ is the optimization of Java Technology to be used in embedded systems and lead to following results:

- A tiny footprint 28KB of Flash memory and 1KB of Ram
- Performances comparable to C systems
- Running on very small microcontrollers with low power modes
- Hard Real Time environment

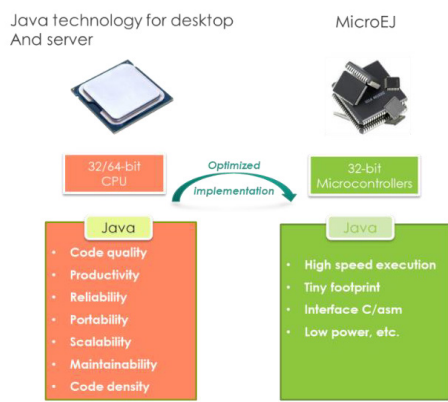


Figure 8: JVM optimization design [5]

MICROEJ INDUSTRIAL PROCESS

Usually develop a JVM for a specific processor is a huge workload (Approx. 10 men/year) but MicroEJ put in place a factory to create JVM in only few weeks with an industrial process.

This process is 80% automatic and allows generating a JVM in 3 or 4 weeks from the processor datasheet. The validation is based on an automatic tests suite running more than 30 000 different tests.

The modern software factory also covers VAP development and validation.

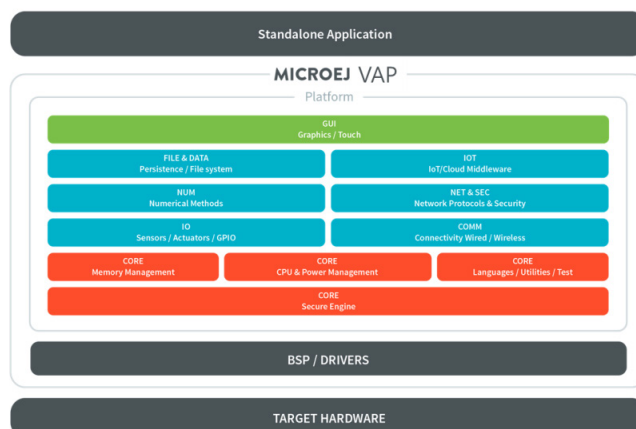


Figure 9: VAP development and validation diagram

Validation should be complex because; depending on the system to develop, the number of services to validate is sometime huge.

- Memory management
- Power Management
- Secure core engine
- Languages and Fonts management
- File system & storage
- Numerical calculations
- IO management
- Communication management
- Network and security
- High level protocols (MQTT, JSON, REST ...)
- Graphical for human management interface (HMI)
- Interface between High level language Java and BSP/drivers

High level APIs are easily tested through tests suite but for testing low level, (BSP, Drivers) MicroEJ define standards Low Level APIs. Using this standardization MicroEJ is able to test automatically the full VAP in order to guarantee the same behaviour from one hardware set to another.

Fig. 10 shows an example of Low Level input devices API:

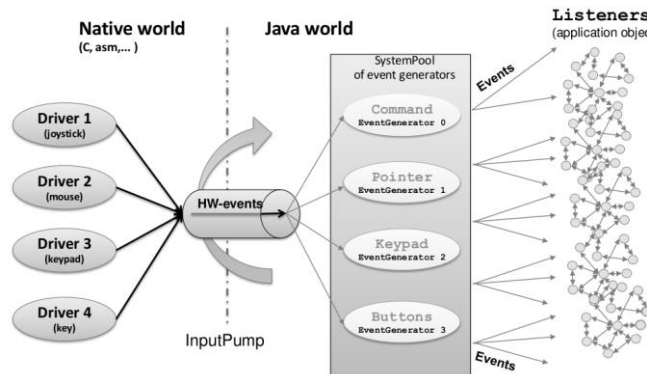


Figure 10: Low level input devices API

HARD REAL TIME AND JAVA

Hard real time is a very specific area in embedded systems and is needed only for few systems. Mainly two approaches exist:

- Hard real time is handled by hardwired logic like FPGA
- Software solutions through operating system

MicroEJ developed another approach. This approach solved the main issue of Java technology regarding hard real time: the automatic memory management is normally not real time.

A hard-real-time system (i.e. software & hardware), is a system that guaranties that expected behaviors will occur at the right times. The distinction between hard and soft real-time is between guaranteeing and best-effort. Hard Real time is not related to speed, but to have the different tasks of a system to complete by their respective deadlines for sure.

Automatic memory management greatly simplifies the development of software, and is one fundamental component of any modern logical processor. In 1992, the first deterministic autonomous concurrent memory management was published [6], a quarter of century ago. In 2006, the memory consumption formulas were mathematically established. In 2009 was published the [7] specification which explicitly targets safety critical system such as avionic DO178C level A, with logical processor.

In 2010, European Space Agency and ASTRIUM [8] studied the hard real-time logical processor of MicroEJ with the following criteria:

1. Defining requirements for space applications of a logical processor,
2. Testing scenarios of use to assess the adequacy of the solution
3. Assessing the efficiency of determinism and I/O handling.

MicroEJ HRTJ logical processor runs cyclic hard real-time task: a HRT task is defined by its period, its priority and a method that is executed at each cycle. Synchronization between HRT tasks is done via monitors that use the priority ceiling protocol. The memory management is semantically done in one extra specific cyclic task that has a priority lower than all HRT tasks: its activity cannot preempt HRT tasks and therefore it does not interfere with the application tasks. This extra task can thus be integrated into the computational model to check resources consumption (memory and time). Then, standard scheduling analysis is conducted to verify (prove) the scheduling properties of the application (including application and garbage collection tasks). The HRT logical processor's memory management of MicroEJ relies on a hard-coded maximal GC speed to get the period of the GC_task and the WCET of the GC_task. Dimensioning it is thus fully integrated to the regular development process of the application. MicroEJ provides a tool that analyzes the cyclic tasks based only on the binary code (and not the source code), computes their worst-case execution time (WCET) and worst-case allocation (WCA) and them deduces the WCET of the GC. Indeed, the WCET of the MicroEJ HRT memory management does not depend on the heap size or the number of objects allocated

but on the live memory size. Since live memory size is bounded (otherwise the application will eventually run out of memory), the WCET of the GC is bounded as well.

(1)

$$WCET_{GC} = \frac{(\sum_i WCA_i) + CPB}{Speed_{GC}}$$

(2)

$$T_{GC} \leq \frac{(HHRT - CPB - 2 \times \sum WCA_i)}{\sum \frac{WCA_i}{T_i}}$$

Figure 11: with T_i the period of the Task_i and WCA_i the worst case allocation of the task_i, and CPB a Cross Period Buffer that may keep live objects across several cycle of one task

The MicroEJ HRT has been tested on the Leon2 processor architecture dedicated to space, with promising results: the footprint of the logical processor including its basic libraries is less than 80 kB, the speed overhead for I/O is negligible compared to non-virtualized technology. The measurements for determinism of the memory management were exactly as predicted by the mathematic formulas.

It results that even in domains that require a safety critical hard realtime system, logical processor technology can be used to save a great amount of energy (cost & time), and to reduce the complexity of the overall system.

CONCLUSION

MicroEJ eliminated and solved all common issues regarding Java technology used in the embedded world and keeps only the advantages.

6. Benefit of virtual application platforms: savings greater than 5% of the overall French defense budget (author: Fred Rivard)

Since the start of the digital revolution (1998-2005), it has been possible to identify the role played by embedded electronics in the major industrial sectors and to follow this evolution. Military equipment obeys to these ratios.

Share of electronic systems (hardware & software) in the overall design cost of an electronic product		Share of all software in the cost of designing an electronic system	Share of all software in the overall design cost of an electronic product (in 2007)	Share of all software in the overall design cost of an electronic product (in 2017)
Automotive	56%	40.9%	22.9%	29.8%
Avionics / Aerospace	54%	29.9%	16.1%	18.6%
Industrial / Automation	48%	54.9%	26.4%	32.9%
Telecom	58%	30.1%	17.5%	26.2%
Consumers / Home	62%	58.5%	36.3%	54.4%
Medical	53%	14.1%	7.5%	8.6%

Source: VDC 2007; IDATE 2006

(2007==>2017 : authors extrapolations)

Table 1: cost breakdown between "firmware" and "software"

At the software level, there are two categories:

- The firmware which is "coupled" to hardware,
- The software which is in fact an abstract model that does not rely on any specific hardware mechanism.

Table II presents the cost breakdown between "firmware" and "software".

Share of cost between firmware and software into embedded electronic product	
firmware	25.7%
software	74.3%

Table 2: cost breakdown between "firmware" and "software"

Thus, it becomes very important to allow ubiquity of application programs (software), representing 75% (in 2008) of the all software costs: their capitalization through the material evolutions allows savings of the same order.

A widespread use of virtualization application platforms, VAP, allows a strong decoupling between the hardware and the software applications running on top of these VAP. It is indeed possible to deploy the same software in its binary form on different hardware and potentially on better hardware in the future. This is possible without any modification of 0 or 1 in the original binary software. This is known as software ubiquity and technology transparency brought by virtualization application platforms, provided by companies like MicroEJ, Android. The hardware and software life cycles are split up. Therefore, decoupling software life cycle from hardware life cycle has two main financial benefits:

- Cost reduction of the system design,
- Reduction of the system maintenance.

The French Defense Department publishes its consolidated multi-year budget for its 5 forces. Although the Maintenance in Operational Condition (MOC) indicator is difficult to apprehend due to the lack of cost accounting which would allow to qualify the total cost of ownership of a military equipment, in 2006, the French Defense Department, in its LPM 2009-2014 (*Loi de programmation militaire 2019-2014*), estimated the MOC to be more than 4 billion euros (as compared to 8 billion euros of investments in 2008). The MOC calculation includes the expenses related to the technical and operational availability, taking also into account the necessary means that allow training under suitable conditions. Over the 2009-2014 period of time, scheduled maintenance of equipment and personnel was budgeted at 17 billion euros, indicating a strong desire to contain the MOC from a budgetary point of view, while maintaining the technical availability level necessary for the needs of military forces.

On the basis of the LPM 2009-2014, the analysis of the programs allowed to qualify between 12% and 14% the savings relative to the MOC if VAP technology is used. The method of analysis consists in determining for different programs an annualized cost of MOC based on the lifespan of the equipment and their revisions and to account for the software part made independent of the hardware. The generalization of the use of virtualization application platforms makes it possible to forecast an annual reduction of more than 500 million euros per year on the French MOC [10] (a possible more optimistic analysis gives about 1 billion euros/year). This is more than 3% of the overall LPM 2009-2014.

More generally, the world of embedded computing is essentially heterogeneous, going hand in hand with the heterogeneity of the missions of the devices. The use of application platform allows preservation of this heterogeneity, while rationalizing the execution target by a standard (virtual) processor: this allows a great variability while benefiting from the advantages, especially economic, of a standard. Indeed, it is no longer acceptable to have to rewrite the software and these possible certifications and / or qualifications when the equipment evolves.

A systematic analysis of the LPM 2009-2014 made it possible to qualify quite precisely the potential benefits on investments, based more particularly on EAs (authorizations of commitments) amounting to more than 8 billion euros in respect of expenditure. This analysis was carried out on the 5 systems of forces, that is the 5 coherent sets defining operational capabilities:

- Deterrence,
- Command and control of information,
- Engagement and combat,
- Mobility and support projection,
- Protection and safeguarding.

The method of financial analysis was applied to program-by-program cost, taking into account ratios by industry, and then deducting a [minimum, maximum] saving interval by applying a prudent coefficient derived from qualified earnings on previous industrial achievements. This analysis makes it possible to identify the orders of magnitude «to a few million euros».

1. Deterrence: ensuring the technical and operational credibility of deterrence and its posture. The authorization commitments (for 2008) amounted to more than 1.5 billion euros. The expected savings vary between 8% and 12%, i.e. between 120 and 190 million euros. In this category, we can find the SNLE, Mirage 2000 N K3, M51, ASMPA programs.
2. Command and control of information: control, drive, communicate, inform, monitor and acquire. The authorization commitments (for 2008) amounted to approximately 500 million euros. The expected savings range from 7% to 12%, i.e. between 38 and 62 million euros. The OE SIC TERRE, DNG 3D, SCCOA and MELCHIOR programs have important positions in this category.
3. Projection, Mobility, Support: projecting forces, ensuring mobility and maintaining this potential. The 2008 authorization commitments were approximately 800 million euros. The expected savings range from 6% to 9%, i.e. between 45 and 70 million euros. The NH90 and PPT programs account for a major share of this force.
4. Commitment and Combat: Targeting and operating in a hostile environment. The authorizations for 2008 commitments amounted to more than 5 billion euros. Expected savings range from 7% to 11%, i.e. between 370 and 590 million euros. Excluding the second aircraft, which represents more than half of the 2008 commitments, the expected savings range from 95 to 150 million euros. This force is very important, which translates into many programs like FELIN, TIGRE, CAESAR, FTL, MDCN, AASM, MU90.
5. 5. Protection and safeguarding: Ensure the safety of approaches and the protection of forces and sites. The 2008 commitment approvals amounted to approximately 300 million euros. The expected savings range from 10% to 17%, i.e. between 30 and 50 million euros. For example, these include the MICA, MISTRAL, FSAF, HORIZON, PAAMS programs.

Table III summarizes some of the potential savings from widespread use of VAP (virtual applications platforms):

Forces	AE in billion euros	Savings Min.	Savings Max
Deterrence	1.5	0.12	0.19
Command and control of information	0.5	0.04	0.06
Projection, Mobility, Support	0.8	0.04	0.59
Commitment and Combat	5	0.04	0.07
Protection and safeguarding	0.3	0.03	0.05
Total 2008 (in billion euros/year)	8.1	0.6	1
Percentage of savings if using massively VAP on Military equipment:		7%	12%

Table 3: savings from the widespread use of VAP

A second analysis of the overall costs of the programs (as opposed to the annual AEs 2008) was undertaken for the LPM 2009-2014 by analyzing the overall costs of the different programs. Expected savings vary between 9.80% and 15.70% of equipment investments, resulting in savings over a period of 5 years, ranging from 5 to 8 billion euros. This global cost analysis confirms the expected savings resulting from the analysis of EAs over 2008, which over 5 years would give an interval of savings ranging from 4 to 6.5 billion. Table IV summarizes year by year the potential savings on equipment investments.

Billion € 2008	2009	2010	2011	2012	2013	2014	Total
Full LPM year	31.25	30.86	30.19	30.52	30.83	31.14	184.79
Including equipment	16.59	16.53	16.21	16.64	17.27	18.01	101.25
Savings: worst case scenario	0.89	0.89	0.87	0.9	0.93	0.97	5.45
Gains: best case scenario	1.44	1.43	1.4	1.44	1.49	1.56	8.76
% average / LPM	3.73%	3.76%	3.76%	3.83%	3.92%	4.06%	3.84%

Table 4: potential savings year by year

These figures do not consider the savings on certification, which significantly increases the reuse capacity of software bricks, which in turn increases the potential savings.

Thus, the use of VAP leads to a reasonable 5% saving on the LPM 2009-2014, an amount of nearly 10 billion euros spread over 5 years for investments on new programs of the French military budget.

7. References

- [1] MicroEJ is a French software editor for embedded systems. Founded by Fred Rivard (PhD in computer science) in 2004. Before MicroEJ, Fred was a core member of IBM Java technology team. MicroEJ injected \$20M in R&D to make possible the usage of Java technology in embedded system even with low material resources. <http://www.microej.com>.
- [2] A. Meyer: Management of component obsolescence in the military electronic support environment, June 2002
- [3] <https://www.exail.com/product/opsys>
- [4] <https://www.exail.com/product-range/platform-management-command-control-ship>
- [5] Vincent Perrier and Fred Rivard: What are the Differences between Java Platforms from Desktops to Wearables: Electronic Design, 23/05/2016.
- [6] Baker Treadmill; Real-Time Garbage Collection Without the Motion Sickness; 1992
- [7] www.e-s-r.net, Hard Real-Time Java specification for safety critical systems; 2010
- [8] Fred Rivard (MicroEJ), & Marek Prochazka (ESA) & Thomas Pareaud, (ASTRIUM): Java for On-Board Software; DASIA 2011.
- [9] Martin Schoeberl: Real-Time Garbage Collection for Java; ISORC 2006
- [10] Of the more than 4 billion euros MOC costs, the staff costs (about 40,000 men assigned to this task) amount to about 1 billion euro (about 25%). The reduction of the MOC relative to hardware (and the software functions carried by the latter) is in fact greater than 30%.