



Success Brief
Intel® Parallel Studio XE
 Video Compression

envivio®



“Having such a tool this early in the development stage frees the validation from trivial bug reports and gives our engineers the opportunity to code more efficiently from the very beginning of the product cycle.”

Jean Kypreos
 Advanced Video Processing
 Team Manager
 Envivio

Helping ensure the best video quality and performance

Intel® Parallel Studio XE helps Envivio create safe and secured code

Company	<p>Any video—Anytime, Anywhere</p> <p>Envivio’s goal is to make the world’s video content universally enjoyable by all viewers, on any device, across any network, at any time. Since its founding in 2000 as an inventor of video encoding technology, Envivio has amassed dozens of patents, pioneered video-over-IP methods, and continually leads with support for emerging technology. Today Envivio solutions deliver millions of content streams to hundreds of different styles of mobile phones, set top boxes, and PC platforms, on behalf of content owners, telecomm operators, cable and satellite companies and mobile service providers in every market in the world. Using codecs from standard H.264 through the Elite and Extreme codecs developed in the Envivio labs, Envivio optimizes the viewing experience for every screen, on every network. Deployment and support are backed by a global staff that is familiar with local technical, regulatory and market guidelines encountered around the world. Envivio is headquartered in South San Francisco, California and has offices in Rennes, France; Beijing, China; Tokyo, Japan and Singapore.</p>
Mission	To grow the quality of the 4Caster C4 ^{2*} product to achieve the best video quality, best performance, and fastest time to market
Product	4Caster C4 ² is a real-time transcoding application that supports one channel of high-definition or multiple channels of standard-definition encoding for IPTV, Internet TV encoding up to VGA resolution, and 3G Mobile TV encoding.
Challenge	As code size grows, a simple IDE is no longer enough to properly follow large sections of code. Larger code generates more bugs of an increased complexity, compromising reliability and time to market.
Results	Sections of unused but executed code were removed, improving performance. Many areas of the code also saw improved safety and quality, and were delivered on time.
Impact	The likelihood that a bug will return to the encoding team has been lowered, while both developer ramp-up time and time to market have improved.

Challenge: Why Envivio’s products benefit from utilizing parallelism

Depending on the formats, several cores are required to perform a single encoding. In particular, for high-definition content repurposing from MPEG2 to H.264, a single core is not enough to achieve the encoding task in real time. In that case, both multicore and multiprocessor (DP) technologies are required to provide the minimum horsepower for the application.

For standard-definition content, a high level of video quality is required for the broadcast market segment. This quality is obtained using heavy-processing algorithms while keeping the real-time constraint. At the same time, in order to keep a modular level of density (i.e., 2/4 SD channels per platform), a parallelization (2/4 cores per channels) schema is mandatory to take advantage of the platform capabilities.

Intel® Parallel Studio XE combines Intel's industry-leading C++ and Fortran compilers, libraries, and error-checking, security, and profiling tools into a single tool suite that enables high-performance computing and enterprise developers to boost application performance, security, and reliability.

Used independently or together to meet the needs of workflows and development teams, this ultimate all-in-one performance tool suite helps produce fast, reliable, and secure applications that take advantage of advanced processor features like Intel® AVX and multicore, and scale for manycore.

Intel® Parallel Studio XE includes Intel® Composer XE, Intel® Inspector XE, and Intel® VTune™ Amplifier XE.

Build High-Performance Serial and Parallel Applications

Intel® Composer XE delivers continued improvements to boost performance of C++ and Fortran applications with additional built-in optimizations, powerful new parallelism models, and libraries. The enhanced Fortran compiler offers key Fortran 2008 features, including co-arrays and more complete support of Fortran 2003.

Prevent Coding Defects Early in the Development Cycle

Intel® Inspector XE is a powerful, easy-to-use serial and parallel error-checking tool for C/C++ and Fortran developers. It detects challenging threading and memory errors to help improve application reliability. When used in Intel® Parallel Studio XE, it offers static security analysis (SSA) for improved code integrity.

Tune Applications for Scalable Multicore Performance

Intel® VTune™ Amplifier XE is a powerful threading and performance profiler for C/C++ and Fortran developers who need to understand an application's serial and parallel behavior to improve performance and scalability.

To enjoy a free 30-day evaluation of Intel® Parallel Studio XE, download it here: www.intel.com/software/products/eval.

The density problematic is induced by the economic equation of the BOM cost: more channels per platform lead to smaller COGS. In Envivio applications, the H.264 codec is the most "time-critical" software component. Particular care is taken in the parallelization balance of this module.

Results

Using the static security analysis (SSA) capability in Intel Parallel Studio XE to analyze sample code, all the falsely inserted bugs were detected. SSA was then introduced into the process with a real test case: the encoding core of the 4Caster C4². Based on the results of the analysis, some sections of unused but executed code were removed, thus improving performance.

Even if no examples of potential memory corruption were found, many areas of the code saw improved safety, adding error management and tracking the potential null pointer errors. Special care was taken with the in-and-out interface as the last test for SSA allowed the front end of the encoder to be secured. Misuse in the product of this critical module will result in immediate detection.

On its first launch, SSA found approximately 300 errors, most related to potential pointer usage. Five critical bugs were discovered, in addition to three minor bugs and approximately 50 vulnerabilities. As a result, some detected areas were rewritten, most of the dead code was eliminated, and many critical pointers were secured.

Globally, more reliable code was delivered to the integrators, minimizing the possibility that a bug will return to the encoding team.

How Intel® Software Development Products assisted

The SSA capability in Intel Parallel Studio XE was used to analyze the core module of the product: the H264 encoder. It was set to its maximum sensitivity, helping reduce the floating validation time, while improving product quality and facilitating on-time delivery.

SSA also offers more resilience to the code. This protection promoted developer confidence in the code, making it safe to connect one piece of software to the other. Safe and secured code will return errors immediately, preventing long, demanding debugging sessions. Used in the early stages of development, SSA fixed bugs before they appeared, saving time and resources. By combining with Intel Inspector XE, also in the early stage of development, greater dynamic error resilience was achieved.

The SSA tool was easy to set up, launch, and use. And because results are displayed within Intel Inspector XE, there was only a single interface to master. In addition, access to the source code was fast, and problems could be found and fixed within minutes.

SSA is fully integrated with the compiler, and results could be manipulated in many ways. The user-friendly interface and filters enabled the developers to focus on a category of problems, while the documentation clarified tricky and advanced coding issues.

Optimization Notice

Intel® compilers, associated libraries and associated development tools may include or utilize options that optimize for instruction sets that are available in both Intel® and non-Intel microprocessors (for example SIMD instruction sets), but do not optimize equally for non-Intel microprocessors. In addition, certain compiler options for Intel compilers, including some that are not specific to Intel micro-architecture, are reserved for Intel microprocessors. For a detailed description of Intel compiler options, including the instruction sets and specific microprocessors they implicate, please refer to the “Intel® Compiler User and Reference Guides” under “Compiler Options.” Many library routines that are part of Intel® compiler products are more highly optimized for Intel microprocessors than for other microprocessors. While the compilers and libraries in Intel® compiler products offer optimizations for both Intel and Intel-compatible microprocessors, depending on the options you select, your code and other factors, you likely will get extra performance on Intel microprocessors.

Intel® compilers, associated libraries and associated development tools may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include Intel® Streaming SIMD Extensions 2 (Intel® SSE2), Intel® Streaming SIMD Extensions 3 (Intel® SSE3), and Supplemental Streaming SIMD Extensions 3 (Intel® SSSE3) instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors.

While Intel believes our compilers and libraries are excellent choices to assist in obtaining the best performance on Intel® and non-Intel microprocessors, Intel recommends that you evaluate other compilers and libraries to determine which best meet your requirements. We hope to win your business by striving to offer the best performance of any compiler or library; please let us know if you find we do not.

Notice revision #20101101

