

**WORK PRACTICE INSTRUCTION
FOR
SOFTWARE INTERFACE DESIGN STANDARDS
WPI0035**

Prepared By:

**WSR-88D Radar Operation Center
1200 Westheimer Drive
Norman, OK 73069**

SUBMITTED BY: _____ **DATE:** _____

**Peter J. Grant
Team Lead, Documentation
WSR-88D Radar Operation Center**

**APPROVED FOR
USE AS PRODUCT
BASELINE BY:** _____

DATE: _____

**Rex Reed
Engineering Branch Chief
WSR-88D Radar Operation Center**

DATE: _____

**Edward L. Berkowitz
Program Branch Chief
WSR-88D Radar Operation Center**

WORK PRACTICE INSTRUCTIONS
WPI 0035

Software Interface Design Standards

Purpose and Scope

This Work Practice Instruction (WPI) establishes certain guidelines for software developed by the ROC, including window layout, text, and terminology. Objectives include:

- Ease the work load of software engineers and the RST by reducing the number of consistency-related issues to be addressed.
- Provide software engineers with a standard reference, although the WPI will remain a living document.
- Reduce the need to replace documents for cosmetic changes.
- Alleviate user confusion by maintaining consistency and eliminating ambiguity in software and documentation.
- Encourage the design and implementation of a standard format.

TABLE OF CONTENTS

Paragraph		Page
SECTION 1.	USABILITY PRINCIPLES	1
1.1	Consistency	1
1.2	Keep the User Informed	1
1.3	Keep it Simple	1
SECTION 2.	MENU ITEM NAMES	3
2.1	Include a Functional Description in the Menu Name	3
2.2	Put Only Useful Information in the Menu Name	3
2.3	Menu Name Formats	4
SECTION 3.	CONTROLS	5
3.1	Text Entry Fields	5
3.2	Sliders	6
3.3	Buttons	6
3.4	Check Boxes	7
3.5	Radio Buttons	8
3.6	Toggle Buttons	9
3.7	Lists	9
3.8	Drop Down Lists	11
3.9	Scrollbars	11
3.10	Tabbed Notebooks	12
SECTION 4.	VISUAL DESIGN	13
4.1	Window Titles	13
4.2	Window Layout	13
4.3	Text Labels: Spacing and Alignment	16
4.4	Capitalization	17
SECTION 5.	MISCELLANEOUS	19
5.1	Language and Labels	19
5.2	Units	19
5.3	Response Times	28
5.4	Date/Time Display	28
5.5	Background Color Codes	29
5.6	Sensitized/Desensitized Controls	29
5.7	Values and Ranges	29

SECTION 1. USABILITY PRINCIPLES

1.1 Consistency.

Make the application internally consistent and consistent with other applications in both appearance and behavior.

Consistency enables users to apply their existing knowledge of their computing environment to new applications. This not only allows users to become familiar with new applications more quickly, but also helps create a sense of comfort and trust in the overall environment.

In particular, maintain consistency in the application of basic format elements, including:

- Buttons
- Window Titles
- Location Inside Windows
- Units
- Date/Time Display
- Colors
- Fonts
- Acronyms
- Capitalization
- Singular/Plural Usage
- Punctuation
- Tooltips
- Left/Right Justification

1.2 Keep the User Informed.

Always let the user know what is happening in your application by using appropriate feedback at an appropriate time. When the user performs an action, provide feedback to indicate that the system has received the input and is operating on it.

Employ feedback styles (e.g., hourglass, warning popup, progress meter, spinning globe) consistently within an application. Require confirmation for all major actions to safeguard the system against inadvertent mouse clicks.

1.3 Keep it Simple.

SECTION 2. MENU ITEM NAMES

2.1 Include a Functional Description in the Menu Name.

Use words, phrases, and concepts that are familiar to the user rather than terms from the underlying system.

Example 2-1. Include a Functional Description

Original Menu Item	Revised Menu Item
Adaptation	Adaptation Data Editor

2.2 Put Only Useful Information in the Menu Name.

Do not include words like "OPUP", "RPG", "RDA" or other platform details in Application menu names. The user already knows what platform they are using, and if they don't, then application names are not the place to inform them.

Example 2-2. Remove Non-essential Information

Original Menu Item	Revised Menu Item
RPG Image Viewer	Image Viewer
OPUP Map Editor	Map Editor

Do not include technical details when the user does not need to know them, or can infer them from context. Avoid technical jargon. For example, when both a client and a server for something are listed in the menus, remove the word "Client" from the menu name for the client.

Example 2-3. Remove Technical Jargon

Original Menu Item	Revised Menu Item
Hydro Video Client	Movie Player

2.3 Menu Name Formats.

If the application's proper name succinctly describes its functionality, use it.

If there is a succinct functional description of the application, include the descriptive term in the name.

Example 2-4. Using a Functional Description in the Menu Name

Application Name	Menu Name
Clutter	Clutter Map Editor

SECTION 3. CONTROLS

3.1 Text Entry Fields.

Text entry fields are used for entering one or more lines of plain text.

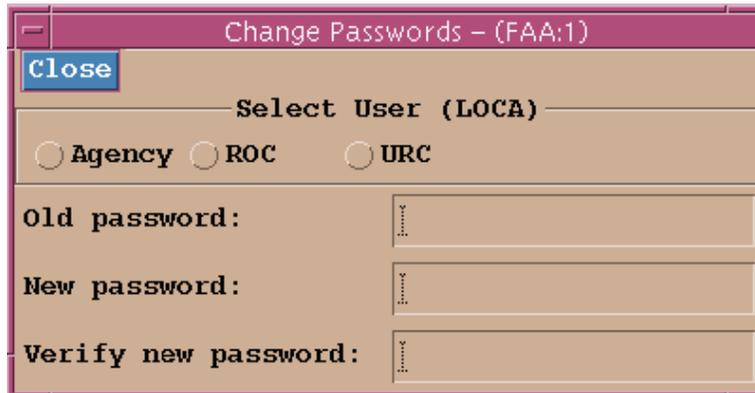


Figure 3-1. Text Entry Fields

Guidelines:

- Label the entry field with a text label above it or to its left, using sentence capitalization.
- Right-justify the contents of entry fields that are used only for numeric entry. This is useful in windows where the user might want to compare two numerical values in the same column of controls. In this case, ensure the right edges of the relevant controls are also aligned.

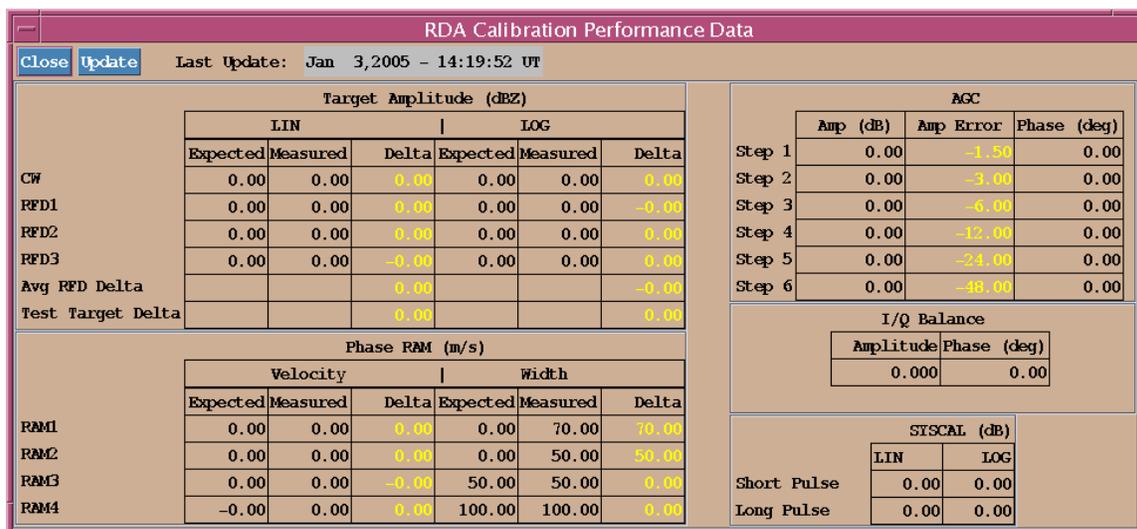


Figure 3-2. Numeric Entry Fields

3.2 Sliders.

A slider allows the user to quickly select a value from a fixed, ordered range, or to increase or decrease the current value.

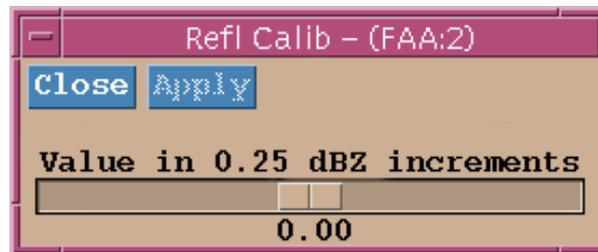


Figure 3-3. A Simple Slider Control

Guidelines:

- Use a slider when:
 - adjusting the value relative to its current value is more important than choosing an absolute value.
 - it is useful for the user to control the rate of change of the value in real time.
- Label the slider with a text label above it or to its left, using sentence capitalization.

3.3 Buttons.

A button initiates an action when the user clicks it.



Figure 3-4. Typical Buttons in a Modal Dialog

Guidelines:

- Label all buttons with imperative verbs, using header capitalization (e.g., Apply, Cancel, Close, Open, Send, Yes, No).
- Do not use more than one or two different widths of button in the same window, and make all of them the same height.
- Make invalid buttons desensitized, rather than popping up an error message when the user clicks them.

- Use standard terms with standard meanings. Examples include the following:

Close: Close the window.

Start: Start the specific process.

Stop: Stop the specific process.

Save: Save the input.

Cancel: Close the window and discard any changes.

3.4 Check Boxes.

Check boxes are used to show or change a setting. Its two states, set and unset, are shown by the presence or absence of a checkmark in the labelled box.

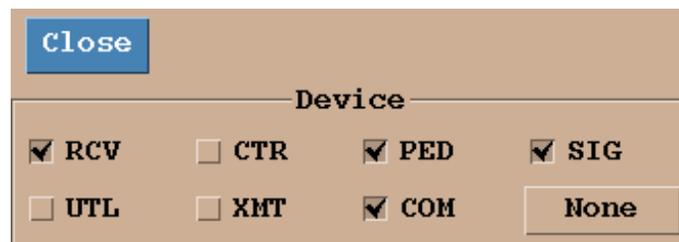


Figure 3-5. Check Boxes

Guidelines:

- Use sentence capitalization for check box labels.
- Label check boxes to clearly indicate the effects of both their checked and unchecked states, for example, Show icons in menus. Where this proves difficult, consider using two radio buttons instead.

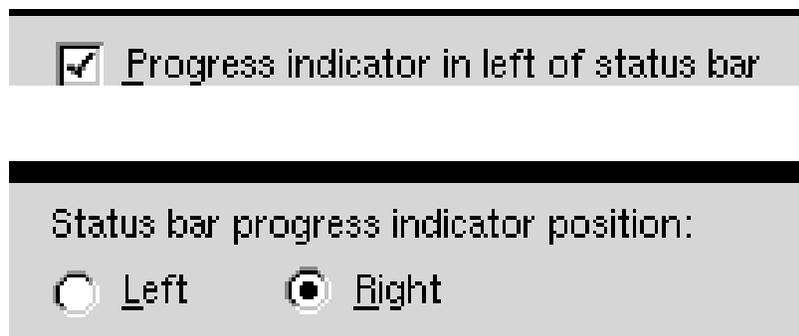


Figure 3-6. Ambiguous Check Box (top), Radio Buttons (bottom)

The single check box in Figure 3-6 is ambiguous, as it is not clear where the "progress indicator" will go if the box is unchecked. Two radio buttons are better in this case, as they make the options clear.

- Try to align groups of check boxes vertically rather than horizontally, as this makes them easier to scan visually.

3.5 Radio Buttons.

Radio buttons are used in groups to select from a mutually exclusive set of options. Only one radio button within a group may be set at any one time.

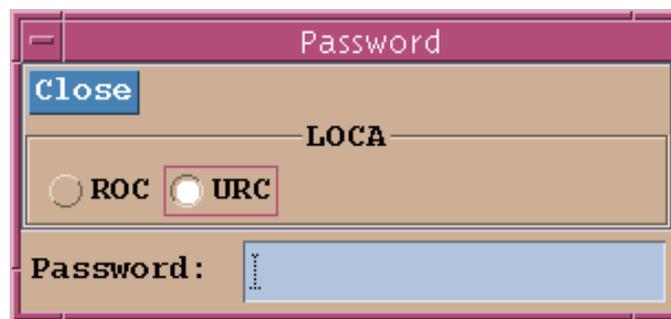


Figure 3-7. Radio Buttons

Guidelines:

- Only use radio buttons in groups of at least two. Never use a single radio button on its own. To represent a single setting, use a check box or two radio buttons, one for each state.
- Clicking a radio button should not affect the values of any other controls. It may sensitize, desensitize, hide or show other controls, however.
- Use sentence capitalization for radio button labels.
- Label a group of radio buttons with a descriptive heading above or to the left of the group.
- Do not place more than about eight radio buttons under the same group heading. If you need more than eight, consider using a single-selection list instead.
- Try to align groups of radio buttons vertically rather than horizontally, as this makes them easier to scan visually.

3.6 Toggle Buttons.

Toggle buttons look similar to regular buttons, but are used to show or change a state rather than initiate an action.

Guidelines:

- Do not use groups of toggle buttons in dialogs unless required by space constraints. Check boxes or radio buttons are usually preferable, as they allow more descriptive labels and are less easily-confused with other types of control.
- Label a group of toggle buttons with a descriptive heading above or to the left of the group, as you would with a group of check boxes or radio buttons.
- Try to align groups of toggle buttons horizontally rather than vertically.

3.7 Lists.

A list control allows the user to inspect, manipulate or select from a list of items. Lists may have one or more columns, and contain text, graphics, simple controls, or a combination of all three.

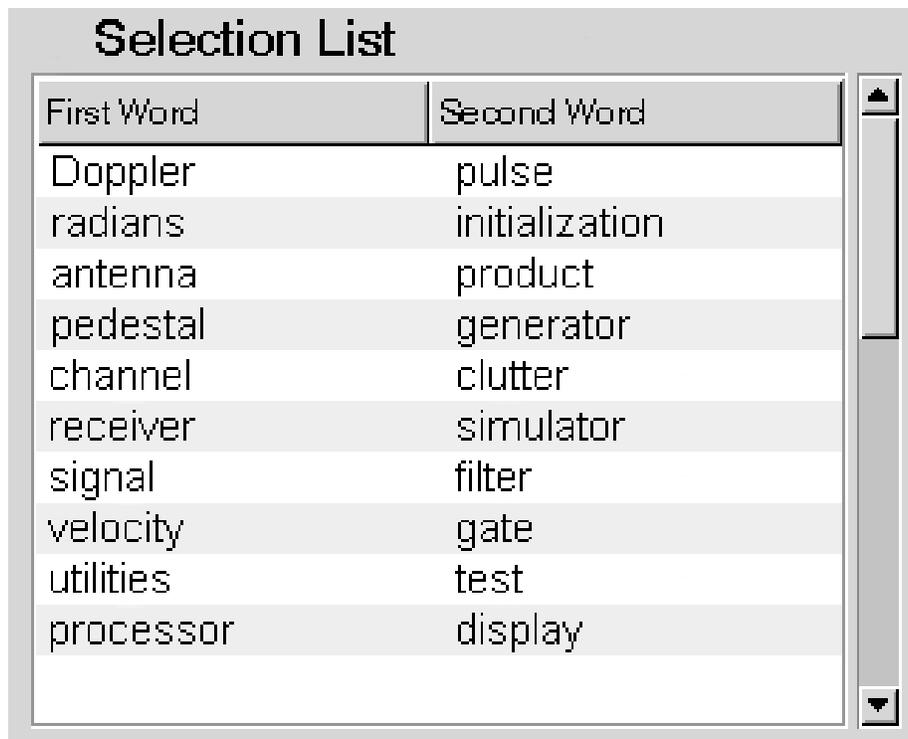


Figure 3-8. List Control

Guidelines:

- Always give list controls a label, positioned above or to the left of the list, in sentence capitalization.
- Make the list control large enough that it can show at least four items at a time without scrolling. For lists of ten or more items, increase this minimum size as appropriate.
- If the list appears in a dialog or utility window, consider making the window and the list within it resizable so that the user can choose how many list items are visible at a time without scrolling. Each time the user opens this dialog, set its dimensions to those that the user last resized it to.
- Do not use lists with less than about five items, unless the number of items may increase over time. Use check boxes, radio buttons or a drop-down list if there are fewer items.
- Only use column headers when the list has more than one column.
- Always label column headers when used. If the column is too narrow for a sensible label, provide a tooltip for the column instead.
- Consider providing Select All and Deselect All buttons beside multiple selection lists, if appropriate
- Ensure window size fits within the ROC Baseline standard for monitors at field sites.

3.8 Drop Down Lists.

Drop-down lists are used to select from a mutually exclusive set of options.

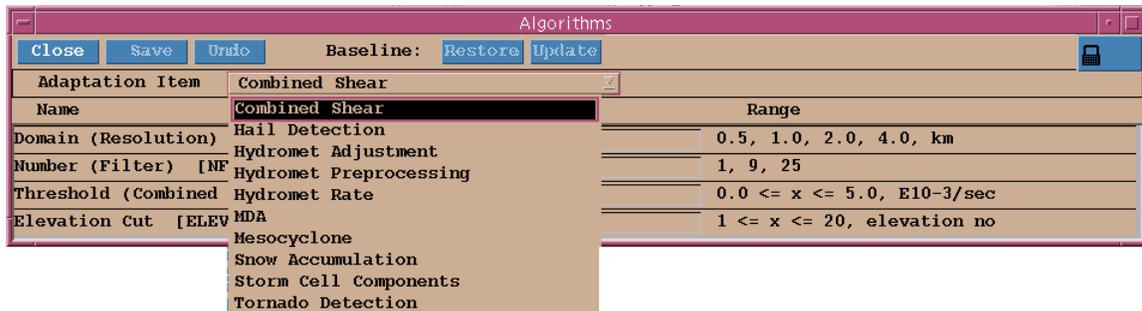


Figure 3-9. Drop Down List - Adaptation Items

Guidelines:

- If possible, avoid the use of drop-down lists with more than ten items.
- Label the drop-down list with a text label above it or to its left, using sentence capitalization.
- Use sentence capitalization for drop-down list items (drop down list items in Figure 3-9 are capitalized because they are titles).

3.9 Scrollbars.

A scrollbar alters which part of the object is currently visible inside the viewer control: it slides the view across the object in one axis (horizontal or vertical).

Guidelines:

- Only display scrollbars when they are required for sliding the view. If an object fits inside the viewer control, don't draw scrollbars.
- Do not use scrollbars as a replacement for a slider. Scrollbars should only be used affixed to a view that they actively alter, not used as a generic continuous input control.
- Affix scrollbars to the right side of a viewer control (to slide the view vertically), or to the bottom side (to slide the view horizontally).
- Scrollbars should be aligned in both directions with the view they are affixed to on the axis they control. In other words, horizontal scrollbars should span the full length of the viewer control, and vertical scrollbars should span the full height of the viewer control.

3.10 Tabbed Notebooks.

A tabbed notebook control is a convenient way of presenting related information in the same window, without having to display it all at the same time

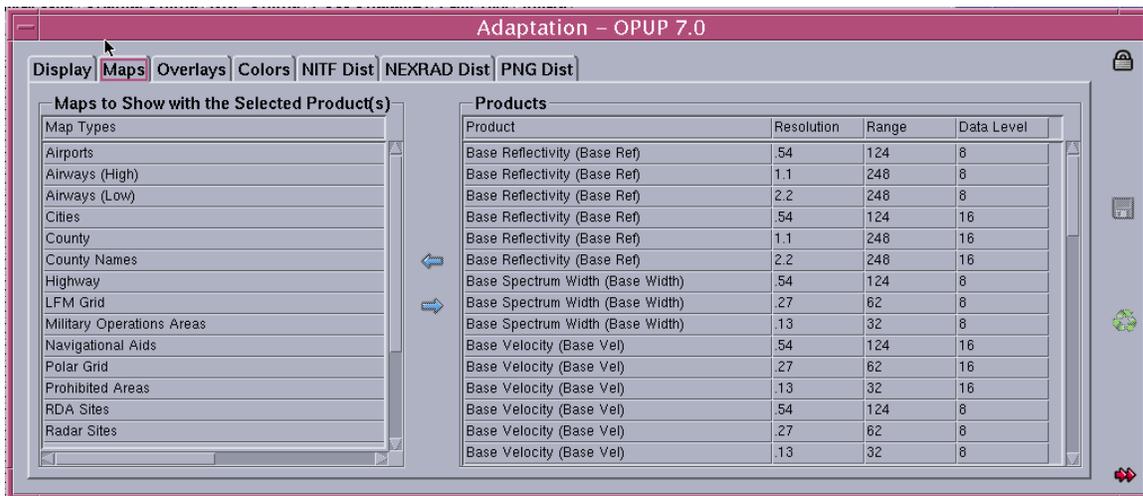


Figure 3-10. Tabbed Notebook

Guidelines:

- Do not put too many pages in the same notebook. If you cannot see all the tabs without scrolling or splitting them into multiple rows, consider alternative controls (e.g., list controls, button groups, or pull-down menus).
- Label tabs with header capitalization, and use nouns rather than verbs. Try to keep all labels in a notebook the same general length.
- Use tabs that are proportional to the width of their labels (Figure 3-11). Don't set all the tabs to the same width, as this makes them harder to scan visually and limits the number of tabs you can fit into the notebook without scrolling.

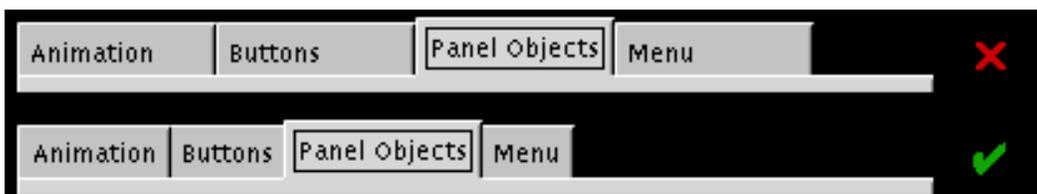


Figure 3-11. Fixed-Width and Proportional-Width Tabs

SECTION 4. VISUAL DESIGN

Good visual design is about communication. A well-designed application will make it easy for the user to understand the information that is being presented and show them clearly how they can interact with that information.

4.1 Window Titles.

Give every window a title. Make the title consistent with the label on the control (e.g., menu item, button) used to open it, although an exact match is not necessary.

A good window title contains information that is relevant to the user and distinguishes a particular window from other open windows. Omit information that does not assist in this selection, such as the application's version number or vendor name. These consume space, making titles in limited spaces such as the system window list less useful, and they add more text the user has to scan to find useful information.

4.2 Window Layout.

If labels are all similar in length, they should be left-aligned. This gives the user a firm left margin to anchor the eye and scan the list of items vertically more easily. If most of the labels in a group greatly differ in length, right-align them instead, so that the controls do not end up too far away from their corresponding labels.

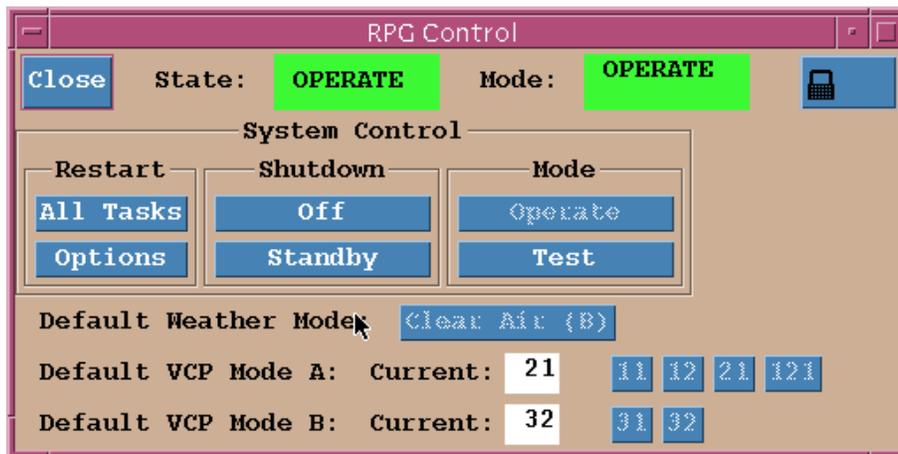


Figure 4-1. RPG Window with Left-Aligned Labels

Guidelines:

- Keep the location of the Close button consistent within each system.
- If a Padlock button is used, place it in the upper right corner.

- Labels must be concise and make sense when taken out of context.
- Lay out components left-to-right, top-to-bottom.
- Align controls in your layout exactly. The eye is very sensitive to aligned and unaligned objects. If nothing lines up with anything else in a window, it will be difficult for the user to scan the contents and find the desired information. Two things that almost line up, but not quite, are equally distracting.
- Be consistent. Use the same spacing, alignment, and component sizes in all dialogs appearing in your application. The Close and Cancel buttons, for example, should appear exactly the same in all dialog windows.
- Do not design windows that are more than 50% longer in one dimension than in the other. People are more comfortable looking at windows and dialogs whose dimensions stay within the golden ratio (about 1.6 to 1).
- Provide adequate space between controls and groups of controls. This white space will make it easier for the user to find the information they need.
- Leave a 12-pixel border between the edge of the window and the nearest controls.
- Leave a 12-pixel horizontal gap between a control and its label. The gap may be bigger for other controls in the same group due to differences in label lengths

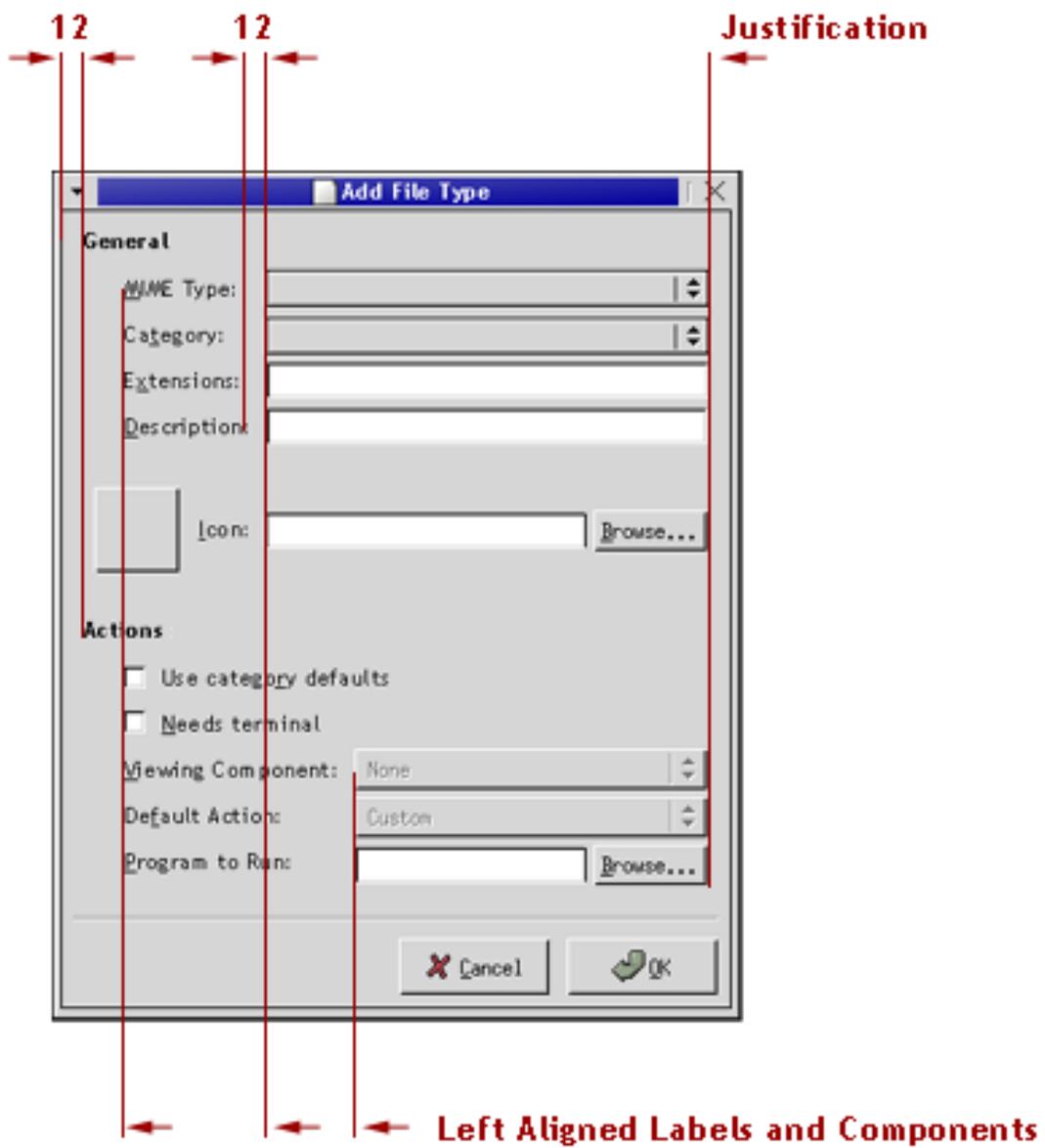
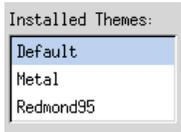
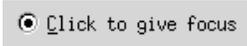
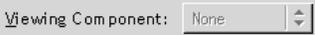


Figure 4-2. Layout Specifications (Measured in Pixels).

4.3 Text Labels: Spacing and Alignment.

Text provides the majority of the information and feedback in most applications. Choose and position text carefully to enable all users to use the application effectively. Use spacing and alignment of text uniformly.

Table 4-1. Alignment and Spacing of Text Elements

Element	Placement	Example
Large icons	Horizontally centered with and (6 pixels, if specification necessary) below large icon	
Small icons	Vertically centered with and (6 pixels, if specification necessary) to the right of small icons	
List control label	6 pixels above and horizontally left-aligned with list control	
Radio button and check box labels	6 pixels to the right of and vertically center-aligned with radio button	
Textfield labels	6 pixels to the left of and vertically center-aligned with textfield control	
Button labels	6 pixels of padding to either side of centered text (and any accompanying graphic). If appearing in a group of buttons, longest button label sets button size, and center all other button labels and accompanying graphics in same-sized buttons	
Other component labels (e.g., text fields)	12 pixels between the longest text label and its associated component, all other text labels in component grouping left-aligned with the longest label. All labels vertically center-aligned with associated components	

Guidelines:

- If the label precedes the control it is labelling, end the label with a colon. This helps identify it as a control's label rather than an independent item of text.
- Ensure that a label with a mnemonic is associated with the control it labels.
- Left-align components and labels, unless all the labels in a group have very different lengths. If they do, right-align the labels instead, to ensure that no controls end up too far away from their corresponding labels.
- Label objects with names that make sense when taken out of context.
- Be consistent with label usage and semantics. For example, if you use the same label in different windows, it should mean the same thing in both windows.
- Don't use the same label more than once in the same window.
- Do not use more than two or three different fonts and sizes in your application, and choose visually distinct rather than similar-looking fonts in one window.

4.4 Capitalization.

Two styles of capitalization are standard in user interface elements:

Header Capitalization - Capitalize all words in the element, with the following exceptions:

- Articles: a, an, the.
- Conjunctions: and, but, for, not, so, yet ...
- Prepositions of three or fewer letters: at, for, by, in, to ...

Sentence Capitalization - Capitalize the first letter of the first word, and any other words normally capitalized in sentences, such as application names.

Table 4-2. Capitalization Style Guidelines for User Interface Elements

Element	Style
Check box labels	Sentence
Command button labels	Header
Column heading labels	Header
Desktop background object labels	Header
Dialog messages	Sentence
Drop-down combination box labels	Sentence
Drop-down list box labels	Sentence
Field labels	Sentence
Filenames	Sentence
Graphic equivalent text: for example, Alt text on web pages	Sentence
Group box or frame labels	Header
Items in drop-down combination boxes, drop-down list boxes, and list boxes	Sentence
List box labels	Sentence
Menu items	Header
Menu items in applications	Header
Menu titles in applications	Header
Radio button labels	Sentence
Slider labels	Sentence
Spin box labels	Sentence
Tabbed section titles	Header
Text box labels	Sentence
Titlebar labels	Header
Toolbar button labels	Header
Tooltips	Sentence
Webpage titles and navigational elements	Header

SECTION 5. MISCELLANEOUS

5.1 Language and Labels.

Consistent labelling creates a familiar environment that the user can navigate comfortably. The more familiar the environment, the easier the task of finding information.

Clear, consistent and concise labelling of controls helps users to work out the purpose of a window or dialog they have never seen before.

- Keep labels short. Do not shorten labels to the point of losing meaning, however. A three-word label that provides clear information is better than a one-word label that is ambiguous or vague. Try to find the fewest possible words to satisfactorily convey the meaning of your label.
- Use standard terms with standard meanings. Examples include the following:

Close: Close the window.

Start: Start the specific process.

Stop: Stop the specific process.

Save: Save the input.

Cancel: Close the window and discard any changes.

- Apply standard capitalization rules.
- Do not use personal pronouns (you, we).

5.2 Units.

Use standard symbols and abbreviations for units of measure.

Table 5-1. Units and Abbreviations

Quantity	Unit	Symbol/ Abbreviation
angle	radian	rad
	degree	deg
angular velocity	radian per second	rad/s
	degree per second	deg/s
area	square meter	m ²

Table 5-1. Units and Abbreviations (Continued)

Quantity	Unit	Symbol/ Abbreviation
computer data	byte	byte
	octet	octet
	halfword	halfword
electrical current	ampere	A
electrical potential difference	volt	V
	kilovolt	kV
	millivolt	mV
frequency	hertz	Hz
	megahertz	MHz
length	centimeter	cm
	meter	m
	kilometer	km
	foot	ft
	1,000 feet	kft
	statute mile	mi
	nautical mile	nm
mass	gram	g
	kilogram	kg
percent	percent	%
power	watt	W
	kilowatt	kW
	megawatt	MW
	milliwatt	mW
	decibels above one milliwat	dBm
pressure	bar	bar
	millibar	mb

Table 5-1. Units and Abbreviations (Continued)

Quantity	Unit	Symbol/ Abbreviation
reflectivity	decibels of equivalent reflectivity	dBZ
speed	meter per second	m/s
	miles per hour	mph
	knot (nautical miles per hour)	kt
temperature	degrees Celsius	deg C
	degrees kelvin	K
time	second	s
	microsecond	usec
	millisecond	msec
	nanosecond	nsec
	minute	min
	hour	h
	day	d
	month	mo
	year	yr
volume	cubic meter	m ³

Guidelines:

- Unit symbols and abbreviations are not followed by a period, unless at the end of a sentence.
- Plural and singular forms are identical; do not add the letter “s” to symbols or abbreviations. Knots (kts) is the exception to this rule.
- Units with exponents are expressed with a caret ^ preceding the exponent (e.g., m² = m²). See Figure 5-1.
- In the expression for the value of a quantity, a space is left between the numerical value and the unit symbol (e.g., 10 m, not 10m)

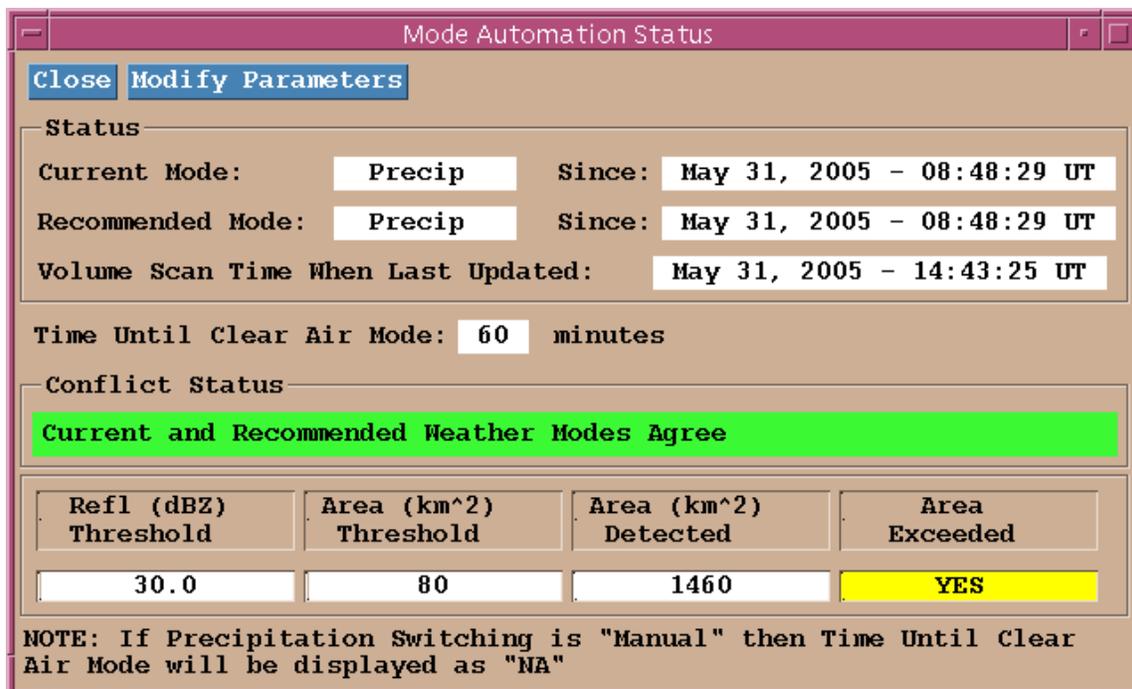


Figure 5-1. RPG Window with Exponents Indicated by Carets

Table 5-2. General Abbreviations

Acronym/ Abbreviation	Description
A/D	Analog/Digital
ABS	Absolute Value
AC	Air conditioner
AIS	Alarm Indication Signal
ANSI	American National Standards Institute
ANT	Antenna
ARC/VSWR	Arc/Voltage Standing Wave Ratio
ARP	Address Resolution Protocol
ASCII	American Standard Code for Information Interchange
ATTEN	Attenuator
AVG	Average

Table 5-2. General Abbreviations (Continued)

Acronym/ Abbreviation	Description
AZ	Azimuth
BAM	Binary Angular Measure
BDDS	Base Data Distribution System
BIT	Built-In Test
BITE	Built-in-Test-Equipment
CAL	Calibration
CF	Clutter Filter
CHAN	Channel
CI	Configuration Item (hardware)
CMD	Command
COHO	Coherent Oscillator
COM	Communications
CPCI	Computer Program Configuration Item
CSU	Channel Service Unit
CTRL	Control
CW	Contiguous Wave
DAU	Data Acquisition Unit
DCU	Digital Control Unit
DEV	Deviation
DOC	Department of Commerce
DoD	Department of Defense
DOT	Department of Transportation
DSP	Digital Signal Processor
ED	Edge Detected
EDG	Edge Detected
EL	Elevation

Table 5-2. General Abbreviations (Continued)

Acronym/ Abbreviation	Description
EQUIP	Equipment
EST	Estimate
FAA	Federal Aviation Administration
FE	Front End
FOC	Filtered Occurrence
FREQ	Frequency
GEN	Generator
GPS	Global Positioning System
HCI	Human Computer Interface
I/O	Input/Output
ICD	Interface Control Document
ICMP	Internet Control Message Protocol
ID	Identification
IF	Intermediate Frequency
IFD	Intermediate Frequency Digitizer
IHL	Internet Header Length
INIT	Initialization
INOP	Inoperable
IP	Internet Protocol
IPC	Inter-Processor Communications
KD	Delayed Klystron
KLY	Klystron
LAN	Local Area Network
LOG	Logarithmic
LSB	Least Significant Bit
MAINT	Maintenance

Table 5-2. General Abbreviations (Continued)

Acronym/ Abbreviation	Description
MLOS	Microwave Line-Of-Sight
MM	Maintenance Mandatory
MR	Maintenance Required
MSB	Most Significant Bit
MSCF	Master Station Console Function
N/A	Not Applicable
NTP	Network Time Protocol
NWS	National Weather Service
O	Occurrence
OCC	Occurrence
OSI	Open System Interconnect
PCU	Pedestal Control Unit
PFN	Pulse Forming Network
PMC	Program Management Committee
PPP	Point-to-Point Protocol
PRF	Pulse Repetition Frequency
PS	Power Supply
PVC	Permanent Virtual Channel
PWA	Printed Wire Assembly
PWR	Power
R,V,W	Reflectivity, Velocity, Spectrum Width
RAI	Resource Availability Indication
RAS	Remote Access Server
RCP	Radar Control Processor
RCV	Another representation for Receiver
RDA	Radar Data Acquisition area (hardware and software)

Table 5-2. General Abbreviations (Continued)

Acronym/ Abbreviation	Description
RDASC	RDA Status and Control
REG	Regulator
REL	Release
REQ	Request, Required
RF	Radiated Frequency
RFD	Radio Frequency Drive
RMS	Remote Monitoring Subsystem
ROC	Radar Operations Center
RPG	Radar Product Generation area (hardware and software)
RVP	Radar Video Processor
SEC	Secondary Alarm
SEQ	Sequence
SG	Sigmat
SIG	Signal
SNMP	Simple Network Management Protocol
SNR	Signal to Noise Ratio
SP	Signal Processor
STALO	Stable Local Oscillator
STD	Standard
STS	System Test Software
SW	Spectrum Width
SYS	System Information
T1	Type 1 communications carrier link (1.544 megabits/second)
TCM	Trellis Encoded Modulation
TCP	Transmission Control Protocol
TEMP	Temperature

Table 5-2. General Abbreviations (Continued)

Acronym/ Abbreviation	Description
TOUTS	Time-outs
TOW	Tower/Utilities
TPS	Transitional Power Source
TR	Another designator for the Transmitter
TST	Test
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram
UPS	Uninterruptible Power Supply
UTIL	Utility or Utilities
V & V	Verification & Validation
VCP	Volume Coverage Pattern
VDC	Volts Direct current
VEL	Velocity
VSWR	Voltage Standing Wave Ratio
WG	Wave Guide
WSR-88D	Weather Service Radar - 88 Doppler
XMT	Another representation for Transmitter

5.3 Response Times.

Display feedback as soon as possible. SS or SRS requirements supersede the guidance in the table below, if they conflict.

Table 5-3. Recommended Response Times for Typical Events

User Interface (UI) Event	Recommended Response Time
Mouse click, pointer movement, window movement or resizing, keypress, button press, drawing gesture, other UI input event involving hand-eye co-ordination	0.1 second
Displaying progress indicators, completing ordinary user commands (e.g. closing a window), completing background tasks (e.g. reformatting a table)	1.0 second
Displaying a graph or anything else a typical user would expect to take time.	10.0 seconds
Accepting and processing all user input to any task	10.0 seconds

The acceptable response delay for each event is based on a typical user's sense that the event is a logical point at which to stop or pause. Delay times also depend on the requirements of the platform.

5.4 Date/Time Display.

For decimal notation, use the following 24-hour formats to display date and time:

Date: **MM/DD/YYYY** (upper case) and specific date: 00/00/0000

Time: **hh:mm:ss** (lower case) and specific time: 00:00:00

For alphanumeric notation of the date, use the three-letter abbreviation for the month, followed by the day and year (e.g., Feb 4, 2006).

5.5 Background Color Codes.

Specific background colors indicate the status of alarms or other elements of the system. These colors are viewable from a distance, enabling users to more effectively monitor changes in status. Table 5-4 lists each color with its indication.

Table 5-4. Status Background Color Codes

Color	Indication
Green	Good. Status is normal.
White	Benign. No problem is indicated.
Yellow	Caution. Used with alarms to indicate maintenance required. Also used to indicate a transition state.
Orange	Used with alarms to indicate maintenance is mandatory.
Red	Item has failed and/or is inoperable. May also indicate shutdown of software or components, intentional or not.

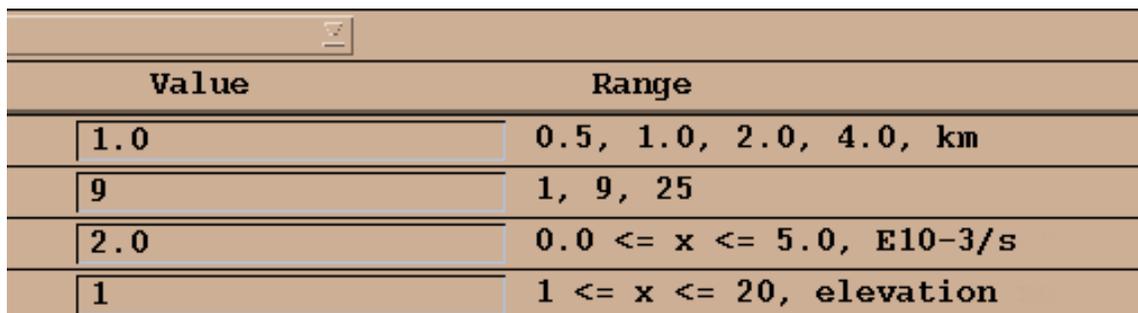
5.6 Sensitized/Desensitized Controls.

It is usually better to desensitize a control than to hide it altogether. This way, users can learn about functionality they may be able to use later.

5.7 Values and Ranges.

When describing specific values, use commas to separate them (0, 10, 20).

Describe a range of values, such as 1 to 20, as simply “1 to 20”, or use the symbols <, >, and = (e.g., a range from 1 to 20 is expressed as $1 \leq x \leq 20$). Be consistent within the application, however. See Figure 5-2.



Value	Range
1.0	0.5, 1.0, 2.0, 4.0, km
9	1, 9, 25
2.0	$0.0 \leq x \leq 5.0$, E10-3/s
1	$1 \leq x \leq 20$, elevation

Figure 5-2. Section of RPG Window with Specific Values and Ranges

INDEX

Title	Page
A	
Abbreviations	22
B	
Buttons	
Format	6
Placement	13
C	
Capitalization Styles	17
Check Boxes	7
Color	29
Consistency	1
D	
Date/Time	28
Desensitized Controls	29
Drop Down Lists	11
E	
Exponents	21
F	
Feedback Response Times	28
Format Elements	1
M	
Menu Item Names	3
Formats	4
N	
Numeric Entry Fields	5
R	
Radio Buttons	8
Range of Values	29
S	
Scrollbars	11
Sliders	6
T	
Tabbed Notebooks	12

INDEX

Title	Page
Text Entry	5
Text Labels	
Language	19
Spacing and Alignment	16
Toggle Buttons	9
U	
Units	19
W	
Windows	
Layout	13
Titles	13