MapuSoft OS Abstractor™

- A White Paper by Kyle Craig, Product Manager, MapuSoft

Developing software is complicated. A software project has several critical elements governing its success. Issues related to budgeting, software tools, project schedules, and application requirements are just a few of the elements to be handled. In many cases, all of these factors are necessary, but not sufficient, to a successful software project. Having all of these project aspects aligned and moving forward can be overwhelming, and missing one minor piece can make or break a product launch.

In choosing the specific software tools for the project, there are several variables to consider. You must spend precious time evaluating the capabilities, verifying availability, confirming pricing, and making an overall decision for the fit of the tool. Significant time is invested in this process. You need to make sure that your time in evaluation leads to an efficient and practical solution. A poor choice in tools could lead to project delays or a project cancellation. Many fancy software features that looked good in the initial demonstration may lead to project delays and problems later in the development cycle. It is critical to perform the necessary due diligence to ensure that the proper tools are chosen.

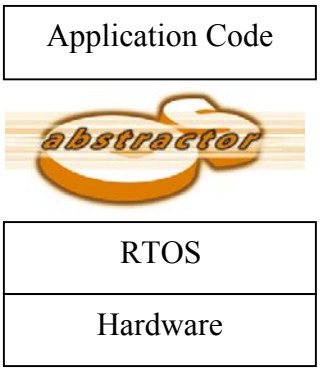
One of the major software decisions you must make is the type of real-time operating system (RTOS) that will be used in the project. The RTOS is a core component of the application design, and the RTOS capabilities must adequately fit the needs of the project. As with the other software tools, the choices surrounding the RTOS are numerous. There are over 100 different commercial RTOS vendors to choose from. You must take into consideration many variables: What is the RTOS code size? How fast is the kernel? What is the interrupt latency? Are the price and business model right for my project? What are the APIs?

Using OS Abstractor provides you a solid architecture for application development and eliminates the risk associated with the RTOS choice. By acting as a layer between your code and the application, our abstraction technology allows a seamless transition from the application code to the operating system. OS Abstractor helps to alleviate issues surrounding the RTOS choice including the support of updated hardware for your project or achieving the performance you expected from the RTOS. If you need to move to another RTOS vendor as a result of these situations, OS Abstractor becomes a critical component. Alternatively, if you have to support older application code that runs on an outdated RTOS, the MapuSoft solution can help you in your transition to a new operating system.

OS Abstractor - Overview

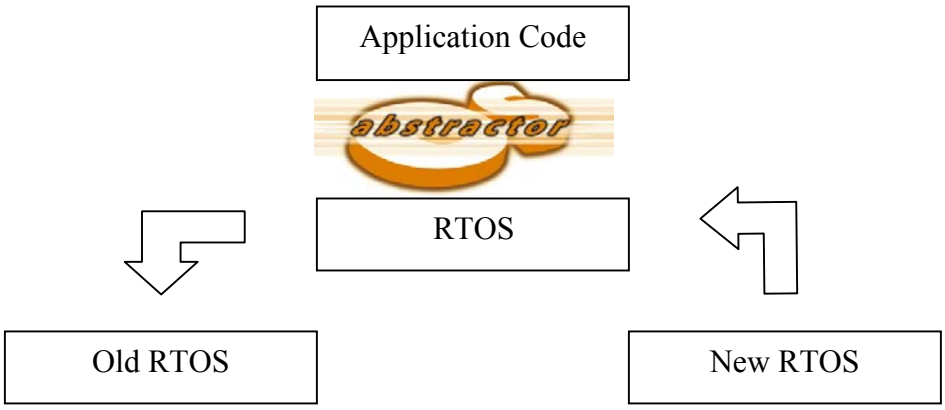
In software programming practices, defining the software architecture is a vital element for the proper design. A strong architecture simplifies downstream activities such as adding new features, maintenance, bug fixing, and optimization.

In a software project using OS Abstractor, the MapuSoft product fits into the application similar to this diagram:



From an architecture viewpoint, your application has a clear and distinct separation according to solid programming practices. Your application code is making calls to the underlying operating system, but the untidy coupling between your application and the RTOS is eliminated.

Should you need to move your code to another operating system, the new target RTOS is swapped out:



Your code has been completely unaffected and you have moved your code to a new RTOS without modifications to your application. Your project has eliminated potential delays due to the existence of a solid architecture.

OS Abstractor – Design Philosophy

OS Abstractor provides a level of abstraction to the underlying RTOS while minimizing the overhead associated with other abstraction products. The MapuSoft product is designed as a library, so adding the code to your application is as easy as modifying your project parameters for the linker/loader. You do not have to spend time updating each of your project files, since you can quickly add the library to your project and get started.

To lower the impact on your application, the code overhead is kept to a minimum. Many "roll-your-own" abstraction solutions add significant overhead to the application code. OS Abstractor's code size is between 4K – 30K, depending on the architecture, tools, and operating system. Since the Abstractor product is written in "C," it easily integrates into your existing application development.

Many home grown and other translation products perform the abstraction by using a wrapper implementation. Wrappers are easy to implement and straightforward to understand, however they suffer from several problems. First, the in-house wrapper methods do not always cover the majority of the operating system function calls. OS Abstractor provides strong coverage for the most popular functions used in the real time operating system.

Another drawback of an in-house wrapper is that a proprietary solution only abstracts the function calls. OS Abstractor abstracts all aspects of the operating system, not just the function calls. The control blocks, APIs, header files and data types are all abstracted, providing you with a complete separation from your operating system.

When possible, OS Abstractor uses the compiler preprocessor to parse the application code for the MapuSoft function calls, and then translates them to the appropriate operating system call. By leveraging the compiler preprocessor, we are able to eliminate the overhead associated with the wrapper solution. The preprocessor performs the necessary mapping and translation between the application code and operating system calls. As an application writer, you can write your code using "C" style function calls and a consistent interface and let the MapuSoft technology perform all the necessary translation for the specific operating system.

OS Abstractor - The Solution

OS Abstractor provides an excellent choice for your software project and offers several benefits for your software application.

OS Abstrator reduces the learning curve

One of the most under-estimated tasks in a software project is the ramp-up time that developers must go through to become productive on a new operating system. With the majority of the operating systems, the cost of learning a new one can be significant. OS Abstractor offers an easy-to-learn interface that can be re-used across projects, reducing the learning curve and allowing people to start coding sooner.

Good programming techniques reduce un-necessary bugs and help maintenance

Having a solid architecture brings many benefits to a software project. A solid architecture allows for better testing during both the unit testing and system testing phases. The downstream project activities of adding new features and maintenance are significantly easier when an application has a solid architecture.

OS Abstractor eliminates choice of RTOS risk

Coupling too closely to the RTOS can lead to too much dependency on the future of the operating system vendor. As vendors drop support for your hardware platform or prices change, your project could be jeopardized. Using the OS Abstractor helps reduce dependency on the RTOS vendors.

OS Abstractor provides additional functionality to your RTOS

Vendors vary widely, not only on the terms they used to describe their operating system, but also on the other "extra" features that are provided with the offerings. With OS Abstractor, you can use the well-established UNIX standard I/O subsystem and a set of debugging C library calls with any RTOS that you choose.

The UNIX I/O interface comes with every abstraction product that is distributed. The I/O interface provides the standard I/O interface calls like open, create, link, ulink, read, write, ioctl. The I/O system can add a standard interface and can fill missing holes in other operating system offerings.

Many RTOS implementations lack a simple printf functions. Printf is a primitive, yet commonly used debugging outlet. OS Abstractor provides a reentrant printf function that can be used even if the operating system does not support a printf.

About OS Abstractor:

With OS Abstractor, developers can instantly have their code running under multiple RTOSs. They can also develop new code using the OS Abstractor APIs to have multiple RTOS support with the new application code. Our OS Abstractor solution provides insurance for a RTOS change in the future because the application developed using the OS Abstractor APIs will provide a good portability solution for the application.

Abstraction solutions are available for:

- ✓ Nucleus PLUS from Mentor Graphics
- ✓ Precise/MQX from ARC International
- ✓ ThreadX from Express Logic
- ✓ Linux from several vendors
- ✓ µITRON from several vendors

About MapuSoft:

MapuSoft Technologies provides royalty-free tools and support that allow customers to quickly move their products from one Real Time Operating System (RTOS) to another, and we believe that our advanced software and vision will revolutionize the embedded software industry.

Our vision at MapuSoft Technologies is to give our customers added flexibility and control in their respective product development life cycles. The flexibility and control that MapuSoft can provide your company will allow a MapuSoft customer to:

- ✓ Avoid high cost associated with royalty base RTOS vendors or customer services
- ✓ Have access to a full source code format. As a result, customers will be able to customize or optimize their application to their specific needs, and problems can be pin pointed and resolved quickly
- ✓ Take full advantage of the existing code base and knowledge base without rebuilding or relearning new software. This will cut down on time to market and allow the MapuSoft customer to stay ahead of the competition
- ✓ Select a suitable OS for their products without feeling locked into a particular one The OS Abstractor allows the customer to choose the RTOS that best fits their product lines' unique needs
- ✓ Have no restrictions on the target OS, supporting tools and the hardware as a result of using our products. The OS Abstractor will work on any CPU, and in most cases we can provide missing functionalities
- ✓ Easily expand product lines to support more than one RTOS

Please download our evaluation copy of the OS Abstractor product at:

http://www.mapusoft.com/latest_info/FREE_EVAL_FORM.html

or email us at info@mapusoft.com for more information