

Графические Системы. Часть II

Практическое Занятие № 1

**Программирование графического
пользовательского интерфейса
средствами X-WINDOW.
ИПВУ. ТсI/ТК**

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ

Основное назначение тех средств разработки пользовательских графических интерфейсов, которые разрабатываются и поставляются отдельно от оконной системы, является облегчение создания нового графического интерфейса за счет использования существующих параметризованных заготовок - **widget'ов**. Как видно, в принципе это те же самые идеи, на которых основана объектно-ориентированная библиотека X Window System **Xt Intrinsics**.

Наиболее распространенный пакет, предназначенный для быстрой и качественной разработки графических пользовательских интерфейсов, это **OSF Motif**. Спроектирован и разработан в **OSF**- консорциуме и в основном является развитием идей **Xt Intrinsics**. **Motif** является сугубо коммерческим продуктом. Дело дошло до того, что компания OSF запатентовала внешний интерфейс продуктов, входящих в состав **Motif**, чтобы не дать кому-нибудь возможность воспроизвести этот интерфейс.

Это привело к настоящему скандалу в сообществе американских программистов, потому что создало опасный прецедент патентования интерфейсов. Если можно закрыть своим авторским правом возможность повторения графического интерфейса, то почему нельзя запатентовать синтаксис языка программирования, интерфейс операционной системы и т.д.? Однако строгость американских законов не оправдывается необязательностью их исполнения, и поэтому недовольные свободолюбивые американские программисты ропчат, но терпят, а тем временем пытаются сагитировать восточно-европейских программистов (не так сильно зависящих от американских законов) на нелегальную свободно доступную реализацию интерфейсов **Motif**.

С другой стороны, сравнительно недавно в Калифорнийском университете г. Беркли был создан альтернативный механизм под названием **Tcl/Tk**. Этот механизм основан на наличии специализированного **командного языка**, предназначенного для описания графических пользовательских интерфейсов, **соответствующего интерпретатора и библиотеки ранее разработанных заготовок интерфейсов - widget'ов**. Пакет **Tcl/Tk** распространяется (вместе с полной документацией) свободно, и многие профессиональные программисты находят его более удобным, чем **Motif**.

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ OSF/Motif

Motif (официальное название этого продукта - **OSF/Motif**) представляет собой программный пакет, включающий оконный менеджер, набор вспомогательных утилит, а также библиотеку классов, построенных на основе **Xt Intrinsics**. Для конечных пользователей оконных систем, опирающихся на **Motif**, основной интерес представляет менеджер окон, хотя, скорее всего, вы не сможете определить, применяется ли оконный менеджер **Motif** в используемой вами установке. Для разработчиков же графических интерфейсов важны все три компонента **Motif**. Новый интерфейс разрабатывается в графическом же режиме с использованием оконного менеджера. При этом полезно использование утилит **Motif** и необходимо использование библиотеки классов **Motif**.

Библиотека классов **Motif** является расширением библиотеки **Xt Intrinsics** с целью предания этой библиотеке практического смысла (по-другому можно сказать, что **Motif** - это то, чем должен был бы быть **Xt**, если бы при его создании ставились коммерческие цели). Все графические объекты (правильнее сказать, классы) **Xt Intrinsics** включаются в библиотеку классов **Motif**, хотя в ней используются другие имена.

Но **Motif** существенно расширяет возможности **Xt Intrinsics**. В его библиотеке поддерживается большое число классов, позволяющих создавать меню, "нажимаемые" кнопки и т.д. Основное назначение этих классов - определение новых **widget'ов**, связанных с окнами.

Однако в **Motif** поддерживается и новый вид графических объектов (их классов) - так называемые **гаджеты (gadgets)**. **Гаджет** отличается от **widget'a** тем, что соответствующий класс также может использоваться для создания элементов интерфейса, но графический объект не привязывается к определенному окну. При отображении на экран **гаджета** используется окно объекта, относящегося к суперклассу класса **гаджета**.

Понятно, что здесь мы не можем привести подробное описание **Motif** (еще раз повторим, что соответствующий материал содержится в нескольких солидных книгах). Однако основная идея должна быть понятна: развитая библиотека классов языка **Си++**, возможности применения этих классов при использовании обычного стиля программирования и поддержка визуального программирования с немедленным отображением получающихся графических объектов.

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ Tcl/Tk

Продукт **Tcl/Tk** в действительности представляет собой два связанных программных пакета, которые совместно обеспечивают возможность разработки и использования приложений с развитым графическим пользовательским интерфейсом. Название **Tcl** относится к "командному языку инструментальных средств - **tool command language**", и его рекомендуется произносить "**тикл**". Это простой **командный язык** для управления приложениями и расширения их возможностей. Язык **Tcl** является "встраиваемым": его интерпретатор реализован в виде библиотеки функций языка **Си**, так что интерпретатор может быть легко пристыкован к любой прикладной программе, написанной на языке **Си**.

Tk (рекомендуемое произношение – «**ти-кей**») является библиотекой **Си-функций**, ориентированной на облегчение создания пользовательских, **кросс-платформенных** графических интерфейсов в среде **X Window System** (т.е., по сути дела, некоторый аналог **Xt Intrinsics**). С другой стороны, аналогично тому, как это делается в командных языках семейства **shell**, функции библиотеки **Tk** являются командами языка **Tcl**, так что любой программист может расширить командный репертуар языка **Tcl** путем написания новой функции на языке **Си**.

Совместно, Tcl и Tk обеспечивают четыре преимущества для разработчиков приложений и пользователей :

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ Tcl/Tk

Во-первых, наличие командного языка **Tcl** дает возможность в каждом приложении использовать мощный командный язык. Все, что требуется от разработчика приложения, чтобы удовлетворить его/ее специфические потребности, - это создать несколько новых команд **Tcl**, требующихся приложению. После этого нужно связать прикладную программу с интерпретатором **Tcl** и пользоваться полными возможностями командного языка.

Вторым преимуществом использования **Tcl/Tk** является возможность быстрой разработки графических интерфейсов. Многие интересные оконные приложения могут быть написаны в виде скриптов языка **Tcl** без привлечения языков **Cи** или **Cи++** (а **Tcl** позволяет скрыть многие несущественные детали). Как утверждают разработчики **Tcl/Tk**, пользователи оказываются способными к созданию новых графических интерфейсов уже после нескольких часов знакомства с продуктом. Другой особенностью языка **Tcl**, способствующей быстрой разработке оконных приложений, является то, что язык является интерпретируемым. Можно опробовать новую идею интерфейса, выражающуюся в сотнях или тысячах строк кода на языке **Tcl**, без потребности вызова новых программных средств, путем простого нажатия на клавишу мыши (не наблюдая существенных задержек при использовании современных рабочих станций).

Третьим преимуществом языка **Tcl** является то, что его можно применять в качестве языка "склейки" приложений. Например, любое основанное на **Tcl** и использующее **Tk** оконное приложение может направить свой скрипт любому другому аналогично ориентированному приложению. С использованием **Tcl/Tk** можно создавать приложения, работающие в стиле мультимедиа, и опять же они смогут обмениваться скриптами, поскольку пользуются общим интерпретатором командного языка **Tcl** и общей внешней библиотекой **Tk**.

Наконец, четвертым удобством интегрированного пакета **Tcl/Tk** является удобство пользователей. Для написания нового приложения в среде **Tcl/Tk** достаточно выучить несколько совершенно необходимых команд, и этого окажется достаточно. Другими словами оказывается возможным инкрементальный (пошаговый) стиль погружения в предмет. Такая возможность всегда радует сердце и греет душу.

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ Tcl/Tk. Основы

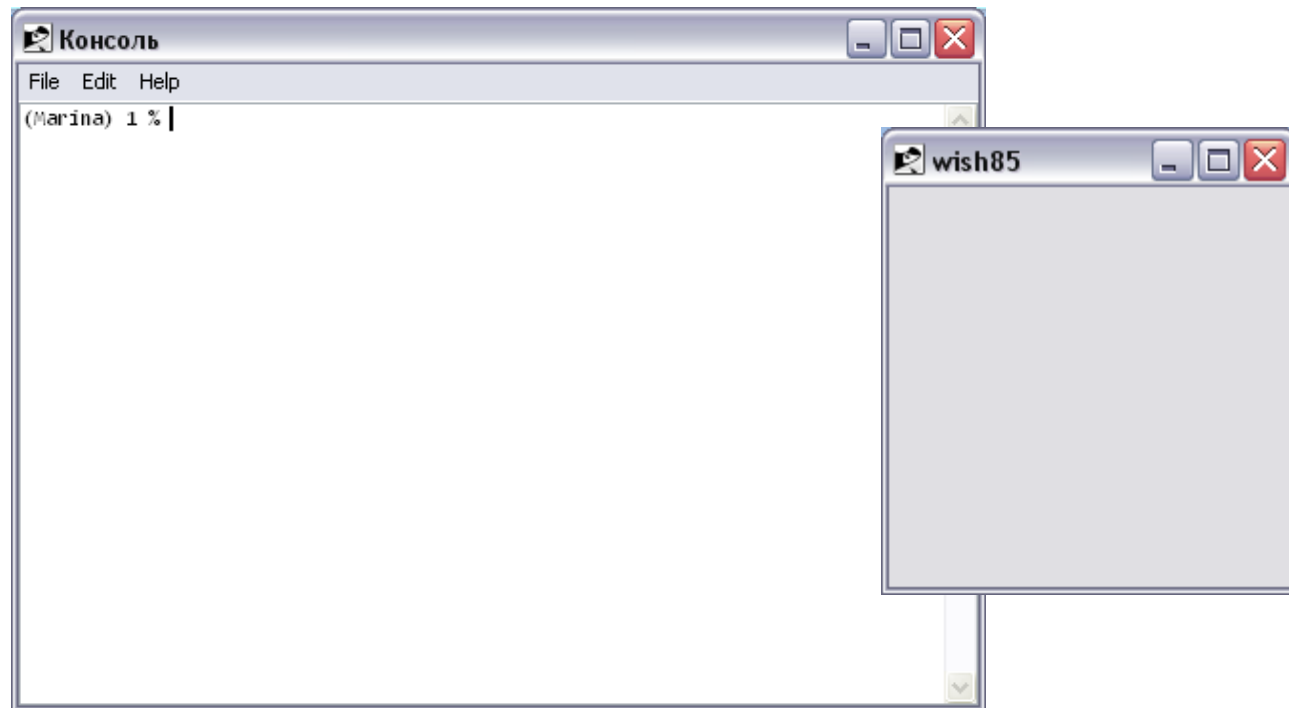
Разработанный **John Ousterhout**, язык **Tcl** гораздо больше похож на командный, скриптовый язык, чем на язык программирования высокого уровня. Иначе говоря, программировать с помощью **Tcl/Tk** также легко, как в **Bash shell'e**, **C shell'e** или **Perl'e**, чем в **C** или **C++**.

К примеру, классическая программка «Hello World» в **Tcl/Tk** будет выглядеть просто элементарно:

```
# This is a comment.  
button .b -text "Hello World" -command exit  
pack .b
```

ПРОБУЕМ!

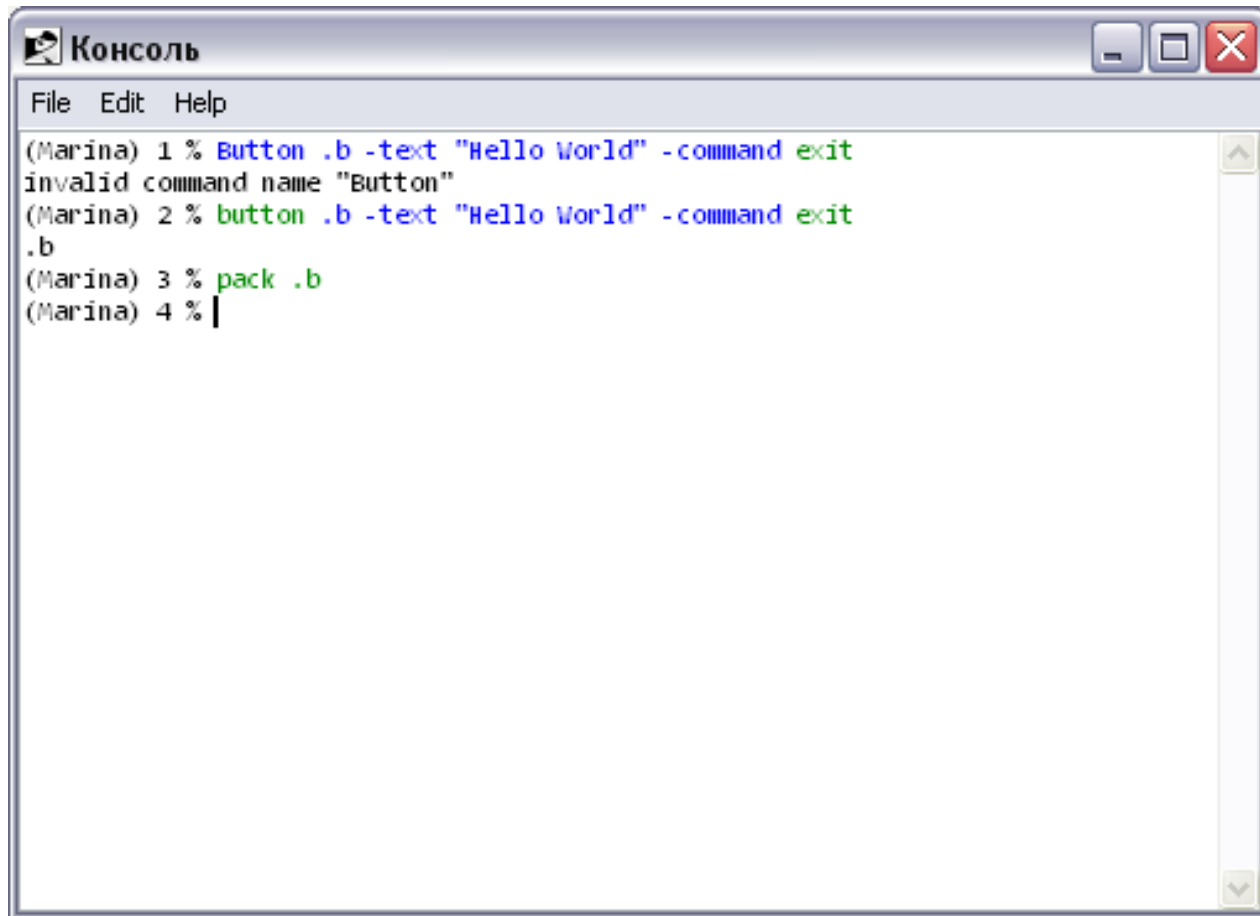
**Вызываем
Интерпретатор
wish -**



Средства разработки графических интерфейсов в системе X Window System.

ИПВУ ТсІ/ТК. Основы

После ввода командных строк получаем:



```
Консоль
File Edit Help
(Marina) 1 % Button .b -text "Hello World" -command exit
invalid command name "Button"
(Marina) 2 % button .b -text "Hello World" -command exit
.b
(Marina) 3 % pack .b
(Marina) 4 % |
```



А теперь – в реале:

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ Tcl/Tk. Быстрое введение

Tcl имеет очень простую структуру. В каждой строке стартуется команда, как например в нашем примере – команда **button**. Каждая команда выполняется как C-функция, отвечающая за обработку всех аргументов, следующих за ней.

Синтаксис:

`command arg1 arg2`

Выполняет команду с заданными аргументами;

`"Text in quotes"`

Передает текст между кавычками в как один аргумент выполняемой команды;

`{Text in braces}`

Передает текст между кавычками в как один аргумент выполняемой команды, откладывая подстановку на потом; допускает вложения;

`$variable`

Подставляет значение заданной переменной;

`command arg1 \
arg2`

Продолжение командной строки;

Более подробно основы синтаксиса команд **Tcl/Tk** рассмотрены здесь:

http://a0601.narod.ru/Zelenizina_2008/command_syntax.htm

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ Tcl/TK. Запуск программ

Интерпретатор wish. Это программа с двумя рабочими окнами, главным и выходным. Интерпретатор может работать в интерактивном или пакетном режиме.

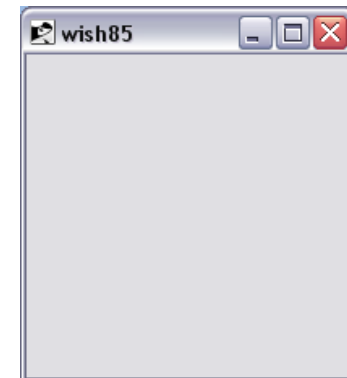
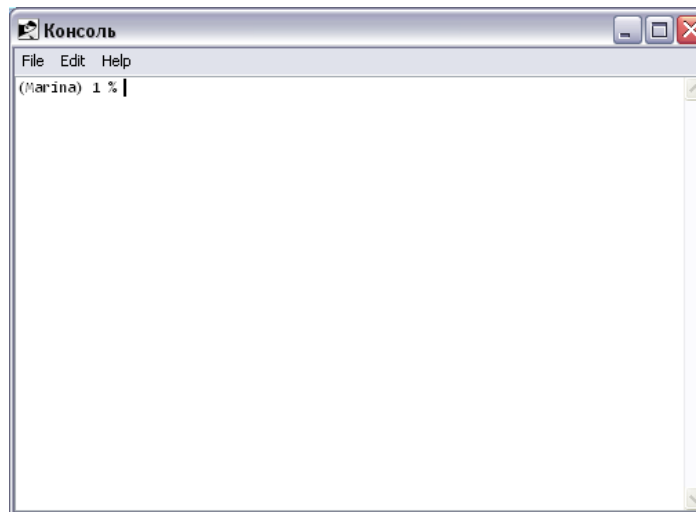
В первом случае он считывает команды **Tcl** со стандартного входа: пользователь вводит команды с клавиатуры в главном окне, **wish** обрабатывает их и выводит результат или сообщение об ошибке в выходном окне.

Во втором случае источником команд для обработки служит указанный дисковый файл. В обоих режимах интерпретатор будет работать до тех пор, пока не будут удалены все окна приложения, или пока на стандартный вход не поступит символ конца файла.

Для запуска интерпретатора необходимо в ответ на приглашение операционной системы подать команду

wish ?fileName arg arg ...?

Если никаких аргументов указано не было, или если первый аргумент начинается с символа «-», то интерпретатор запустится в интерактивном режиме, изображая на дисплее приглашение для ввода команд в виде знака процента «%». В ответ на приглашение следует ввести команду **Tcl** и нажать клавишу **Enter**. Затем ввести следующую команду и снова нажать **Enter**.



Более подробно Интерпретатор wish рассмотрен здесь:

http://a0601.narod.ru/Zelenizina_2008/wish.htm

Средства разработки графических интерфейсов в системе X Window System.

ИПВУ Tcl/Tk. Widget'ы – встроенные команды TK

Итак, библиотека **Tk** в системе программирования **Tcl/Tk**, представляет из себя набор параметризованных заготовок графического интерфейса - **widget'ов**, или встроенных команд.

Widget'ы

Описание

toplevel	Контейнер для других Widget'ов , управляется непосредственно WM
frame	Контейнер для других Widget'ов
label	Текстовая метка
entry	Окно для ввода текстовых данных
message	Отображает текстовое сообщение, возможно состоящее из нескольких строк
text	Сложный Widget для ввода и редактирования текста
checkboxbutton	Отображает сообщение, ассоциированное с переменной, имеющей два значения
radiobutton	Отображает сообщение, позволяет выбрать опцию из предопределенного набора
scale	«шкала», позволяет легко изменять действительные значения
listbox	Отображает список элементов для выбора
scrollbar	Вертикальная и горизонтальная прокрутка, может быть связана с одним или более Widget'ами
canvas	Отображает фигуры и графики

Более подробно встроенные команды **Tk** рассмотрены здесь:

<http://velociraptor.mni.fh-giessen.de/TclTk/index.html>