TRAITIFYING
MOTIF

Daniel Dardailler *(dar-da-yé)*
Open Software Foundation.
CAMBRIDGE, USA.

AGENDA

- What, Why, Where, How
- Motif API and List
- Discussion and examples

What is a trait

A *characteristic of an object that can be expressed by a set of operations/methods which can be applied to this object.*

- **Same as a "character trait" for a person:**
  - **- being** *dynamic*
  - **- being** *happy*

- **A widget trait:**
  - **- being a** *button*
  - **- being a** *scroller*

Our goal

- Remove the incestuous dependencies of Motif

  - permit extensibility
  - improve link profile
  - simplify our code

Our job

- Decide on a trait architecture
  - API
  - Implementation

- Provide a set of traits
  - find the candidates
  - choose the trait names
  - define the trait methods
  - implement callee and caller

traits are good

- A trait of traits: they are *small*

- Objects usually carry/support multiple traits.

- Traits are light:

  - to implement

  - to refer to.

- Multiple-inheritance

History

- Xerox PARC 1980: Star project

  - Add multiple-inheritance to a SIMULA-67 base.

  - everything is a trait

  - traits can be composed, synthesized, restricted, etc.

  - Similarity to LISP flavors.

- **OSF 1992/93: Motif 2.0 project**

- Part of the extensibility project:

Widget writer book: **Xme API, XmPrimitive/XmManager class methods, Traits (+ Exm widget set).**

- Keep the single-inheritance Xt model, add traits.

Holding textual data

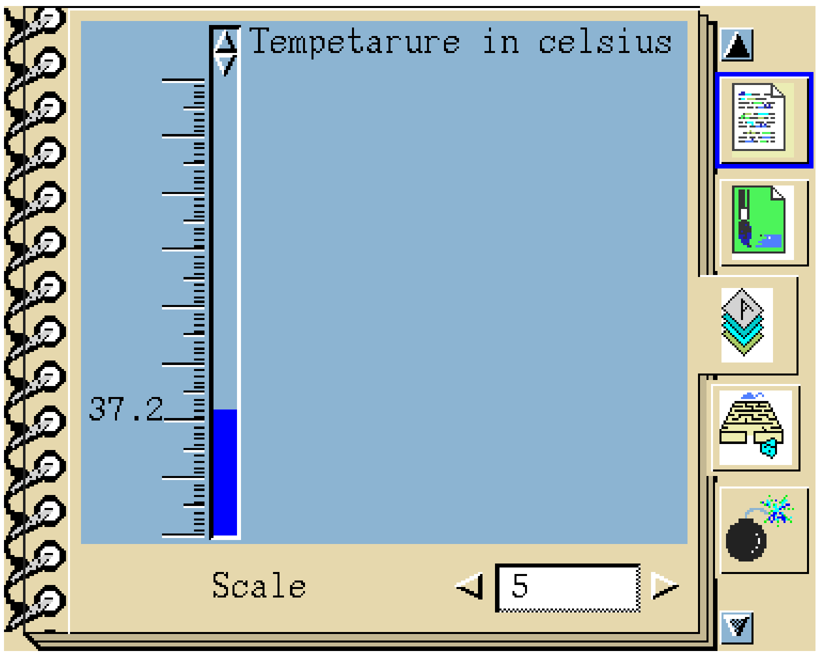
- **Define the methods for this trait**
  - **- Get/Set the text in a given format**
  - **- Ask for preferred format**

- *Implement* **the trait on:**
  - **- Label: `XmString` is preferred.**
  - **- Text: `char*` is preferred.**

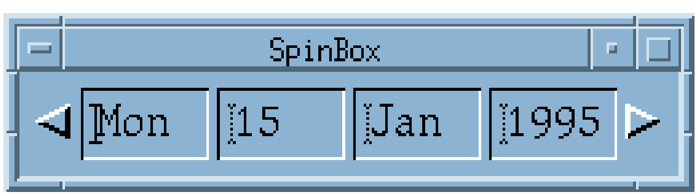
Example (2)

- *Use* **the trait with:**

- Notebook: default StatusArea



- SpinBox: update child value



Motif API for traits

- Two functions.

```
Boolean XmeTraitSet(XtPointer,
                    XrmQuark,
                    XtPointer);
```

```
XtPointer XmeTraitGet(XtPointer,
                      XrmQuark);
```

Motif Trait API semantics

```
Boolean XmeTraitSet(WidgetClass
                    XrmQuark,
                    XmTraitRec*);

XmTraitRec *XmeTraitGet(WidgetClass,
                        XrmQuark);

#include <Xm/TraitP.h>

#include <Xm/TraitNameT.h>
```

Example using Xme

In Label.c

```
#include <Xm/AccTextT.h>

...

XmAccessTextualTraitRec accTextT = {
    0, /* version */
    AccTextGetValue,
    AccTextSetValue,
    AccTextPreferredFormat };

...

void ClassPartInitialize(WidgetClass w_class){
    XmeTraitSet (w_class,
                 XmQTaccessTextual,
                 &accTextT);
}

...
```

Get the trait signature

Fill a trait record

Install the trait on the class

Example using Xme (2)

In Label.c

```
static XtPointer AccTextGetValue (Widget w,
                                  int format){
```

Make the appropriate conversion

```
  if (format == XmFORMAT_XmSTRING)
      return XmStringCopy(LAB_Value(w));

  if (format == XmFORMAT_MBYTE)
      return _XmStringGetText(LAB_Value(w));

  if (format == XmFORMAT_WCS)
      return ToWideChar(LAB_Value(w));

  return NULL;
}
```

Example using Xme (3)

In SpinB.c

```
static void UpdateChildText (Widget child) {

  XmAccessTextualTrait textT =
          XmeTraitGet(XtClass(child),
                          XmQTaccessTextual);

  if (SB_ChildIsNumeric(child)){
     ...
     sprintf(buffer,"%d",SB_NumericValue(child));
     textT->setValue (child, buffer,
                          XmFORMAT_MBYTE);
  } else {

     ...

     textT->setValue (child,
                          SB_CurStringValue(child),
  }                       XmFORMAT_XmSTRING);
}
```

Get the trait record

Set the new text in the desired format

List of 2.0 Traits

- Navigator
- ScrollFrame

- Container
- ContainerItem

- MenuSystem
- MenuSavvy

- Activatable
- TakesDefault

List of 2.0 Traits (2)

- Transfer
- AccessTextual
- DialogShellSavvy
- CareAboutParentVisual
- SpecifyRenderTable
- JoinSide

Trait and Xt

- Traits *vs*. Xt resources

  - support existing widgets

  - no Xt resources available

  - performance

- Use of Xt ClassExtension

  - no API available

  - widget class trait only

  - performance

Conclusion

- **Traits have brought us:**

  - extensibility

  - flexibility

  - internal clean-up

  - better link profile

- **The future of Traits**

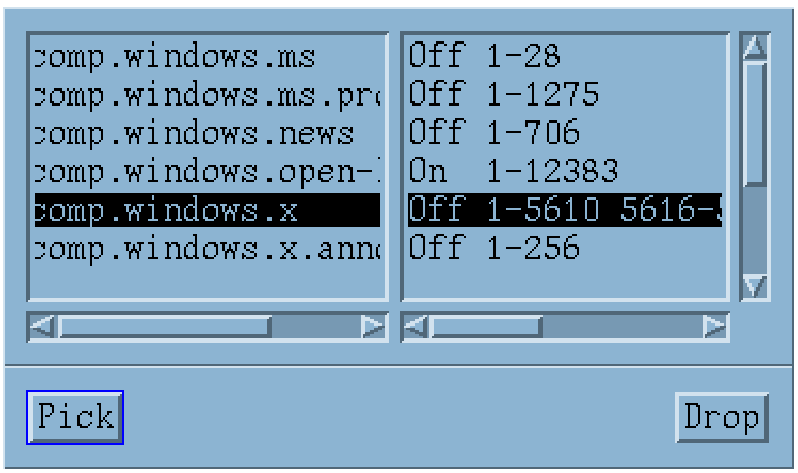
  - extension of existing traits

  - new traits

  - ***Requests For Traits!***

Navigator examples

- **Synchronized Lists**



Navigator examples

- **Multiple scrollers**