Universal Pointer

The universal pointer is a new design pattern that abstracts reading and writing from:

- Variable or object member
- Pointer, reference, or smart pointer
- Get/set method pair
- Get/set function pair

It can be implemented using existing features in most languages, including C++, C#, and Java, using pointers or reflection for indirection and inheritance to provide adapters for the input permutations. This enables continuously evaluated (a.k.a. data-bound, reactive [3,4]) expressions like those from C#/WPF without new language features. Furthermore, any invertable function, such as $1-x$, $kx$, or $\sqrt{x}$ can be used to wrap the pointers.

Stateless GUI

Most GUI systems use change events, dirty rectangles, and explicit data models [1] to synchronize state. These increase runtime efficiency at the cost of duplicate state and boilerplate code, which made sense 30 years ago. Today, real-time interactive programs like games and CAD run on fast machines. Conserving GUI computation at the expense of code complexity is now a poor tradeoff.

We created a new GPU-efficient, themed extension to Barrett’s Immediate Mode GUIs [2]. Ours has a stateless data model where universal pointers directly connect application variables to controls. For example, there is no “checkbox.getValue()”; instead, a checkbox is bound to a boolean variable at creation. This eliminates most events. To further eliminate boilerplate, we automatically manage GUI memory and layout. In terms of MVC[1], we have a transparent model, a smart controller, and a very fast view. These simple changes make an API remarkably easier than MFC, Swing, and WxWidgets while naturally integrating with 3D rendering.

Universal Pointer

The universal pointer is a new design pattern that abstracts reading and writing from:

- Variable or object member
- Pointer, reference, or smart pointer
- Get/set method pair
- Get/set function pair

It can be implemented using existing features in most languages, including C++, C#, and Java, using pointers or reflection for indirection and inheritance to provide adapters for the input permutations. This enables continuously evaluated (a.k.a. data-bound, reactive [3,4]) expressions like those from C#/WPF without new language features. Furthermore, any invertable function, such as $1-x$, $kx$, or $\sqrt{x}$ can be used to wrap the pointers.

Efficient Rendering

Modern graphics applications are not limited by overdraw but by texture map changes and shader performance. To minimize those, we use alpha blending for anti-aliasing, glows, and drop shadows, and pack everything into two textures: one for all controls and one for all font characters.