Issue #3:

Improving the Tk Binding Mechanism

John Ousterhout

Computer Science Division
Department of EECS

University of California at Berkeley

Overview

Problem:

- Hard to combine independently-developed bindings for widgets.

Solution:

- Allow more than one binding to trigger for widget.
- Add tag-like mechanism for specifying related groups of bindings.
- Provide control over order of execution of bindings.

Current Bindings

Three kinds of bindings:

- Widget-specific bindings: **bind .a.b.c ...**
- Class bindings: **bind Button ...**
- "All" bindings: **bind all ...**

A single binding triggers for each event:

- Widget-specific has priority over class.
- Class has priority over "all".
- Within type, "most specific" binding wins:

  <Escape> beats **<KeyPress>**

  <Control-B1-Motion> beats **<Motion>**

Class bindings used for default widget behavior.

Widget-specfic bindings used for overriding, extending default behavior.

Problems with Current Approach

Bindings don’t compose gracefully.

Example: add additional widget-specific action to <Enter> for buttons.

- **bind .b <Enter> foo**
- Replaces class binding for <Enter>, breaks normal button behavior.
- Current solution:

```
bind .b <Enter> \
    "[bind Button <Enter>]; foo"
```

- But what if class binding changes to <Any-Enter>?

"All" bindings don’t really work very well.

Proposed Solution

(based on suggestion by Michael Halle)

Let multiple bindings trigger on a single event:

- First find and trigger one widget-specific binding.
- Then trigger one class binding.
- Then trigger one "all" binding.

Add support for `break` command to skip remaining bindings.

Only one binding can trigger within a group.

Disadvantage: not backwards compatible.

Proposed Solution, cont'd

Provide new command to change binding tags for a widget:

- Change widget's class for binding purposes:

```
bind MyButton ...
bind tags .b .b MyButton all
```

- Add a new independent set of bindings:

```
bind special ...
bind tags .b special .b Button all
```

- Replace all bindings for widget:

```
bind tags .b special
```

- Make widget-specific bindings trigger after class bindings:

```
bind tags .b Button .b all
```

Conclusion

- Does this solve all known problems with bindings?
- Anyother suggestions for bindings?