Modernizing UI Visual Development

The Rig project enables more dynamic UI visuals with flexible, real-time interactivity using the GPU. It makes workflows more intuitive for designers by giving them immediate feedback on their work, showing actual capabilities of the target device. This approach reduces the burden on application developers, speeding the workflow between designers and engineers.

Rig is an open-source tool and community, still in its early stages, that seeks to enrich UIs using the capabilities of modern GPUs. It offers benefits to people in the following roles:

• **UI designers** get greater insight into how their designs work on the target hardware, better connecting the design process to the final product.

• **Application developers** avoid the burden of advanced GPU programming, while GPU programming specialists retain full freedom to develop advanced graphics algorithms within Rig itself.

• **Business decision makers** can offer UIs based on real-time interactivity, helping differentiate their products for a competitive advantage.

**Challenges in UI Design**

Despite the pervasiveness of feature-rich GPUs, UIs rarely use these resources to their full potential. The technologies that underlie current UIs are all based on the legacy PostScript rendering model, which is over 25 years old, and at its heart is based solely on 2D shapes (mostly rectangles), images (including gradients), and text. These primitives are simply combined in layers, which significantly limits UI designers.

Building UIs that get the full value out of the features and capabilities of modern GPUs requires very specific programming approaches, which are precluded by the traditional PostScript model. By tapping more of the GPU, UI designers can explore new perspectives, textures, and lighting, to create more visually dynamic and creative results.

Another factor that holds UI designers back from innovation is that UI toolkits tend to be designed to favor developer convenience over visual expressiveness. Both markup languages (e.g., HTML, CSS, and QML) and toolkit APIs (e.g., GTK and QT) tend to be built with engineers in mind, with value to designers considered as only an afterthought (if at all).

Popular tools favored by designers (e.g., Adobe Photoshop*, Adobe AfterEffects*, Adobe Illustrator*) focus on static image creation and offline video processing. Moreover, the sophisticated capabilities of these tools, such as ray tracing, are not suited to resource-constrained mobile devices. Therefore, a gap in communication exists between designers and engineers, which can necessitate more iterations and increase a product’s time to market.
Moreover, designers have no systematic or direct way to connect their work in detail to the actual physical device it ultimately will be used on. They can only make informed speculation about the actual user experience, and they also do not have an easy, routine means of understanding the effects of design changes on performance and power.

Rig seeks to address these limitations by providing a tool approach that tailors itself specifically to the experience the device will provide, rather than relying on the use of separate, off-the-shelf toolsets by designers and engineers. While people in both roles will continue to use their familiar, preferred tools, they will use Rig to connect the two roles. For example, designers may use Illustrator and Photoshop (as well as various sources for 3D models, audio, video, or modules of logic) to create assets that they will then use in Rig, as shown in Figure 1.

To make use of these assets, Rig uses advanced algorithms, largely pioneered in the gaming industry, that take advantage of GPU-specific capabilities. Importantly, the GPU programming is handled behind the scenes by Rig, exposing the functionality of the algorithms to designers and application developers in an intuitive form that does not require them to have or develop specialized GPU programming knowledge.

**Bridging the Gap Between Designers and Engineers**

Application-development teams tend to include a number of specialized, interconnected roles, including joint work between designers and engineers on UI development. While the individuals who fill these different roles may have significant common ground where they relate to each other, they also have different main concerns, in terms of their focus within a project and the associated priorities. Key aspects of the two roles can be characterized as follows:

- **UI designers** with expertise in human-machine interaction and visual design typically create the first representation of the UI, which identifies the controls, their functionality, and their visual appearance.

- **Development engineers** work from those designs to prototype and develop a software representation that provides the specified appearance and functionality.

Significant potential exists for gaps in communication between these two roles, as suggested by the differently focused tools used by designers and engineers, which in fact represent a larger mismatch within the workflow. For example, designers typically begin their work free from many of the technical constraints that will ultimately need to be addressed by the engineering phase.

Because of that factor, problems often arise in the coding phase of UI creation that could not easily have been foreseen during the design phase. Functional disparities may arise in an actual, functioning UI versus the hypothetical one in the design phase, power usage may prove to be a barrier, or the design may not fit the device form factor as intended.

![Figure 1. Rig prototype.](image)
Designing to an Experience, Not a Tool

From a workflow perspective, a central challenge in the interactions between designers and engineers is that typical design tools such as Photoshop are not intended to place constraints on designers to ensure the art’s suitability for real-time rendering and specific design constraints. Rig therefore takes the approach of exposing the real-time rendering technology used for deploying the UIs on target devices, explicitly constraining the design process based on the capabilities of those devices.

Related to providing constraints as guidance to design teams, Rig connects the capabilities of the target hardware into the design tool. Changes made in the visual-design environment are immediately synchronized with the device, with no explicit deploy step. Therefore, immediately after performing an operation on a UI, the designer can pick up the device he or she is designing for and experience the associated changes in how the UI looks, feels, and performs.

Rig will also extend this real-time feedback to designers using profiling data that will quantify the impact of visual-design changes on CPU and GPU performance and power consumption. This capability draws greater value out of existing profiling by using it earlier in the UI-development process. Profile data will become a design aid, in addition to being a diagnostic tool. Designers will also gain an intuitive understanding of the abilities and limitations of the target hardware and more tightly tie their designs to the environment where those designs will actually be used.

Planning the Road Ahead

The Rig project is still at an early stage, with most of the development being done at Intel’s Open Source Technology Center. This work is a good example of how open-source work at Intel complements core Intel® Graphics hardware and software technologies, with an eye toward advancing the capabilities of the industry as a whole.

Early work on Rig to extend the UIs’ use of GPU resources has focused on algorithms widely used in the gaming industry that emphasize photorealistic effects. Additional algorithms will also be added in the future, but examples of Rig’s present capabilities include the following:

- **Depth-of-field effect**, changing focus on an object based on its distance from the viewer
- **Real-time lighting**, including basic effects such as shadows and how different materials react to light
- **Loading 3D models** exported from tools such as Autodesk Maya*, Autodesk 3ds Max*, and Blender*
- **Bump mapping**, for efficient modeling of how light plays on bumpy surfaces

Present work on visual effects also includes non-photorealistic effects such as those that systematically abstract images, as well as purely artistic styles. The next priorities for the Rig project are to add the capabilities needed to build complete UIs, beginning with the ability to respond to events and handle imperative application logic. Rig also plans to be able to handle code snippets as assets and integrate data sources into the design process, which will further aid collaboration between designers and engineers.
INTEL AND OPEN SOURCE:

People think of Intel as a hardware company, and of course, that’s true; but we are also consistently one of the top two contributors to Linux* and one of the top employers of software engineers in the world.

The Open Source Technology Center is the heart of Intel's open source work, which complements closed-source solutions, for an ecosystem that shapes the future of user experiences.

Learn more at 01.org

Join Our Community: Help Shape Rig and the Future of UI Creation

Rig has taken its first steps toward replacing the legacy PostScript approach to creating UIs. The project aspires to tap the resources of the GPU, without requiring designers and application developers to be GPU programmers. But its vision goes further, hoping to unify the UI development workflow by providing designers with far greater insight into how their work functions on the target device.

To realize Rig's potential, the Open Source Technology Center is soliciting feedback, feature suggestions, and code contributions. The growing community is looking for perspectives from all over, including academics, students, professional UI designers, engineers, user experience researchers, and business people, among others.

Participate in the community.
Help create a better world of UIs.

www.01.org/rig

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information. The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order. Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or by visiting Intel's Web Site http://www.intel.com.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark* and MobileMark*, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

*Other names and brands may be claimed as the property of others.

Copyright © 2013 Intel Corporation. All rights reserved. Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

0713/SS/Mesh/PDF 329110-001US