

Programmer's Guide to the MUT Facility

A Facility Which Provides Motif Utility Functions

Stephen M. Moore

Mallinckrodt Institute of Radiology
Electronic Radiology Laboratory
510 South Kingshighway Boulevard
St. Louis, Missouri 63110
314/362-6965 (Voice)
314/362-6971 (FAX)

Version 2.10.0

August 3, 1998

Copyright (c) 1995, 1998 RSNA, Washington University

1 Introduction

The Motif toolkit provides a number of widgets for developing user interfaces. This facility provides some convenience functions for loading values into widgets and extracting values from widgets.

2 Data Structures

3 Include Files

To use MUT functions, applications need to include these files in the order given below:

```
#include "dicom.h"  
#include "condition.h"  
#include "mut.h"
```

4 Return Values

The following returns are possible from the MUT facility:

MUT_NORMAL	Normal return from MUT facility.
MUT_SCANFAILURE	Function failed to scan an ASCII field. For example, caller was scanning for an integer and the value to be scanned was "abc".

5 MUT Routines

This section provides detailed documentation for each MUT facility routine.

MUT_LoadList

Name

MUT_LoadList - load a Motif scrolled list with a set of items.

Synopsis

```
void MUT_LoadList(Widget w, LST_HEAD *lst, void (*format)(), char *buf)
```

<i>w</i>	The MOTIF scrolled list widget which will be loaded with text items.
<i>lst</i>	A list of user items to be placed in the scrolled list.
<i>format</i>	A callback function which is invoked for each item in the caller's <i>lst</i> . This function is expected to fill in a text buffer with the ASCII data to be placed in the widget.
<i>buf</i>	A scratch buffer which will be passed to the <i>format</i> function. This buffer should be large enough to hold the ASCII data created by <i>format</i> .

Description

MUT_LoadList is used to provide a uniform mechanism for loading text items into a Motif scrolled list widget. For each item in the caller's *lst*, *MUT_LoadList* invokes the caller's *format* function with the arguments:

<i>node</i>	The node in the list.
<i>i</i>	The index of the node in the list (0 - N-1).
<i>buffer</i>	The ASCII buffer to be filled by <i>format</i> .

format is expected to examine the data in *node* and create a line of text to be placed in the scrolled list widget. This line of text should be written into *buffer*.

For each line of text it receives from *format*, *MUT_LoadList* creates the appropriate *XmString* variable and loads it into the list widget.

Notes

The user's *callback* function should know what type of structure to expect. The current implementation uses *XmSTRING_DEFAULT_CHARSET* as the character set when creating the *XmString* variable.

It is assumed the caller has knowledge of the structure to be formatted and how the output of the *format* function, allowing the caller to allocate sufficient space in *buf*.

Return Values

None

MUT_ScanWidget

Name

MUT_ScanWidget - scan a Motif text widget for data in one of several formats (text, int, float)

Synopsis

CONDITION MUT_ScanWidget(Widget w, MUT_DATATYPE type, CTNBOOLEAN *nullFlag, void *d)

<i>w</i>	The widget containing text to be scanned.
<i>type</i>	An enumerated type that indicates the format of the data expected by the user.
<i>nullFlag</i>	Will be set by this function to indicate if the data in widget w is null.
<i>d</i>	Address of caller's data.

Description

MUT_ScanWidget reads the text from a text widget, w, and converts the text into a format that is easily used by the caller. The caller indicates the type of data expected to be in the text widget:

MUT_TEXT	Text
MUT_INT	Integer
MUT_FLOAT	Float
MUT_US	Unsigned short

MUT_ScanWidget expects the caller to have allocated enough space to hold the return data. This could be a problem for text data.

Return Values

MUT_NORMAL
MUT_SCANFAILURE