

Assembler to C Software Migration Tool

In today's business, there are several compelling reasons to migrate software applications from old legacy computer systems to current technology. These reasons include

- The high cost of maintaining the old aging hardware systems
- Lack of knowledge of the legacy system software and hardware
- Older systems are more prone to the date problems

Despite these reasons, many legacy systems are still critical to a company's daily operation. This fact is perhaps the most compelling reason to migrate - the company depends on this application to conduct business, but as each day passes, the likelihood of system failure increases due to the above reasons.

Dot4 understands this scenario, and is dedicated to helping clients migrate from their legacy system to an open system solution. Many legacy applications were developed on the so-called "super mini-computer" technology of the late '70s and early '80s. The premier mini-computers were manufactured by companies such as:

- Digital Equipment Corp. (DEC)
- Interdata / Perkin Elmer / Concurrent Computer Corp.
- Harris Computer Systems (now merged with Concurrent)
- SEL Computer Systems / Gould / Encore

These applications are well suited for migration since the vendors have, by and large, shifted their focus to open system (i.e. UNIX or Windows/NT) solutions. The legacy hardware and operating systems are no longer available, and maintenance contracts, while available, are extremely expensive.

With today's technology, these mini-computers can be replaced with smaller, less expensive, microprocessor-based systems. These systems are typically based on an open operating system, with some level of Portable Operating System Interface (POSIX) compliance.

One marketplace that mini-computers were very well suited for is the Real-Time market. This market comprises applications such as:

- Financial Trading Systems
- Flight Simulators
- Process Monitoring
- On-line Gaming

Why were these apparently diverse applications and mini-computers a good match? Each application, though different in nature, has a common core of system requirements. From a hardware perspective, each needed a considerable processor performance which sometimes resulted in a multi-processor configuration, large physical memory which could be shared among multiple processors, and substantial disk space.

From a software perspective, each of these types of applications have several threads of execution (requiring true multi-tasking), intertask communication and synchronization, coordination of access to critical data by any task (usually a Test and Set instruction), large use of proprietary operating system services, and finally, most were written in FORTRAN and or ASSEMBLY language containing between 100K to millions of lines code.

What is the challenge for a company such as Dot4 to migrate, port or "rehost" a legacy system? The answer takes on many facets of knowledge and experience:

- Knowledge of the legacy computer system and its' architecture
- Knowledge of the legacy operating system and its' services
- Knowledge of the application language
- Knowledge of the target computer system architecture
- Knowledge of the target open system operating system
- Experience to determine the migration of old to new system services

From experience based on past and current migrations, Dot4 is constantly developing various "in-house" software migration tools for their engineering staff. The staff is mainly composed of senior engineers, each with 15+ years of experience. This places Dot4 in the unique position of developing software tools to challenge a very complex problem. One such software tool worth discussing here takes a sophisticated approach to convert low level assembly language source code to ANSI "C" source code.

Having analyzed this problem many times, Dot4 concludes that a quality conversion tool for ASSEMBLY language to "C" can not be a straight line by line translation. A line by line solution will produce unreadable, and un-maintainable, "C" code. That is to say, the resulting "C" code will look nearly identical to the ASSEMBLY code, but it will compile with a "C" compiler. A conversion should maintain closeness to the original code in recognition of validation problems and integrate the code in recognition of eventual maintenance issues. The solution that Dot4 has undertaken incorporates the following capabilities:

- Maintenance of both comment blocks and inline comments
- Integration of multiple lines of ASSEMBLER to generate Expressions
- Integration of multiple lines of ASSEMBLER to generate Conditional Branch's
- Recognition of register and variable data types based on instruction usage
- Recognition of both low level (register) and high level (parameter) calling conventions
- Recognition of calling convention parameter passing and parameter receiving
- Recognition and tracking of "standard" register convention usage
- Recognition of type casting and other hardware/firmware intrinsic instructions
- Recognition of assembly pseudo-opcodes, including conditional compilation
- Resolution of pointer and/or index register usage ambiguity by heuristics
- Translation of "include" filenames, including path specification
- Flagging of suspicious or questionable coding, sometimes an indication of original code bugs

Dot4 believes at this time, it can produce a 90% to 95% conversion of ASSEMBLY source code to "C" source code. While a 100% conversion is desirable, it is not a practical goal - any experienced engineer in this field would say that it is nearly impossible due to the extreme liberties that an ASSEMBLY language programmer can take.

As an example of the increased productivity that has been realized with this tool, consider an application consisting of 100,000 lines of original ASSEMBLY source code. By reducing the level of effort down to about 5,000 to 10,000 lines of ASSEMBLY language, hundreds of thousands of dollars can be saved in a large project.

Whether you need to change programming languages, computers, or operating systems, or want to analyze or re-engineer your existing software, our software and services can help you.

For further information on this specific tool and Dot4's services and products go to the following page.

<http://www.dot4.com/salesandcontact.html>