

Open Technology License

The step beyond open source

"Open" is one of those words that is used a lot. It came into being as a reaction to the closed software offered by many software vendors. Since then Open and Free have become intermingled although it is not by lack of variants of open source licensing schemes. So why did we create another "Open Technology Licensing" scheme? It all goes back to some 10 years ago when we conducted a market research for one of our customers in the space sector. They wanted to find an alternative for a distributed RTOS. There aren't that many RTOS around that can handle multi-processing systems in a transparent way. The shocking discovery however was that the bulk of all RTOS, commercial or open source could not be certified, not even when available for years and that is true for a lot of software products. There are exceptions, but to start with, design documents are a rarity. These can be reverse engineered. Decent comments in the source code even less, not to speak of obfuscation techniques. The software might have been tested, but then there is a license that says "right to use at your own risk". And what about a user manual? What about (formal) proofs that the software meets the specifications? Not a wonder that the software industry, except for a few giants, is often a service industry. Thus, is there a better alternative?

The first observation to make is that proving and certifying software is close to impossible, after it was developed. Testing only proves the absence of errors the specific tests can detect. The tested code can still fail if the usage pattern is different from the one used by the test. If we look at the requirements of safety standards, then one sees that the development process is as important as the code itself. It comes down to minimising the risk that the software doesn't meet the specifications. Even formal proofing of source code has its limits. If one line of code is changed, how can one be certain that all specifications and model properties are still met?

So we came up with the logical step beyond "open source". We call it the "**Open Technology License**". First of all, it is a software license, copyright applies. It comes with source code, the formal models, the design documents, the build system, the user manual, the unit tests, examples, simulators, supporting tools like code generators, visual editors and debugging tools. It's not just software, it is technology. Licensees can reuse, modify, rebrand and create themselves binary licenses in exchange for a royalty fee. And if needed, there is an optional **Qualification Package**.

Altreonic is a developer of technology with a focus on the high reliability / safety engineering market. Hence, the focus is on Research & Development. We have proposed Open Technology type Licenses from the very beginning based on the previous 25 years of experience that standard distribution doesn't work very well for high-end software. Some companies address this issue by using the software to sell expensive engineering services. But this means that the software is sometimes deliberately kept simple, not well documented, etc. But if you focus like Altreonic on high reliability ("trust"), this is not a good option as it conflicts with the requirements for safety engineering. It can also turn out to be a costly and risky option for the customer.

This is where the Open Technology License comes in. What licensees acquire is the technology (fully documented) allowing higher margins, more added value but also more flexibility. Licensees have full access and are able to support the customer much better. Altreonic will support the licensees, but is often not involved with licensee's customers. A small royalty applies because support is proportional to the number of users. The licensee will handle most of the support and its engineers will be well trained. The annual support and maintenance provide the licensee with new versions and better support tools.

There is an investment up front but this is a small fraction of the time and cost to develop the equivalent technology from scratch. Once the licensee integrates the technology with the other products and engineering services he offers, he will become more productive and will enjoy better margins. Hence, the initial investment is quickly recovered.

What our customers / technology partners acquire is an integrated, unified approach, leading to more reliability and higher productivity. It allows Altreonic to focus and to deliver new leading edge technology. Hence, it is a win-win proposal. Available for OpenComRTOS Designer, Safe Virtual Machine and GoedelWorks

Besides the Open Technology License we offer binary with source code licenses as well to the occasional user. Interested in an Open Technology License? Contact us. **OpenLicense** (@) altreonic.com

Altreonic NV - Gemeentestraat 61A b1, B-3210 Linden, Belgium.

info.request@altreonic.com, www.altreonic.com, tel. +32 16 20 20 59



"From Deep Space To Deep Sea"

Annex: detailed list of Open Technology License (as for OpenComRTOS Designer).

- **Formally modeled and verified network-centric OpenComRTOS**
 - Represents 7 years of development, but has a history of 3 previous generations since 1991
 - Fully distributed, supporting heterogeneous targets (processor as well as communication medium)
 - Fully transparent and straightforward concurrent and parallel application development
 - Very resource efficient (typically 5KB to 15KB/node), therefore very good performance
 - Formal models of the services (TLA+)
 - Source code with meaningful comments.
 - Design documents
 - User API manual (generated from the source code)
 - Tutorials
 - Example code
 - Test suites
 - Build system
- **Simulator and cross development:**
 - Allows cross developing multi-node systems on a PC (Windows or Linux)
 - Source code only needs to be recompiled for mapping on heterogeneous embedded targets
 - Host node integrates seamlessly with an embedded target network.
- **Visual Designer:**
 - Visual development environment for OpenComRTOS (Windows or Linux)
 - Context sensitive help system
 - Code generators:
 - Used in OpenVE or command line
 - Generate most of the source code based on application and platform metamodels
 - Generate routing tables
- **Visual Event Tracer:**
 - Customisable event trace visualization tool
 - Also calculates resource statistics
 - Displays interprocessor communication
- **Open System Debugger:**
 - Allows to retrieve OpenComRTOS status data from any node
 - System wide Peek and Poke capability.
- **Application support modules:**
 - Host server for stdio and graphics across a network
 - Windows multimode simulation / cross compilation environment
- **Drivers like TCP/IP, CAN, UART, ...**
- **Target support included for your selected processor**
- **Optional:** Safe Virtual Machine for C.
- **License Conditions:**
 - Once-off license fee costing less than a year of development
 - No runtime royalties
 - Right to customize the software
 - Right to rebrand under a different name
 - Right to produce binary licenses
 - Right to set the reselling price
 - Each binary license created carries a small royalty fee to Altreonic.
 - Annual maintenance and support: 20%/yr (starting 2nd year) on all licenses.
 - Porting and customer specific developments upon request.
 - Two weeks training in our offices or at licensee's premises.
 - Special conditions for academic users available.

“From Deep Space To Deep Sea”

