

TkDND: a cross-platform drag'n'drop package

Georgios Petasis

Software and Knowledge Engineering Laboratory,
Institute of Informatics and Telecommunications,
National Centre for Scientific Research "Demokritos",
Athens, Greece
petasis@iit.demokritos.gr



Institute of Informatics & Telecommunications – NCSR "Demokritos"



Overview

- Drag and Drop (DnD)
- DnD and Tk
 - Intra & inter application DnD
- TkDND
 - History and current status
 - Usage
 - Supported platforms
- Conclusions and future work



Drag and Drop (1)

- The action of clicking on a virtual object, and:
 - either dragging it to a different location,
 - or onto a different virtual object
- Nowadays, an important element of modern UI development
 - Thus, several extensions exist
- Drag and Drop in Tk: [<http://wiki.tcl.tk/571>]
 - Lists more than 5 extensions



Drag and Drop (2)

- Two main categories:
 - Approaches that target DnD within the same application (intra-application)
 - Approaches that target DnD among different applications (inter-application)
- Intra-application:
 - Not difficult to implement
 - A communication mechanism among virtual objects needs to be devised
 - ✓ Feasible even in Tcl
 - Several intra-application approaches are available
 - ✓ “megawidgets” (i.e. Bwidget)



Inter-application DnD

- A more challenging task
 - Requires an inter-application communication scheme
- Initial approaches targeted other Tk applications
 - i.e. BLT, with a communication scheme based on Tk's "sent"
- More recent approaches tried to exploit standardised DnD communication protocols
 - i.e. Olednd



Olednd

- Olednd
 - Developed by Gordon Chaffee
- Amongst the first DnD approaches for Tk that can be characterised as complete:
 - Inter-application DnD under Microsoft Windows
 - ✓ By exploiting OLE DnD, the platform's native DnD protocol
- TkDND was inspired by Olednd
 - Enhance functionality under Windows (i.e. files)
 - Support more operating systems
 - ✓ Gnu/Linux
 - ✓ Mac OS X.



TkDND 1.x series (1)

- Development started around 2000
 - With a new implementation for Windows
 - ✓ Several data types: ASCII-UNICODE text transfers, file names, etc.
- Supporting Linux was attempted
 - Two protocols at that time:
 - ✓ Motif DND (Motif/Lesstiff, GTK/GNOME)
 - ✓ XDND (Qt/KDE)
- TkDND tried to compromise the two protocols:
 - Support for both dragging/dropping actions for (XDND)
 - Support for dropping actions (Motif)



TkDND 1.x series (2)

- The result was not satisfactory
 - Stability issues
 - ✓ Inadequate knowledge of Xlib programming
 - Frequent revisions of the XDND protocol
 - ✓ XDND support was a moving target
 - Incompatible implementations of the protocols
 - ✓ Motif applications worked ok
 - ✓ GTK/GNOME applications exhibit random behaviour
 - ✓ XDND support was changing with each Qt release
- TkDND was too ambitious
 - Too many types to be supported
 - ✓ i.e. plain text, Unicode text, files, links, images



TkDND 2.x series

- TkDND 1.x was abandoned in 2006
 - In favor for TkDND 2.x
- TkDND 2.x
 - Yet another DnD API
 - ✓ TkDND 1.x API still supported though
 - A new implementation for all supported operating systems
 - Implemented in Tcl (as much as possible)
 - ✓ A support library in Tcl
 - Support for cross-platform types
 - ✓ DND_Text
 - ✓ DND_Files



Using TkDND

- Two categories of operations:
 - Accepting a drop operation (“Drop Targets”)
 - Initiating a drag operation (“Drag Sources”)



Drop targets

- Drop targets

```
tkdnd::drop_target register window ?type-list?
```

- type-list: DND_Text, DND_Files, etc.

- DnD events delivered as Tk virtual events

- <<DropEnter>>: returns action

- <<DropPosition>>: returns action

- <<DropLeave>>

- <<Drop>>: returns action

- <<Drop:**type**>>: returns action

- ✓ Actions: copy, move, link, ask, private, and refuse_drop



Drag sources

- Drop targets

```
tkdnd::drag_source register window ?type-list? ?mouse-button?
```

- type-list: DND_Text, DND_Files, etc.
- mouse-button: 1 (default), 2, ...

- DnD events delivered as Tk virtual events

- <<DragInitCmd>>: returns action-list, type-list, data
 - ✓ Actions: copy, move, link, ask, private

```
bind .drag_source <<DragInitCmd>> \  
    {list copy DND_Text {Hellow world!}}
```

- <<DragEndCmd>>



Supported platforms: Windows

- Fairly complete support
 - DND_Text (CF_UNICODETEXT - CF_TEXT)
 - ✓ Both ANSI & Unicode supported
 - DND_Files (CF_HDROP)
 - Mouse modifiers supported
- Uses OLE DnD
 - The native DnD protocol of the platform
 - Implements the needed data objects
 - ✓ IDropTarget and IDropSource
 - Native events converted to TkDND events
 - ✓ DragEnter, DragOver, DragLeave, Drop, QueryContinueDrag, GiveFeedback



Supported platforms: Mac OS X

- Fairly complete support
 - DND_Text (NSStringPboardType)
 - ✓ Both ANSI & Unicode supported
 - DND_Files (NSFileNamesPboardType)
 - Mouse modifiers not supported
- Uses Cocoa DnD protocol
 - The native DnD protocol of the platform
 - Great resemblance to XDND
 - ✓ A convenient API is available
 - Native events converted to TkDND events
 - ✓ draggingEntered, draggingUpdated, draggingExited, prepareForDragOperation, performDragOperation



Unsupported platforms: Linux

- Currently, TkDND does not work under Linux
- Linux lacks a generic implementation of the XDND protocol
 - Instead, each toolkit (i.e. Qt, GTK+) contain their private implementation
 - Compatibility issues among the toolkits
- Support for accepting drops exists
 - And seems working with Qt/KDE applications, with the exception of retrieving the dragged data
 - XDND requires to retrieve data from the selection with a specific timestamp
 - ✓ Relates to TIP 370



Conclusions – Future work

- TkDND offers inter-application DnD for several platforms
 - Windows, Mac OS X, and hopefully Linux
 - The dominant DnD protocol is used for each platform
 - Cross-platform DnD types are provided
- Future work will concentrate on:
 - Improving Linux support for dropping actions
 - Implementing Linux support for dragging actions
 - Fixing bugs



Aknowledgments

Several contributors to TkDND:

- Gordon Chaffee
- Laurent Riesterer
- Kevin Walzer
- Daniel A. Steffen



Thank you!
