AUTODESK SOFTIMAGE INTERACTIVE CREATIVE ENVIRONMENT (ICE)
TECHNOLOGY OVERVIEW

Intuitive new platform for developing powerful tools and complex 3D effects

EXECUTIVE SUMMARY
Autodesk® Softimage® Interactive Creative Environment (ICE) is an open, scalable platform that helps users extend the capabilities of the Autodesk Softimage software quickly and intuitively using a node-based dataflow diagram. This paradigm means that 3D artists may create complex 3D effects and tools extremely quickly without writing code. Powering ICE is a high-performance parallel processing engine that takes advantage of multi-core workstations – a first for a general 3D architecture – giving users high performance and scalability. The result is a giant leap forward in production efficiency and consistency, enabling higher production standards.

INDUSTRY BACKGROUND
3D professionals have almost always been limited by computing power. Host platforms for Digital Content Creation (DCC) applications historically have been unable to provide enough computing power to deliver the level of interactive performance required to create the complex visual effects and simulations used in movies, games and 3D visualization.

Recent advances in computing technology have changed all this. Increased processor-core density, coupled with new ways to leverage the Graphics Processing Unit (GPU) for general computation, have created a rich computing environment. Now, the challenge is no longer to find enough computing power for DCC tools, but to build DCC tools capable of harnessing the newly available computing power.

In order to produce unique, high-calibre work, modern games, film, and television production teams regularly develop custom tools and effects that push the boundaries of what’s possible. Harnessing computing power is essential for studios that differentiate themselves through invention and innovation – for them, power results in creativity.

A philosophical shift has begun to take place where studios have started viewing DCC applications as extensible platforms, rather than products ready to be used off the shelf. Developers and Technical Directors (TDs) tend to customize DCC applications to create specialized tools used by artists. While this kind of customization gets the job done, it is peripheral to the core DCC technology and rarely yields solutions that can deliver robust performance and a high level of reusability.
Traditional DCC Application Architecture (left) - This diagram illustrates the traditional DCC application, with closed data—an environment where the tools developed by the manufacturer use a preferential development environment, while customers and third party developers are left to extend the application through application programming interfaces (APIs) and scripting; slow processes which don’t always provide the performance or robustness required for re-use.

ICE - EXTENDING AUTODESK SOFTIMAGE FROM PRODUCT TO PLATFORM

The ICE environment represents Autodesk Softimage functionality using a collection of nodes, each with specific capabilities. Developers and TDs can connect nodes together, visually representing the data flow, to create powerful tools and effects that can be packaged and distributed for use by artists.

The ICE environment (above) – Softimage functionality is displayed using a node-based dataflow diagram. Each node has specific capabilities. Users connect nodes to create a desired effect. A group of nodes can be packaged together into a “compound”, which is displayed as a single node. As a result, highly complex tools can be deployed and re-used without writing code.

This graph-based approach is faster and far more accessible than traditional development using scripting and/or compiled code. It offers a cohesive development environment that large facilities can use to quickly develop complete tools from scratch. Smaller teams can use and modify tools provided by Autodesk and third party developers to suit the needs of a production.
ICE is also the internal development platform being used by the developers of Softimage. This means that rather than developing ‘closed’ systems that give developer’s limited access to the software, Autodesk will help give user control. If a user is not obtaining a desired effect using the internal Softimage feature set, the user no longer has to wait for someone to write a plug-in or for the next version of the software to be released. He can simply open it in ICE and modify it to obtain the desired effect.

ICE is fast, interactive and offers significant advantages over traditional scripting and coding:

- It’s **visual** – ICE uses a node-based graph for creating and modifying 3D tools and effects. Users can see exactly what’s happening without examining code.
- It’s **interactive** – Users can create and modify their effects, and see the results executing immediately.
- It’s **fast** – Tools developed in ICE run as fast as, or faster than compiled code. ICE takes advantage of multi-core processing and helps to scale almost linearly.
- It’s **predictable** – There’s no custom code or scripts where a lot of time is spent debugging and compiling. Users can see the results immediately and develop quickly.
- It’s **flexible** – As mentioned above, users don’t have to wait for someone to write a plug-in, if they want a feature altered or extended. They can simply open it in ICE and edit it.
- It’s **safe** - ICE comes with a versioning system, so that users can roll back changes easily.
- It **integrates** nicely – ICE comes with a suite of import/export tools to allow for data interoperability with current pipelines.
- It gives **control** – ICE comes with a visual debugging system for solving complex problems. It also has performance analysis tools to help users optimize effects.
- **It enables sharing** – Once a tool is built, it’s easy to package the compound and share it.

**ICE architecture (left)**

With ICE, tools become equal citizens. Unlike the traditional approach where tools created by the application developers have a distinct advantage, ICE gives everyone access to the full underlying 3D platform.

While the tools that ship with Softimage provide stellar performance and amazing creative capabilities out of the box, the open architecture means that users can customize and extend these tools as needed.
ICE - SCALABLE PERFORMANCE

Underlying the ICE development environment is a high-performance parallel processing engine optimized for multi-core computers.

ICE leverages multiprocessing using a two-stage graph-evaluation process:

Stage 1 — dynamic graph analysis: this is akin to code compilation, but occurs at run-time. Graph analysis happens only when the ICE graph changes—for example, when the user adds or removes nodes, makes new connections, and breaks existing connections.

Stage 2 — execution: this happens whenever the ICE graph requests an output. The key to the ICE engine’s highly responsive user environment is found in the way it evaluates the graph in parallel without locking.

The result is extremely high performance for complex 3D scenes. Effects often execute in real time or faster, even while a user interactively adjusts parameters or components of the scene. This performance scales to take advantage of all available computing power.

Above: An ICE flocking simulation takes advantage of the eight cores on a workstation. The simulation ran to hundreds of thousands of particles, each instanced to fully rigged and animated characters.

Since ICE is lock-free during evaluation, as the amount of data increases it can scale near-linearly to the number of processor cores available. For instance, moving from a four-core machine to an eight-core machine allows ICE to produce effects on datasets of twice the size and achieve nearly identical performance.

A small additional overhead is required to stream the data to graphics hardware and/or other targets (usually done in a single-threaded context). The cost of this overhead shrinks as the complexity of the effect increases. In real-world examples of low-to-medium complexity it is common to see 90%+ CPU usage across four CPUs.
ICE AND THE CREATIVE PROCESS

ICE is an ideal creative environment, offering distinct advantages over traditional script and plug-in based customization:

**Faster and Easier Development** - Writing, testing and debugging with proprietary scripting languages or complex programming languages takes a lot of time and specialized knowledge. Developing in ICE is visual, interactive, and requires no code compilation or optimization. It's easy to make changes, even after production starts.

**Facilitates Collaboration Between Artists and Developers** - ICE facilitates the creative process by providing a collaborative framework for artists, developers and TDs. Traditionally, artists needed to communicate a creative vision to developers whose job it was to turn that vision into code. Without a true collaborative environment, important details were often lost in translation. With ICE artists can not only create and refine custom tools and effects with developers, they can make the final adjustments themselves.

**Easy Deployment** - ICE was designed from the ground up for easy deployment in pipelines serving from a few users to hundreds. It has powerful versioning and revision capabilities built in, so TDs and developers can package tools and effects, distribute them to a single user or an entire facility, or even share and sell tools online. The ICE versioning system allows backwards compatibility, allows multiple versions of a tool to coexist, and provides a coherent process to propagate changes to any number of users.

How ICE relates to individuals and teams (above) – Because ICE is visual and interactive, it means that artists and developers can collaborate much more easily than with traditional scripting and plug-in based development. With built-in packaging and versioning, ICE tools can be deployed easily to entire teams.