

# G2 GUIDE/UII

## Procedures Reference Manual

Version 2015



G2 GUIDE/UII Procedures Reference Manual, Version 2015

December 2015

The information in this publication is subject to change without notice and does not represent a commitment by Gensym Corporation.

Although this software has been extensively tested, Gensym cannot guarantee error-free performance in all applications. Accordingly, use of the software is at the customer's sole risk.

Copyright (c) 1985-2015 Gensym Corporation

All rights reserved. No part of this document may be reproduced, stored in a retrieval system, translated, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Gensym Corporation.

Gensym®, G2®, Optegrity®, and ReThink® are registered trademarks of Gensym Corporation.

NeurOn-Line™, Dynamic Scheduling™, G2 Real-Time Expert System™, G2 ActiveXLink™, G2 BeanBuilder™, G2 CORBALink™, G2 Diagnostic Assistant™, G2 Gateway™, G2 GUIDE™, G2GL™, G2 JavaLink™, G2 ProTools™, GDA™, GFI™, GSI™, ICP™, Integrity™, and SymCure™ are trademarks of Gensym Corporation.

Telewindows is a trademark or registered trademark of Microsoft Corporation in the United States and/or other countries. Telewindows is used by Gensym Corporation under license from owner.

This software is based in part on the work of the Independent JPEG Group.

Copyright (c) 1998-2002 Daniel Veillard. All Rights Reserved.

SCOR® is a registered trademark of PRTM.

License for Scintilla and SciTE, Copyright 1998-2003 by Neil Hodgson, All Rights Reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).

All other products or services mentioned in this document are identified by the trademarks or service marks of their respective companies or organizations, and Gensym Corporation disclaims any responsibility for specifying which marks are owned by which companies or organizations.

Gensym Corporation  
52 Second Avenue  
Burlington, MA 01803 USA  
Telephone: (781) 265-7100  
Fax: (781) 265-7101

Part Number: DOC023-1200

# Contents Summary

---

	<b>Preface</b>	<b>xxi</b>
<b>Chapter 1</b>	<b>Introduction</b>	<b>1</b>
<b>Chapter 2</b>	<b>Border Objects</b>	<b>7</b>
<b>Chapter 3</b>	<b>Buttons</b>	<b>23</b>
<b>Chapter 4</b>	<b>Check Boxes</b>	<b>45</b>
<b>Chapter 5</b>	<b>Check Buttons</b>	<b>59</b>
<b>Chapter 6</b>	<b>Dialogs</b>	<b>73</b>
<b>Chapter 7</b>	<b>Dialog Titles</b>	<b>195</b>
<b>Chapter 8</b>	<b>Edit Boxes, Combo Boxes, and Spin Control Boxes</b>	<b>207</b>
<b>Chapter 9</b>	<b>Graphical Messages</b>	<b>251</b>
<b>Chapter 10</b>	<b>Graphical Objects</b>	<b>263</b>
<b>Chapter 11</b>	<b>Message Objects</b>	<b>309</b>
<b>Chapter 12</b>	<b>Navigation Buttons</b>	<b>361</b>
<b>Chapter 13</b>	<b>Push Buttons</b>	<b>377</b>
<b>Chapter 14</b>	<b>Radio Boxes</b>	<b>395</b>
<b>Chapter 15</b>	<b>Radio Buttons</b>	<b>411</b>
<b>Chapter 16</b>	<b>Scroll Areas</b>	<b>425</b>
<b>Chapter 17</b>	<b>Selection Boxes</b>	<b>469</b>

<b>Chapter 18</b>	<b>Selection Buttons</b>	<b>487</b>
<b>Chapter 19</b>	<b>Separators</b>	<b>499</b>
<b>Chapter 20</b>	<b>Text Objects</b>	<b>509</b>
<b>Chapter 21</b>	<b>Toggle Buttons</b>	<b>525</b>
<b>Chapter 22</b>	<b>Items and Relations</b>	<b>541</b>
<b>Chapter 23</b>	<b>Upgrade and Utility Procedures</b>	<b>551</b>
<b>Appendix A</b>	<b>System-Defined Relations</b>	<b>635</b>
<b>Appendix B</b>	<b>Common Object Attributes</b>	<b>643</b>
	<b>Index</b>	<b>651</b>

# Contents

---

## **Preface xxi**

About this Manual **xxi**

Audience **xxii**

Organization **xxii**

How to Use the Chapters in this Manual **xxiv**

    Description of UIL Objects **xxiv**

    Class Hierarchy Figure **xxiv**

    Attribute Tables **xxv**

    Operations Table **xxvi**

    Object Relations Table **xxvii**

A Note About the API **xxviii**

Conventions **xxviii**

Related Documentation **xxx**

Customer Support Services **xxxii**

## **Chapter 1 Introduction 1**

Introduction **1**

Common Tasks Performed with UIL Procedures **1**

Procedure Priority Setting **6**

## **Chapter 2 Border Objects 7**

Introduction **7**

    Border Object Class Hierarchy **8**

    Attributes of Border Objects **9**

    Operations Performed on Border Objects **10**

Procedures that Operate on Border Objects **13**

    uil-adjust-border-on-workspace **14**

    uil-anchor-box-corner-to-workspace-edge **15**

    uil-create-border-for-text **16**

    uil-create-box-border **17**

    uil-get-border **18**

    uil-move-related-border **19**

uil-release-box-corner-from-workspace-edge 20

Additional Procedures that Support Border Objects 21

### **Chapter 3 Buttons 23**

Introduction 24

Button Class Hierarchy 24

Button Relations 25

Procedures that Operate on Buttons 26

uil-add-label-to-button 27

uil-center-label-in-text-button 28

uil-create-custom-button 29

uil-create-text-for-button-from-label 30

uil-handle-button 31

uil-handle-button-method 32

uil-highlight-button 33

uil-reset-button 34

uil-return-size-of-button-as-symbol 35

uil-select-button 36

uil-toggle-button 37

uil-update-button-label 38

uil-update-button-togglelabel 39

Additional Procedures that Support Buttons 40

Superseded Procedures 41

uil-disable-button 42

uil-enable-button 43

### **Chapter 4 Check Boxes 45**

Introduction 45

Check Box Class Hierarchy 46

Attributes of Check Boxes 46

Operations Performed on Check Boxes 48

Check Box and Check Button Relations 50

Procedures that Operate on Check Boxes 51

uil-create-check-box 52

uil-get-check-box-values 53

uil-set-check-box-values 54

Additional Procedures that Support Check Boxes 55

Superseded Procedures 56

uil-delete-check-box 57

## **Chapter 5 Check Buttons 59**

### **Introduction 59**

- Check Button Class Hierarchy 60
- Attributes of Check Buttons 60
- Operations Performed on Check Buttons 63
- Check Button and Check Box Relations 64

### **Procedures that Operate on Check Buttons 65**

- uil-create-styled-check-button 66
- uil-create-styled-check-button-with-different-sized-label 67

### **Additional Procedures that Support Check Buttons 68**

### **Superseded Procedures 69**

- uil-create-check-button 70
- uil-delete-check-button 71

## **Chapter 6 Dialogs 73**

### **Introduction 76**

- Dialog Class Hierarchy 76
- Tab Dialog Classes 76
- Dialog Attributes 77
- Tab Dialog Attributes 83
- Operations Performed on Dialogs 84
- Dialog Relations 86
- Tab Dialog Relations 87

### **Procedures that Operate on Dialogs 88**

- uil-add-apply-button-to-dialog 90
- uil-add-cancel-button-to-dialog 91
- uil-add-grobject-to-dialog 92
- uil-add-grobject-to-tab-page 93
- uil-add-item-to-dialog 94
- uil-add-message-to-dialog 95
- uil-add-ok-button-to-dialog 96
- uil-add-standard-buttons-to-dialog 97
- uil-add-tab-page-to-dialog 98
- uil-cancel-edit-session-in-window-if-any 99
- uil-check-module-consistency-for-dialog 100
- uil-clear-transient-dialog-pool 101
- uil-control-dialog-callback 102
- uil-create-confirm-dialog 104
- uil-create-dialog-border 105
- uil-create-message-dialog 106
- uil-create-query-dialog 107
- uil-create-tab-dialog 108
- uil-create-tab-page 109

uil-create-tab-pushbutton 110  
uil-create-tailored-dialog 111  
uil-delete-permanent-copies-in-transient-dialog-pool 112  
uil-end-dialog-edit 113  
uil-find-destination-object 114  
uil-find-initiating-object 115  
uil-find-initiating-object-for-parent-dialog 116  
uil-find-source-object 117  
uil-find-target-object 118  
uil-find-temporary-storage-object 119  
uil-find-workspace-object-maxima 120  
uil-flush-dialog-event-queue 121  
uil-generate-customized-dialog 122  
uil-generate-default-dialog 124  
uil-get-active-dialog-on-window-if-one-exists 125  
uil-get-child-dialogs 126  
uil-get-dialog-for-this-temporary-storage-object 127  
uil-get-dialog-launched-for-item 128  
uil-get-dialogs-managed-on-window 129  
uil-get-grobject-from-id 130  
uil-get-grobject-from-id-on-dlg-or-wksp 131  
uil-get-master-dialog 132  
uil-get-parent-dialog 133  
uil-get-window-on-which-dialog-is-managed 134  
uil-get-windows-on-which-dialog-is-managed 135  
uil-hide-managed-dialog 136  
uil-insert-actions-into-dialog-event-queue 137  
uil-is-dialog-managed-on-window 138  
uil-is-this-a-child-dialog-of-this-dialog 139  
uil-is-this-dialog-managed-for-this-item 140  
uil-is-this-the-parent-dialog 141  
uil-locate-first-edit-box-on-dialog 142  
uil-locate-next-edit-box-on-dialog 143  
uil-locate-next-rows-edit-box-on-dialog 144  
uil-move-grobject-to-tab-page-layer 145  
uil-override-initiating-object-for-dialog 146  
uil-permanent-copies-exist-in-transient-dialog-pool 147  
uil-post-delay-notification 148  
uil-post-generic-dialog 149  
uil-post-notification-dialog 151  
uil-quit-dialog-edit 152  
uil-raise-tab-page-to-top 153  
uil-raise-tab-page-to-top-callback 154  
uil-release-dialog 155  
uil-remove-all-remaining-actions 156  
uil-remove-delay-notification-if-any 157  
uil-remove-grobject-from-tab-page 158



- uil-remove-notification-dialog 159
- uil-remove-tab-page-from-dialog 160
- uil-reserve-dialog 161
- uil-restore-dialog-grobject-list 163
- uil-select-tab-page 164
- uil-set-destination-object-for-dialog 165
- uil-show-managed-dialog 166
- uil-shrink-wrap-dialog 167
- uil-simulate-play-mode 168
- uil-start-dialog 169
- uil-start-dialog-processing 170
- uil-start-or-refocus-dialog 172
- uil-store-dialog-in-home-bin 173
- uil-translate-dialog 174
- uil-unsimulate-play-mode 175
- uil-update-controls-for-specific-attribute 176
- uil-update-dialog-for-initiating-object 177
- uil-update-notification-text 178
- uil-update-wait-clock-face 179
- uil-utility-create-permanent-copies-for-module 180
- uil-utility-create-permanent-copy-for-dialog 181
- uil-wait-for-dialog 182

Additional Procedures that Support Dialogs 183

Superseded Procedures 184

- uil-apply-edit-dialog-callback 185
- uil-clear-transient-dialog-pool-and-permanent-copies 186
- uil-delete-dialog 187
- uil-get-query-dialog-value 188
- uil-manage-and-show-dialog 189
- uil-popdown-and-delete-dialog-callback 190
- uil-popdown-and-hide-managed-dialog-callback 191
- uil-popdown-and-keep-edit-dialog-callback 192
- uil-popdown-dialog-callback 193
- uil-update-wait-semaphore 194

## **Chapter 7 Dialog Titles 195**

Introduction 195

- Dialog Title Class Hierarchy 196
- Dialog Title Attributes 196
- Internationalization of Dialog Titles 198
- Operations Performed on Dialogs Titles 198

Procedures that Operate on Dialog Titles 201

- uil-create-and-manage-dialog-title 202
- uil-create-dialog-title 203

uil-create-styled-dialog-title 204  
uil-justify-dialog-title 205  
uil-title-button-callback 206

## **Chapter 8 Edit Boxes, Combo Boxes, and Spin Control Boxes 207**

### **Introduction 208**

Edit Box Class Hierarchy 209  
Edit Box, Combo Box, and Spin Control Attributes 210  
Operations Performed on Edit Boxes 214  
Edit Box, Spin Control, and Combo Box Relations 216  
Combo Box Relations 217  
Spin Control Relations 217

### **Procedures that Operate on Edit Boxes 218**

uil-commit-edit-session-result-to-edit-box 219  
uil-create-styled-edit-box 220  
uil-get-control-with-focus 221  
uil-get-control-with-focus-on-window 222  
uil-edit-box-selection 223  
uil-edit-box-selection-method 224  
uil-edit-box-unselection 225  
uil-edit-box-unselection-method 226  
uil-get-edit-box-value 227  
uil-launch-edit-session 228  
uil-register-edit-session-results 229  
uil-select-edit-box 230  
uil-select-edit-box-on-dialog 231  
uil-unselect-edit-box 232  
uil-update-edit-box-label 233

### **Procedures that Operate on Combo Boxes 234**

uil-add-to-combo-box-list 235  
uil-clear-combo-box-list 236  
uil-combo-box-message-selection 237  
uil-combo-box-no-selection 238  
uil-combo-box-show-list 239  
uil-combo-box-unselection 240  
uil-create-combo-box 241  
uil-update-combo-box-list 242

### **Procedures that Operate on Spin Control Boxes 243**

uil-create-spin-control 244  
uil-delete-spin-control 245

### **Additional Procedures that Support Edit Boxes 246**

#### **Superseded Procedures 247**

uil-create-edit-box 248

uil-create-sized-edit-box 249  
uil-delete-edit-box 250

## **Chapter 9 Graphical Messages 251**

Introduction 251  
    Graphical Message Class Hierarchy 252  
    Operations Performed on Graphical Messages 253

Procedures that Operate on Graphical Messages 255  
    uil-create-custom-text 256  
    uil-get-message-for-border 257  
    uil-return-size-of-grmes-as-symbol 258  
    uil-update-text 259  
    uil-update-unformatted-text 260

Additional Procedures that Support Graphical Messages 261

## **Chapter 10 Graphical Objects 263**

Introduction 264  
    Graphical Object Class Hierarchy 265  
    Operations Performed on Graphical Objects 265

Procedures that Operate on Graphical Objects 268  
    uil-call-conclude-method-for-grobj 269  
    uil-call-update-method-for-grobj 270  
    uil-call-validate-method-for-grobj 271  
    uil-change-size-of-grobj 272  
    uil-change-window-style-of-grobj 273  
    uil-clone-grobj 274  
    uil-clone-grobj-method 275  
    uil-conclude-grobj 276  
    uil-conclude-grobj-method 277  
    uil-configure-grobj 278  
    uil-configure-grobj-method 279  
    uil-delete-grobj 280  
    uil-delete-grobj-method 281  
    uil-delete-grobj-with-confirmation 282  
    uil-disable-grobj 283  
    uil-disable-grobj-method 284  
    uil-drop-border-around-grobj 285  
    uil-enable-grobj 286  
    uil-enable-grobj-method 287  
    uil-initialize-grobj 288  
    uil-initialize-grobj-method 289  
    uil-make-grobj-permanent 290  
    uil-manage-grobj 291  
    uil-manage-grobj-method 292

- uil-move-grobj 293
- uil-set-slider-value 294
- uil-size-of-grobj 295
- uil-size-of-grobj-method 296
- uil-unmanage-grobj 297
- uil-update-grobj 298
- uil-update-grobj-method 299
- uil-utility-restore-grobj 300
- uil-validate-grobj 301
- uil-validate-grobj-method 302

Procedures that Operate on Sliders 303

- uil-create-slider 304
- uil-get-slider-value 305
- uil-rotate-slider 306

Additional Procedures that Support Graphical Objects 307

## **Chapter 11 Message Objects 309**

Introduction 310

- Message Object Class Hierarchy 310
- Message Object Attributes 311
- Operations Performed on Message Objects 314
- Scroll Area and Message Object Relations 316

Procedures that Operate on Message Objects 317

- uil-add-message-to-list 318
- uil-add-message-to-list-at-position 319
- uil-append-item-to-message 321
- uil-clear-messages 322
- uil-copy-message-object-system-attributes 323
- uil-copy-messages 324
- uil-create-message 325
- uil-create-message-with-appended-item 327
- uil-delete-selected-messages 328
- uil-display-message-list 329
- uil-does-message-belong-to-scroll-area 330
- uil-find-excess-message-and-remove 331
- uil-generate-time-stamp-for-message 332
- uil-get-first-message 333
- uil-get-index-of-scroll-message 334
- uil-get-item-appended-to-message 335
- uil-get-last-message 336
- uil-get-message-appended-to-this-item 337
- uil-get-message-from-index 338
- uil-get-message-object-state 339
- uil-get-scroll-area-for-this-message 340

- uil-get-selected-message 341
- uil-get-selected-message-objects 342
- uil-insert-message-in-ordered-list 343
- uil-is-message-selected 344
- uil-message-selection 345
- uil-message-selection-method 346
- uil-message-unselection 347
- uil-message-unselection-method 348
- uil-move-message 349
- uil-move-messages 350
- uil-on-double-click 351
- uil-on-single-click 352
- uil-remove-message 353
- uil-remove-message-from-list 354
- uil-select-message 355
- uil-select-message-on-window 356
- uil-sort-scroll-area-messages 357
- uil-unselect-message 358
- uil-update-message-list-with-message 359

Additional Procedures that Operate on Message Objects 360

## **Chapter 12 Navigation Buttons 361**

Introduction 361

- Navigation Button Class Hierarchy 362

- Navigation Button Attributes 364

- Operations Performed on Navigation Buttons 365

- Navigation Button Relations 367

Procedures that Operate on Navigation Buttons 368

- uil-create-navigation-button 369

- uil-handle-navigation-button-method 370

- uil-handle-workspace-button-method 371

- uil-select-navigation-button 372

Additional Procedures that Support Navigation Buttons 373

Superseded Procedures 374

- uil-delete-navigation-button 375

## **Chapter 13 Push Buttons 377**

Introduction 377

- Push Button Class Hierarchy 378

- Push Button Attributes 378

- Operations Performed on Push Buttons 383

- Push Button Relations 385

Procedures that Operate on Push Buttons 386

- uil-create-pushbutton-to-launch-dialog 387
- uil-create-styled-pushbutton 388
- uil-unselect-button 389

Additional Procedures that Support Push Buttons 390

Superseded Procedures 391

- uil-create-pushbutton 392
- uil-delete-pushbutton 393

## **Chapter 14 Radio Boxes 395**

Introduction 396

- Radio Box Class Hierarchy 396
- Radio Box Attributes 397
- Operations Performed on Radio Boxes 398
- Radio Box and Radio Button Relations 400

Procedures that Operate on Radio Boxes 402

- uil-create-radio-box 403
- uil-get-radio-box-value 404
- uil-reset-radio-box 405
- uil-set-radio-box-value 406

Additional Procedures that Support Radio Boxes 407

Superseded Procedures 408

- uil-delete-radio-box 409

## **Chapter 15 Radio Buttons 411**

Introduction 411

- Radio Button Class Hierarchy 411
- Radio Button Attributes 412
- Operations Performed on Radio Buttons 415
- Radio Box and Radio Button Relations 416

Procedures that Operate on Radio Buttons 417

- uil-create-styled-radio-button 418
- uil-create-styled-radio-button-with-different-sized-label 419

Additional Procedures that Support Radio Buttons 421

Superseded Procedures 422

- uil-create-radio-button 423
- uil-delete-radio-button 424

## **Chapter 16 Scroll Areas 425**

Introduction 426

- Scroll Area Class Hierarchy 426

Scroll Area Attributes	426
Operations Performed on Scroll Areas	433
Scroll Area and Message Object Relations	435
Procedures that Operate on Scroll Areas	436
uil-add-resize-corner-to-scroll-area	437
uil-change-scroll-area-message-font-size	438
uil-create-customized-scroll-area	439
uil-create-scroll-area-from-array	440
uil-create-scroll-area-from-list	441
uil-create-sized-scroll-area-from-array	442
uil-create-sized-scroll-area-from-list	443
uil-create-styled-scroll-area	444
uil-get-related-scroll-area	445
uil-reset-scroll-area	446
uil-resize-scroll-area	447
uil-return-module-name-for-scroll-area-definition	448
uil-scroll-area-reflect	449
uil-scroll-decrement	450
uil-scroll-decrement-line	451
uil-scroll-decrement-line-method	452
uil-scroll-decrement-method	453
uil-scroll-increment	454
uil-scroll-increment-line	455
uil-scroll-increment-line-method	456
uil-scroll-increment-method	457
uil-scroll-to-position	458
uil-scroll-to-position-method	459
uil-select-matching-values-in-scroll-area	460
uil-update-scroll-area-from-array	461
uil-update-scroll-area-from-list	462
uil-update-scroll-area-message	463
uil-update-scroll-area-messages	464
Additional Procedures that Support Scroll Areas	465
Superseded Procedures	466
uil-create-scroll-area	467
uil-delete-scroll-area	468
<b>Chapter 17 Selection Boxes</b>	<b>469</b>
Introduction	469
Selection Box Class Hierarchy	470
Selection Box Attributes	470
Operations Performed on Selection Boxes	471
Selection Box and Selection Button Relations	473
Procedures that Operate on Selection Boxes	475

- uil-add-button-to-selection-box 476
- uil-get-buttons-for-this-selection-box 477
- uil-get-buttons-grouped-by-this-selection-box 478
- uil-get-selection-box-for-button 479
- uil-get-selection-box-last-button 480
- uil-get-selection-box-lead-button 481
- uil-utility-check-for-empty-selection-box 482

Additional Procedures that Support Selection Boxes 483

Superseded Procedures 484

- uil-frame-selection-box 485

## **Chapter 18 Selection Buttons 487**

Introduction 487

- Selection Button Class Hierarchy 488

- Selection Button Attributes 488

- Operations Performed on Selection Buttons 491

- Selection Box and Selection Button Relations 492

Procedures that Operate on Selection Buttons 493

- uil-get-next-selection-button-y-position 494

- uil-get-selection-button 495

- uil-is-this-button-grouped-by-this-selection-box 496

Additional Procedures that Support Selection Buttons 497

## **Chapter 19 Separators 499**

Introduction 499

- Separator Class Hierarchy 499

- Separator Attributes 501

- Operations Performed on Separators 501

Procedures that Operate on Separators 504

- uil-create-line-separator 505

- uil-rotate-line-separator 506

Additional Procedures that Support Separators 507

## **Chapter 20 Text Objects 509**

Introduction 509

- Text Object Class Hierarchy 509

- Text Object Attributes 510

- Operations Performed on Text Objects 512

- Text Object Relations 514

Procedures that Operate on Text Objects 515

- uil-create-styled-text 516



	uil-get-grobject-for-label	517
	uil-get-label-text	518
	uil-select-text	519
	uil-translate-text	520
	Additional Procedures that Support Text Objects	521
	Superseded Procedures	522
	uil-create-text	523
	uil-delete-text	524
<b>Chapter 21</b>	<b>Toggle Buttons</b>	<b>525</b>
	Introduction	525
	Toggle Button Class Hierarchy	526
	Toggle Button Attributes	527
	Operations Performed on Toggle Buttons	530
	Toggle Button Relations	532
	Procedures that Operate on Toggle Buttons	533
	uil-create-styled-toggle-button	534
	uil-get-toggle-button-state	535
	uil-set-toggle-button-state	536
	Additional Procedures that Support Toggle Buttons	537
	Superseded Procedures	538
	uil-create-toggle-button	539
	uil-delete-toggle-button	540
<b>Chapter 22</b>	<b>Items and Relations</b>	<b>541</b>
	Procedures for Accessing UIL Objects in Relations	542
	Procedures that Access Information about Relations	545
	uil-are-any-related-to	546
	uil-is-related-to	547
	uil-get-related-item	548
	uil-get-related-items	549
<b>Chapter 23</b>	<b>Upgrade and Utility Procedures</b>	<b>551</b>
	Upgrade Procedures	553
	Upgrade Procedures	554
	uil-30r1-upgrades	555
	uil-30r3-and-30r4-upgrades	556
	uil-30r3-upgrade-button-label-texts	557
	uil-30r3-upgrade-radio-and-check-button-label-texts	558
	uil-30r3-upgrade-scroll-area-keys	559
	uil-40r0-upgrades	560

uil-convert-button-from-4.1-to-5.0 **561**  
uil-convert-buttons-from-4.1-to-5.0 **562**  
uil-convert-dialog-subworkspaces-from-4.1-to-5.0 **563**  
uil-convert-scroll-area-to-41r0 **564**

#### Utility Procedures **565**

uil-add-prefix-to-user-criteria-prefixes **567**  
uil-create-uilib-module-setting **568**  
uil-date-verification **569**  
uil-determine-if-guide-is-loaded **570**  
uil-display-color-selector-dialog **571**  
uil-do-nothing **573**  
uil-generate-rules-for-attributes **574**  
uil-get-attribute **575**  
uil-get-configuration **576**  
uil-get-text-extent **577**  
uil-get-user-criteria-prefixes **578**  
uil-guidata-launch-dialog-for-item **579**  
uil-is-this-item-near-another **580**  
uil-remove-guide-application **581**  
uil-remove-prefix-from-user-criteria-prefixes **582**  
uil-reset-uilib-module-setting-to-defaults **583**  
uil-reset-user-criteria-prefixes **584**  
uil-return-bin-workspace-for **585**  
uil-return-configuration-for-color **586**  
uil-set-attribute **587**  
uil-time-verification **588**  
uil-utility-build-symbol-list-of-values-for-class-attribute **589**  
uil-utility-change-module **591**  
uil-utility-check-for-orphaned-selection-buttons **592**  
uil-utility-cleanup-dialog-bin **593**  
uil-utility-convert-from-edit-box-to-integer **594**  
uil-utility-convert-from-edit-box-to-quantity **595**  
uil-utility-convert-from-edit-box-to-symbol **596**  
uil-utility-convert-from-edit-box-to-value **597**  
uil-utility-create-procedure-xref-listing **598**  
uil-utility-return-current-real-time **599**  
uil-utility-synchronize-text-fields-for-uil-text **600**  
uil-utility-unique-id **601**  
uil-utility-unique-name **602**  
utl-clone-through-copying **603**  
utl-convert-from-text-to-integer **604**  
utl-convert-from-text-to-quantity **605**  
utl-convert-from-text-to-symbol **606**  
utl-convert-from-text-to-value **607**  
utl-convert-list-to-text **608**  
utl-copy-object-attributes **609**

- utl-copy-symbol-array-to-initial-values 610
- utl-copy-symbol-array-to-symbol-array 611
- utl-copy-symbol-array-to-symbol-list 612
- utl-copy-symbol-list-to-symbol-array 613
- utl-copy-symbol-list-to-symbol-list 614
- utl-copy-text-array-to-initial-values 615
- utl-copy-text-list-to-text-list 616
- utl-create-instance-of-class 617
- utl-get-module-information 618
- utl-get-module-owning-this-workspace 619
- utl-get-top-level-workspace-of-this-workspace 620
- utl-goto-superior 621
- utl-goto-workspace 622
- utl-hide-workspace 623
- utl-insert-text-in-alphabetized-list 624
- utl-insert-text-in-reversed-alphabetized-list 625
- utl-instances-exist 626
- utl-lift-to-top 627
- utl-remove-hyphens-from-text 628
- utl-remove-module-from-hierarchy 629
- utl-return-clipped-string 630
- utl-show-superior-workspace 631
- utl-transfer-to-at 632

Additional Utility Procedures 633

## **Appendix A System-Defined Relations 635**

Introduction 635

Relations among UIL Objects 636

Relations among Dialogs and Supporting Objects 638

## **Appendix B Common Object Attributes 643**

**Index 651**



# Preface

---

*Describes this manual and the conventions that it uses.*

About this Manual	<b>xxi</b>
Audience	<b>xxii</b>
Organization	<b>xxii</b>
How to Use the Chapters in this Manual	<b>xxiv</b>
A Note About the API	<b>xxviii</b>
Conventions	<b>xxviii</b>
Related Documentation	<b>xxx</b>
Customer Support Services	<b>xxxii</b>



## About this Manual

This manual describes the G2 GUIDE User Interface Library (GUIDE/UIL). GUIDE/UIL provides an application programmer's interface (API) to procedures that control dialogs and other elements of a graphical user interface.

This manual describes G2 GUIDE/UIL. If you are running an earlier version of G2 GUIDE/UIL, consult the documentation for that version.

Users of this manual also need the *G2 GUIDE User's Guide*. GUIDE is a development tool that enables you to create graphical user interfaces for G2 applications.

# Audience

This manual is written for GUIDE application developers. It addresses application developers as “you”, and refers to end-users of GUIDE applications as “the user” or “users”.

# Organization

This manual contains 23 chapters and two appendixes:

	<b>Title</b>	<b>Description</b>
1	Introduction	Describes some common tasks that you can perform using UIL procedures and describes how the procedure priority for UIL procedures is set.
2	Border Objects	Describes the UIL procedures that perform operations on border objects.
3	Buttons	Describes the UIL procedures that perform operations on buttons.
4	Check Boxes	Describes UIL procedures that operate on check boxes.
5	Check Buttons	Describes UIL procedures that operate on check buttons.
6	Dialogs	Describes the UIL procedures that operate on dialogs.
7	Dialog Titles	Describes the UIL procedures that operate on dialog titles.
8	Edit Boxes, Combo Boxes, and Spin Control Boxes	Describes UIL procedures that operate on edit boxes, combo boxes, and spin control boxes.
9	Graphical Messages	Describes procedures that operate on graphical messages.
10	Graphical Objects	Describes procedures that operate on graphical objects belonging to subclasses of the class <code>uil-grobj</code> .

	<b>Title</b>	<b>Description</b>
11	Message Objects	Describes procedures that operate on message objects belonging to subclasses of the class <code>uil-message-object</code> .
12	Navigation Buttons	Describes procedures that operate on navigation buttons.
13	Push Buttons	Describes procedures that operate on push buttons.
14	Radio Boxes	Describes procedures that operate on radio boxes.
15	Radio Buttons	Describes procedures that operate on radio buttons.
16	Scroll Areas	Describes procedures that operate on scroll areas.
17	Selection Boxes	Describes procedures that operate on selection boxes.
18	Selection Buttons	Describes procedures that operate on selection buttons.
19	Separators	Describes procedures that operate on separators.
20	Text Objects	Describes procedures that operate on text objects.
21	Toggle Buttons	Describes procedures that operate on toggle buttons.
22	Items and Relations	Summarizes UIL procedures for accessing UIL objects in relations, and describes procedures that access information about relations themselves.
23	Upgrade and Utility Procedures	Describes procedures that upgrade elements of your GUIDE user interface.

	<b>Title</b>	<b>Description</b>
A	System-Defined Relations	Describes the relations between UIL objects.
B	Common Object Attributes	Summarizes the attributes that are common to all UIL objects, to all graphical message objects (uil-grmes), and to all UIL objects with values that can be updated or concluded.

## How to Use the Chapters in this Manual

This manual describes the system-defined UIL procedures that perform operations on UIL objects. Each chapter contains the following sections:

- A brief description of UIL objects of a particular UIL class.
- A figure showing the class hierarchy of this class.
- A table listing the attributes common to UIL objects of this class.
- A table listing basic operations that you can perform on UIL objects of this class, and the procedures that perform these operations.
- A table of the system-defined relations between UIL objects of this particular class and other UIL objects.
- Detailed descriptions of the procedures that perform operations on UIL objects of this class.

The following sections describe how to use these sections.

### Description of UIL Objects

This section briefly describes the nature and purpose of UIL objects of the particular UIL class. For a more information about UIL objects of this class, see the *G2 GUIDE User's Guide*.

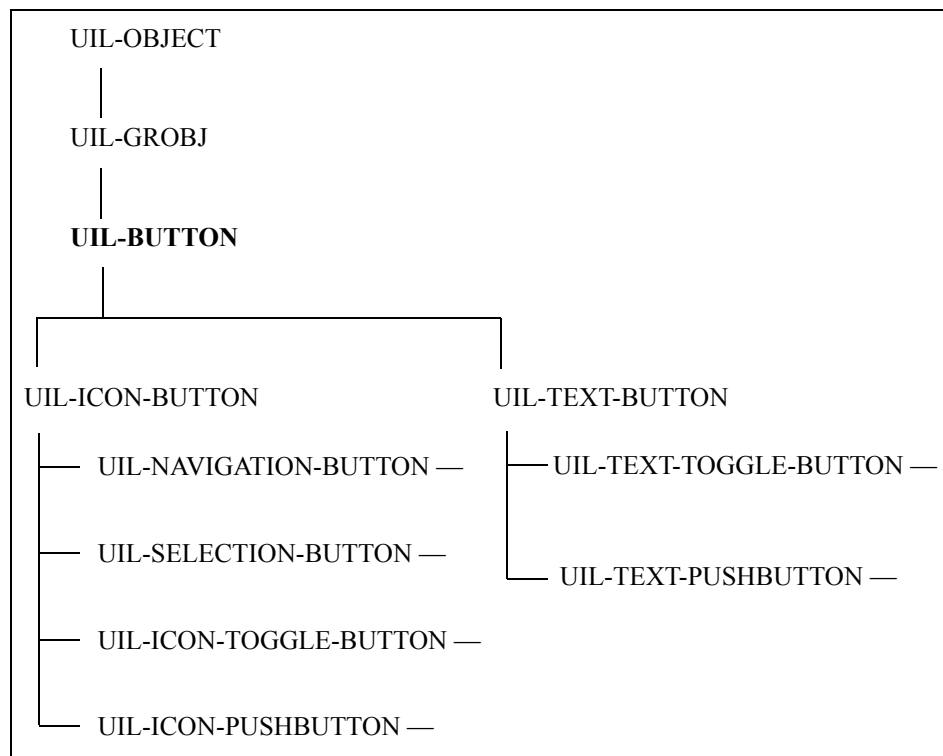
### Class Hierarchy Figure

The class hierarchy figure shows the UIL classes from which the UIL class described in the chapter inherits attributes. It also shows any system-defined subclasses of this UIL class.

You can refer to this figure for class names if you need to manipulate classes of UIL objects programmatically. You can also refer to this figure to find appropriate UIL classes from which to create customized subclasses. For information about how to create customized subclasses of system-defined UIL classes, see the *G2 GUIDE User's Guide*.



For example, the class hierarchy figure for buttons looks like this:



## Attribute Tables

The attributes table lists the attributes that are unique to, or have special meaning for, UIL objects of a particular class. You can access the online table of these attributes by choosing **table** from the menu of the UIL object.

For information about attributes that are common to all UIL classes, see [Appendix B, Common Object Attributes](#).

For example, the first row of the attributes table for border objects looks like this:

Attribute	Description
<b>uil-box-border-margin-left</b>	Specifies width of the left margin of the border. This margin is used when the border is anchored to a workspace.

*Allowable values:* Any integer

*Default value:* 10

## Operations Table

The operations table lists the procedures and methods used to perform common operations on UIL objects of a particular class. Each row in the table describes a particular operation.

For example, the first five lines of the operations table for border objects looks like this:

**Operations on Border Objects**

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
	Change Size	n/a	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a

The left-most column in a row contains a check mark if the procedure that performs the operation is referenced from the UIL object's attribute table. For example, when an UIL object is cloned, the UIL procedure responsible for handling all cloning operations calls the procedure named by the UIL object's `uil-clone-method` attribute. Thus, you see a check mark in the left-most column of the row that describes the cloning operation.

The Operation column lists common operations performed on the UIL object. The procedure listed with the operation in the Procedure column is the system-defined UIL procedure that performs the operation on UIL objects of the given class. Procedures followed by asterisks (\*\*\*) are described in the section immediately following the table. Procedures not followed by an asterisk are described in another chapter.

The Method column lists procedures that are called as methods for some of the common operations. Methods are referenced from the object attributes that correspond to the operations performed by the methods. Each method by default calls the lower-level procedure listed in the Procedure column.

You can create your own methods to use in place of the system-defined methods that appear in the attribute tables by default. The following table illustrates two possible values of the attribute that corresponds to the conclude operation.

<b>Table Attribute for Conclude Operation</b>	<b>Some Possible Values</b>
uil-conclude-method	uil-conclude-grobj-method (the default)
uil-conclude-method	my-conclude-method

In the first example shown in the table above, the attribute `uil-conclude-method` references `uil-conclude-grobj-method`. This is the default conclude method. When `uil-conclude-grobj-method` is run, it calls the lower-level UIL procedure `uil-conclude-grobj`, which performs the conclude operation.

In the second example, the attribute value is `my-conclude-method`. This is an example of a user-defined method. When the conclude operation is performed on the object, the procedure `my-conclude-method` is called, rather than the default conclude method.

---

**Caution** Calling a method from a user-defined method results in an infinite loop. For example, an infinite loop results if `my-conclude-method` calls `uil-conclude-grobj-method`.

However, for most purposes, a user-defined method such as `my-conclude-method` needs to call the lower-level procedure, such as `uil-conclude-grobj`, that performs the basic operation, such as concluding the object.

---

For more information about how methods work, see the *G2 GUIDE User's Guide*.

## Object Relations Table

This section includes a table that lists the system-defined relations between UIL objects of a particular class and other objects.

For example, the following table lists the standard relations between buttons and text objects used as their labels:

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-text	the-uil-text-label-of	item, with text label  (buttons, edit boxes, selection boxes)	one-to-one	the-uil-button-owning-this-text-label

For detailed information about system-defined relations between dialogs and other objects, see [Appendix A, System-Defined Relations](#).

## A Note About the API

The GUIDE/UIL API, as described in this manual, is not expected to change significantly in future releases, but exceptions may occur. A detailed description of any changes will accompany the *product* release that includes them.

Therefore, it is essential that you use GUIDE/UIL exclusively through its API, as described in this manual. If you bypass the API, you cannot rely on your code to work in the future, since GUIDE/UIL may change, or in the present, because the code may not correctly manage the internal operations of GUIDE/UIL.

If GUIDE/UIL does not seem to provide the capabilities that you need, call Gensym Customer Support for more information.

## Conventions

This guide uses the following typographic conventions and conventions for defining system procedures.

### Typographic

Convention Examples	Description
g2-window, g2-window-1, ws-top-level, sys-mod	User-defined and system-defined G2 class names, instance names, workspace names, and module names
history-keeping-spec, temperature	User-defined and system-defined G2 attribute names
true, 1.234, ok, "Burlington, MA"	G2 attribute values and values specified or viewed through dialogs
Main Menu > Start KB Workspace > New Object create subworkspace Start Procedure	G2 menu choices and button labels

Convention Examples	Description
conclude that the x of y ...	Text of G2 procedures, methods, functions, formulas, and expressions
<i>new-argument</i>	User-specified values in syntax descriptions
<u><i>text-string</i></u>	Return values of G2 procedures and methods in syntax descriptions
File Name, OK, Apply, Cancel, General, Edit Scroll Area	GUIDE and native dialog fields, button labels, tabs, and titles
File > Save	GMS and native menu choices
Properties	
<b>workspace</b>	Glossary terms
<i>c:\Program Files\Gensym\</i>	Windows pathnames
<i>/usr/gensym/g2/kbs</i>	UNIX pathnames
<i>spreadsh.kb</i>	File names
<i>g2 -kb top.kb</i>	Operating system commands
<i>public void main() gsi_start</i>	Java, C and all other external code

---

**Note** Syntax conventions are fully described in the *G2 Reference Manual*.

---

## Procedure Signatures

A procedure signature is a complete syntactic summary of a procedure or method. A procedure signature shows values supplied by the user in *italics*, and the value (if any) returned by the procedure *underlined*. Each value is followed by its type:

```
g2-clone-and-transfer-objects
(list: class item-list, to-workspace: class kb-workspace,
 delta-x: integer, delta-y: integer)
-> transferred-items: g2-list
```

# Related Documentation

## **G2 Core Technology**

- *G2 Bundle Release Notes*
- *Getting Started with G2 Tutorials*
- *G2 Reference Manual*
- *G2 Language Reference Card*
- *G2 Developer's Guide*
- *G2 System Procedures Reference Manual*
- *G2 System Procedures Reference Card*
- *G2 Class Reference Manual*
- *Telewindows User's Guide*
- *G2 Gateway Bridge Developer's Guide*

## **G2 Utilities**

- *G2 ProTools User's Guide*
- *G2 Foundation Resources User's Guide*
- *G2 Menu System User's Guide*
- *G2 XL Spreadsheet User's Guide*
- *G2 Dynamic Displays User's Guide*
- *G2 Developer's Interface User's Guide*
- *G2 OnLine Documentation Developer's Guide*
- *G2 OnLine Documentation User's Guide*
- *G2 GUIDE User's Guide*
- *G2 GUIDE/UIIL Procedures Reference Manual*

## **G2 Developers' Utilities**

- *Business Process Management System Users' Guide*
- *Business Rules Management System User's Guide*
- *G2 Reporting Engine User's Guide*
- *G2 Web User's Guide*
- *G2 Event and Data Processing User's Guide*

- *G2 Run-Time Library User's Guide*
- *G2 Event Manager User's Guide*
- *G2 Dialog Utility User's Guide*
- *G2 Data Source Manager User's Guide*
- *G2 Data Point Manager User's Guide*
- *G2 Engineering Unit Conversion User's Guide*
- *G2 Error Handling Foundation User's Guide*
- *G2 Relation Browser User's Guide*

### **Bridges and External Systems**

- *G2 ActiveXLink User's Guide*
- *G2 CORBALink User's Guide*
- *G2 Database Bridge User's Guide*
- *G2-ODBC Bridge Release Notes*
- *G2-Oracle Bridge Release Notes*
- *G2-Sybase Bridge Release Notes*
- *G2 JMail Bridge User's Guide*
- *G2 Java Socket Manager User's Guide*
- *G2 JMSLink User's Guide*
- *G2 OPCLink User's Guide*
- *G2 PI Bridge User's Guide*
- *G2-SNMP Bridge User's Guide*
- *G2 CORBALink User's Guide*
- *G2 WebLink User's Guide*

### **G2 JavaLink**

- *G2 JavaLink User's Guide*
- *G2 DownloadInterfaces User's Guide*
- *G2 Bean Builder User's Guide*

## G2 Diagnostic Assistant

- *GDA User's Guide*
- *GDA Reference Manual*
- *GDA API Reference*

# Customer Support Services

You can obtain help with this or any Gensym product from Gensym Customer Support. Help is available online, by telephone, by fax, and by email.

### To obtain customer support online:

➔ Access G2 HelpLink at [www.gensym-support.com](http://www.gensym-support.com).

You will be asked to log in to an existing account or create a new account if necessary. G2 HelpLink allows you to:

- Register your question with Customer Support by creating an Issue.
- Query, link to, and review existing issues.
- Share issues with other users in your group.
- Query for Bugs, Suggestions, and Resolutions.

### To obtain customer support by telephone, fax, or email:

➔ Use the following numbers and addresses:

	<b>Americas</b>	<b>Europe, Middle-East, Africa (EMEA)</b>
<b>Phone</b>	(781) 265-7301	+31-71-5682622
<b>Fax</b>	(781) 265-7255	+31-71-5682621
<b>Email</b>	<a href="mailto:service@gensym.com">service@gensym.com</a>	<a href="mailto:service-ema@gensym.com">service-ema@gensym.com</a>



# Introduction

---

*Describes some common tasks that you can perform using UIL procedures and describes how the procedure priority for UIL procedures is set.*

Introduction 1

Common Tasks Performed with UIL Procedures 1

Procedure Priority Setting 6



## Introduction

The G2 GUIDE User Interface Library (GUIDE/UIL) provides programmatic support for G2 GUIDE, a development tool that enables you to create graphical user interfaces for G2 applications.

GUIDE/UIL provides an application programmer's interface (API) to procedures that perform basic operations on the dialogs and UIL controls that you use with GUIDE to create a user interface. For detailed information about GUIDE, see the *G2 GUIDE User's Guide*.

## Common Tasks Performed with UIL Procedures

The following table lists:

- Common tasks that you can perform using UIL procedures.
- The UIL procedure or procedures that you can use to perform each task.
- Ways to execute the UIL procedures — for example, through callbacks, update and conclude actions, or selection and unselection methods.

The following table does not describe *all* the tasks that UIL procedures can perform. It only describes some of the tasks that you can perform using some of the more commonly used UIL procedures.

### UIL Procedures for Common Tasks

<b>To perform this task...</b>	<b>Use these UIL Procedures...</b>	<b>And invoke the UIL procedures from a user-defined...</b>
Launch a dialog	uil-control-dialog-callback  You can invoke uil-control-dialog-callback directly, or through the following UIL procedures:  uil-start-dialog, uil-start-or-refocus-dialog, uil-start-dialog-processing	action button, procedure, or user-menu choice
Launch a message dialog, confirmation dialog, query dialog, or notification dialog	uil-post-generic-dialog	action button or procedure
Launch, update, and remove Delay Notification Icon	uil-post-delay-notification  uil-update-wait-clock-face, uil-update-wait-semaphore  uil-remove-delay-notification-if-any	procedure-invocation
Manage, unmanage, and release dialogs	uil-show-managed-dialog, uil-hide-managed-dialog, uil-release-dialog	action or procedure

### UIL Procedures for Common Tasks

<b>To perform this task...</b>	<b>Use these UIL Procedures...</b>	<b>And invoke the UIL procedures from a user-defined...</b>
Update, conclude, or validate dialog programmatically	uil-update-dialog-for-initiating-object  uil-call-conclude-method-for-grobj, uil-call-update-method-for-grobj, uil-call-validate-method-for-grobj	action, procedure, or rule
Remove events from or add events to dialog event queue	uil-flush-dialog-event-queue, uil-remove-all-remaining-actions, uil-insert-actions-into-dialog-event-queue	action, callback, or procedure
Find or set the source object of a dialog	uil-find-initiating-object, uil-find-source-object, uil-override-initiating-object-for-dialog	update action or procedure
Find or set the target object	uil-set-destination-object-for-dialog, uil-find-destination-object, uil-find-initiating-object, uil-find-target-object	conclude action or procedure
Locate an item on a dialog or workspace to perform some action upon	uil-get-grobj-from-id-on-dlg-or-wksp	conclude action, update action, or callback
Enable, disable, or configure items on a dialog	uil-enable-grobj-method, uil-disable-grobj-method, uil-configure-grobj-method	update action or callback
Set value of a text object, edit box, or message object	uil-update-text  Example: call uil-update-text(T, "the new text");	conclude action, update action, procedure, or callback
Get value of a text object, edit box, or message object	Use the following grammar:  T: class uil-grmes; txt: text:  txt = the message-contents of T;	conclude action or callback

### UIL Procedures for Common Tasks

To perform this task...	Use these UIL Procedures...	And invoke the UIL procedures from a user-defined...
Set and get values of a check box	uil-set-check-box-values, uil-get-check-box-values	update action (set values), conclude action (get values), or callback (set or get values)
Set and get values of a radio box	uil-set-radio-box-value, uil-get-radio-box-value	update action (set values), conclude action (get values), or callback (set or get values)
Select and unselect buttons (any uil-button object)	uil-select-button, uil-unselect-button	update or conclude action, or callback
Select and unselect buttons (any except push buttons)	uil-toggle-button	update or conclude action, or callback
Start and end edit session on edit box	uil-select-edit-box, uil-unselect-edit-box  <b>Note:</b> A user-defined select or unselect method needs to call these UIL procedures to perform the default select or unselect operations on the edit box.	update or conclude action, or callback
Get index of a message object	uil-get-index-of-scroll-message	update or conclude action, or callback
Get message object with specified index	uil-get-message-from-index	update or conclude action, or callback
Update a scroll area to display messages starting at a specific index	uil-display-message-list	callback, conclude action, update action, or procedure
Select and unselect message objects	uil-select-message uil-message-selection, uil-message-unselection	update or conclude action, or callback

### UIL Procedures for Common Tasks

To perform this task...	Use these UIL Procedures...	And invoke the UIL procedures from a user-defined...
Sort message objects in scroll area	uil-sort-scroll-area-messages	update or conclude action, or callback
Create, add, and delete message objects	uil-create-message-object, uil-delete-selected-messages, uil-delete-grobject-method	update or conclude action, or callback
Add messages to a scroll area	uil-update-scroll-area-from-array, uil-update-scroll-area-from-list, uil-create-message	update action or callback
Access message objects in a scroll area	<p>To access <i>all</i> message objects managed by the scroll area, use this grammar:</p> <pre>MO: class uil-message-object; SA: class uil-scroll-area; for MO = each uil-message-object that is a-message-belonging-to SA do {Some user-defined activity. This loop iterates over all message objects.} end;</pre> <p>To access all <i>selected</i> message objects managed by the scroll area, use this grammar:</p> <pre>for MO = each uil-message-object that is a-selected-message-of SA do {Some user defined activity. This loop iterates over all selected message objects.} end;</pre>	conclude action, update action, procedure, or callback
Enable and disable graphical objects (any uil-grobject object)	uil-enable-grobject-method, uil-disable-grobject-method	conclude action, update action, or callback

## Procedure Priority Setting

Procedures started by UIL (using the G2 **start** statement) are started at the priority indicated by the UIL global variable named `uil-global-default-procedure-priority`. Wherever UIL or GUIDE uses the **start** action, the **start** action is followed by the extra syntax: "... at priority `uil-global-default-procedure-priority`".

When UIL procedures are called (using the G2 **call** statement) by user applications, they are called according to the priority of the calling procedure in the user's application.

# Border Objects

---

*Describes the UIL procedures that perform operations on border objects.*

Introduction 7

Procedures that Operate on Border Objects 13

    uil-adjust-border-on-workspace 14

    uil-anchor-box-corner-to-workspace-edge 15

    uil-create-border-for-text 16

    uil-create-box-border 17

    uil-get-border 18

    uil-move-related-border 19

    uil-release-box-corner-from-workspace-edge 20

Additional Procedures that Support Border Objects 21



## Introduction

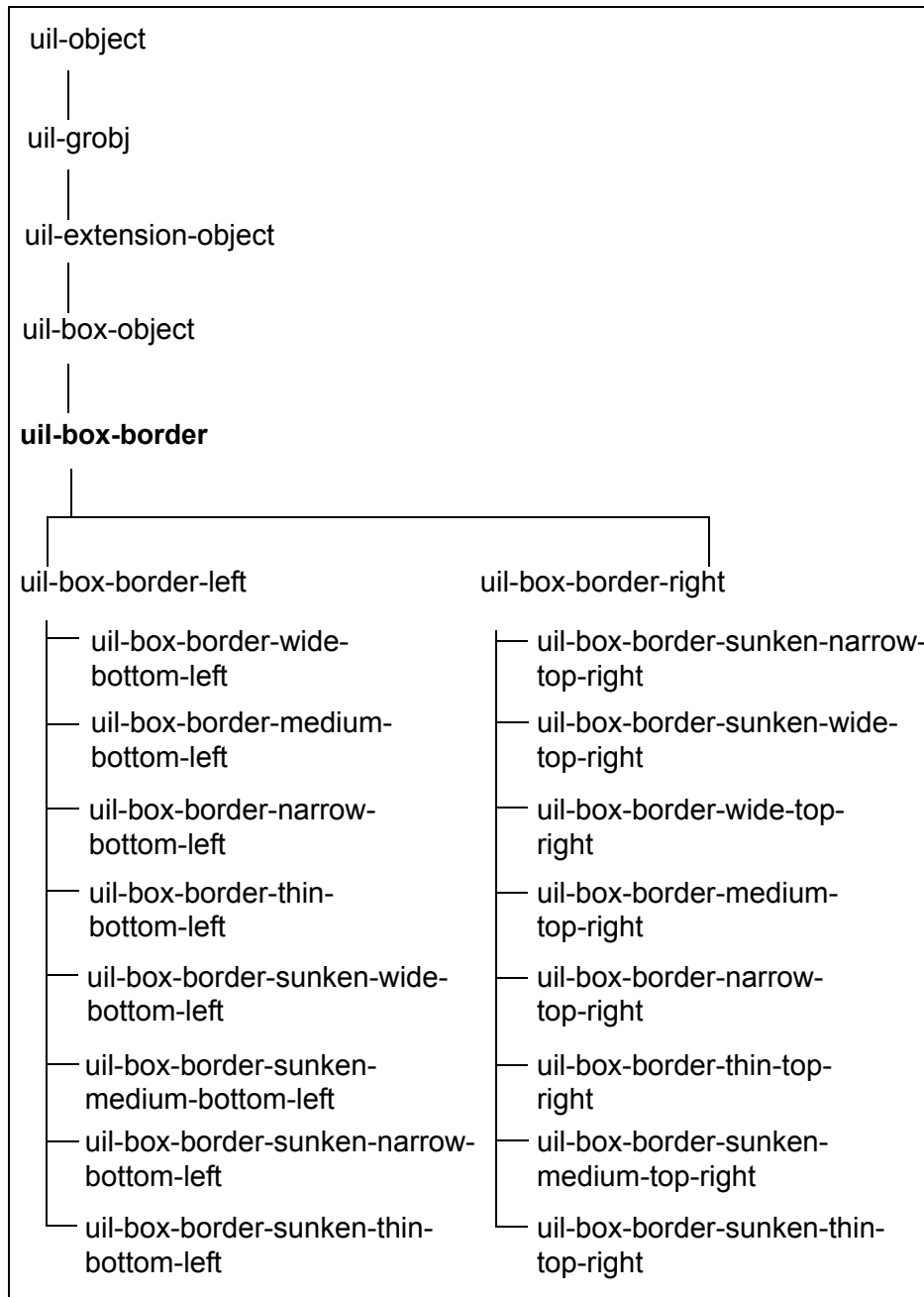
All border objects inherit from the class `uil-box-object`.

A border can stand alone, be attached to the edge of a workspace, or enclose a text object or edit box.

## Border Object Class Hierarchy

The following figure illustrates the UIL class hierarchy for border objects. All classes shown are object definitions.

Border Object Class Hierarchy





## Attributes of Border Objects

The following table lists attributes that are unique to border objects or have special meaning for them. See [Appendix B, Common Object Attributes](#), for information about the attributes that border objects have in common with other UIL objects.

### Attributes of Border Objects

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this border object.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-line-and-border-configuration
<b>uil-box-border-margin-left</b>	Specifies width of the left margin of the border. This margin is used when the border is anchored to a workspace.  <i>Allowable values:</i> Any integer  <i>Default value:</i> 10
<b>uil-box-border-margin-right</b>	Specifies width of the right margin of the border. This margin is used when the border is anchored to a workspace.  <i>Allowable values:</i> Any integer  <i>Default value:</i> 10
<b>uil-box-border-margin-top</b>	Specifies width of the top margin of the border. This margin is used when the border is anchored to a workspace.  <i>Allowable values:</i> Any integer  <i>Default value:</i> 10

## Attributes of Border Objects

Attribute	Description
<b>uil-box-border-margin-bottom</b>	Specifies a bottom margin for the border. This margin is used when the border is anchored to a workspace.  <i>Allowable values:</i> Any integer  <i>Default value:</i> 10
<b>uil-corner-anchored-to-workspace</b>	Specifies whether border is attached to its workspace, with specified margins.  <i>Allowable values:</i> <b>true</b> (attach border) or <b>false</b> (do not attach border)  <i>Default value:</i> <b>false</b>

## Operations Performed on Border Objects

The following table lists the common operations that can be performed on border objects, and the procedures and methods used to perform these operations.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations on Border Objects

	Operation	Procedure	Method
	Change Size	n/a	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method

### Operations on Border Objects

	Operation	Procedure	Method
	Create	uil-create-box-border ***	n/a
✓	Delete	uil-delete-grobject	uil-delete-grobject-method
✓	Disable	n/a	n/a
✓	Enable	n/a	n/a
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobject	uil-initialize-grobject-method
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	n/a	n/a
	Update Wrapper	n/a	n/a

### Operations on Border Objects

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Procedures that Operate on Border Objects

This section describes procedures that you can use to manipulate uil-box-object objects.

- uil-adjust-border-on-workspace
- uil-anchor-box-corner-to-workspace-edge
- uil-create-border-for-text
- uil-create-box-border
- uil-get-border
- uil-move-related-border
- uil-release-box-corner-from-workspace-edge

# uil-adjust-border-on-workspace

Adjusts the border on a workspace to surround all items on the workspace.

## Synopsis

uil-adjust-border-on-workspace  
(*workspace*: class kb-workspace)

Argument	Description
<i>workspace</i>	The workspace on which the border appears.

## Description

The adjustment takes into account the top, bottom, left and right margins specified on the border object. For information about how to set border margins, see the description of the Edit Border Margins Dialog in the *G2 GUIDE User's Guide*.

## uil-anchor-box-corner-to-workspace-edge

Attaches the border to the edges of the workspace on which it currently appears.

### Synopsis

uil-anchor-box-corner-to-workspace-edge  
(*box-object*: class uil-box-object)

Argument	Description
<i>box-object</i>	The lower left corner of the border to adjust.

# uil-create-border-for-text

Creates a border for specified text object of the given type.

## Synopsis

uil-create-border-for-text

(*gobj*: class uil-grmes, *type*: symbol, *style*: item-or-value)

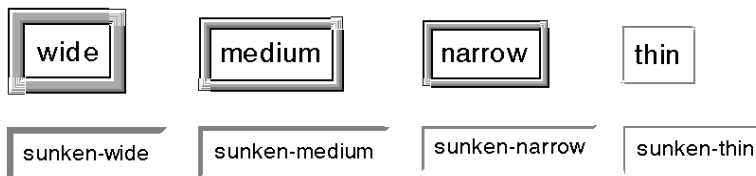
-> {*border-object*: class uil-box-border | no-border-created: symbol}

Argument	Description
<i>gobj</i>	The object for which a border is created. This object must be a graphical message (object of class uil-text or uil-edit-box).
<i>type</i>	The style of border created. Specify the symbol followed by: wide, medium, narrow, thin, sunken-wide, sunken-medium, sunken-narrow, or sunken-thin.
<i>style</i>	Specify the symbol none. (This is a place-holder for a style object).

Return Value	Description
<u><i>border-object</i></u>	The new border object. If none is created, the symbol no-border-created is returned instead.

## Description

The following figure illustrates the border types that you can specify for *type*:





# uil-create-box-border

Creates a box border of the specified type.

## Synopsis

uil-create-box-border (*type*: symbol)  
 -> *box-border*: class uil-box-object

Argument	Description
<i>type</i>	The style of border created. Specify the symbol followed by: wide, medium, narrow, thin, sunken-wide, sunken-medium, sunken-narrow, or sunken-thin.
Return Value	Description
<u><i>box-border</i></u>	The new box border.

# uil-get-border

Returns the border of a specified object.

## Synopsis

uil-get-border (*object*: item)

-> {*border*: class uil-box-border | not-found: symbol}

Argument	Description
<i>object</i>	The object whose border is to be returned.

Return Value	Description
<u><i>border</i></u>	The border of the specified object. If the object has no border, the symbol not-found is returned instead.

## uil-move-related-border

Finds the border for this graphical message object and moves it so that it tightly wraps around its text.

### Synopsis

uil-move-related-border  
(*txt*: class uil-grmes)

Argument	Description
<i>txt</i>	The graphical message being moved.

# uil-release-box-corner-from-workspace-edge

Releases the border from the edges of the workspace. The border no longer snaps to the edge of the workspace as the workspace contents change.

## Synopsis

uil-release-box-corner-from-workspace-edge

(*box-obj*: class uil-box-object)

Argument	Description
<i>box-obj</i>	The lower left corner of the managed box border.

## Description

The uil-release-box-corner-from-workspace-edge procedure releases the border from the edges of the workspace. The border longer snaps to the edge of the workspace as the workspace contents change.

## Additional Procedures that Support Border Objects

The following list provides additional procedures that support borders:

[uil-create-dialog-border](#)  
[uti-copy-object-attributes](#)  
[uti-lift-to-top](#)



# Buttons

---

*Describes the UIL procedures that perform operations on buttons.*

Introduction **24**

Procedures that Operate on Buttons **26**

- uil-add-label-to-button **27**
- uil-center-label-in-text-button **28**
- uil-create-custom-button **29**
- uil-create-text-for-button-from-label **30**
- uil-handle-button **31**
- uil-handle-button-method **32**
- uil-highlight-button **33**
- uil-reset-button **34**
- uil-return-size-of-button-as-symbol **35**
- uil-select-button **36**
- uil-toggle-button **37**
- uil-update-button-label **38**
- uil-update-button-togglelabel **39**

Additional Procedures that Support Buttons **40**

Superseded Procedures **41**

- uil-disable-button **42**
- uil-enable-button **43**



# Introduction

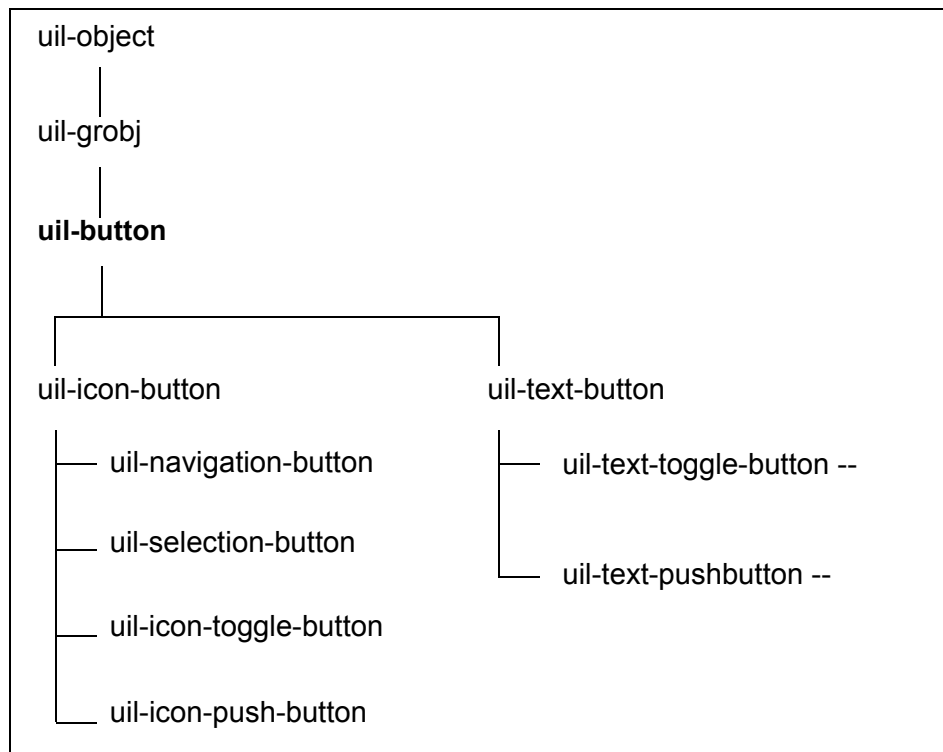
This chapter describes the UIL procedures that can perform operations on buttons of all classes: `uil-check-button`, `uil-icon-push button`, `uil-icon-toggle-button`, `uil-navigation-button`, `uil-radio-button`, `uil-text-pushbutton`, and `uil-text-toggle-button`.

All buttons inherit from the class `uil-button`.

## Button Class Hierarchy

The following figure illustrates the UIL class hierarchy for buttons. All classes shown are object definitions. A class whose name is followed by dashes (for example, `uil-text-pushbutton --`) has subclasses that are not shown in the diagram.

**Button Class Hierarchy**





## Button Relations

The following table lists the standard relations between buttons and other objects.

### Relations Between Buttons and Other Objects

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label
<p><b>Note:</b> The the-uil-text-label-of relation applies only to buttons whose labels are contained in separate text objects.</p> <p>Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UII earlier than Version 5.0 and were not converted to have their label text in their icons.</p>				

## Procedures that Operate on Buttons

This section describes procedures that can operate on all classes of buttons. Buttons are objects of classes that inherit from the class `uil-button`.

- `uil-add-label-to-button`
- `uil-center-label-in-text-button`
- `uil-create-custom-button`
- `uil-create-text-for-button-from-label`
- `uil-handle-button`
- `uil-handle-button-method`
- `uil-highlight-button`
- `uil-reset-button`
- `uil-return-size-of-button-as-symbol`
- `uil-select-button`
- `uil-toggle-button`
- `uil-update-button-label`
- `uil-update-button-togglelabel`

---

**Note** For descriptions of UIL procedures that enable you to convert buttons in your GUIDE applications when you upgrade to newer versions of GUIDE/UIL, see [Upgrade and Utility Procedures](#).

---

## uil-add-label-to-button

Adds a label with the specified size and text to the specified button.

### Synopsis

uil-add-label-to-button

(*button*: class uil-button, *text*: text, *size*: symbol)

Argument	Description
<i>button</i>	The button to which the label is added.
<i>text</i>	The text of the label to add to button.
<i>size</i>	The size of the label. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .

# uil-center-label-in-text-button

Centers label text in a text button.

## Synopsis

uil-center-label-in-text-button  
(*button*: class uil-text-button)

Argument	Description
<i>button</i>	The text button whose label is centered.

## uil-create-custom-button

Creates an instance of a button of the specified class with the specified label.

### Synopsis

uil-create-custom-button (*class-name*: symbol, *label*: text)  
-> {*button*: class uil-button | not-found: symbol}

Argument	Description
<i>class-name</i>	The class of the button that is created.
<i>label</i>	The text of the button's label.

Return Value	Description
<u><i>button</i></u>	The new button. If none is created, the symbol not-found is returned instead.

# uil-create-text-for-button-from-label

Creates a text object containing the text of the button's label.

## Synopsis

uil-create-text-for-button-from-label  
(*button*: class item, *size*: symbol)

Argument	Description
<i>button</i>	The button to create the text object for.
<i>size</i>	The size of the text. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .

## Description

The `uil-create-text-for-button-from-label` procedure creates a text object (instance of class `uil-text`) containing the text of the button's label based on the contents of the button's `label` attribute.

## uil-handle-button

Calls the callback procedure referenced by the button's `callback` attribute when a user selects or unselects the button. Sets the state of the button and configures it when it is selected or unselected.

### Synopsis

uil-handle-button

(*button*: class uil-button, *window*: class item, *workspace*: class item)

Argument	Description
<i>button</i>	The button to be handled by a select or unselect action.
<i>window</i>	The window on which the workspace containing the button appears.
<i>workspace</i>	The workspace on which the button appears.

## uil-handle-button-method

Calls the procedure named by the `uil-handler-method` attribute of the button whenever the button is selected.

### Synopsis

`uil-handle-button-method`

(*button*: class uil-button, *window*: class item, *workspace*: class item)

Argument	Description
<i>button</i>	The button to be handled by a select or unselect action.
<i>window</i>	The window on which the workspace containing the button appears.
<i>workspace</i>	The workspace on which the button appears.



## uil-highlight-button

Sets the colors of the specified button to reflect its current state. This procedure is called from uil-select-button.

### Synopsis

uil-highlight-button  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The button to highlight.

# uil-reset-button

Resets the button to its specified configuration, based on its current state.

## Synopsis

uil-reset-button

(*button*: class uil-button)

Argument	Description
<i>button</i>	The button to be reset.

# uil-return-size-of-button-as-symbol

Returns the size of the specified button as a symbol.

## Synopsis

uil-return-size-of-button-as-symbol

(*button*: class item)

-> *size*: symbol

Argument	Description
<i>button</i>	The button whose size is to be returned.

Return Value	Description
<u><i>size</i></u>	The value of <u><i>size</i></u> is either small, medium, or large.

## uil-select-button

Handles selection of a button. Highlights the button, passes actions to the target dialog, if one exists, and calls the handler method (which executes the callback if one is specified). Use this procedure to select a button programmatically.

### Synopsis

uil-select-button

(*button*: class uil-button, *window*: class item,  
*dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that is selected.
<i>window</i>	The window on which the dialog, or workspace, containing the button appears.
<i>dialog-or-workspace</i>	The dialog or workspace containing the button.

## uil-toggle-button

Toggles the specified radio button, check button, or toggle button. This procedure cannot be used with push buttons, which do not have a state that can be toggled.

### Synopsis

uil-toggle-button

(*button*: class uil-button, *window*: class item, *workspace*: class item)

Argument	Description
<i>button</i>	The button to be toggled.
<i>window</i>	The window on which the workspace containing the button is managed.
<i>workspace</i>	The workspace on which the button appears.

## uil-update-button-label

Updates the text value of the button's label with the value specified in the button's `label` attribute.

### Synopsis

uil-update-button-label  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The button whose label is updated.

## uil-update-button-togglelabel

Updates the text value of the button's label with the value specified in the button's togglelabel attribute.

### Synopsis

uil-update-button-togglelabel  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The button whose label is updated.

## Additional Procedures that Support Buttons

[utl-copy-object-attributes](#)

[utl-lift-to-top](#)

Also, see the chapters in this manual that describe procedures that operate on particular classes of buttons, such as push buttons, toggle buttons, check buttons, and radio buttons.



## Superseded Procedures

The following procedures are supported by this version of GUIDE/UIL, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

- uil-disable-button
- uil-enable-button

## uil-disable-button

Disables the specified button. Use instead: uil-disable-grobject-method.

### Synopsis

uil-disable-button  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The button that is disabled.

## uil-enable-button

Enables a button. Use instead: uil-enable-grobject-method

### Synopsis

uil-enable-button  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The button that is enabled.



# Check Boxes

---

*Describes UIL procedures that operate on check boxes.*

Introduction **45**

Procedures that Operate on Check Boxes **51**

    uil-create-check-box **52**

    uil-get-check-box-values **53**

    uil-set-check-box-values **54**

Additional Procedures that Support Check Boxes **55**

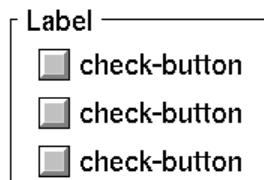
Superseded Procedures **56**

    uil-delete-check-box **57**



## Introduction

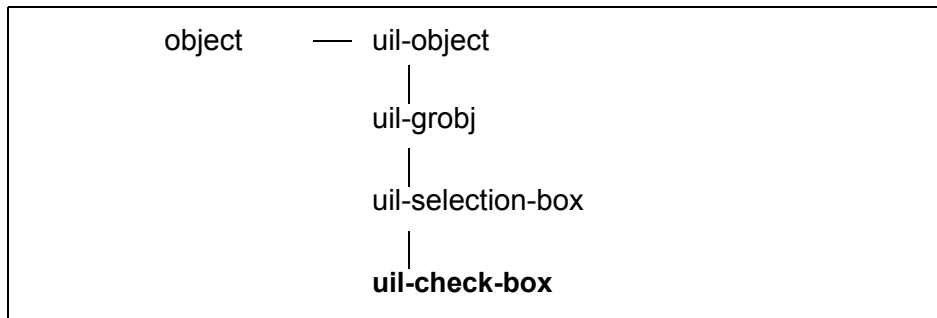
This chapter describes UIL procedures that operate on check boxes. A check box manages a collection of check buttons. Check boxes have borders, which are visible by default. You can edit a check box to specify a label for it. The following figure illustrates a check box with a label.



## Check Box Class Hierarchy

The following figure illustrates the UIL class hierarchy for check boxes. All classes shown are object definitions.

Check Box Class Hierarchy



## Attributes of Check Boxes

The following table lists attributes that are unique to or have special meaning for check boxes. See [Common Object Attributes](#) for information about the attributes that check boxes have in common with other UIL objects.

Attributes of Check Boxes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this check box.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-check-box-configuration
<b>members</b>	Name of an array containing the names of the check buttons managed by this check box.  <i>Allowable values:</i> Any valid check button name.  <i>Default value:</i> a text-array

### Attributes of Check Boxes

Attribute	Description
<b>members-index</b>	The number of buttons in the check box.  <i>Allowable values:</i> Integer > 0 <i>Default value:</i> 3
<b>uil-update-value</b>	Indicates whether or not the update method should be run on the check box.  <i>Allowable values:</i> <i>Default values:</i> <b>true</b> (run update method) or <b>false</b> (do not run update method)
<b>state</b>	Indicates whether check box is enabled or disabled.  <i>Allowable values:</i> <b>enabled, disabled</b> <i>Default values:</i> <b>enabled</b>
<b>label</b>	Specifies the text of the label that appears with the check box. You can specify this text using the Edit Dialog dialog. For information about how to use this dialog, see the <i>G2 GUIDE User's Guide</i> .  <i>Allowable values:</i> Any valid text. <i>Default value:</i> ""

## Operations Performed on Check Boxes

The following table lists the common operations that application developers can perform on check boxes.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations on Check Box Objects

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
✓	Change Style	uil-change-window-style-of-grobj	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-check-box ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	n/a	n/a
	Get State	uil-get-check-box-values ***	n/a
	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method



### Operations on Check Box Objects

	Operation	Procedure	Method
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-set-check-box-values ***	n/a
	Set State	n/a	n/a
✓	Size Method (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	uil-validate-grobject-method

## Check Box and Check Button Relations

Application developers can access sibling uil-check-buttons through the member-list of the check-box that is the-group-master-of the buttons.

### Relations Between Check Boxes and Other Objects

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-check-box	the-group-master-of	uil-check-button	one-to-many	grouped-by
uil-check-button	grouped-by	uil-check-box	many-to-one	the-group-master-of
uil-selection-box	the-group-master-of	uil-selection-button	one-to-many	grouped-by
uil-selection-button	grouped-by	uil-selection-box	many-to-one	the-group-master-of
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UII earlier than Version 5.0 and were not converted to have their label text in their icons

## Procedures that Operate on Check Boxes

This section describes procedures that you can use to manipulate check boxes. Check boxes are objects of the class `uil-check-box`.

- `uil-create-check-box`
- `uil-get-check-box-values`
- `uil-set-check-box-values`

## uil-create-check-box

Creates and returns a check box with the specified ID.

### Synopsis

uil-create-check-box (*id*: text)  
-> *check-box*: class uil-grobj

Argument	Description
<i>id</i>	The ID of the new check box.

Return Value	Description
<u><i>check-box</i></u>	The new check box.

## uil-get-check-box-values

Returns an array containing the values of selected check buttons.

### Synopsis

```
uil-get-check-box-values
(dialog-or-workspace: class item, id: text)
-> text-array: text-array
```

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace that contains the check box.
<i>id</i>	The ID of the check box whose buttons' values are returned.
Return Value	Description
<u>text-array</u>	A text array containing the on-values of selected buttons.

### Description

`uil-get-check-box-values` creates and returns a text array containing the on-value(s) of selected check buttons. The calling application is responsible for deleting the text array that is created.

## uil-set-check-box-values

Finds check box with a specified ID. Turns on all the check buttons in that check box whose on values are listed in a specified text array.

### Synopsis

uil-set-check-box-values

(*dialog-or-workspace*: class item, *id*: text, *values*: class text-array)

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace on which the check box is managed.
<i>id</i>	The ID of the check box whose values are to be set.
<i>values</i>	An array containing on-values for buttons to be toggled on in the check box.

### Description

The `uil-set-check-box-values` procedure iterates over all check boxes associated with check buttons on the given dialog or workspace, looking for one whose ID matches the given ID. If it finds a check box with the specified *id*, then all check buttons whose on-values are listed in the *values* text array are toggled on.

# Additional Procedures that Support Check Boxes

[utl-copy-object-attributes](#)

[utl-lift-to-top](#)

[uil-utility-check-for-empty-selection-box](#)

## Superseded Procedures

The following procedure is supported by this version of GUIDE/UII, but does not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

uil-delete-check-box



## uil-delete-check-box

Deletes a specified check box. Use instead: uil-delete-grobject-method.

### Synopsis

uil-delete-check-box  
(*check-box*: class uil-check-box)

Argument	Description
<i>check-box</i>	The check-box to delete.

### Description

The uil-delete-check-box procedure is supported but does not take advantage of the functionality introduced in GUIDE 4.0.



# Check Buttons

---

*Describes UIL procedures that operate on check buttons.*

Introduction **59**

Procedures that Operate on Check Buttons **65**

    uil-create-styled-check-button **66**

    uil-create-styled-check-button-with-different-sized-label **67**

Additional Procedures that Support Check Buttons **68**

Superseded Procedures **69**

    uil-create-check-button **70**

    uil-delete-check-button **71**



## Introduction

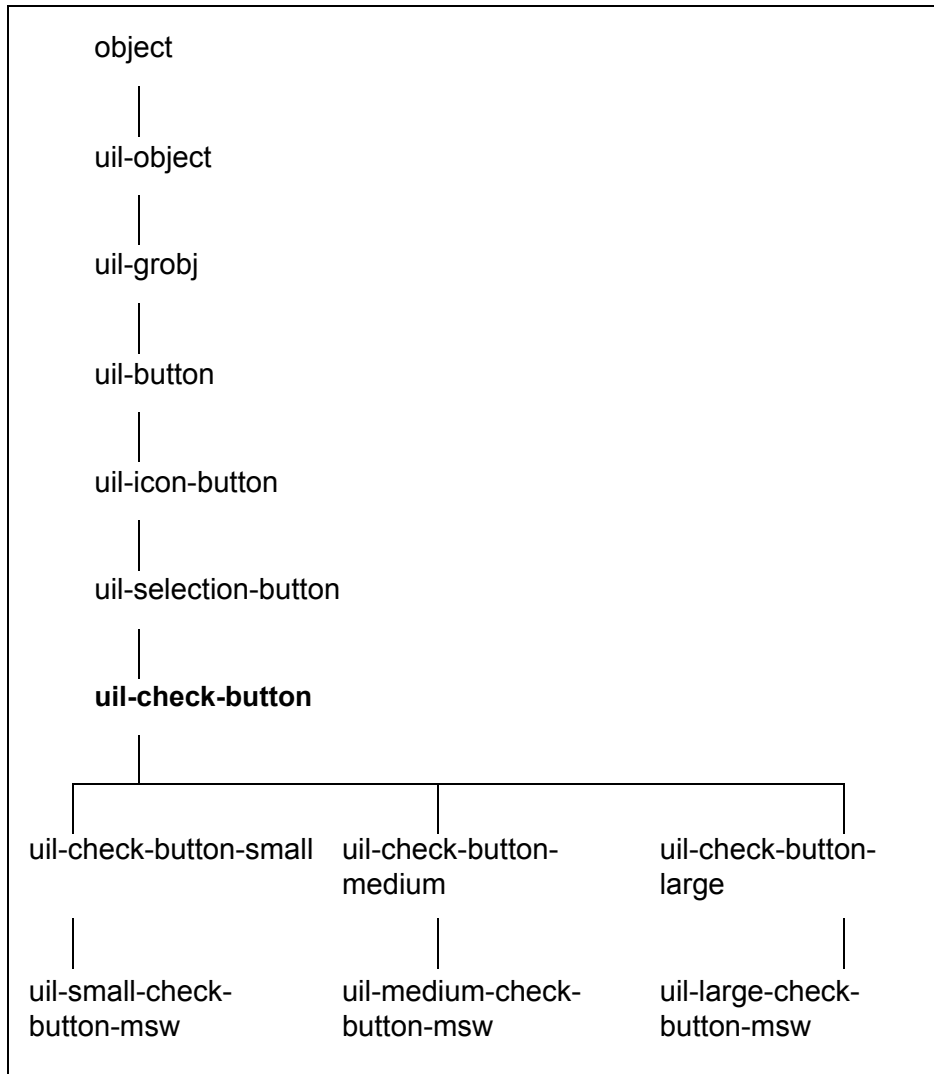
This chapter describes UIL procedures that operate on check buttons. All check buttons inherit from the class `uil-check-button`.

A check button has an on or off value that reflects its state at any given time. Any number of check buttons in a check box can be on or off, because their values are not mutually exclusive.

## Check Button Class Hierarchy

The following figure illustrates the UIL class hierarchy for check buttons. All classes shown are object definitions.

Check Button Class Hierarchy



## Attributes of Check Buttons

The following table lists attributes that are unique to or have special meaning for check buttons. See [Common Object Attributes](#) for information about the attributes that check buttons have in common with objects of other UIL classes.

## Attributes of Check Buttons

Attribute	Description
<b>configuration</b>	<p>If you want to display this check button in a non-standard color, you can specify a customized configuration here to provide that color.</p> <p><i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Default value:</i> The configuration of the check box that owns this check button.</p>
<b>uil-label-constant</b>	<p>Stores a symbol used as a look-up constant that GUIDE/UII uses to provide different natural language versions of text in text objects (<b>uil-text</b>) and buttons (<b>uil-button</b>). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> Any symbol.</p> <p><i>Default value:</i> unspecified</p>
<b>uil-translation-constraints</b>	<p>Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.</p> <p><i>Allowable values:</i> left, right, or center</p> <p><i>Default value:</i> left</p>
<b>label</b>	<p>The text that is used as the label for this check button.</p> <p><i>Allowable values:</i> Any text value</p> <p><i>Default values:</i> "check button"</p>

## Attributes of Check Buttons

Attribute	Description
<b>on-value</b>	The value of the object when the button is selected.
<i>Allowable values:</i>	text or symbol
<i>Default values:</i>	check-button
<b>off-value</b>	The value of the button when the button is not selected.
<i>Allowable values:</i>	text or symbol
<i>Default values:</i>	check-button
<b>toggle-state</b>	Either on or off where on means the button is selected and off means the button is not selected.
<i>Allowable values:</i>	on or off
<i>Default values:</i>	off
<b>callback</b>	Names a procedure that is called when the button is selected by the button's handler method.
<i>Allowable values:</i>	A callback procedure. For information about callback procedures, see the <i>G2 GUIDE User's Guide</i> .
<i>Default values:</i>	uil-do-nothing

## Operations Performed on Check Buttons

The following table lists the common operations that application developers can perform on check buttons.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---

### Operations on Check Button Objects

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	uil-change-window-style-of-grobj	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-check-button ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-button-as-symbol	n/a
	Get State	n/a	n/a
✓	Handler	uil-handle-button	uil-handle-button-method
	Highlight	uil-highlight-button	n/a

## Operations on Check Button Objects

	Operation	Procedure	Method
✓	Initialization	uil-initialize-grobject	uil-initialize-grobject-method
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	uil-reset-button	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-toggle-button	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Check Button and Check Box Relations

For information about the relations between check buttons and check boxes, see [Check Box and Check Button Relations](#).



## Procedures that Operate on Check Buttons

This section describes procedures that you can use to manipulate check buttons. Check buttons are objects of classes that inherit from `uil-check-button`.

`uil-create-styled-check-button`

`uil-create-styled-check-button-with-different-sized-label`

# uil-create-styled-check-button

Creates and returns a check button and assigns it to a check box.

## Synopsis

uil-create-styled-check-button

(*id*: text, *size*: symbol, *check-box*: class uil-check-box, *label*: text,  
*on-value*: value, *off-value*: value, *callback*: symbol, *style*: item-or-value)  
-> *button*: class uil-button

Argument	Description
<i>id</i>	Specifies a unique ID for the button.
<i>size</i>	The size of check button to create.  For Motif <sup>tm</sup> style buttons, specify <i>small</i> , <i>medium</i> , or <i>large</i> .  For Windows <sup>tm</sup> style buttons, specify <i>small-msw</i> , <i>medium-msw</i> , or <i>large-msw</i> .
<i>check-box</i>	The check box containing the button. The check box is the <i>group-master-of</i> the created check button, where the <i>group-master-of</i> is a relation between a selection box and a button.
<i>label</i>	Text to appear to the right of the button.
<i>on-value</i>	The value of the button when it is selected (state is on).
<i>off-value</i>	The value of the button when it is not selected (state is off).
<i>callback</i>	A symbol specifying the name of the callback procedure executed when the button is selected.
<i>style</i>	The symbol <i>none</i> . This symbol is a place holder for style object.
Return Value	Description
<u><i>button</i></u>	The created button.

# uil-create-styled-check-button-with-different-sized-label

Creates and returns a check button with a label of a specified size.

## Synopsis

uil-create-styled-check-button-with-different-sized-label

(*id*: text, *button-size*: symbol, *label-size*: symbol,  
*check-box*: class uil-check-box, *label*: text, *on-value*: value,  
*off-value*: value, *callback*: symbol, *style*: item-or-value)  
 -> *button*: class uil-button)

Argument	Description
<i>id</i>	Specifies a unique ID for button.
<i>button-size</i>	The size of check button to create. For Motif <sup>tm</sup> style buttons, specify <b>small</b> , <b>medium</b> , or <b>large</b> . For Windows <sup>tm</sup> style buttons, specify <b>small-msw</b> , <b>medium-msw</b> , or <b>large-msw</b> .
<i>label-size</i>	The size of uil-text object to create to contain label for button. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>check-box</i>	The check box containing the button. The check box is the <b>group-master-of</b> the created check button, where the <b>group-master-of</b> is a relation between a selection box and a button.
<i>label</i>	Text to appear to the right of the button.
<i>on-value</i>	Value of button when it is selected (state is on).
<i>off-value</i>	Value of button when not selected (state is off).
<i>callback</i>	A symbol specifying the name of the callback procedure executed when the button is selected.
<i>style</i>	The symbol <b>none</b> (place holder for style object).
Return Value	Description
<i>button</i>	The created button.

# Additional Procedures that Support Check Buttons

[uil-30r3-upgrade-radio-and-check-button-label-texts](#)

[uil-utility-check-for-orphaned-selection-buttons](#)

[util-copy-object-attributes](#)

[util-lift-to-top](#)

[uil-add-button-to-selection-box](#)

[uil-add-label-to-button](#)

[uil-create-custom-button](#)

[uil-create-text-for-button-from-label](#)

[uil-disable-button](#)

[uil-enable-button](#)

[uil-handle-button](#)

[uil-handle-button-method](#)

[uil-highlight-button](#)

[uil-reset-button](#)

[uil-return-size-of-button-as-symbol](#)

[uil-select-button](#)

[uil-toggle-button](#)

[uil-update-button-label](#)

[uil-update-button-togglelabel](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UIL, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

uil-create-check-button

uil-delete-check-button

# uil-create-check-button

Creates and returns a check button and assigns it to a check box. Use instead: uil-create-styled-check-button.

## Synopsis

uil-create-check-button

(*id*: text, *check-box*: class uil-check-box, *callback*: symbol, *label*: text,  
*on-value*: item-or-value)

-> *button*: class uil-button)

Argument	Description
<i>id</i>	A text string specifying a unique ID for this button.
<i>check-box</i>	The check box containing the button. The check box is the-group-master-of the created check button, where the group-master-of is a relation between a selection box and a button.
<i>callback</i>	A symbol specifying the name of the callback procedure executed when the button is selected.
<i>label</i>	Text to appear to the right of the button.
<i>on-value</i>	The value of the button when it is selected.

Return Value	Description
<i>button</i> :	The created button.

## uil-delete-check-button

Deletes the specified check-button. Use instead: uil-delete-grobject-method.

### Synopsis

uil-delete-check-button

(*button*: class uil-check-button)

Argument	Description
<i>button</i>	Declares an object used by this procedure





# Dialogs

---

*Describes the UIL procedures that operate on dialogs.*

Introduction **76**

Procedures that Operate on Dialogs **88**

- uil-add-apply-button-to-dialog **90**
- uil-add-cancel-button-to-dialog **91**
- uil-add-grobject-to-dialog **92**
- uil-add-grobject-to-tab-page **93**
- uil-add-item-to-dialog **94**
- uil-add-message-to-dialog **95**
- uil-add-ok-button-to-dialog **96**
- uil-add-standard-buttons-to-dialog **97**
- uil-add-tab-page-to-dialog **98**
- uil-cancel-edit-session-in-window-if-any **99**
- uil-check-module-consistency-for-dialog **100**
- uil-clear-transient-dialog-pool **101**
- uil-control-dialog-callback **102**
- uil-create-confirm-dialog **104**
- uil-create-dialog-border **105**
- uil-create-message-dialog **106**
- uil-create-query-dialog **107**
- uil-create-tab-dialog **108**
- uil-create-tab-page **109**
- uil-create-tab-pushbutton **110**
- uil-create-tailored-dialog **111**
- uil-delete-permanent-copies-in-transient-dialog-pool **112**
- uil-end-dialog-edit **113**
- uil-find-destination-object **114**
- uil-find-initiating-object **115**
- uil-find-initiating-object-for-parent-dialog **116**
- uil-find-source-object **117**
- uil-find-target-object **118**
- uil-find-temporary-storage-object **119**
- uil-find-workspace-object-maxima **120**

uil-flush-dialog-event-queue 121  
uil-generate-customized-dialog 122  
uil-generate-default-dialog 124  
uil-get-active-dialog-on-window-if-one-exists 125  
uil-get-child-dialogs 126  
uil-get-dialog-for-this-temporary-storage-object 127  
uil-get-dialog-launched-for-item 128  
uil-get-dialogs-managed-on-window 129  
uil-get-grobject-from-id 130  
uil-get-grobject-from-id-on-dlg-or-wksp 131  
uil-get-master-dialog 132  
uil-get-parent-dialog 133  
uil-get-window-on-which-dialog-is-managed 134  
uil-get-windows-on-which-dialog-is-managed 135  
uil-hide-managed-dialog 136  
uil-insert-actions-into-dialog-event-queue 137  
uil-is-dialog-managed-on-window 138  
uil-is-this-a-child-dialog-of-this-dialog 139  
uil-is-this-dialog-managed-for-this-item 140  
uil-is-this-the-parent-dialog 141  
uil-locate-first-edit-box-on-dialog 142  
uil-locate-next-edit-box-on-dialog 143  
uil-locate-next-rows-edit-box-on-dialog 144  
uil-move-grobject-to-tab-page-layer 145  
uil-override-initiating-object-for-dialog 146  
uil-permanent-copies-exist-in-transient-dialog-pool 147  
uil-post-delay-notification 148  
uil-post-generic-dialog 149  
uil-post-notification-dialog 151  
uil-quit-dialog-edit 152  
uil-raise-tab-page-to-top 153  
uil-raise-tab-page-to-top-callback 154  
uil-release-dialog 155  
uil-remove-all-remaining-actions 156  
uil-remove-delay-notification-if-any 157  
uil-remove-grobject-from-tab-page 158  
uil-remove-notification-dialog 159  
uil-remove-tab-page-from-dialog 160  
uil-reserve-dialog 161  
uil-restore-dialog-grobject-list 163  
uil-select-tab-page 164  
uil-set-destination-object-for-dialog 165  
uil-show-managed-dialog 166  
uil-shrink-wrap-dialog 167  
uil-simulate-play-mode 168  
uil-start-dialog 169  
uil-start-dialog-processing 170

- uil-start-or-refocus-dialog 172
- uil-store-dialog-in-home-bin 173
- uil-translate-dialog 174
- uil-unsimulate-play-mode 175
- uil-update-controls-for-specific-attribute 176
- uil-update-dialog-for-initiating-object 177
- uil-update-notification-text 178
- uil-update-wait-clock-face 179
- uil-utility-create-permanent-copies-for-module 180
- uil-utility-create-permanent-copy-for-dialog 181
- uil-wait-for-dialog 182

Additional Procedures that Support Dialogs 183

Superseded Procedures 184

- uil-apply-edit-dialog-callback 185
- uil-clear-transient-dialog-pool-and-permanent-copies 186
- uil-delete-dialog 187
- uil-get-query-dialog-value 188
- uil-manage-and-show-dialog 189
- uil-popdown-and-delete-dialog-callback 190
- uil-popdown-and-hide-managed-dialog-callback 191
- uil-popdown-and-keep-edit-dialog-callback 192
- uil-popdown-dialog-callback 193
- uil-update-wait-semaphore 194



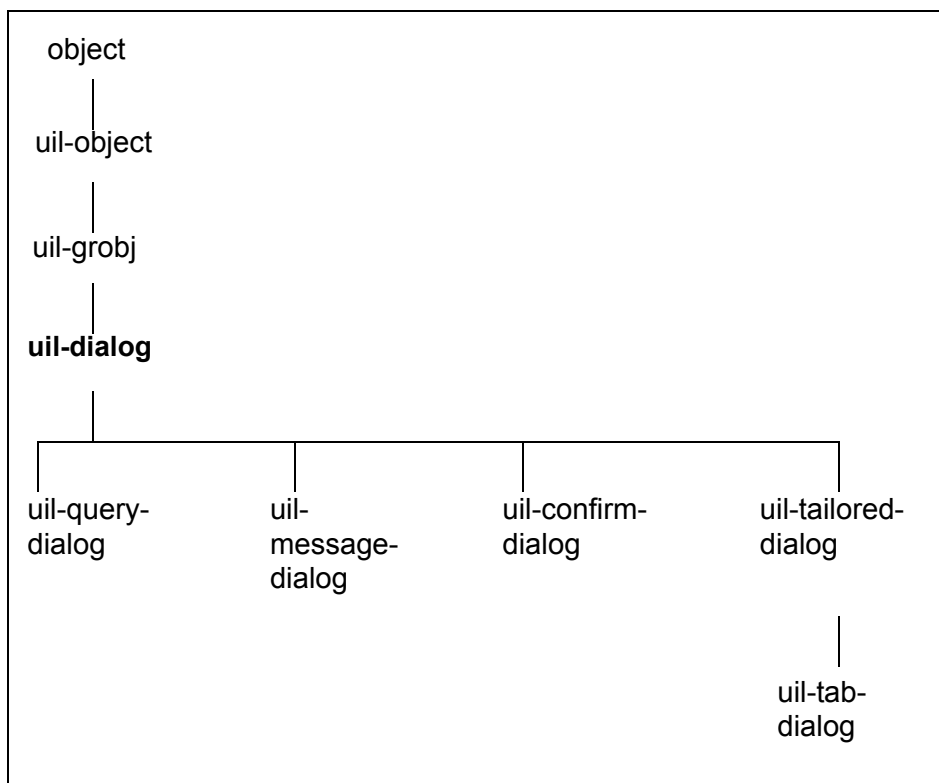
# Introduction

Dialogs add structure and flexibility to a user interface constructed with GUIDE. For information about how to create dialogs, see the *G2 GUIDE User's Guide*.

## Dialog Class Hierarchy

All dialogs inherit from the class `uil-dialog`. The following figure illustrates the UIL class hierarchy for dialogs. All classes shown are object definitions.

Dialog Class Hierarchy



Dialogs that you create using the GUIDE Dialog Generator or by cloning icons on the G2 GUIDE palette are of the class `uil-tailored-dialog`.

## Tab Dialog Classes

GUIDE/UIL now includes classes that support tabbed dialogs:

- `uil-tab-dialog`
- `uil-tab-page`
- `uil-tab-pushbutton`

These classes reside in the optional module `uiltdlg`.

### **uil-tab-dialog**

`uil-tab-dialog` is a subclass of `uil-tailored-dialog`, which is the basis for almost all dialogs found in *GUIDE*. A `uil-tab-dialog` has the same Edit Dialog dialog as other *GUIDE* dialogs, and it is launched and closed in the same ways.

It is recommended that the API procedures be used to modify the values of these attributes, and that they are accessed directly only when it is necessary to observe the values in them.

### **uil-tab-page**

`uil-tab-page` is a subclass of `uil-grmes` that defines the tab pages used in a `uil-tab-dialog`.

### **uil-tab-pushbutton**

`uil-tab-pushbutton` is a subclass of `uil-text-pushbutton` that defines the tab pushbuttons used on tab pages. The sole purpose of a tab pushbuttons is to lift the appropriate tab page to the top of the stack.

## **Dialog Attributes**

The following tables list attributes that are unique to or have special meaning for dialogs:

**Dialog Attributes**

<b>Attribute</b>	<b>Description</b>
<b>configuration</b>	Specifies the configuration that determines the color of this dialog.
<i>Allowable values:</i>	Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .
<i>Default value:</i>	<code>uil-dialog-configuration</code>

## Dialog Attributes

Attribute	Description
<b>uil-initiating-class-for-dialog</b>	For automatically generated dialogs, this attribute is set to the name of the G2 class for which the dialog was generated.  <i>Allowable values:</i> A valid user-defined G2 class.  <i>Default value:</i> none
<b>uil-temporary-storage-object</b>	Refers to a class of object that can be used to store values updated to and concluded from the dialog.  <i>Allowable values:</i> Valid class name  <i>Default value:</i> unspecified
<b>uil-which-button-pressed</b>	Contains the ID of the button selected on the dialog.  <i>Allowable values:</i> Valid button ID.  <i>Default value:</i> none
<b>uil-return-key-behavior</b>	Defines the return key's behavior during edit sessions on edit boxes.  <i>Allowable values:</i> end-and-tab-to-next-row, end-and-tab-to-next-field, end-and-exit-field-edit  <i>Default value:</i> end-and-tab-to-next-row
<b>uil-tab-key-behavior</b>	Defines the tab key's behavior during edit sessions on edit boxes.  <i>Allowable values:</i> end-and-tab-to-next-row, end-and-tab-to-next-field, end-and-exit-field-edit  <i>Default value:</i> end-and-tab-to-next-field

## Dialog Attributes

Attribute	Description
<b>uil-abort-key-behavior</b>	Defines the abort key's behavior during edit sessions on edit boxes.  <i>Allowable values:</i> cancel-and-tab-to-next-row, cancel-and-tab-to-next-field, cancel-and-exit-field-edit  <i>Default value:</i> cancel-and-tab-to-next-field
<b>uil-dialog-has-movable-workspace</b>	Indicates whether or not users can move the dialog when it is managed.  <i>Allowable values:</i> true (the dialog can be moved) false (the dialog cannot be moved)  <i>Default value:</i> true
<b>uil-dialog-edit-boxes-have-menus</b>	Indicates whether or not edit menus are displayed on the edit box during edit sessions when the dialog is managed.  <i>Allowable values:</i> true (edit menus are displayed) false (edit menus are not displayed)  <i>Default value:</i> false
<b>uil-origin</b>	Indicates whether the dialog is a master dialog.  <i>Allowable values:</i> master or permanent-copy  <i>Default value:</i> master
<b>uil-allow-multiple-copies-per-window</b>	Indicates whether or not multiple copies of a dialog can be displayed on the same window at the same time.

## Dialog Attributes

Attribute	Description
<p><i>Allowable values:</i> true (multiple copies can be displayed) false (multiple copies cannot be displayed)</p> <p><i>Default value:</i> false</p>	
<p><b>uil-dialog-allow-outside-text-selection</b></p> <p><i>Allowable values:</i> true or false</p> <p><i>Default value:</i> false</p>	<p>If set to <b>true</b>, users can overwrite the current contents of the currently selected edit box on the dialog by selecting text in another object on the same G2 window.</p>
<p><b>uil-dialog-home-module-name</b></p> <p><i>Allowable values:</i> name of an existing module</p> <p><i>Default value:</i> the name of the module that contains the workspace on which the dialog icon resides</p>	<p>Refers to the name of the module in which the dialog resides.</p>
<p><b>uil-dialog-instance-pool-limit</b></p> <p><i>Allowable values:</i> 0 or any positive integer</p> <p><i>Default value:</i> 1</p>	<p>Indicates the number of copies of a dialog that are allowed to accumulate in the dialog pool.</p>
<p><b>uil-dialog-in-use</b></p> <p><i>Allowable values:</i> true (dialog is managed) false (dialog is not managed)</p> <p><i>Default value:</i> false</p>	<p>Indicates whether or not a dialog is currently managed</p>



## Dialog Attributes

Attribute	Description
<b>uil-dialog-window-list</b>	An item list that contains all the windows on which the dialog is currently managed. The default value cannot be changed.  <i>Default value:</i> an uil-no-duplicate-item-list
Uil dialog grobj list	An item list containing the controls on the subworkspace of the dialog. The default value cannot be changed.  <i>Default value:</i> an uil-no-duplicate-item-list
<b>uil-always-update-when-managed</b>	Indicates whether or not the values of all graphical components on the dialog are to be updated whenever the dialog is displayed (managed), regardless of the state of the update flags for the controls.  <i>Allowable values:</i> <b>true</b> (update values) or <b>false</b> (do not update values)  <i>Default value:</i> <b>true</b>
<b>uil-x-position</b>	Moves the center of the dialog subworkspace to this x position on the G2 window.  <i>Allowable values:</i> Any valid x coordinate on the G2 window.  <i>Default value:</i> 0 (middle of G2 window)
<b>uil-y-position</b>	Moves the center of the dialog subworkspace to this y position on the G2 window.  <i>Allowable values:</i> Any valid y coordinate on the G2 window.  <i>Default value:</i> 0 (middle of G2 window)

## Dialog Attributes

Attribute	Description
<b>uil-use-default-positioning</b>	<p>When set to true, Uil x position and Uil y position remain at their default values (0,0) unless you edit them directly.</p> <p>When set to false, the location of the dialog when it is dismissed is written to the Uil x position and Uil y position attributes, so that the dialog is displayed in the same location the next time it is opened.</p> <p><i>Allowable values:</i> true or false</p> <p><i>Default value:</i> false</p>
<b>uil-allow-lift-to-top</b>	<p>When this attribute is set to true, the dialog is lifted above other dialogs or objects when a user clicks on its title or background.</p> <p>When this attribute is set to false, clicking the dialog does not lift it.</p> <p><i>Allowable values:</i> true or false</p> <p><i>Default value:</i> true</p>
<b>uil-translate-dialog</b>	<p>When set to True, UIL attempts to translate the dialog before displaying it. UIL translates text during execution of the uil-show-managed-dialog action.</p> <p><i>Allowable values:</i> true, false</p> <p><i>Default value:</i> false</p>
<b>uil-gfr-text-resource-group</b>	<p>Stores the name of the gfr-text-resource-group used by the application.</p> <p><i>Allowable values:</i> Any valid GFR text resource group (gfr-text-resource-group).</p> <p><i>Default value:</i> unspecified</p>

## Tab Dialog Attributes

The following table summarizes the attributes added to `uil-tab-dialog` to support tab dialogs.

Tab Dialog Attributes	
Attribute	Description
<b>uil-tab-page-button-placement</b>	Specifies the placement of tab page buttons on tab pages.  <i>Allowable values:</i> top, left, right  <i>Default value:</i> top
<b>uil-tab-page-list</b>	An item list containing the tab pages on the subworkspace of the dialog. The default value cannot be changed.  <i>Default value:</i> an uil-no-duplicate-item-list

The following table lists an attribute added to `uil-grobject` to support tab pages.

Attribute of uil-grobject to Support Tab Pages	
Attribute	Description
<b>uil-tab-page-contents-list</b>	An ordered list of the UIL items contained in the tab page.  <i>Default value:</i> uil-no-duplicate-item-list

It is recommended that these attributes not be modified directly, with one exception. Normally, only UIL objects can be added to the tab pages, so that the interactive manipulation of tab pages can work. A user who needs to put some other G2 object into the tab page can add it to this list programmatically.

## Operations Performed on Dialogs

The following table lists the common operations that application developers can perform on dialogs.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations on Dialogs

	Operation	Procedure	Method
	Change Size	n/a	n/a
	Change Style	uil-change-window-style-of-grobject	n/a
✓	Clone	uil-clone-grobject	uil-clone-grobject-method
✓	Conclude	uil-conclude-grobject	uil-conclude-grobject-method
	Conclude Wrapper	uil-call-conclude-method-for-grobject	n/a
✓	Configure	uil-configure-grobject	uil-configure-grobject-method
	Create	uil-create-tailored-dialog ***	n/a
✓	Delete	uil-delete-grobject	uil-delete-grobject-method
✓	Disable	uil-disable-grobject	uil-disable-grobject-method
✓	Enable	uil-enable-grobject	uil-enable-grobject-method
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobject	uil-initialize-grobject-method

### Operations on Dialogs

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	uil-validate-grobject	uil-validate-grobject-method
	Validate Wrapper	uil-call-validate-method-for-grobject	n/a

## Dialog Relations

The following table summarizes the standard relations between dialogs and other UIL objects. You can reference these relations to access dialogs and related objects programmatically.

**Relations Between Dialogs and Other Objects**

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-dialog	a-uil-dialog-presently-in-use-on-window	g2-window	many-to-many	the-g2-window-now-showing-dialog
g2-window	the-g2-window-now-showing-dialog	uil-dialog	many-to-many	a-uil-dialog-presently-in-use-on-window
uil-dialog	the-uil-dialog-resulting-from-action	item (initiating object or push button that launched dialog)	many-to-one	the-object-initiating-dialog-activity-for
item (initiating object or push button that launched dialog)	the-object-initiating-dialog-activity-for	uil-dialog	one-to-many	the-uil-dialog-resulting-from-action
uil-dialog	the-uil-parent-dialog-of-this-cascaded-dialog	uil-dialog	one-to-many	a-uil-child-cascaded-dialog-of
uil-dialog	a-uil-child-cascaded-dialog-of	uil-dialog	many-to-one	the-uil-parent-dialog-of-this-cascaded-dialog
message	the-current-edit-field-of	uil-dialog	one-to-one	none

## Tab Dialog Relations

The following table summarizes relations that associate tab dialogs with tab pages, and tab pages with the tab buttons and objects that are on the tab pages:

**Relations Between Tab Dialogs and Other Objects**

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-tab-page	the-tab-button-of-the-tab-page	uil-button	one-to-one	the-tab-page-launched-by-the-tab-button
uil-tab-dialog	the-uil-selected-tab-page-of	uil-tab-page	one-to-one	the-uil-dialog-of-the-selected-tab-page

the-uil-selected-tab-page-of can be used to find out which tab page on a given tab dialog is currently selected and on top of the stack.

the-tab-button-of-the-tab-page associates the tab page with its button, so that each tab button can be accessed through its tab page and changed programmatically.

## Procedures that Operate on Dialogs

This section describes procedures that you can use to manipulate dialogs. Dialogs are objects of the class `uil-dialog`.

- `uil-add-apply-button-to-dialog`
- `uil-add-cancel-button-to-dialog`
- `uil-add-grobject-to-dialog`
- `uil-add-grobject-to-tab-page`
- `uil-add-item-to-dialog`
- `uil-add-message-to-dialog`
- `uil-add-ok-button-to-dialog`
- `uil-add-standard-buttons-to-dialog`
- `uil-add-tab-page-to-dialog`
- `uil-check-module-consistency-for-dialog`
- `uil-clear-transient-dialog-pool`
- `uil-control-dialog-callback`
- `uil-create-confirm-dialog`
- `uil-create-dialog-border`
- `uil-create-message-dialog`
- `uil-create-query-dialog`
- `uil-create-tab-dialog`
- `uil-create-tab-page`
- `uil-create-tab-pushbutton`
- `uil-create-tailored-dialog`
- `uil-delete-permanent-copies-in-transient-dialog-pool`
- `uil-end-dialog-edit`
- `uil-find-destination-object`
- `uil-find-initiating-object`
- `uil-find-initiating-object-for-parent-dialog`
- `uil-find-source-object`
- `uil-find-target-object`
- `uil-find-temporary-storage-object`
- `uil-find-workspace-object-maxima`
- `uil-flush-dialog-event-queue`
- `uil-generate-customized-dialog`
- `uil-generate-default-dialog`
- `uil-get-active-dialog-on-window-if-one-exists`
- `uil-get-child-dialogs`
- `uil-get-dialog-for-this-temporary-storage-object`
- `uil-get-dialog-launched-for-item`
- `uil-get-dialogs-managed-on-window`
- `uil-get-grobject-from-id`
- `uil-get-grobject-from-id-on-dlg-or-wksp`
- `uil-get-master-dialog`
- `uil-get-parent-dialog`
- `uil-get-window-on-which-dialog-is-managed`



uil-get-windows-on-which-dialog-is-managed  
uil-hide-managed-dialog  
uil-insert-actions-into-dialog-event-queue  
uil-is-dialog-managed-on-window  
uil-is-this-a-child-dialog-of-this-dialog  
uil-is-this-dialog-managed-for-this-item  
uil-is-this-the-parent-dialog  
uil-locate-first-edit-box-on-dialog  
uil-locate-next-edit-box-on-dialog  
uil-locate-next-rows-edit-box-on-dialog  
uil-move-grobject-to-tab-page-layer  
uil-override-initiating-object-for-dialog  
uil-permanent-copies-exist-in-transient-dialog-pool  
uil-post-delay-notification  
uil-post-generic-dialog  
uil-post-notification-dialog  
uil-quit-dialog-edit  
uil-raise-tab-page-to-top  
uil-raise-tab-page-to-top-callback  
uil-release-dialog  
uil-remove-all-remaining-actions  
uil-remove-delay-notification-if-any  
uil-remove-grobject-from-tab-page  
uil-remove-notification-dialog  
uil-remove-tab-page-from-dialog  
uil-restore-dialog-grobject-list  
uil-select-tab-page  
uil-set-destination-object-for-dialog  
uil-show-managed-dialog  
uil-shrink-wrap-dialog  
uil-simulate-play-mode  
uil-start-dialog  
uil-start-dialog-processing  
uil-start-or-refocus-dialog  
uil-store-dialog-in-home-bin  
uil-translate-dialog  
uil-unsimulate-play-mode  
uil-update-controls-for-specific-attribute  
uil-update-dialog-for-initiating-object  
uil-update-notification-text  
uil-update-wait-clock-face  
uil-utility-create-permanent-copies-for-module  
uil-utility-create-permanent-copy-for-dialog  
uil-wait-for-dialog

# uil-add-apply-button-to-dialog

Adds an Apply button with the label “Apply” to a dialog. The button is configured by default with the action `uil-call-conclude-method`.

## Synopsis

`uil-add-apply-button-to-dialog`

(*dialog*: class uil-dialog, *size*: symbol, *x*: integer, *y*: integer)

-> *apply-button*: class uil-button

Argument	Description
<i>dialog</i>	The dialog the button will be added to.
<i>size</i>	The size of button to create. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>x</i>	The x position for the button.
<i>y</i>	The y position for the button.

Return Value	Description
<u><i>apply-button</i></u> :	The Apply button that is added.

## uil-add-cancel-button-to-dialog

Adds a Cancel button with the label “Cancel” to a dialog. By default, the button is configured with the actions `uil-unsimulate-play-mode`, `uil-hide-dialog`, and `uil-release-dialog`.

### Synopsis

`uil-add-cancel-button-to-dialog`

(*dialog*: class uil-dialog, *size*: symbol, *x*: integer, *y*: integer)

-> *cancel-button*: class uil-button

Argument	Description
<i>dialog</i>	The dialog to which the button is added.
<i>size</i>	The size of button to create. Specify <code>small</code> , <code>medium</code> , or <code>large</code> .
<i>x</i>	The x position for the button.
<i>y</i>	The y position for the button.
Return Value	Description
<u><i>cancel-button</i></u>	The created Cancel button.

# uil-add-grobj-to-dialog

Adds the specified object to the dialog's subworkspace at the given location. The object added (*grobj*) must be an instance of one of the predefined UIL classes, or a user-defined subclass of one of these classes.

## Synopsis

uil-add-grobj-to-dialog

(*dialog*: class uil-dialog, *grobj*: class item, *x*: integer, *y*: integer)

-> *item-or-value*

Argument	Description
<i>dialog</i>	The dialog to add the object to.
<i>grobj</i>	The object to add to dialog. This object is expected to be a UIL object.
<i>x</i>	The x position on the dialog's subworkspace at which to add object.
<i>y</i>	The y position on the dialog's subworkspace at which to add object.
Return Value	Description
<u><i>item-or-value</i></u>	The object added to the dialog.

## uil-add-grobject-to-tab-page

Adds an item to a specified tab page.

### Synopsis

uil-add-grobject-to-tab-page(*item*: class item, *tab-page*: class uil-tab-page)

Argument	Description
<i>item</i>	The item to add to the specified tab page.
<i>tab-page</i>	The tab page to which the item is added.

### Description

uil-add-grobject-to-tab-page() adds the specified *item* to the tab page specified by *tab-page*.

# uil-add-item-to-dialog

Adds the specified item to the dialog's subworkspace at the given location.

## Synopsis

uil-add-item-to-dialog

(*dialog*: class uil-dialog, *item*: class item, *x*: integer, *y*: integer)

Argument	Description
<i>dialog</i>	The dialog to which the item is added.
<i>item</i>	The item added to the dialog. This can be any G2 item.
<i>x</i>	The x position on the dialog's subworkspace at which to add the item.
<i>y</i>	The y position on the dialog's subworkspace at which to add item.

## Description

The uil-add-item-to-dialog procedure enables you to add items to an existing dialog programmatically. It adds the specified item to the dialog's subworkspace at the given location.

---

**Note** To add UIL objects to a dialog, use uil-add-grobj-to-dialog.

---

## Related Procedures

[uil-add-message-to-dialog](#)

[uil-add-grobj-to-dialog](#)

## uil-add-message-to-dialog

Adds a text object (object of the class `uil-text`) of the specified size with the given text to the dialog at the specified location.

### Synopsis

`uil-add-message-to-dialog`

(*dialog*: class `uil-dialog`, *message*: text, *size*: symbol, *x*: integer, *y*: integer)

-> *text-object*: class `uil-text`

Argument	Description
<i>dialog</i>	The dialog to which the message is added.
<i>message</i>	The text of the message to add.
<i>size</i>	Specify <code>small</code> , <code>large</code> , or <code>extra-large</code> .
<i>x</i>	The x position on the dialog where the text is added.
<i>y</i>	The y position on the dialog where the text is added.
Return Value	Description
<u><i>text-object</i></u>	The text object added to the dialog.

## uil-add-ok-button-to-dialog

Adds an OK button with the label "OK" to a dialog. The button is configured by default with the actions: uil-call-conclude-method, uil-unsimulate-play-mode, uil-hide-dialog, uil-release-dialog.

### Synopsis

uil-add-ok-button-to-dialog

(*dialog*: class uil-dialog, *size*: symbol, *x*: integer, *y*: integer)

-> *ok-button*: class uil-button

Argument	Description
<i>dialog</i>	The dialog to which the button is added.
<i>size</i>	The size of button to create. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>x</i>	The x position on the dialog where the button is added.
<i>y</i>	The y position on the dialog where the button is added.

Return Value	Description
<u><i>ok-button</i></u>	The OK button that is added.



# uil-add-standard-buttons-to-dialog

Adds the standard OK, Apply and Cancel buttons to the dialog. These buttons are configured with appropriate actions.

## Synopsis

uil-add-standard-buttons-to-dialog

(*dialog*: class uil-dialog)

-> *ok-button*: class uil-button, *apply-button*: class uil-button,

*cancel-button*: class uil-button

Argument	Description
<i>dialog</i>	The dialog to which the OK, Apply and Cancel buttons are added.

Return Values	Description
<u><i>ok-button</i></u> , <u><i>apply-button</i></u> , <u><i>cancel-button</i></u>	Returns standard OK, Apply and Cancel buttons in the following order: OK, Apply, Cancel.

## Related Procedures

[uil-add-apply-button-to-dialog](#)

[uil-add-cancel-button-to-dialog](#)

[uil-add-ok-button-to-dialog](#)

# uil-add-tab-page-to-dialog

Adds a tab page to a dialog.

## Synopsis

uil-add-tab-page-to-dialog

(*tab-page*: class uil-tab-page, *dialog*: class uil-tab-dialog)

Argument	Description
<i>tab-page</i>	The tab page that is added to the dialog.
<i>dialog</i>	The dialog to which the tab page is added.

## Description

uil-add-tab-page-to-dialog() adds the tab page specified by *tab-page* to the tab dialog specified by *dialog*.

## uil-cancel-edit-session-in-window-if-any

Cancels the most recent edit session in a particular client window.

### Synopsis

**uil-cancel-edit-session-in-window-if-any**

(*client*: class ui-client-item)

Argument	Description
<i>client</i>	The client window in which the dialog is displayed.

# uil-check-module-consistency-for-dialog

Verifies that the specified dialog is in the appropriate home bin. If the dialog is not in the appropriate bin, this procedure moves it to that bin.

## Synopsis

uil-check-module-consistency-for-dialog  
(*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog to check the module consistency of.

# uil-clear-transient-dialog-pool

Deletes copies of a specified master dialog from the dialog pool.

## Synopsis

uil-clear-transient-dialog-pool

(*dialog*: class uil-dialog)

-> *status*: symbol

Argument	Description
<i>dialog</i>	The master dialog.

Return Value	Description
<u><i>status</i></u>	One of the following symbols: successfully-deleted-transient-dialogs-from-pool, dialog-is-not-a-master-copy, permanent-copies-found-in-pool, no-dialogs-deleted.

# uil-control-dialog-callback

Reserves a dialog and runs actions on the dialog.

## Synopsis

uil-control-dialog-callback (*target-object*: item-or-value,  
*target-attribute*: symbol, *action-array*: item-or-value,  
*initiating-item*: item-or-value, *window*: class G2-window)  
-> {*dialog*: class uil-dialog | dialog-not-found: symbol},  
    {*button*: class uil-button | none: symbol}

Argument	Description
<i>target-object</i>	The dialog on which actions are run. Specify either: <ul style="list-style-type: none"><li>• A text value representing a valid dialog ID, enclosed in quotation marks, or</li><li>• the symbol host-dialog</li></ul> Specify the symbol host-dialog if you are invoking uil-control-dialog-callback from a button on a dialog and you want to run actions on the dialog that contains the button.
<i>target-attribute</i>	Not used. Specify the symbol NONE.
<i>action array</i>	An action description array specifying the actions run on the dialog, or the symbol NONE.

Argument	Description
<i>initiating-item</i>	The object that launches the dialog (for example, an action button, push button, or user menu choice), or the symbol NONE. Specify this item if you call a procedure from a user menu choice.
<i>window</i>	The G2 window on which the dialog is managed. Specify this window if you call a procedure from a user menu choice.  <b>Note:</b> This procedure creates an inoperable dialog if <i>window</i> specifies an invalid G2 window. An invalid G2 window is one whose <code>g2-connection-status</code> is <i>not</i> equal to the symbol connected.
Return Value	Description
<u><i>dialog</i></u>	The dialog, or the symbol dialog-not-found.
<u><i>button</i></u>	The button that was activated on the dialog, or the symbol none.

## Description

`uil-control-dialog-callback` analyzes the button's `uil-target-object`, which is either a dialog ID, or the symbol `host-dialog`, and adds its actions into the event handler for the target dialog. If no event handler is active, `uil-control-dialog-callback` starts one.

The only dialog action that returns the dialog to the launching procedure is `uil-return-dialog`. If `uil-return-dialog` is used, users are responsible for calling `uil-release-dialog` to return the dialog to its dialog pool.

# uil-create-confirm-dialog

Creates a confirm dialog.

## Synopsis

uil-create-confirm-dialog

(*id*: text, *obj*: item-or-value, *message*: text, *size*: symbol)

-> (*dialog*: class uil-dialog, *yes-button*: class uil-button,

*no-button*: class uil-button, *cancel-button*: class uil-button)

Argument	Description
<i>id</i>	The ID for the confirm dialog.
<i>obj</i>	The symbol none, or an item to establish as the-dialog-object-of for the dialog.
<i>message</i>	The text to add to the dialog.
<i>size</i>	The size text to create. Specify small, large, or extra-large.

Return Values	Description
<u><i>dialog</i></u>	The confirm dialog that is created.
<u><i>yes-button</i></u> , <u><i>no-button</i></u> , <u><i>cancel-button</i></u>	The Yes, No, and Cancel buttons on the confirm dialog.



# uil-create-dialog-border

Creates a border of the given style for the specified dialog.

## Synopsis

uil-create-dialog-border

(*style*: symbol, *configuration*: symbol, *dialog*: class uil-dialog)

-> *border*: uil-box-object

Argument	Description
<i>style</i>	Specify wide, medium, narrow, thin, sunken-wide, sunken-medium, sunken-narrow, or sunken-thin.  For a description of these different border styles, see <a href="#">uil-create-border-for-text</a> .
<i>configuration</i>	The name of a uil-configuration-class to use as the border's configuration.
<i>dialog</i>	The dialog for which a border is created.
Return Value	Description
<u><i>border</i></u>	The created border.

# uil-create-message-dialog

Creates a message dialog.

## Synopsis

uil-create-message-dialog

(*id*: text, *obj*: item-or-value, *message*: text, *size*: symbol)

-> *dialog*:class uil-dialog, *button*: class uil-button

Argument	Description
<i>id</i>	The ID for the message dialog.
<i>obj</i>	The symbol none, or an item to be established as the dialog-object-of for the message dialog.
<i>message</i>	The text to add to the dialog.
<i>size</i>	The size text to create, one of small, large, extra-large.

Return Value	Description
<u><i>dialog</i></u>	The message dialog created.
<u><i>button</i></u>	The OK button on the message dialog.

# uil-create-query-dialog

Creates a query dialog with standard OK, Apply, and Cancel buttons.

## Synopsis

uil-create-query-dialog

(*id*: text, *obj*: item-or-value, *message*: text, *size*: symbol,  
*initial-value*: item-or-value)

-> *dialog*: class uil-dialog, *ok-button*: class uil-button,  
*apply-button*: class uil-button, *cancel-button*: class uil-button

Argument	Description
<i>id</i>	The ID for the query dialog.
<i>obj</i>	The symbol <code>none</code> , or an item to be established as the <code>dialog-object-of</code> for the message dialog.
<i>message</i>	The text to add to the dialog.
<i>size</i>	The size text to create, one of <code>small</code> , <code>large</code> , <code>extra-large</code> .
<i>initial-value</i>	The value to display in the edit box on the query dialog.
Return Values	Description
<i>dialog</i>	The query dialog created.
<i>ok-button</i> , <i>apply-button</i> , <i>cancel-button</i>	The OK, Apply, and Cancel buttons on the query dialog.

# uil-create-tab-dialog

Creates a tab dialog.

## Synopsis

uil-create-tab-dialog (*id*: text)  
-> *dialog*: class uil-tab-dialog

Argument	Description
<i>id</i>	The ID assigned to the tab dialog.

Return Value	Description
<i>dialog</i>	The tab dialog.

## Description

uil-create-tab-dialog() creates a tab dialog (an instance of uil-tab-dialog) and sets the ID of the dialog to the value specified for *id*. The return value of the procedure is the tab dialog.

# uil-create-tab-page

Creates a tab page with a tab button.

## Synopsis

uil-create-tab-page

(*id*: text, *size*: symbol, *style*: symbol, *label*: text)

-> *page*: class uil-tab-page

Argument	Description
<i>id</i>	The object ID of the tab page created.
<i>size</i>	The size of the tab page. Specify one of: small, medium, or large.
<i>style</i>	Not used in this release.
<i>label</i>	The text of the label on the tab button created for this tab page.
Return Value	Description
<i>page</i>	The tab page (instance of uil-tab-page) created by this procedure.

## Description

uil-create-tab-page() creates a tab page (an instance of uil-tab-page) and a push button (an instance of uil-tab-pushbutton) and returns the tab page.

The tab page and tab button are joined through the relation the-tab-button-of-the-tab-page. The *size* and *label* arguments specify the size and label text of the tab pushbutton. The *style* argument is not used in this release.

# uil-create-tab-pushbutton

Creates a tab pushbutton.

## Synopsis

uil-create-tab-pushbutton  
(*size*: symbol, *label*: text)  
-> *push-button*: class uil-tab-pushbutton

<b>Argument</b>	<b>Description</b>
<i>size</i>	The size of the tab pushbutton. Specify one of: small, medium, or large.
<i>label</i>	The text of the label on the tab pushbutton.

<b>Return Value</b>	<b>Description</b>
<u><i>push-button</i></u>	The tab push button (instance of uil-tab-pushbutton) created by this procedure.

# uil-create-tailored-dialog

Creates and returns a tailored dialog box.

## Synopsis

uil-create-tailored-dialog

(*id*: text, *related-item*: symbol)

-> tailored-dialog: class uil-dialog

Argument	Description
<i>id</i>	The ID for the dialog.
<i>related-item</i>	A symbol specifying the name of a G2 item for which the relation <code>the-dialog-of</code> is established with the dialog created. Specify the symbol <code>none</code> to have no related item.
Return Value	Description
<u>tailored-dialog</u> :	The created dialog.

## Description

The procedure `uil-create-tailored-dialog` creates an empty master dialog, without buttons or title bar.

This procedure also defines the relation `the-dialog-of` between the tailored dialog and the G2 item that you specify as `related-item`. This relation is included in the description of the dialog that you see when you choose `describe` from the dialog's menu.

# uil-delete-permanent-copies-in-transient-dialog-pool

Deletes permanent copies of a dialog from the dialog pool.

## Synopsis

uil-delete-permanent-copies-in-transient-dialog-pool  
(*master-dialog*: class uil-dialog)

Argument	Description
<i>master-dialog</i>	The permanent copies of this master dialog are deleted.

## Description

This procedure checks the dialog pool for permanent copies of the specified master dialog. If it finds any, it deletes them.

It does not check for or delete non-permanent copies.

## Related Procedure

[uil-clear-transient-dialog-pool](#)



## uil-end-dialog-edit

Ends the edit session for the specified dialog on the given window, if one exists.

### Synopsis

uil-end-dialog-edit

(*dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>dialog</i>	The dialog to terminate edit session for.
<i>window</i>	The window on which dialog is managed.

# uil-find-destination-object

Finds the destination object of a specified dialog, starting with the specified dialog and working up the parent-child dialog hierarchy.

## Synopsis

uil-find-destination-object

(*dialog*: class uil-dialog)

-> {*destination-object*: item-or-value | not-found: symbol}

Argument	Description
<i>dialog</i>	The dialog to find the destination-object of.

Return Value	Description
<u><i>destination-object</i></u>	The destination object, or the symbol not-found.

## uil-find-initiating-object

Finds the initiating object of the dialog, starting with the specified dialog and working up the parent-child dialog hierarchy.

### Synopsis

uil-find-initiating-object

(*dialog*: class uil-dialog)

-> *initiating-object*: item-or-value | not-found: symbol}

Argument	Description
<i>dialog</i>	The dialog to find the initiating-object of.

Return Value	Description
<u><i>initiating-object</i></u>	The initiating object, or the symbol not-found.

# uil-find-initiating-object-for-parent-dialog

Finds the initiating object for the specified dialog, starting with the highest level parent dialog and working down the parent dialog hierarchy.

## Synopsis

uil-find-initiating-object-for-parent-dialog

(*dialog*: class uil-dialog)

-> {*initiating-object*: item-or-value | not-found: symbol}

Argument	Description
<i>dialog</i>	The dialog to find the initiating-object of.

Return Value	Description
<u><i>initiating-object</i></u>	The initiating object, or the symbol not-found.

## uil-find-source-object

Finds the source object of a specified object, starting with a specified dialog and working up the parent-child dialog hierarchy.

### Synopsis

uil-find-source-object

(*item*: class item, *dialog-or-workspace*: item-or-value)

-> {source-object: item-or-value | not-found:symbol}

Argument	Description
<i>item</i>	The object to find the source-object of.
<i>dialog-or-workspace</i>	The dialog or workspace on which the object resides.
Return Value	Description
<u>source-object</u>	The source object, or the symbol not-found.

# uil-find-target-object

Finds the target object of a specified object, starting with a specified dialog and working up the parent-child dialog hierarchy.

## Synopsis

uil-find-target-object

(*item*: class item, *dialog-or-workspace*: item-or-value)

-> {*target-object*: item-or-value | not-found: symbol}

Argument	Description
<i>item</i>	The object to find the target object of.
<i>dialog-or-workspace</i>	The dialog or workspace on which the object resides.

Return Value	Description
<u><i>target-object</i></u>	The target object, or the symbol not-found.

## uil-find-temporary-storage-object

Finds the temporary storage object of a specified object, starting with the specified dialog and working up the parent-child dialog hierarchy.

### Synopsis

```
uil-find-temporary-storage-object
  (itm: item-or-value, dialog: class uil-dialog, from: symbol)
  -> result: item-or-value
```

Argument	Description
<i>itm</i>	The symbol <code>none</code> , or the item to look for a match against either the <code>uil-event-target-attribute</code> or <code>uil-event-source-attribute</code> in the temporary-storage objects found while searching up the parent-child dialog hierarchy.
<i>dialog</i>	The dialog from which to starting looking for the temporary storage object.
<i>from</i>	One of two values: <code>SOURCE</code> or <code>TARGET</code> .  If <code>SOURCE</code> , then a match is looked for between the temporary storage object and the <code>uil-event-source-attribute</code> of the specified item ( <i>itm</i> ).  If <code>TARGET</code> , a match is looked for between the temporary storage object and the <code>uil-event-target-attribute</code> of the item ( <i>itm</i> ).
Return Value	Description
<u>result</u>	The temporary storage object or the symbol <code>not-found</code> .

# uil-find-workspace-object-maxima

Returns the minimum and maximum X and Y values of the items on the workspace or dialog subworkspace, not including borders.

## Synopsis

uil-find-workspace-object-maxima

(*workspace*: class kb-workspace)

-> *min-x*: integer, *min-y*: integer, *max-x*: integer, *max-y*: integer

Argument	Description
<i>workspace</i>	The workspace to return the extents of.

Return Values	Description
<i>min-x</i>	The minimum X value.
<i>min-y</i>	The minimum Y value.
<i>max-x</i>	The maximum X value.
<i>max-y</i>	The maximum Y value.



# uil-flush-dialog-event-queue

Removes pending actions from the specified dialog's event queue.

## Synopsis

uil-flush-dialog-event-queue

(*dialog*: class uil-dialog, *window*: class g2-window, *actions*: class symbol-list)

Argument	Description
<i>dialog</i>	The dialog whose event queue is to be flushed.
<i>window</i>	The window on which dialog is managed.
<i>actions</i>	The actions pending in the dialog's event queue.

## Related Procedures

[uil-insert-actions-into-dialog-event-queue](#)

[uil-remove-all-remaining-actions](#)

# uil-generate-customized-dialog

Generates a customized master dialog for displaying and editing specified attributes of a user-defined class.

## Synopsis

uil-generate-customized-dialog

(*user-class*: symbol, *dialog-id*: text, *title*: text, *target-object*: symbol,  
*attribute-mapping*: class uil-object-attribute-to-dialog-control-mappings)  
-> *dialog*: class uil-dialog

Argument	Description
<i>user-class</i>	The class for which the dialog is generated.
<i>dialog-id</i>	The object ID of the dialog generated.
<i>title</i>	The title of the dialog generated.
<i>target-object</i>	Specify initiating-object.  Represents the uil-event-source-object and uil-event-target-object for the UIL controls on the generated dialog.
<i>attribute-mapping</i>	A list of objects, each of which specifies an attribute of <i>user-class</i> and the class of UIL control that is added to the generated dialog to represent that attribute.  Each object in the list is an instance of uil-object-to-dialog-control-map.
Return Value	Description
<i>dialog</i>	The customized dialog generated by this procedure.

## Description

The class `uil-object-attribute-to-dialog-control-mappings` is a list of objects of type `uil-object-to-dialog-control-map`, each of which has two attributes:

- `attribute-name`, representing the name of an attribute of the class for which one is generating the dialog.
- `control-type`, the name of a class of UIL control that you want to use to represent the attribute. For example: `uil-radio-box`, `uil-edit-box`.

# uil-generate-default-dialog

Generates a dialog for a specified class that contains an edit box to represent attributes of that class.

## Synopsis

uil-generate-default-dialog

(*dialog-class*: symbol, *dialog-id*: text, *title*: text,  
*target-object*: symbol, *attributes*: item-or-value)

-> *dialog*: class uil-dialog

Argument	Description
<i>dialog-class</i>	The class of the dialog generated.
<i>dialog-id</i>	The object ID of the dialog generated.
<i>title</i>	The title of the dialog generated.
<i>target-object</i>	Specify initiating-object.
	Represents the uil-event-source-object and uil-event-target-object for the UIL controls on the generated dialog.
<i>attributes</i>	A list of the attributes of <i>dialog-class</i> for which you want to add edit boxes to the generated dialog.
Return Value	Description
<i>dialog</i>	The generated dialog.

## uil-get-active-dialog-on-window-if-one-exists

Finds an active (managed) dialog with a specified ID on the specified window. If it finds such a dialog, it returns the dialog.

### Synopsis

```
uil-get-active-dialog-on-window-if-one-exists
(id: text, window: class g2-window)
-> {dialog: class uil-dialog | not-found: symbol}
```

Argument	Description
<i>id</i>	The ID of the dialog being checked for.
<i>window</i>	The window on which to check for the dialog.
Return Value	Description
<u><i>dialog</i></u>	A dialog or the symbol not-found.

# uil-get-child-dialogs

Populates an item-list with all the managed child dialogs for the specified dialog.

## Synopsis

uil-get-child-dialogs

(*parent*: item-or-value, *children*: class item-list)

Argument	Description
<i>parent</i>	The dialog to get the child dialogs for.
<i>children</i>	An item-list that is populated with the child dialogs of parent.

# uil-get-dialog-for-this-temporary-storage-object

Returns the dialog associated with the temporary storage object.

## Synopsis

uil-get-dialog-for-this-temporary-storage-object

(*temp-storage*: item-or-value)

-> result: item-or-value

Argument	Description
<i>temp-storage</i>	The temporary storage object to return the dialog for. If no dialog is found, the symbol no-dialog-found is returned.

Return Value	Description
<u>result</u>	The temporary storage object or the symbol no-dialog-found.

# uil-get-dialog-launched-for-item

Returns the dialog that was launched as a result of selecting a particular item. The specified item is the initiating object for the dialog that is returned.

## Synopsis

```
uil-get-dialog-launched-for-item  
  (itm: item-or-value)  
  -> result: item-or-value
```

Argument	Description
<i>itm</i>	The item that is the initiating object.

Return Value	Description
<u>result</u>	The dialog that was launched as a result of selecting the specified item. If no dialog is found, then the symbol dialog-not-found is returned.



## uil-get-dialogs-managed-on-window

Populates an item-list with the dialogs currently managed on the specified window.

### Synopsis

uil-get-dialogs-managed-on-window  
(*window*: item-or-value, *dialogs*: class item-list)

Argument	Description
<i>window</i>	The window to check for managed dialogs.
<i>dialogs</i>	The item-list to populate with dialogs that are managed on the specified window.

# uil-get-grobj-from-id

Finds the item with a specified ID.

## Synopsis

uil-get-grobj-from-id (*id*: text)

-> {*object*: class uil-grobj | item-not-found: symbol}

Argument	Description
<i>id</i>	The ID of the object to find.

Return Value	Description
<u><i>object</i></u>	The object whose ID matches the specified ID, or the symbol item-not-found.

## Related Procedure

[uil-get-grobj-from-id-on-dlg-or-wksp](#)

## uil-get-grobj-from-id-on-dlg-or-wksp

Searches for an object with a specified ID on a specified dialog or workspace. Use this procedure to find an item on a managed dialog.

### Synopsis

```
uil-get-grobj-from-id-on-dlg-or-wksp
  (id: text, dialog-or-workspace: class item)
  -> {object: item | item-not-found: symbol}
```

Argument	Description
<i>id</i>	The ID of the object to find.
<i>dialog-or-workspace</i>	The dialog or workspace on which to look for the object.

Return Value	Description
<u><i>object</i></u>	The object whose ID matches the specified ID, or the symbol <code>item-not-found</code> .

# uil-get-master-dialog

Finds and returns the master copy of a dialog.

## Synopsis

uil-get-master-dialog

(*id*: text)

-> {master-dialog: class uil-dialog | not-found:symbol}

Argument	Description
<i>id</i>	The ID of the master dialog to find.

Return Value	Description
<u>master-dialog</u>	The master dialog, or the symbol not-found.

# uil-get-parent-dialog

Returns the parent dialog for the specified dialog.

## Synopsis

```
uil-get-parent-dialog  
  (child: item-or-value)  
  -> result: item-or-value
```

Argument	Description
<i>child</i>	The dialog to return the parent dialog of.

Return Value	Description
<u>result</u>	The dialog that is the parent dialog of <i>child</i> . If no parent dialog is found then the symbol no-parent-dialog-found is returned.

# uil-get-window-on-which-dialog-is-managed

Returns the window on which the specified dialog is managed.

## Synopsis

uil-get-window-on-which-dialog-is-managed

(*dialog*: item-or-value)

-> result: item-or-value

Argument	Description
<i>dialog</i>	The dialog to check if managed.

Return Value	Description
<u>result</u>	The g2-window on which the specified dialog is managed. If the dialog is not currently managed on a window, then the symbol dialog-not-managed is returned.

# uil-get-windows-on-which-dialog-is-managed

Populates an item-list with the windows on which the specified dialog is managed.

## Synopsis

uil-get-windows-on-which-dialog-is-managed  
(*dialog*: item-or-value, *windows*: class item-list)

Argument	Description
<i>dialog</i>	The dialog to check if managed.
<i>windows</i>	An item-list that is populated with the g2-windows on which the dialog is managed.

# uil-hide-managed-dialog

Hides a dialog that is shown (managed) on a specified G2 window.

## Synopsis

uil-hide-managed-dialog

(*dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>dialog</i>	The dialog to unmanage.
<i>window</i>	The window on which the dialog is managed.



# uil-insert-actions-into-dialog-event-queue

Inserts actions into event queue for a specified dialog on a specified window.

## Synopsis

uil-insert-actions-into-dialog-event-queue

(*action-array*: class symbol-array, *dialog*: class uil-dialog,  
*window*: class g2-window)

Argument	Description
<i>action-array</i>	Array containing symbolic names of procedures to insert into event queue. Procedures are listed in the order in which they are to be executed.
<i>dialog</i>	The dialog to receive the actions.
<i>window</i>	The window on which the dialog is managed.

## Related Procedures

[uil-flush-dialog-event-queue](#)

[uil-remove-all-remaining-actions](#)

# uil-is-dialog-managed-on-window

Returns a truth-value indicating whether the specified dialog is managed on the specified window.

## Synopsis

uil-is-dialog-managed-on-window

(*dialog*: item-or-value, *window*: item-or-value)

-> result: truth-value

Argument	Description
<i>dialog</i>	The dialog to check for being managed on the window.
<i>window</i>	The window on which to check if the dialog is managed.

Return Value	Description
<u>result</u>	If the dialog is managed on the window, then TRUE, else FALSE.

# uil-is-this-a-child-dialog-of-this-dialog

Returns a truth-value indicating whether or not a dialog is a child dialog of the other.

## Synopsis

uil-is-this-a-child-dialog-of-this-dialog  
(*child*: item-or-value, *parent*: item-or-value)  
-> *result*: truth-value

Argument	Description
<i>child</i>	The dialog being checked as the child dialog of parent
<i>parent</i>	The dialog being checked as the parent dialog of child

Return Value	Description
<u><i>result</i></u>	If child is a child dialog of parent, then TRUE, else FALSE.

# uil-is-this-dialog-managed-for-this-item

Returns a truth-value indicating whether or not the specified item is the initiating object for the specified dialog.

## Synopsis

uil-is-this-dialog-managed-for-this-item  
(*dialog*: item-or-value, *itm*: item-or-value)  
-> result: truth-value

Argument	Description
<i>dialog</i>	The dialog to check against the item as the initiating object.
<i>itm</i>	The item to check as the initiating object for the dialog.

Return Value	Description
<u>result</u>	If the specified item is the initiating object for the specified dialog, then TRUE, else FALSE.

# uil-is-this-the-parent-dialog

Returns a truth-value indicating whether or not a dialog is the parent dialog of another.

## Synopsis

uil-is-this-the-parent-dialog

(*child*: item-or-value, *parent*: item-or-value)

-> result: truth-value

Argument	Description
<i>child</i>	The dialog to check to see if it is a child dialog of parent
<i>parent</i>	The dialog to check to see if it is a parent dialog of child

Return Value	Description
<u>result</u>	If child is a child dialog of parent, then TRUE, else FALSE.

# uil-locate-first-edit-box-on-dialog

Returns the first edit box on the subworkspace of the specified dialog. Edit boxes are ordered from top-left to bottom-right.

## Synopsis

uil-locate-first-edit-box-on-dialog

(*dialog*: class uil-dialog)

-> {*edit-box*: class uil-edit-box | no-edit-box-found: symbol}

Argument	Description
<i>dialog</i>	The dialog for which to return the first edit box

Return Value	Description
<u><i>edit-box</i></u>	An edit box or the symbol no-edit-box-found.

## uil-locate-next-edit-box-on-dialog

Locates the next edit box in a dialog subworkspace. Edit boxes are ordered from top-left to bottom-right.

### Synopsis

uil-locate-next-edit-box-on-dialog

(*edit-box*: class uil-edit-box, *dialog*: class uil-dialog, *window*: class g2-window)  
 -> {*edit-box*: class uil-edit-box | no-edit-box-found: symbol}

Argument	Description
<i>edit-box</i>	Edit box to use for determining the next edit box.
<i>dialog</i>	The dialog for which to return the next edit box.
<i>window</i>	Not used. Specify the symbol none.

Return Value	Description
<u><i>edit-box</i></u>	An edit box or the symbol no-edit-box-found.

# uil-locate-next-rows-edit-box-on-dialog

Returns the first (leftmost) edit box in the next row on the dialog.

## Synopsis

uil-locate-next-rows-edit-box-on-dialog

(*box*: class uil-edit-box, *dialog*: class uil-dialog, *window*: class g2-window)

-> *next-box*: class uil-edit-box

Argument	Description
<i>box</i>	An edit box.
<i>dialog</i>	The dialog on which the edit boxes are displayed.
<i>window</i>	Not used. Specify the symbol none.

Return Value	Description
<u><i>next-box</i></u>	The leftmost edit box in the row immediately below the row that contains <i>box</i> .

## Description

uil-locate-next-rows-edit-box-on-dialog returns the first (leftmost) edit box in the row immediately below the row that contains the edit box specified by *box*.



# uil-move-grobject-to-tab-page-layer

Moves an item to a specified tab page.

## Synopsis

uil-move-grobject-to-tab-page-layer  
(*item*: class item, *tab-page*: class uil-tab-page)

Argument	Description
<i>item</i>	The item to be moved to the tab page.
<i>tab-page</i>	The tab page to which the item is moved.

## Description

uil-move-grobject-to-tab-page-layer() moves the item specified by *item* to the tab page specified by *tab-page*.

# uil-override-initiating-object-for-dialog

Designates an object as the initiating object of a dialog. This object becomes the-object-initiating-dialog-activity-for the given dialog.

## Synopsis

uil-override-initiating-object-for-dialog  
(*obj*: class item, *dialog*: class uil-dialog)

Argument	Description
<i>obj</i>	The object to establish as the initiating object.
<i>dialog</i>	The dialog to establish the relationship with.

# uil-permanent-copies-exist-in-transient-dialog-pool

Checks the dialog pool for permanent copies of the specified dialog.

## Synopsis

uil-permanent-copies-exist-in-transient-dialog-pool

(*dialog*: class uil-dialog)

-> *found*: truth-value, *copies*: integer)

Argument	Description
<i>dialog</i>	The dialog to check for permanent copies of.

Return Values	Description
<i>found</i>	true or false.
<i>copies</i>	If <i>found</i> is true, <i>copies</i> is the number of copies of the dialog. If <i>found</i> is false, <i>copies</i> is 0.

# uil-post-delay-notification

Displays the delay notification icon while a procedure invocation runs to completion.

## Synopsis

uil-post-delay-notification

(*window*: class g2-window, *proc-invocation*: class procedure-invocation,  
*message*: text)

-> *delay-icon*: class uil-wait- semaphore

Argument	Description
<i>window</i>	The window associated with the g2-session.
<i>proc-invocation</i>	The procedure invocation to post the clock for.
<i>message</i>	An optional message to post with the clock.

Return Values	Description
<i>delay-icon</i>	An instance of the delay notification icon.

## Description

This procedure posts the delay notification icon in the upper left corner of the specified window to indicate that work is in progress for the given procedure invocation.

For information about the delay notification icon, see the *G2 GUIDE User's Guide*.

## Related Procedures

[uil-update-wait-clock-face](#)

[uil-remove-delay-notification-if-any](#)

# uil-post-generic-dialog

Posts a confirmation, message, notification, or query dialog.

## Synopsis

uil-post-generic-dialog

(*dialog-text*: text, *type*: symbol, *size*: symbol, *window*: class g2-window,  
*icon*: symbol)

-> *button-label*: text, *returned-value*: text,  
{*dialog*: class uil-dialog | *dialog-not-available*: symbol}

Argument	Description
<i>dialog-text</i>	The text to post in the dialog.
<i>type</i>	The type of generic dialog to use. Specify confirmation, message, notification, or query.
<i>size</i>	The size of the text. Specify small, medium, or large.
<i>window</i>	The window on which to post the dialog.  <b>Note:</b> This procedure creates an inoperable dialog if <i>window</i> specifies an invalid G2 window. An invalid G2 window is one whose <i>g2-connection-status</i> is <i>not</i> equal to the symbol connected.
<i>icon</i>	The class-name of an icon to display with text. UIL provides the following icons: uil-information-icon, uil-question-icon, uil-warning-icon.
Return Values	Description
<i>button-label</i>	The label of the button selected: Yes, No, OK, or Cancel. Query dialogs return Yes (if the OK button is selected) or No (if the Cancel button is selected.)

<b>Return Values</b>	<b>Description</b>
<u><i>returned-value</i></u>	The value returned by a query dialog, or "" (empty string).
<u><i>dialog</i></u>	The dialog, if type is notification, or the symbol dialog-not-available.

## **Description**

This procedure posts one of GUIDE's built-in generic dialogs (Confirmation, Message, Notification, Query).

If you use this procedure to post a notification dialog, you must also call `uil-remove-notification-dialog` to hide and release the dialog.

# uil-post-notification-dialog

Posts a notification dialog.

## Synopsis

uil-post-notification-dialog

(*dialog-text*: text, *size*: symbol, *window*: class g2-window, *icon*: symbol)

-> *dialog*: item-or-value, *msg*: item-or-value

Argument	Description
<i>dialog-text</i>	The text to post in the dialog.
<i>size</i>	The size of the text. Specify small, medium, or large.
<i>window</i>	The window on which to post the dialog.
<i>icon</i>	The class-name of an icon to display with text. UIL provides the following icons: uil-information-icon, uil-question-icon, uil-warning-icon.

Return Value	Description
<i>dialog</i>	The notification dialog, or the symbol dialog-not-available.
<i>msg</i>	The message in the Notification Dialog.

## Related Procedures

[uil-remove-notification-dialog](#)

[uil-post-generic-dialog](#)

[uil-update-notification-text](#)

# uil-quit-dialog-edit

Cancels the edit session currently active on the specified dialog in the given window, if an edit is in progress.

## Synopsis

uil-quit-dialog-edit

(*dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>dialog</i>	The dialog to cancel edit session on.
<i>window</i>	The window on which the dialog is shown (managed).



## uil-raise-tab-page-to-top

Raises a tab page to the top of the stack without selecting it.

### Synopsis

uil-raise-tab-page-to-top

(*dialog*: class uil-tab-dialog, *tab-page*: class uil-tab-page,  
*window*: item-or-value)

Argument	Description
<i>dialog</i>	The dialog that contains the tab page to be raised.
<i>tab-page</i>	The tab page that this procedure raises.
<i>window</i>	The G2 window where the dialog with the tab page is displayed.

### Description

uil-raise-tab-page-to-top() raises the tab page specified by *tab-page* to the top of the stack without selecting it.

# uil-raise-tab-page-to-top-callback

Called when a user presses a specified button.

## Synopsis

uil-raise-tab-page-to-top-callback

(*button*: class uil-button, *window*: class g2-window, *dialog*: class item)

Argument	Description
<i>button</i>	The button that the user presses to raise the tab page.
<i>window</i>	The G2 window where the dialog with the tab page is displayed.
<i>dialog</i>	The dialog with the tab page.

## Description

uil-raise-tab-page-to-top-callback() is called when a user presses the button specified by *button*. This procedure calls uil-select-tab-page().

If a tab page contains non-UIL objects, you can write a customized callback procedure that raises objects to the top after first calling uil-raise-tab-page-to-top().

# uil-release-dialog

Releases a dialog and its child dialogs and returns them to the dialog pool.

## Synopsis

uil-release-dialog

(*dialog*: class uil-dialog, *window*: class g2-window,  
*delete-related-item*: truth-value)

Argument	Description
<i>dialog</i>	The dialog to release.
<i>window</i>	The window on which the dialog is managed.
<i>delete-related-item</i>	The related item to be deleted, if one exists.

# uil-remove-all-remaining-actions

Removes the pending actions from the dialog's event queue.

## Synopsis

uil-remove-all-remaining-actions  
(*action-queue*: class symbol-list)

Argument	Description
<i>action-queue</i>	The list of pending actions for a particular dialog.

## Related Procedures

[uil-insert-actions-into-dialog-event-queue](#)

[uil-flush-dialog-event-queue](#)

## uil-remove-delay-notification-if-any

Removes any posted delay notification icon for the given window.

### Synopsis

uil-remove-delay-notification-if-any  
(*window*: class g2-window)

Argument	Description
<i>window</i>	The window to remove delay notification for.

### Related Procedures

[uil-post-delay-notification](#)

[uil-update-wait-clock-face](#)

# uil-remove-grobject-from-tab-page

Removes an item from a tab page.

## Synopsis

uil-remove-grobject-from-tab-page

(*item*: class item, *tab-page*: class uil-tab-page)

Argument	Description
<i>item</i>	The item removed from the tab page.
<i>tab-page</i>	The tab page from which the item is removed.

## Description

uil-remove-grobject-from-tab-page() removes the item specified by *item* from the tab page specified by *tab-page*.

# uil-remove-notification-dialog

Hides and releases the specified notification dialog on the given window.

## Synopsis

uil-remove-notification-dialog

(*dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>dialog</i>	The notification dialog.
<i>window</i>	The window on which the dialog is managed.

## Related Procedure

[uil-post-notification-dialog](#)

# uil-remove-tab-page-from-dialog

Removes a tab page from a dialog.

## Synopsis

uil-remove-tab-page-from-dialog

(*tab-page*: class uil-tab-page, *dialog*: class uil-tab-dialog)

Argument	Description
<i>tab-page</i>	The tab page that is removed.
<i>dialog</i>	The dialog from which the tab page is removed.

## Description

uil-remove-tab-page-from-dialog() removes the tab page specified by *tab-page* from the tab dialog specified by *dialog*.



# uil-reserve-dialog

Locates the currently active dialog and returns that dialog.

## Synopsis

### uil-reserve-dialog

(*dialog-id*: text, *check-active-and-pool*: truth-value,  
*client*: class uil-client-item)

-> (*dialog-or-status*: item-or-value, *status*: symbol)

Argument	Description
<i>dialog-id</i>	The id of the dialog to reserve.
<i>check-active-and-pool</i>	<ul style="list-style-type: none"> <li>• If true:           <ul style="list-style-type: none"> <li>- (Case A) If multiple copies per window are not allowed, UIL looks for an active dialog and returns that dialog.</li> <li>- (Case B) UIL looks in the dialog pool for an unmanaged copy and returns the copy.</li> <li>- (Case C) UIL creates a copy of the dialog and returns the copy.</li> </ul> </li> <li>• (Case D) If <b>false</b>, UIL creates a copy of the dialog and returns the copy.</li> </ul>
<i>client</i>	The current window or client.

Return Value	Description
<u>dialog-or-status</u>	If a dialog is found, the dialog is returned. If no dialog is found, the symbol DIALOG-RESERVATION-IN-PROGRESS is returned, which occurs in Case A above, while the dialog is still in use.
<u>status</u>	Returns EXISTING (Case A), ORIGINAL (Case B), NEW (Case C or D), or UNABLE-TO-RESERVE-DIALOG.  If the master dialog cannot be found, returns NO-MASTER-DIALOG and the following message is signalled as an error:  "No master dialog with the id <ID> could be located"

## Description

If no currently active dialog exists, this procedure creates a copy of the dialog and returns that.

## uil-restore-dialog-grobj-list

Iterates over all objects on the subworkspace of a specified dialog and puts them into the item list referenced by the attribute uil-dialog-grobj-list.

### Synopsis

uil-restore-dialog-grobj-list  
 (*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog whose uil-dialog-grobj-list attribute is restored.

### Description

The uil-restore-dialog-grobj-list procedure iterates over all objects on the subworkspace of a specified dialog and puts them into the item list referenced by the attribute uil-dialog-grobj-list.

Edit boxes are listed first, in the order in which they appear on the dialog. All other objects follow in no particular order. Selection buttons are represented by their selection boxes.

# uil-select-tab-page

Selects a tab page and raises it to the top of the stack.

## Synopsis

uil-select-tab-page

(*dialog*: class uil-tab-dialog, *tab-page*: class uil-tab-page,  
*window*: item-or-value)

Argument	Description
<i>dialog</i>	The dialog that contains the tab page to be selected.
<i>tab-page</i>	The tab page that this procedure selects.
<i>window</i>	The G2 window where the dialog that contains the tab page is displayed.

## Description

uil-select-tab-page() raises the tab page specified by *tab-page* to the top of the stack in the dialog specified by uil-tab-dialog. For *window*, specify a G2 window or the symbol no window.

# uil-set-destination-object-for-dialog

Specifies the destination object for a dialog.

## Synopsis

uil-set-destination-object-for-dialog  
 (*obj*: class item, *dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>obj</i>	The object to establish as the destination object.
<i>dialog</i>	The dialog to establish the relationship for.
<i>window</i>	The G2 window on which the dialog is managed.

## Description

This procedure establishes the relationship between the specified object and dialog as the destination object. UIL objects on the subworkspace of the dialog can conclude their value(s) to the destination object.

# uil-show-managed-dialog

Displays the specified dialog on the given window at the specified x, y position.

## Synopsis

uil-show-managed-dialog

(*dialog*: class uil-dialog, *window*: class g2-window,  
*x-position*: integer, *y-position*: integer)

Argument	Description
<i>dialog</i>	The dialog to display.
<i>window</i>	The window on which to display the dialog.
<i>x-position</i>	The x-position on the window at which to display the dialog.
<i>y-position</i>	The y-position on the window at which to display dialog.

# uil-shrink-wrap-dialog

Shrink wraps the subworkspace of a dialog to contain its items, but maintains the margins of the dialog's border.

## Synopsis

uil-shrink-wrap-dialog  
(*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog on which to perform the shrink-wrap operation.

# uil-simulate-play-mode

Simulates running a dialog in a user mode.

## Synopsis

uil-simulate-play-mode  
(*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog to simulate play mode for.

## Description

This procedure enforces restrictions on the specified dialog so that the menus are not accessible. Selecting any item implies select.



# uil-start-dialog

Displays a specified dialog on a specified window. If a copy of the specified dialog does not exist in the dialog bin, a copy is generated and displayed.

## Synopsis

uil-start-dialog

(*dialog-id*: text, *initiating-item*: item-or-value, *window*: class g2-window)

-> {*dialog*: class uil-dialog | dialog-not-found: symbol}, *button*: class uil-button

Argument	Description
<i>dialog-id</i>	The ID of the dialog to display.
<i>initiating-item</i>	The item responsible for launching dialog (or the symbol none).
<i>window</i>	The window on which to manage the dialog.

Return Value	Description
<i>dialog</i>	The dialog, or the symbol dialog-not-found.
<i>button</i>	The button that was activated on the dialog, or the symbol none.

## Description

Note that the only action that returns a dialog to its launching procedure is uil-return-dialog. If uil-return-dialog returns a dialog to its launching procedure, the procedure must call uil-release-dialog to return the dialog to the dialog bin when it finishes processing the dialog.

## Related Procedures

[uil-release-dialog](#)

[uil-start-or-refocus-dialog](#)

[uil-start-dialog-processing.](#)

[uil-control-dialog-callback.](#)

# uil-start-dialog-processing

Displays or refocuses the dialog whose ID matches the specified ID on the given window.

## Synopsis

uil-start-dialog-processing

(*dialog-id*: text, *initiating-item*: item-or-value, *window*: class g2-window, *start-actions*: item-or-value, *refocus*: truth-value, *conclude-values*: truth-value, *refocus-actions*: item-or-value, *prompt-for-conclude*: truth-value)

-> {*dialog*: class uil-dialog | dialog-not-found: symbol}, *button*: class uil-button

Argument	Description
<i>dialog-id</i>	The ID of the dialog to display or refocus.
<i>initiating-item</i>	The item to bind as initiating-object (or the symbol none).
<i>window</i>	The window on which to manage the dialog, or to check for if already managed.
<i>start-actions</i>	Specifies action(s) to run if dialog is started. One of: a symbol naming an existing action, the name of a uil-action-description-array, or the symbol none.
<i>refocus</i>	If true, attempt to refocus dialog. If false, display a new copy.
<i>conclude-values</i>	If true, run the conclude method before refocusing dialog.
<i>refocus-actions</i>	Specifies action(s) to run if dialog is refocused. One of: a symbol naming an existing action, the name of a uil-action-description-array, or the symbol none.
<i>prompt-for-conclude</i>	If true, prompt user to run conclude method before refocusing dialog, or not to run conclude.

Return Value	Description
<i>dialog</i>	The dialog, or the symbol dialog-not-found.
<i>button</i>	The button that was activated on the dialog, or the symbol none.

---

**Note** The only dialog action that returns the dialog to the launching procedure is `uil-return-dialog`. If you use `uil-return-dialog` to return a dialog to its launching procedure, you must call `uil-release-dialog` to return the dialog to the dialog bin when the launching procedure finishes processing the dialog.

---

## Related Procedures

[uil-control-dialog-callback](#)

[uil-start-dialog](#)

[uil-start-or-refocus-dialog](#)

# uil-start-or-refocus-dialog

Updates or starts a dialog.

## Synopsis

uil-start-or-refocus-dialog

(*dialog-id*: text, *initiating-item*: item-or-value, *window*: class g2-window)

-> {*dialog*: class uil-dialog | dialog-not-found: symbol}, *button*: class uil-button

Argument	Description
<i>dialog-id</i>	The ID of the dialog to display.
<i>initiating-item</i>	The item responsible for launching the dialog (or the symbol none).
<i>window</i>	The window on which to manage the dialog.

Return Value	Description
<i>dialog</i>	The dialog, or the symbol dialog-not-found.
<i>button</i>	The button that was activated on the dialog, or the symbol none.

## Description

If the specified dialog is managed, this procedure updates the dialog to display information associated with the new *initiating-item*. If the dialog is not managed, this procedure searches the dialog bin for a copy to launch. If the dialog bin does not contain such a dialog, a copy is generated from the master copy of the dialog, and this copy is launched.

---

**Note** The only dialog action that returns the dialog to the launching procedure is `uil-return-dialog`. If you use `uil-return-dialog` to return a dialog to its launching procedure, you must call `uil-release-dialog` to return the dialog to the dialog bin when the launching procedure finishes processing the dialog.

---

This procedure terminates all other open dialogs.

## Related Procedure

[uil-release-dialog](#)

## uil-store-dialog-in-home-bin

Returns and stores the dialog in the dialog pool for the module in which it resides.

### Synopsis

uil-store-dialog-in-home-bin  
(*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog to store in its home bin.

# uil-translate-dialog

Performs the translation of text in a dialog.

## Synopsis

uil-translate-dialog

(*dialog*: class uil-dialog, *window*: class g2-window,  
*gfr-text-resource*: class gfr-text-resource-group)

Argument	Description
<i>dialog</i>	The dialog to be translated.
<i>window</i>	The window on which the dialog to be translated is displayed.
<i>gfr-text-resource</i>	The gfr-text-resource-group associated with the gfr-local-text-resource objects that contain the natural language text for the dialog.

## uil-unsimulate-play-mode

Releases play-mode restrictions on a dialog, and places the specified dialog in the dialog bin associated with the module that it resides in.

### Synopsis

uil-unsimulate-play-mode  
(*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog to unsimulate play mode for.

### Related Procedure

[uil-simulate-play-mode](#)

# uil-update-controls-for-specific-attribute

Updates the UIL control or controls on a dialog that are updated from a specified source attribute.

## Synopsis

uil-update-controls-for-specific-attribute

(*dialog*: class uil-dialog, *item*: class item, *attribute*: symbol)

Argument	Description
<i>dialog</i>	The dialog on which a UIL control is updated.
<i>item</i>	The item that contains the modified attribute.
<i>attribute</i>	The source attribute of the item that is modified.

## Description

uil-update-controls-for-specific-attribute is useful for updating the UIL control or controls on a dialog that specify the same source attribute, when this attribute is known to have changed value.

Using this procedure to update only the UIL control or controls whose source attribute has changed value is more efficient in this case than updating the entire dialog and its UIL controls.



## uil-update-dialog-for-initiating-object

Updates the values of the UIL controls in the dialog to display values of the specified initiating object.

### Synopsis

uil-update-dialog-for-initiating-object  
(*dialog*: class uil-dialog, *object*: class item)

Argument	Description
<i>dialog</i>	The dialog whose values are updated.
<i>object</i>	The initiating object of the dialog. You must specify the initiating object ( <b>the-object-initiating-dialog-activity-for</b> ) of the dialog.

# uil-update-notification-text

Updates the message text in a currently displayed Notification Dialog.

## Synopsis

uil-update-notification-text

(*dialog*: class uil-dialog, *message*: class uil-grmes, *text*: text)

Argument	Description
<i>dialog</i>	The Notification Dialog containing a graphical message object whose text is updated.
<i>message</i>	The message in the Notification Dialog. You can reference the message returned in the <i>msg</i> argument of uil-post-notification-dialog().
<i>text</i>	The text with which <i>message</i> is to be updated.

## Description

uil-update-notification-text() updates the uil-text that contains the message text on a Notification Dialog. This procedure also resizes the border around the Notification dialog.

uil-update-notification-text() enables you to change the text in a currently displayed Notification Dialog dialog dynamically, to provide users with an ongoing display of information about runtime conditions.

For information about Notification Dialogs, see the *G2 GUIDE User's Guide*.

## uil-update-wait-clock-face

Updates the clock face on the delay notification icon to indicate that time has passed and that work is still in progress.

### Synopsis

uil-update-wait-clock-face  
(*semaphore*: class uil-wait-semaphore)

Argument	Description
<i>semaphore</i>	The delay notification icon to update.

### Related Procedures

[uil-post-delay-notification](#)

[uil-remove-delay-notification-if-any](#)

# uil-utility-create-permanent-copies-for-module

Iterates over all dialogs that reside in a specified module. If the Uil origin attribute is the symbol `master`, a copy is created and placed as a permanent copy into the dialog pool for the given module.

## Synopsis

uil-utility-create-permanent-copies-for-module

(*window*: class g2-window, *ModuleName*: symbol)

Argument	Description
<i>window</i>	The window corresponding to the g2-session.
<i>ModuleName</i>	The name of the module to generate permanent copies of all master dialogs for.

## uil-utility-create-permanent-copy-for-dialog

Creates permanent copies of a specified master dialog and places them in the dialog pool. The Uil origin attribute of each copy is set to permanent-copy.

### Synopsis

uil-utility-create-permanent-copy-for-dialog

(*window*: class g2-window, *dialog-id*: text, *NumCopies*: integer)

Argument	Description
<i>window</i>	The window corresponding to the g2-session.
<i>dialog-id</i>	The ID the master dialog to generate permanent copies of.
<i>NumCopies</i>	The number of permanent copies to generate.

# uil-wait-for-dialog

Launches a dialog and puts it in the wait state until a user selects a button on the dialog.

## Synopsis

uil-wait-for-dialog

(*dialog*: class uil-dialog, *window*: item-or-value)

-> *button-id*: text | "dialog-not-available": text

Argument	Description
<i>dialog</i>	The dialog to launch.
<i>window</i>	The window on which to manage dialog.

Return Value	Description
<u><i>button-id</i></u>	The ID of the button selected, or the text "dialog-not-available".

## Additional Procedures that Support Dialogs

[uil-select-edit-box-on-dialog](#)

[utl-copy-object-attributes](#)

[utl-lift-to-top](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE, but they do not take advantage of the dialog mechanism introduced in GUIDE 4.0 or other features introduced in subsequent versions.

[uil-apply-edit-dialog-callback](#)

uil-clear-transient-dialog-pool-and-permanent-copies

uil-delete-dialog

uil-get-query-dialog-value

uil-manage-and-show-dialog

uil-popdown-and-delete-dialog-callback

uil-popdown-and-hide-managed-dialog-callback

uil-popdown-and-keep-edit-dialog-callback

uil-popdown-dialog-callback

[uil-update-wait-semaphore](#)



# uil-apply-edit-dialog-callback

Concludes the current edited value into the edit-field being edited.

## Synopsis

uil-apply-edit-dialog-callback

(*button*: class uil-button, *window*: class G2-window,  
*dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that was selected on the dialog.
<i>window</i>	The window on which the dialog containing the button is managed.
<i>dialog-or-workspace</i>	The dialog, or the workspace of the dialog, that the button appears on.

# uil-clear-transient-dialog-pool-and-permanent-copies

This procedure is superseded by uil-clear-transient-dialog-pool.

## Synopsis

uil-clear-transient-dialog-pool-and-permanent-copies

(*dialog*: class uil-dialog, *window*: class g2-window, *confirm*: truth-value)

Argument	Description
<i>dialog</i>	The master dialog.
<i>window</i>	The window on which to post a dialog asking users to confirm that they want to delete copies.
<i>confirm</i>	If TRUE, users will be asked to confirm the deletion of permanent copies.

## Description

This procedure searches the transient dialog pool for both transient copies and permanent copies of the specified master dialog.

If this procedure finds permanent copies of the specified master dialog, it returns TRUE to the *confirm* argument and posts a dialog asking users to confirm that they want to delete copies.

## uil-delete-dialog

Deletes the specified dialog and all the objects on its subworkspace. Use instead: uil-delete-grobject-method.

### Synopsis

uil-delete-dialog  
(*dialog*: class uil-dialog)

Argument	Description
<i>dialog</i>	The dialog to delete.

# uil-get-query-dialog-value

Returns the text value of the edit box on the query dialog.

## Synopsis

```
uil-get-query-dialog-value  
  (dialog: class uil-dialog)  
  -> edit-box-text: text | not-found: symbol
```

Argument	Description
<i>dialog</i>	The query dialog.

Return Value	Description
<u><i>edit-box-text</i></u>	The text of the edit box on the query dialog, or the symbol not-found.

# uil-manage-and-show-dialog

Displays the dialog on the window at the specified  $x$ ,  $y$  location. Use instead: uil-show-managed-dialog.

## Synopsis

uil-manage-and-show-dialog

(*dialog*: class uil-dialog, *window*: class g2-window,  
*configuration*: symbol, *x*: integer, *y*: integer)

Argument	Description
<i>dialog</i>	The dialog to manage and show.
<i>window</i>	The window on which to manage the dialog.
<i>configuration</i>	The name of a uil-configuration class for the dialog, or the symbol none.
<i>x</i>	The $x$ position on the window at which to manage the dialog.
<i>y</i>	The $y$ position on the window at which to manage the dialog.

# uil-popdown-and-delete-dialog-callback

Pops down and deletes the dialog specified by `dialog-or-workspace` on the indicated window. This procedure does not work with GUIDE 4.0.

## Synopsis

`uil-popdown-and-delete-dialog-callback`  
(*button*: class uil-button, *window*: class g2-window,  
*dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that was selected on the dialog.
<i>window</i>	The window the dialog containing the button is managed on.
<i>dialog-or-workspace</i>	The dialog, or subworkspace of the dialog, the button appears on.

## uil-popdown-and-hide-managed-dialog-callback

Pops down and unmanages the dialog specified by `dialog-or-workspace` on the indicated window. This procedure does not work with GUIDE 4.0.

### Synopsis

`uil-popdown-and-hide-managed-dialog-callback`  
 (*button*: class uil-button, *window*: class G2-window,  
*dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that was selected on the dialog.
<i>window</i>	The window the dialog containing the button is managed on.
<i>dialog-or-workspace</i>	The dialog, or subworkspace of the dialog, the button appears on.

# uil-popdown-and-keep-edit-dialog-callback

Pops down the dialog specified by `dialog-or-workspace` and keeps the edit session going in the background. This procedure does not work with GUIDE 4.0.

## Synopsis

`uil-popdown-and-keep-edit-dialog-callback`  
(*button*: class uil-button, *window*: class G2-window,  
*dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that was selected on the dialog.
<i>window</i>	The window the dialog containing the button is managed on.
<i>dialog-or-workspace</i>	The dialog, or workspace of the dialog, the button appears on.



## uil-popdown-dialog-callback

Pops down the specified dialog on the given window. This procedure does not work with GUIDE 4.0. See uil-update-wait-clock-face.

### Synopsis

uil-popdown-dialog-callback  
 (*button*: class uil-button, *window*: class G2-window,  
*dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that was selected on the dialog.
<i>window</i>	The window the dialog containing the button is managed on.
<i>dialog-or-workspace</i>	The dialog, or workspace of the dialog, the button appears on.

# uil-update-wait-semaphore

Updates the clock's face to indicate that time has passed and work is still in progress.

## Synopsis

uil-update-wait-semaphore

(*window*: class g2-window, *proc-invocation*: class procedure-invocation)

Argument	Description
<i>window</i>	The window corresponding to the g2-session.
<i>proc-invocation</i>	The procedure-invocation to post the wait semaphore for.

# Dialog Titles

---

*Describes the UIL procedures that operate on dialog titles.*

Introduction **195**

Procedures that Operate on Dialog Titles **201**

    uil-create-and-manage-dialog-title **202**

    uil-create-dialog-title **203**

    uil-create-styled-dialog-title **204**

    uil-justify-dialog-title **205**

    uil-title-button-callback **206**



## Introduction

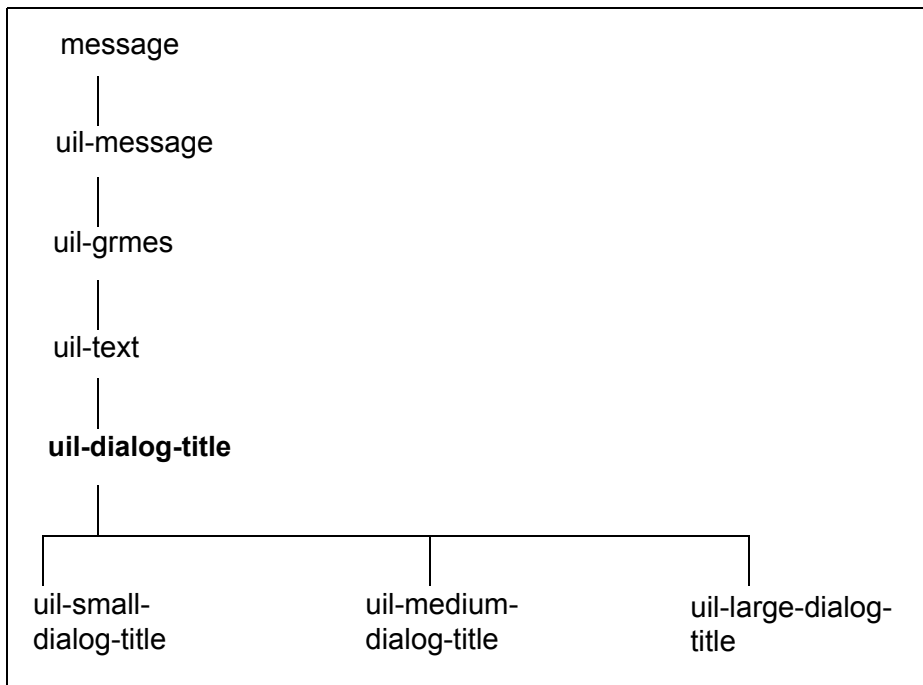
You can add dialog titles to the subworkspaces of dialogs, or to workspaces.

Each dialog can have a single title, which is displayed across the top of the dialog subworkspace. Each dialog title is a graphical message. A dialog can have a title only if a border object is anchored on the subworkspace of the dialog. The dialog title uses the border's corner objects to determine its position.

## Dialog Title Class Hierarchy

All dialog titles inherit from the `uil-dialog-title` class. The following figure illustrates the UIL class hierarchy for dialog titles. `uil-grmes` and all classes below it are message definitions.

Dialog Title Class Hierarchy



## Dialog Title Attributes

The following table summarizes the attributes of dialog titles.

Dialog Title Attributes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this dialog title.
<i>Allowable values:</i>	Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .
<i>Default value:</i>	<code>guide-dialog-gray-title-configuration</code>

## Dialog Title Attributes

Attribute	Description
<b>uil-format-specification</b>	<p>Specifies a format for title text. For information about formats, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i></p> <p><i>Default value:</i> <code>guide-edit-default-format</code></p>
<b>message-contents</b>	<p>Specifies the text of the dialog title. You can edit this text using the Edit Dialog dialog. For information about how to use this dialog, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> Any valid text, enclosed by quotation marks.</p> <p><i>Default value:</i> "Dialog Title"</p>
<b>uil-justification</b>	<p>Specifies how the dialog title is justified across the width of the dialog or subworkspace on which it resides.</p> <p><i>Allowable values:</i> <code>left</code>, <code>right</code>, <code>center</code></p> <p><i>Default value:</i> <code>center</code></p>
<b>uil-title-button-behaviour-enabled</b>	<p>Displays or hides the dialog title hide button, which users can click to hide the dialog subworkspace or workspace that contains the dialog title. For information about dialog title hide buttons, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> <code>true</code> (display dialog title hide button) or <code>false</code> (hide the dialog title hide button)</p> <p><i>Default value:</i> <code>false</code></p>

## Dialog Title Attributes

---

Attribute	Description
<b>uil-title-button-callback</b>	A callback procedure that is invoked when a user clicks the hide button in the dialog title.  <i>Allowable values:</i> Any valid callback method. For information about how to write methods and callbacks, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-title-button-callback

## Internationalization of Dialog Titles

To support internationalization, dialog titles have the attribute **Uil label constant**, which stores a symbol used as a look-up constant that GUIDE/UIIL uses to provide different natural language versions of text in graphical message objects (**uil-grmes**) and buttons (**uil-button**). For example, you can provide versions of text in English, Italian, and Japanese.

For information about how to internationalize dialogs, see the *G2 GUIDE User's Guide*.

---

**Note** GUIDE/UIIL does not support the internationalization of UIL objects that are not on dialog subworkspaces.

---

## Operations Performed on Dialogs Titles

The following table lists the common operations that application developers can perform on dialog titles.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are documented in this chapter.

---

### Operations on Dialog Titles

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-dialog-title ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-gmes-as-symbol	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a

### Operations on Dialog Titles

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-method	uil-update-grobject-method
	Update Wrapper	n/a	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a



## Procedures that Operate on Dialog Titles

This section describes procedures that you can use to create, manage, configure, and justify dialog titles. Dialog titles are objects of the class `uil-small-dialog-title`, `uil-medium-dialog-title`, or `uil-large-dialog-title`.

[uil-create-and-manage-dialog-title](#)

`uil-create-dialog-title`

`uil-create-styled-dialog-title`

`uil-justify-dialog-title`

`uil-title-button-callback`

# uil-create-and-manage-dialog-title

Creates, manages, and configures a dialog title on the dialog or workspace.

## Synopsis

uil-create-and-manage-dialog-title

(*id*: text, *initial-value*: item-or-value, *size*: symbol,

*dialog-or-workspace*: class item)

-> *dialog-title*: uil-dialog-title | no-dialog-title-created: symbol

Argument	Description
<i>id</i>	The ID of the dialog title.
<i>initial-value</i>	The text to use for the title.
<i>size</i>	The size of title to create. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>dialog-or-workspace</i>	The dialog or workspace on which to place the title.

Return Value	Description
<u><i>dialog-title</i></u>	The created dialog title, or the symbol <b>no-dialog-title-created</b> .

# uil-create-dialog-title

Creates and returns a dialog title.

## Synopsis

uil-create-dialog-title

(*id*: text, *initial-value*: item-or-value, *size*: symbol)

-> *dialog-title*: class uil-dialog-title

Argument	Description
<i>id</i>	The ID of the dialog title.
<i>initial-value</i>	The text for the dialog title.
<i>size</i>	Specify small, medium, or large.

Return Value	Description
<u><i>dialog-title</i></u> :	The created dialog title.

# uil-create-styled-dialog-title

Creates a dialog title with the specified size and text.

## Synopsis

uil-create-styled-dialog-title

(*id*: text, *size*: symbol, *initial-value*: item-or-value, *style*: item-or-value)

-> *dialog-title*: class uil-dialog-title

Argument	Description
<i>id</i>	The ID for the dialog title.
<i>size</i>	The size of title to create. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>initial-value</i>	The text to use for the title.
<i>style</i>	The symbol none (place holder for style object).

Return Value	Description
<u><i>dialog-title</i></u> :	The dialog title created.

## uil-justify-dialog-title

Re-justifies the title based on the title's justification attribute value (left, right, or center).

### Synopsis

uil-justify-dialog-title

(*title*: class uil-grmes, *workspace*: class kb-workspace)

Argument	Description
<i>title</i>	The title to justify.
<i>workspace</i>	The workspace on which the title is managed.

# uil-title-button-callback

The default procedure invoked from the `uil-title-button-callback` attribute of a dialog title.

## Synopsis

`uil-title-button-callback`

(*title*: class uil-dialog-title, *window*: class g2-window)

Argument	Description
<i>title</i>	The dialog title bar.
<i>window</i>	The G2 window where the dialog is displayed.

## Description

`uil-title-button-callback()` does the following:

- If the dialog title is on a workspace and there is a **Cancel** button on the workspace, it selects the **Cancel** button. If there is no **Cancel** button on the workspace, it hides the workspace.
- If the dialog title is on a dialog and there is a **Cancel** button on the dialog, it selects the **Cancel** button. If there is no **Cancel** button on the dialog, it inserts the cancel actions (`uil-unsimulate-play-mode`, `uil-hide-dialog`, and `uil-release-dialog`) into the dialog's event queue.

# Edit Boxes, Combo Boxes, and Spin Control Boxes

---

*Describes UIL procedures that operate on edit boxes, combo boxes, and spin control boxes.*

## Introduction **208**

### Procedures that Operate on Edit Boxes **218**

- uil-commit-edit-session-result-to-edit-box **219**
- uil-create-styled-edit-box **220**
- uil-get-control-with-focus **221**
- uil-get-control-with-focus-on-window **222**
- uil-edit-box-selection **223**
- uil-edit-box-selection-method **224**
- uil-edit-box-unselection **225**
- uil-edit-box-unselection-method **226**
- uil-get-edit-box-value **227**
- uil-launch-edit-session **228**
- uil-register-edit-session-results **229**
- uil-select-edit-box **230**
- uil-select-edit-box-on-dialog **231**
- uil-unselect-edit-box **232**
- uil-update-edit-box-label **233**

### Procedures that Operate on Combo Boxes **234**

- uil-add-to-combo-box-list **235**
- uil-clear-combo-box-list **236**
- uil-combo-box-message-selection **237**
- uil-combo-box-no-selection **238**
- uil-combo-box-show-list **239**
- uil-combo-box-unselection **240**
- uil-create-combo-box **241**
- uil-update-combo-box-list **242**

### Procedures that Operate on Spin Control Boxes **243**

- uil-create-spin-control **244**
- uil-delete-spin-control **245**

Additional Procedures that Support Edit Boxes 246

Superseded Procedures 247

    uil-create-edit-box 248

    uil-create-sized-edit-box 249

    uil-delete-edit-box 250



## Introduction

An edit box is an editable text field. The current contents of an edit box is stored in the `message-contents` attribute of the edit box. The formatted contents that users see is stored in the `text` attribute. When a user edits the contents of an edit box, the value of the `message-contents` attribute is brought into the editor.

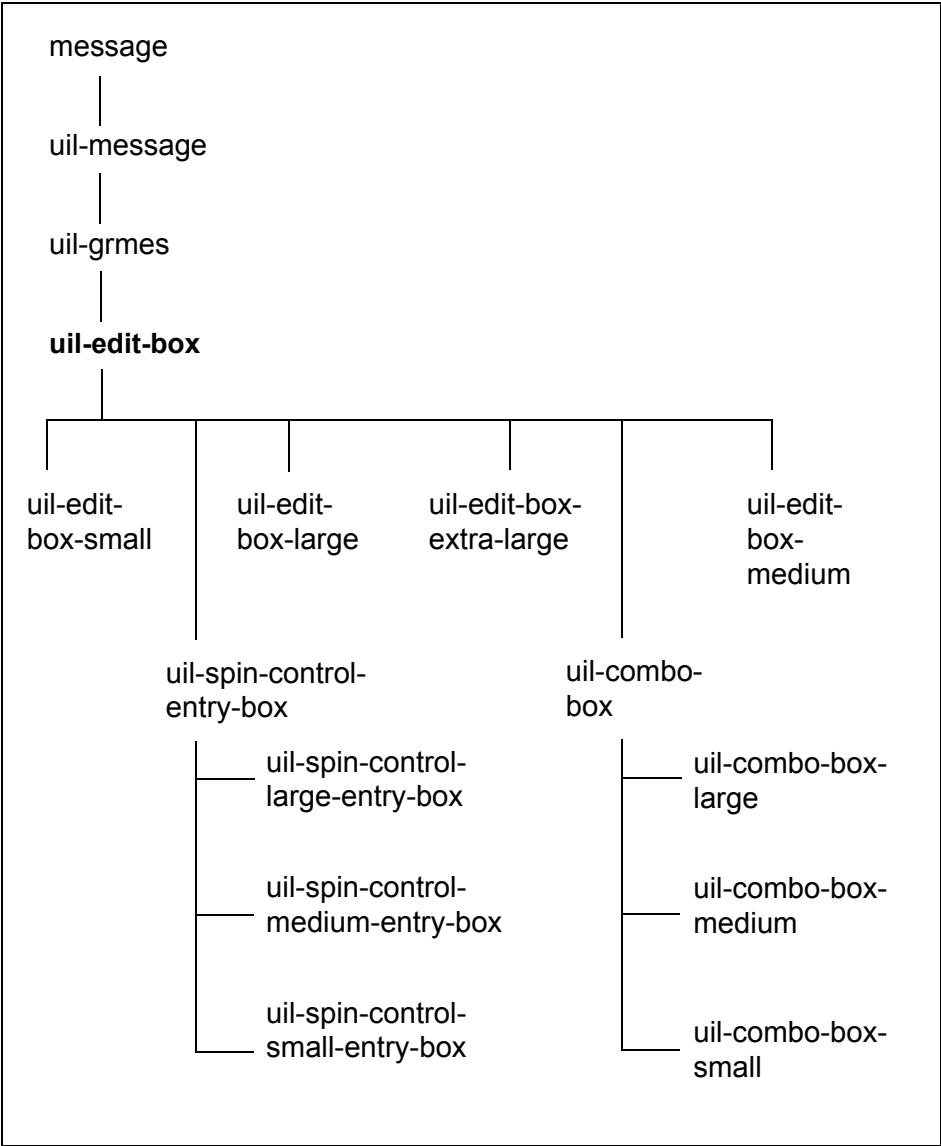
Edit boxes can have labels (which are automatically positioned) and borders.



# Edit Box Class Hierarchy

All edit boxes inherit from the uil-edit-box class. The following figure illustrates the UIL class hierarchy for edit boxes. All classes shown are message-definitions.

Edit Box Class Hierarchy



## Edit Box, Combo Box, and Spin Control Attributes

The following tables list attributes that are unique to or have special meaning for edit boxes, combo boxes, and spin control boxes. See [Appendix B, Common Object Attributes](#), for information about the attributes that edit boxes have in common with other UIL objects.

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this edit box.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> <code>guide-edit-box-configuration</code>
<b>uil-maximum-characters-to-display</b>	Specifies the number of characters to display before clipping the text visible in the edit box. The default value is the symbol unlimited.  <i>Allowable values:</i> 0 or any positive integer  <i>Default value:</i> The symbol unlimited.
<b>uil-edit-style</b>	Refers to a name of an edit style (object of class <code>uil-field-edit-style</code> ).  <i>Allowable values:</i> Name of an existing edit style.  <i>Default value:</i> <code>uil-edit-box-default-field-edit-style</code>
<b>uil-validate-value-immediately</b>	Indicates whether or not the validation method is run on the edit box as soon as the edit session ends.  <i>Allowable values:</i> <code>true</code> (run the validate method as soon as the edit session on the edit box ends) or <code>false</code>  <i>Default value:</i> <code>false</code>

Attribute	Description
<b>uil-allow-field-edit-of-this-message</b>	Indicates whether or not users can navigate to this edit box using return or tab keys.
<i>Allowable values:</i>	<b>true</b> (include the edit box in the group of edit fields that can be navigated by return and tab keys) or <b>false</b>
<i>Default value:</i>	<b>true</b>
<b>uil-clear-text-before-edit</b>	Indicates whether or not the contents of the edit box are removed at the start of an edit session.
<i>Allowable values:</i>	<b>true</b> (clear current contents from the edit box at the start of an edit session) or <b>false</b>
<i>Default value:</i>	<b>false</b>
<b>uil-format-specification</b>	Refers to a name of a format (object of class <code>uil-format-specification</code> ), which specifies how text is formatted when edit session ends.
<i>Allowable values:</i>	Name of an existing format.
<i>Default value:</i>	unspecified  <code>uil-spin-control-default-format-specification</code> (for spin control boxes)
<b>uil-validation-method</b>	Names the procedure used to validate the edit box's contents.
<i>Allowable values:</i>	Name of an existing procedure for handling validation.
<i>Default value:</i>	<code>uil-validate-grobgj-method</code>

<b>Attribute</b>	<b>Description</b>
<b>uil-message-selection-method</b>	Names the procedure called whenever an edit session is initiated on the edit box.  <i>Allowable values:</i> Name of an existing procedure for selecting messages. Any user-defined message selection procedure must call the default method <code>uil-edit-box-selection-method</code> to produce the default selection behavior.  <i>Default value:</i> <code>uil-edit-box-selection-method</code>
<b>uil-message-unselection-method</b>	Names the procedure called whenever an edit session is completed on the edit box. Any user-defined message selection procedure must call the default unselection method to produce the default unselection behavior.  <i>Allowable values:</i> Name of an existing procedure for unselecting messages. Any user-defined message unselection procedure must call <code>uil-edit-box-unselection</code> to produce the default unselection behavior.  <i>Default value:</i> <code>uil-edit-box-unselection-method</code> (for edit boxes and spin control boxes)  <code>uil-combo-box-unselection</code> (for combo boxes)
<b>label</b>	The text used as the label for this edit box.  <i>Allowable values:</i> Any text value.  <i>Default value:</i> "Label:"
<b>message-contents</b>	Contains the current, unformatted value of the edit box.  <i>Allowable values:</i> Any text value.  <i>Default value:</i> ""

Attribute	Description
<b>uil-edit-session-result</b>	Contains the result of the edit session for the edit box.  <i>Allowable values:</i> One of the following: <code>tab-exit</code> , <code>end-exit</code> , <code>cancel-exit</code> , <code>end-action</code> , <code>cancel-action</code> , <code>end-edit-and-do-nothing</code>  <i>Default value:</i> <code>end-edit-and-do-nothing</code>
<b>uil-use-grammar-per-target-specification</b>	 <i>Allowable values:</i> <code>true</code> , <code>false</code> , or <code>none</code>  <i>Default value:</i> <code>none</code>

The following table lists attributes unique to spin control boxes.

Attribute	Description
<b>uil-spin-control-permit-wrapping</b>	Specifies whether the value of the spin control box wraps around when a user attempts to scroll beyond the low value or high value specified for the spin control box. For example, if a user is decrementing the value and scrolls beyond the low value, the first value scrolled to is the high value.  <i>Allowable values:</i> <code>true</code> or <code>false</code>  <i>Default value:</i> <code>false</code> (scrolling stops when a user scrolls to either the low value or the high value)
<b>uil-spin-control-increment</b>	The increment by which the value of the spin control box is increased or decreased when a user scrolls the spin control box.  <i>Allowable values:</i> <code>integer</code>  <i>Default value:</i> <code>1</code>

Attribute	Description
<b>uil-spin-control-hi-value</b>	The highest value that the spin control box can have.  <i>Allowable values:</i> integer  <i>Default value:</i> 99
<b>uil-spin-control-low-value</b>	The lowest value that the spin control box can have.  <i>Allowable values:</i> integer  <i>Default value:</i> 0
<b>uil-format-specification</b>	Specifies the format that determines the color of the spin control box. For information about formats, see the <i>G2 GUIDE User's Guide</i> .  <i>Allowable values:</i> Any valid format.  <i>Default value:</i> uil-spin-control-default-format-specification

## Operations Performed on Edit Boxes

The following table lists the common operations performed on edit boxes by means of procedures.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---

### Operations on Edit Boxes

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method

### Operations on Edit Boxes

	Operation	Procedure	Method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-edit-box ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-gmes-as-symbol	n/a
	Get State	uil-get-edit-box-value ***	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	uil-edit-box-selection ***	uil-edit-box-selection-method ***
✓	Message Unselection	uil-edit-box-unselection ***	uil-edit-box-unselection-method ***
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-select-edit-box ***	n/a

### Operations on Edit Boxes

	Operation	Procedure	Method
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	uil-unselect-edit-box ***	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validation Method	uil-validate-grobj	uil-validate-grobj-method
	Validate Wrapper	uil-call-validate-method-for-grobj	n/a

### Edit Box, Spin Control, and Combo Box Relations

The following table lists the standard relations between edit boxes, spin controls, or combo boxes, and related objects.

#### Relations Between Edit Boxes and Other Objects

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
message	the-current-edit-field-of	uil-dialog	one-to-one	none
message	the-current-uil-message-being-edited-on	g2-window	one-to-one	none
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UII earlier than Version 5.0 and were not converted to have their label text in their icons.



## Combo Box Relations

The following table lists the standard relations between combo boxes and related objects.

**Relations Between Combo Boxes and Other Objects**

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-combo-box	the-combo-box-button-of	uil-combo-box-button	one-to-one	none
uil-combo-box	the-combo-scroll-area-for	uil-scroll-area	one-to-one	none
uil-combo-box	the-combo-list-of	text-list	one-to-one	none

## Spin Control Relations

The following table lists the standard relations between spin controls and related objects.

**Relations Between Spin Controls and Other Objects**

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-spin-button	a-spin-button-controlling	uil-grmes	many-to-one	the-grmes-controlled-by-spin-button
uil-spin-button	the-partner-spin-button-of	uil-spin-button	one-to-one	none

## Procedures that Operate on Edit Boxes

This section describes procedures that you can use to manipulate `uil-edit-box` objects.

- `uil-commit-edit-session-result-to-edit-box`
- `uil-create-styled-edit-box`
- `uil-get-control-with-focus`
- `uil-get-control-with-focus-on-window`
- `uil-edit-box-selection`
- `uil-edit-box-selection-method`
- `uil-edit-box-unselection`
- `uil-edit-box-unselection-method`
- `uil-get-edit-box-value`
- `uil-launch-edit-session`
- `uil-register-edit-session-results`
- `uil-select-edit-box`
- `uil-select-edit-box-on-dialog`
- `uil-unselect-edit-box`
- `uil-update-edit-box-label`

## uil-commit-edit-session-result-to-edit-box

Forces the value just entered during the edit session into the `message-contents` and `text` attributes of the edit box. It returns the edit box that was edited.

### Synopsis

```
uil-commit-edit-session-result-to-edit-box
(edit-mask: class uil-edit-box, dialog: class uil-dialog,
 window: class g2-window, edit-committed: truth-value)
-> {edit-box: class uil-edit-box | edit-box-not-found: symbol}
```

Argument	Description
<i>edit mask</i>	The edit-mask for the edit box whose edit just terminated.
<i>dialog</i>	The dialog on which the edit box (not edit-mask) resides.
<i>window</i>	The g2-window on which the edit session is running.
<i>edit-committed</i>	If <b>true</b> , the edit is to be committed. If <b>false</b> , the edit is canceled.
Return Value	Description
<u><i>edit-box</i></u>	The edit box whose edit-session was just terminated, or the symbol <code>edit-box-not-found</code> .

### Related Procedures

[uil-edit-box-unselection-method](#)

[uil-quit-dialog-edit](#)

[uil-end-dialog-edit](#)

# uil-create-styled-edit-box

Creates and returns an edit box of the specified style.

## Synopsis

uil-create-styled-edit-box

(*id*: text, *size*: symbol, *initial-value*: item-or-value, *label*: text,  
*style*: item-or-value)

-> *edit-box*: class uil-edit-box)

Argument	Description
<i>id</i>	The ID for the edit box.
<i>size</i>	One of small, medium, large, or extra-large.
<i>initial-value</i>	The value to display in the edit box.
<i>label</i>	The text to use as the label for the edit box.
<i>style</i>	The symbol none (place holder for style object).

Return Value	Description
<i>edit-box</i>	The created edit box.

## Description

By default, edit boxes do not have borders. This provides you with the option of creating custom borders for them. Use the uil-create-borders procedure to design borders for the boxes.

# uil-get-control-with-focus

Returns the edit box that has the focus on the specified dialog. (Note that this will not work on workspaces.)

## Synopsis

```
uil-get-control-with-focus
  (dialog: item-or-value)
  -> result: item-or-value
```

Argument	Description
<i>dialog</i>	The dialog to check for focus.

Return Value	Description
<u>result</u>	The edit box that has the focus on the specified dialog. If no edit box has the focus, then the symbol no-item-has-focus is returned.

# uil-get-control-with-focus-on-window

Returns the edit box that has the focus on the specified window.

## Synopsis

uil-get-control-with-focus-on-window

(*window*: item-or-value)

-> result: item-or-value

Argument	Description
<i>window</i>	The window to check for focus.

Return Value	Description
<u>result</u>	The edit box that has the focus on the specified window. If no edit box has the focus, then the symbol no-item-has-focus is returned.

## uil-edit-box-selection

Starts an edit session on the selected edit box.

### Synopsis

uil-edit-box-selection

(*edit-box*: class uil-edit-box, *dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>edit-box</i>	The edit box that was selected.
<i>dialog</i>	The dialog that contains the edit box.
<i>window</i>	The window on which the dialog is managed.

### Description

This procedure is called by the `uil-edit-box-selection-method` in the default case. Users who have specified their own procedure in the `uil-message-selection-method` attribute must call this procedure in order to launch the edit session.

## uil-edit-box-selection-method

Called whenever an edit box is selected by the user.

### Synopsis

uil-edit-box-selection-method

(*edit-box*: class uil-edit-box, *dialog*: class uil-dialog, *window*: class g2-window)

Argument	Description
<i>edit-box</i>	The edit box that was selected.
<i>dialog</i>	The dialog that contains the edit box.
<i>window</i>	The window on which the dialog is managed.

### Description

This procedure calls the procedure named by the `uil-message-selection-method` attribute of the specified edit box. By default, this attribute names the system-defined procedure `uil-edit-box-selection`, which selects the edit box.



## uil-edit-box-unselection

Terminates the edit session for the selected edit box.

### Synopsis

uil-edit-box-unselection

(*edit-box*: class uil-edit-box, *dialog*: class uil-dialog,  
*window*: class g2-window, *edit-committed*: truth-value)

Argument	Description
<i>edit-box</i>	The edit box whose edit session was terminated.
<i>dialog</i>	The dialog that contains the edit box.
<i>window</i>	The window on which the dialog is managed.
<i>edit-committed</i>	<b>true</b> if the edit session should commit the new value. <b>false</b> if the session was canceled.

### Description

This procedure terminates the edit session for the selected edit box. Users who specify their own procedure in the `uil-message-unselection-method` attribute for an edit box must call this procedure in order to end the edit session properly.

This procedure is called by `uil-edit-box-unselection-method`, the default method referenced by the `Uil message unselection method` attribute of the edit box.

# uil-edit-box-unselection-method

Calls the procedure named by the `uil-message-unselection-method` attribute of the specified edit box.

## Synopsis

`uil-edit-box-unselection-method`

(*edit-box*: class uil-edit-box, *dialog*: class uil-dialog,  
*window*: class g2-window, *edit-committed*: truth-value)

Argument	Description
<i>edit-box</i>	The edit box whose edit session was terminated.
<i>dialog</i>	The dialog that contains the edit box.
<i>window</i>	The window on which the dialog is managed.
<i>edit-committed</i>	<b>true</b> if the edit session should commit the new value. <b>false</b> if the session was canceled.

## Description

This procedure is called whenever an edit box loses the focus following an edit-session. It calls the procedure named by the `uil-message-unselection-method` attribute of the specified edit box. The default behavior is to call the procedure `uil-edit-box-unselection`.

# uil-get-edit-box-value

Returns the text of the specified edit box.

## Synopsis

uil-get-edit-box-value

(*dialog-or-workspace*: class item, *id*: text)

-> *contents*: item-or-value

Argument	Description
<i>dialog-or-workspace</i>	The dialog box or workspace containing the edit box.
<i>id</i>	The ID of the edit box.

Return Value	Description
<u><i>contents</i></u>	The value of the message-contents attribute of the edit box.

## uil-launch-edit-session

Launches the appropriate editor for the specified message object, depending on the kind of message object, and whether the message object is on a workspace or a managed dialog.

### Synopsis

uil-launch-edit-session

(*gobj*: class uil-grmes, *window*: class g2-window)

Argument	Description
<i>gobj</i>	The graphical message to edit.
<i>window</i>	The window on which the graphical message is managed.

## uil-register-edit-session-results

Concludes a text value into the edit box that is being edited.

### Synopsis

uil-register-edit-session-results

(*edit-box*: class uil-edit-box, *window*: class g2-window)

Argument	Description
<i>edit-box</i>	The edit box whose edit session is to be registered.
<i>window</i>	The g2-window upon which the edit session is running.

### Description

If there is an active field edit session for the specified edit box, this procedure concludes its current text value into the edit box that is being edited. It does this by calling the procedure `uil-update-text`.

# uil-select-edit-box

Launches the editor for the specified edit box.

## Synopsis

uil-select-edit-box

(*edit-box*: class uil-edit-box, *dialog*: class uil-dialog, *window*: class g2-window)

<b>Argument</b>	<b>Description</b>
<i>edit-box</i>	The edit box to launch the editor for.
<i>dialog</i>	The dialog box containing the edit box.
<i>window</i>	The window on which the edit box is managed.

## uil-select-edit-box-on-dialog

Launches the editor for the specified edit box on a dialog that is managed on the given window. The edit box must be on a dialog.

### Synopsis

uil-select-edit-box-on-dialog

(*edit-box*: class uil-edit-box, *window*: class g2-window)

Argument	Description
<i>edit-box</i>	The edit box to launch the editor for.
<i>window</i>	The window on which the edit box is managed.

# uil-unselect-edit-box

Terminates the specified edit box's edit session.

## Synopsis

uil-unselect-edit-box

(*edit-box*: class edit-box, *dialog*: class uil-dialog,  
*window*: class g2-window, *exit-status*: symbol)

Argument	Description
<i>edit-box</i>	The edit box whose edit session is to be terminated.
<i>dialog</i>	The dialog box containing the edit box.
<i>window</i>	The window on which the dialog containing the edit box is managed.
<i>exit-status</i>	Indicates how the edit session for the edit box should end. Specify <code>tab-exit</code> , <code>end-exit</code> , <code>cancel-exit</code> , <code>end-action</code> , or <code>cancel-action</code> , or <code>end-edit-and-do-nothing</code> (the most commonly used status). You can also specify the symbol <code>none</code> .



## uil-update-edit-box-label

Updates the text object containing the label for the edit box from the `Label` attribute of the edit box.

### Synopsis

uil-update-edit-box-label  
(*edit-box*: class item)

Argument	Description
<i>edit-box</i>	The edit box whose label needs to be updated.

# Procedures that Operate on Combo Boxes

This section describes procedures that operation on uil-combo-box objects.

[uil-add-to-combo-box-list](#)

uil-clear-combo-box-list

uil-combo-box-message-selection

uil-combo-box-no-selection

uil-combo-box-show-list

uil-combo-box-unselection

uil-create-combo-box

uil-update-combo-box-list

## uil-add-to-combo-box-list

Inserts an item at the beginning of the text list of a specified combo box. This procedure is optimal for inserting an individual item. See either the `uil-update-scroll-area-from-list()` procedure or the `uil-update-scroll-area-from-array()` procedure for inserting multiple items.

### Synopsis

`uil-add-to-combo-box-list`

(*combo-box*: class uil-combo-box, *item*: text)

Argument	Description
<i>combo-box</i>	The combo box to which <i>item</i> is to be added.
<i>item</i>	The item to be added to <i>combo-box</i> .

### Description

`uil-add-to-combo-box-list()` adds a text item to the combo box list. This is normally done as part of the initialization of a combo box.

# uil-clear-combo-box-list

Clears all items from the Combo Box list.

## Synopsis

uil-clear-combo-box-list

(*combo-box*: class uil-combo-box)

Argument	Description
<i>combo-box</i>	The combo box whose text list is cleared.

## uil-combo-box-message-selection

Puts the selected scroll area item in the edit box.

### Synopsis

```
uil-combo-box-message-selection
  (message-object: class uil-message-object,
   scroll-area: class uil-scroll-area, window: class g2-window)
```

Argument	Description
<i>message-object</i>	The message object that the user selects.
<i>scroll-area</i>	The scroll area part of the Combo Box.
<i>window</i>	The G2 window where the Combo Box is displayed.

### Description

`uil-combo-box-message-selection()` is the default method referenced by the UIL message selection method attribute of the scroll area part of a Combo Box. When a user selects an item in the scroll area, `uil-combo-box-message-selection()` puts the item in the edit box.

# uil-combo-box-no-selection

A method that causes the contents of the edit box *not* to be added to the Combo Box item list when the edit box loses focus.

## Synopsis

uil-combo-box-no-selection

(*combo-box*: class uil-combo-box, *dialog*: class item,  
*window*: class g2-window)

Argument	Description
<i>combo-box</i>	The Combo Box.
<i>dialog</i>	The dialog or workspace that contains the Combo Box.
<i>window</i>	The G2 window where the dialog or workspace is displayed.

## Description

uil-combo-box-no-selection() can be used as an alternate procedure to reference from the Uil message unselection method attribute of the edit box part of the Combo Box. When this procedure is used, the contents of the edit box are *not* added to the scroll area item list when the edit box loses focus.

The procedure that is referenced from the Uil message unselection method attribute by default, uil-combo-box-unselection(), causes the contents of the edit box to be added to the scroll list. Use uil-combo-box-no-selection() as the procedure referenced from the Uil message unselection method attribute only when you want to change this default behavior of Combo Boxes.

## uil-combo-box-show-list

Displays the items in the Combo Box list

### Synopsis

uil-combo-box-show-list

(*combo-button*: class uil-combo-box-button, *window*: class g2-window,  
*dialog-or-WS*: class item)

Argument	Description
<i>combo-button</i>	The button on the Combo Box whose list is to be displayed.
<i>window</i>	The G2 window where the dialog or workspace is displayed.
<i>dialog-or-WS</i>	The dialog or workspace that contains the Combo Box.

### Description

uil-combo-box-show-list() opens the Combo Box list. It has the same effect as a user clicking the Combo Box button to display the scroll area of items in the Combo Box.

uil-combo-box-show-list() is the procedure that is referenced by default from the Uil handler method attribute of the Combo Box button. You can also invoke this procedure from G2 user code to open the scroll area programmatically.

## uil-combo-box-unselection

Adds the current contents of the edit box part of a Combo Box to the scroll area part when the edit box loses focus.

### Synopsis

uil-combo-box-unselection

(*combo-box*: class uil-combo-box, *dialog*: class item, *window*: class g2-window, *edit-committed* : truth-vlaue)

Argument	Description
<i>combo-box</i>	The combo box that is unselected.
<i>dialog</i>	The dialog containing <i>combo-box</i> .
<i>window</i>	The G2 window where <i>dialog</i> is displayed.
<i>edit-committed</i>	<b>true</b> if the edit session should commit the new value. <b>false</b> if the session was canceled.

### Description

uil-combo-box-unselection() adds the current text of the edit box to the combo box list. This is the default procedure that is referenced from the **Uil message unselection** method attribute of the edit box part of the Combo Box.



# uil-create-combo-box

Creates a Combo Box with a Combo Box button and scroll area.

## Synopsis

uil-create-combo-box)

(*id*: text, *size*: symbol, *initial-value*: text, *style*: symbol)

-> (*box*: class uil-combo-box, *box-button*: class uil-combo-box-button,  
*scroll-area* class uil-scroll-area)

Argument	Description
<i>id</i>	The object ID of the Comb Box created.
<i>size</i>	The size of the Combo Box. Specify one of: <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>initial-value</i>	The initial value of the Combo Box.
<i>style</i>	Not used in this release.

Return Value	Description
<u><i>box</i></u>	The Combo Box created.
<u><i>box-button</i></u>	The Combo Box button that is created as part of the Combo Box.
<u><i>scroll-area</i></u>	The Scroll Area that is created as part of the Combo Box.

## Description

uil-create-combo-box() creates a combo box and returns a handle to each of the separate parts of the combo box.

By default, combo boxes do not have borders. This provides you with the option of creating custom borders for them. Use the uil-create-borders procedure to design borders for the boxes.

## uil-update-combo-box-list

Updates the scroll area part of the Combo Box with the current contents of the Combo Box item list.

### Synopsis

uil-update-combo-box-list  
(*combo-box*: class uil-combo-box

Argument	Description
<i>combo-box</i>	The Combo Box that whose scroll area is updated.

## Procedures that Operate on Spin Control Boxes

This section describes procedures that operation on uil-control-entry-box objects.

uil-create-spin-control  
uil-delete-spin-control

# uil-create-spin-control

Creates a spin control entry box.

uil-create-spin-control

(*id*: text, *size*: symbol, *initial-value*: text, *style*: symbol)

-> *box*: class uil-spin-control-entry-box, *up-button*: class uil-spin-button,

*down-button*: class uil-spin-button

Arguments	Description
<i>id</i>	The desired ID for the new spin control
<i>size</i>	Specify small, medium, or large
<i>initial-value</i>	The value to which the spin control is set initially.
<i>style</i>	currently unused

Return Values	Description
<u><i>box</i></u>	The uil-edit-box part of the spin control
<u><i>up-button</i></u>	The uil-spin-up-button part of the spin control
<u><i>down-button</i></u>	The uil-spin-down part of the spin control

# uil-delete-spin-control

Deletes a spin control entry box.

## Synopsis

uil-delete-spin-control  
(*spin-control*: class uil-spin-control-entry-box)

Arguments	Description
<i>spin-control</i>	The spin control box that this procedure deletes.

## Additional Procedures that Support Edit Boxes

[uil-locate-first-edit-box-on-dialog](#)  
[uil-locate-next-edit-box-on-dialog](#)  
[uil-locate-next-rows-edit-box-on-dialog](#)  
[uil-update-text](#)  
[uil-update-unformatted-text](#)  
[uil-utility-convert-from-edit-box-to-integer](#)  
[uil-utility-convert-from-edit-box-to-quantity](#)  
[uil-utility-convert-from-edit-box-to-symbol](#)  
[uil-utility-convert-from-edit-box-to-value](#)  
[uti-copy-object-attributes](#)  
[uti-lift-to-top](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UIL, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

- uil-create-edit-box
- uil-create-sized-edit-box
- uil-delete-edit-box

# uil-create-edit-box

Creates and returns an edit box. Use instead: uil-create-styled-edit-box.

## Synopsis

uil-create-edit-box

(*id*: text, *dialog-or-workspace*: class item, *initial-value*: item-or-value,  
*x*: integer, *y*: integer)  
-> *edit-box*: class uil-edit-box)

Argument	Description
<i>id</i>	The ID for the edit box.
<i>dialog-or-workspace</i>	The dialog or workspace to place the edit box on.
<i>initial-value</i>	The value to display in the edit box.
<i>x</i>	The x position on the dialog or workspace to position the edit box.
<i>y</i>	The y position on the dialog or workspace to position the edit box.

Return Value	Description
<u><i>edit-box</i></u>	The created edit box.



## uil-create-sized-edit-box

Creates an edit box of a specified size. Use instead: uil-create-styled-edit-box.

### Synopsis

uil-create-sized-edit-box

(*id*: text, *dialog-or-workspace*: class item, *initial-value*: item-or-value,  
*x*: integer, *y*: integer, *size*: symbol)  
 -> edit-box class uil-edit-box)

Argument	Description
<i>id</i>	The ID for the edit box.
<i>dialog-or-workspace</i>	The dialog or workspace to place the edit box on.
<i>initial-value</i>	The value to display in the edit box.
<i>x</i>	The x position on the dialog or workspace to position the edit box.
<i>y</i>	The y position on the dialog or workspace to position the edit box.
<i>size</i>	The size of edit box to create. Specify small, large, or extra-large.
Return Value	Description
<u>edit-box</u>	The created edit box.

## uil-delete-edit-box

Deletes the specified edit box. Use instead: uil-delete-grobj-method.

### Synopsis

uil-delete-edit-box  
(*edit-box*: class uil-edit-box)

Argument	Description
<i>edit-box</i>	The edit box to delete.

# Graphical Messages

---

*Describes procedures that operate on graphical messages.*

Introduction **251**

Procedures that Operate on Graphical Messages **255**

    uil-create-custom-text **256**

    uil-get-message-for-border **257**

    uil-return-size-of-grmes-as-symbol **258**

    uil-update-text **259**

    uil-update-unformatted-text **260**

Additional Procedures that Support Graphical Messages **261**



## Introduction

GUIDE supports the following kinds of graphical messages: text objects, dialog titles, edit boxes, combo boxes, spin control boxes, message objects, and tab pages. All these kinds of messages belong to classes that inherit from the class `uil-grmes`.

---

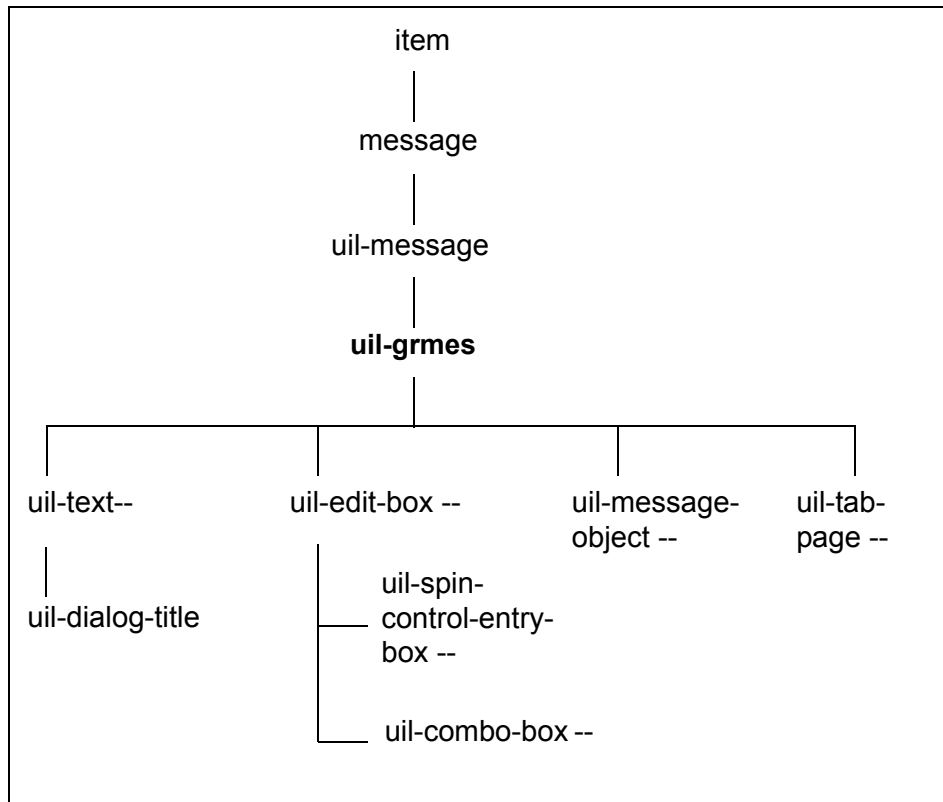
**Note** There are no icons on the GUIDE palette that represents the `message` or `uil-grmes` classes. However, there are icons representing the following classes that inherit from `uil-grmes`: `uil-text`, `uil-dialog-title`, `uil-edit-box`, `uil-spin-control-entry-box`, `uil-combo-box`, `uil-message-object`, and `uil-tab-page`.

---

## Graphical Message Class Hierarchy

The following figure illustrates the UIL class hierarchy for messages. All classes shown are message-definitions. A class whose name is followed by dashes (for example, `uil-text --`) has subclasses that are not shown in the diagram.

Graphical Message Class Hierarchy



## Operations Performed on Graphical Messages

The following table lists the common operations that application developers can perform on graphical messages.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table. Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

(1) Names the procedure called whenever the graphical message object becomes selected.

(2) Names the procedure called whenever the graphical message object becomes unselected.

---

### Operations on Graphical Messages

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-custom-text	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-grmes-as-symbol ***	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a

### Operations on Graphical Messages

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	(1)	n/a
✓	Message Unselection	(2)	n/a
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validation	uil-validate-grobj	uil-validate-grobj-method
	Validate Wrapper	uil-call-validate-method-for-grobj	n/a

## Procedures that Operate on Graphical Messages

This section describes procedures that you can use to manipulate graphical messages. Graphical messages are objects of classes that inherit from the class `uil-grmes`.

- `uil-create-custom-text`
- `uil-get-message-for-border`
- `uil-return-size-of-grmes-as-symbol`
- `uil-update-text`
- `uil-update-unformatted-text`

## uil-create-custom-text

Creates a graphical message object as an instance of a user-defined subclass of uil-grmes.

### Synopsis

uil-create-custom-text

(*class-name*: symbol, *txt*: text)

-> {*message-object*: class uil-grmes | not-found: symbol}

Argument	Description
<i>class-name</i>	The name of the class of message object to create an instance of.
<i>txt</i>	The text for the object's text value.

Return Value	Description
<u><i>message-object</i></u>	The created message object, or the symbol not-found.



## uil-get-message-for-border

Returns the uil-grmes object encompassed by the specified border object.

### Synopsis

```
uil-get-message-for-border
  (border: item-or-value)
  -> result: item-or-value
```

Argument	Description
<i>border</i>	The uil-border object that encompasses either a text or edit box object.

Return Value	Description
<u>result</u>	The uil-text or uil-edit-box that is encompassed by the uil-border object. If no object is found, then the symbol <code>item-not-found</code> is returned.

# uil-return-size-of-grmes-as-symbol

Returns the size of the graphical message.

## Synopsis

uil-return-size-of-grmes-as-symbol

(*txt*: class uil-grmes)

-> size: symbol

Argument	Description
<i>txt</i>	The uil-grmes object to return the size of.

Return Value	Description
<u>size</u>	The size of the graphical message. Possible values: small, medium, large, or extra-large.

## uil-update-text

Updates the specified graphical message object with the given text value.

### Synopsis

uil-update-text

(*message*: class uil-grmes, *input-value*: value)

Argument	Description
<i>message</i>	The graphical message object to update the text of.
<i>input-value</i>	The new text value.

## uil-update-unformatted-text

Updates the specified graphical message object with the given value without running the formatting and clipping procedures.

### Synopsis

uil-update-unformatted-text  
(*message*: class uil-grmes, *input-value*: value)

Argument	Description
<i>message</i>	The graphical message object to update the text of.
<i>input-value</i>	The new text value.

# Additional Procedures that Support Graphical Messages

[uil-launch-edit-session](#)  
[utl-copy-object-attributes](#)  
[utl-lift-to-top](#)



# Graphical Objects

---

*Describes procedures that operate on graphical objects belonging to subclasses of the class `uil-grobj`.*

Introduction **264**

Procedures that Operate on Graphical Objects **268**

`uil-call-conclude-method-for-grobj` **269**

`uil-call-update-method-for-grobj` **270**

`uil-call-validate-method-for-grobj` **271**

`uil-change-size-of-grobj` **272**

`uil-change-window-style-of-grobj` **273**

`uil-clone-grobj` **274**

`uil-clone-grobj-method` **275**

`uil-conclude-grobj` **276**

`uil-conclude-grobj-method` **277**

`uil-configure-grobj` **278**

`uil-configure-grobj-method` **279**

`uil-delete-grobj` **280**

`uil-delete-grobj-method` **281**

`uil-delete-grobj-with-confirmation` **282**

`uil-disable-grobj` **283**

`uil-disable-grobj-method` **284**

`uil-drop-border-around-grobj` **285**

`uil-enable-grobj` **286**

`uil-enable-grobj-method` **287**

`uil-initialize-grobj` **288**

`uil-initialize-grobj-method` **289**

`uil-make-grobj-permanent` **290**

`uil-manage-grobj` **291**

`uil-manage-grobj-method` **292**

`uil-move-grobj` **293**

`uil-set-slider-value` **294**

`uil-size-of-grobj` **295**

`uil-size-of-grobj-method` **296**

`uil-unmanage-grobj` **297**

uil-update-grobject 298  
uil-update-grobject-method 299  
uil-utility-restore-grobject 300  
uil-validate-grobject 301  
uil-validate-grobject-method 302

Procedures that Operate on Sliders 303

uil-create-slider 304  
uil-get-slider-value 305  
uil-rotate-slider 306

Additional Procedures that Support Graphical Objects 307



## Introduction

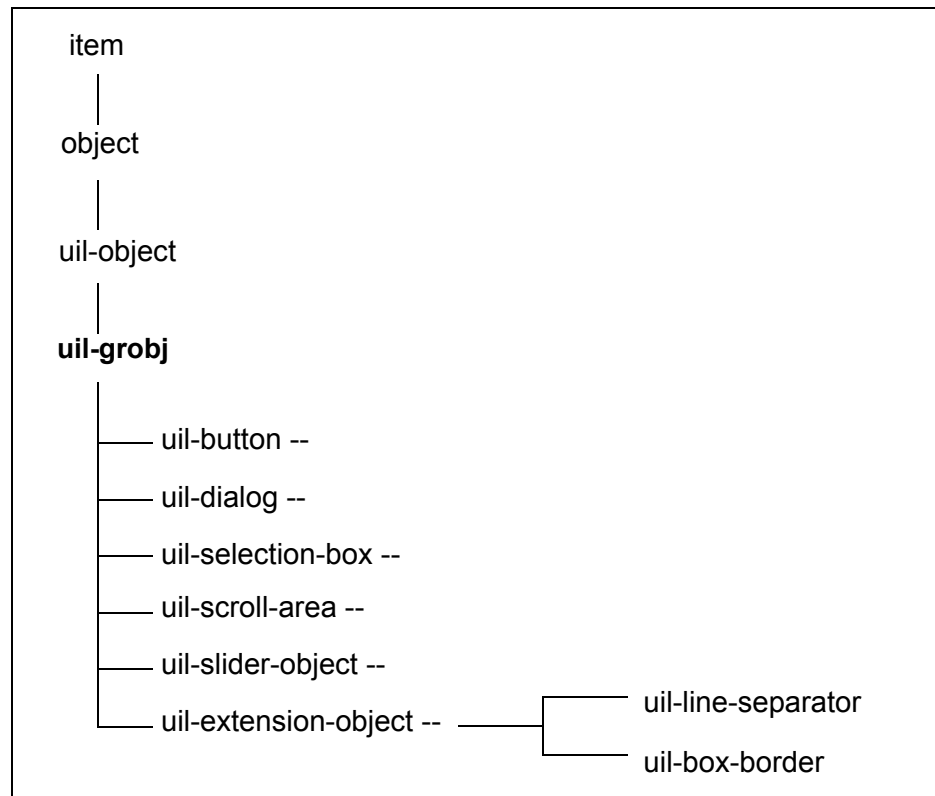
The `uil-grobject` class is the parent class of all classes of graphical objects – dialogs, buttons, edit boxes, scroll areas, border, separators, sliders, and so on. Thus, every graphical object inherits the attributes of the `uil-grobject` class.



## Graphical Object Class Hierarchy

The following figure illustrates the UIL class hierarchy for graphical objects. All classes shown are object-definitions. A class whose name is followed by dashes (for example, `uil-button --`) has subclasses that are not shown in the figure.

Graphical Object Class Hierarchy



## Operations Performed on Graphical Objects

The following table lists the common operations that application developers can perform on graphical objects.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---

## Operations on Graphical Objects

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj ***	n/a
	Change Style	uil-change-window-style- of-grobj ***	n/a
✓	Clone	uil-clone-grobj ***	uil-clone-grobj-method ***
✓	Conclude	uil-conclude-grobj ***	uil-conclude-grobj-method ***
	Conclude Wrapper	uil-call-conclude-method- for-grobj ***	n/a
✓	Configure	uil-configure-grobj ***	uil-configure-grobj-method ***
	Create	n/a	n/a
✓	Delete	uil-delete-grobj ***	uil-delete-grobj-method ***
✓	Disable	uil-disable-grobj ***	uil-disable-grobj-method ***
✓	Enable	uil-enable-grobj ***	uil-enable-grobj-method ***
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj ***	uil-initialize-grobj-method ***
	Make Permanent	uil-make-grobj-permanent ***	n/a
✓	Manage	uil-manage-grobj ***	uil-manage-grobj-method ***
✓	Message Selection	n/a	n/a

### Operations on Graphical Objects

	Operation	Procedure	Method
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj ***	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj ***	uil-size-of-grobj-method ***
	Unmanage	uil-unmanage-grobj ***	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobj ***	uil-update-grobj-method ***
	Update Wrapper	uil-call-update-method-for-grobj ***	n/a
✓	Validation	uil-validate-grobj ***	uil-validate-grobj-method ***
	Validate Wrapper	uil-call-validate-method-for-grobj ***	n/a

# Procedures that Operate on Graphical Objects

This section describes procedures that you can use to manipulate any uil-grobj objects.

- uil-call-conclude-method-for-grobj
- uil-call-update-method-for-grobj
- uil-call-validate-method-for-grobj
- uil-change-size-of-grobj
- uil-change-window-style-of-grobj
- uil-clone-grobj
- uil-clone-grobj-method
- uil-conclude-grobj
- uil-conclude-grobj-method
- uil-configure-grobj
- uil-configure-grobj-method
- uil-delete-grobj
- uil-delete-grobj-method
- uil-delete-grobj-with-confirmation
- uil-disable-grobj
- uil-disable-grobj-method
- uil-drop-border-around-grobj
- uil-enable-grobj
- uil-enable-grobj-method
- uil-initialize-grobj
- uil-initialize-grobj-method
- uil-make-grobj-permanent
- uil-manage-grobj
- uil-manage-grobj-method
- uil-move-grobj
- uil-set-slider-value
- uil-size-of-grobj
- uil-size-of-grobj-method
- uil-unmanage-grobj
- uil-update-grobj
- uil-update-grobj-method
- uil-utility-restore-grobj
- uil-validate-grobj
- uil-validate-grobj-method

## uil-call-conclude-method-for-grobj

Calls uil-conclude-grobj-method, filling in the remaining three arguments.

### Synopsis

uil-call-conclude-method-for-grobj  
(*obj*: class item, *window*: class g2-window)

Argument	Description
<i>obj</i>	The object whose value is to be concluded.
<i>window</i>	The window on which the dialog or workspace containing the object appears.

### Related Procedures

[uil-conclude-grobj-method](#)

[uil-conclude-grobj](#)

# uil-call-update-method-for-grobj

Calls uil-update-grobj-method, filling in the remaining three arguments.

## Synopsis

uil-call-update-method-for-grobj  
(*grobj*: class item, *window*: class g2-window)

Argument	Description
<i>grobj</i>	The item to be updated.
<i>window</i>	The window on which the dialog or workspace containing the object appears.

## Related Procedures

[uil-update-grobj-method](#)

[uil-update-grobj](#)

[uil-update-dialog-for-initiating-object](#)

## uil-call-validate-method-for-grobj

Calls uil-validate-grobj-method, filling in the remaining three arguments.

### Synopsis

```
uil-call-validate-method-for-grobj
  (grobj: class item, window: class g2-window)
  -> validation: text
```

Argument	Description
<i>grobj</i>	The item to be validated.
<i>window</i>	The window on which the dialog or workspace containing the object appears.
Return Value	Description
<u>validation</u>	OK, if the validation is successful. Otherwise, an error message explaining why the validation failed.

### Related Procedures

[uil-validate-grobj-method](#)

[uil-validate-grobj](#)

# uil-change-size-of-grobj

Changes the size of the specified object to a specified size.

## Synopsis

uil-change-size-of-grobj

(*grobj*: class uil-object, *size*: symbol, *workspace*: class kb-workspace)

-> *object*: class uil-object

Argument	Description
<i>grobj</i>	The graphical object whose size is to be changed.
<i>size</i>	The size to which the object is changed. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>workspace</i>	The workspace containing the graphical object.

Return Value	Description
<u><i>object</i></u>	The resized graphical object.



## uil-change-window-style-of-grobj

Changes the window style of an object of the class uil-button to resemble Motif or Windows.

### Synopsis

```
uil-change-window-style-of-grobj
(grobj: class uil-grobj, style: symbol)
-> button: class uil-button)
```

Argument	Description
<i>grobj</i>	The graphical object whose window style is to be changed.
<i>style</i>	The window size and style. For a Motif style window, specify <b>small</b> , <b>medium</b> , or <b>large</b> . For a Windows style window, specify <b>small-msw</b> , <b>medium-msw</b> , or <b>large-msw</b> .
Return Value	Description
<u><i>button</i></u>	The re-styled button.

### Description

Currently, this procedure supports only UIL objects whose superior class is uil-button.

# uil-clone-grobj

Clones and returns the specified graphical object and all its attributes.

## Synopsis

uil-clone-grobj

(*grobj*: class item)

-> {clone: item-or-value | unable-to-clone: symbol}

Argument	Description
<i>grobj</i>	The object to clone.

Return Value	Description
<u>clone</u>	The cloned item, or the symbol <code>unable-to-clone</code> .

## uil-clone-grobj-method

Clones and returns the specified graphical object and all its attributes. The object is cloned by the procedure named in the object's `uil-clone-method` attribute. The default method named by this attribute calls `uil-clone-grobj`.

### Synopsis

`uil-clone-grobj-method`

(*grobj*: class item)

-> {*clone*: item-or-value | unable-to-clone: symbol}

Argument	Description
<i>grobj</i>	The item to be cloned.

Return Value	Description
<i>clone</i>	The cloned item, or the symbol <code>unable-to-clone</code> .

# uil-conclude-grobj

Concludes a graphical object's value into the target attribute of the graphical object's target object.

## Synopsis

`uil-conclude-grobj` (*grobj*: class item, *window*: class g2 window, *dlg-or-wksp*: class item, *button*: item-or-value, *action-queue*: item-or-value)

Argument	Description
<i>grobj</i>	The item to be concluded.
<i>window</i>	The window on which the dialog box containing the item is managed.
<i>dlg-or-wksp</i>	The dialog box or workspace containing the item.
<i>button</i>	The button initiating the conclude action (optional).
<i>action-queue</i>	The list of pending actions for the specified dialog initiated by button (optional).

## Description

If the graphical object *grobj* is a toggle button, radio button, or check button, the value concluded is the on value or off value of the button — depending on whether the button is currently selected or unselected. These values are specified in the `On value` and `Off value` attributes of the button.

If the graphical object is an edit box, message object, or scroll area, the value concluded is the content of the graphical object.

For more information about how graphical objects conclude their values, see the *G2 GUIDE User's Guide*.

---

**Note** For objects residing on workspaces (not dialog subworkspaces), the conclude method works only if the target object is referenced by name.

---

## uil-conclude-grobj-method

Concludes a graphical object's value into the attribute of the target object specified by the graphical object's Uil event target attribute attribute.

### Synopsis

uil-conclude-grobj-method (*grobj*: class item, *window*: class g2-window, *dlg-or-wksp*: class item, *button*: item-or-value, *action-queue*: item-or-value)

Argument	Description
<i>grobj</i>	The item to be concluded.
<i>window</i>	The window on which the dialog containing the item is managed.
<i>dlg-or-wksp</i>	The dialog box or workspace containing the item.
<i>button</i>	The button initiating the conclude action (optional).
<i>action-queue</i>	The list of pending actions for the specified dialog initiated by button (optional).

### Description

uil-conclude-grobj-method runs the procedure referenced by the Uil conclude method attribute.

If the graphical object *grobj* is a toggle button, radio button, or check button, the value concluded is the on value or off value of the button — depending on whether the button is currently selected or unselected. These values are specified in the On value and Off value attributes of the button.

If the graphical object is an edit box, message object, or scroll area, the value concluded is the content of the graphical object.

---

**Note** For objects residing on workspaces (not dialog subworkspaces), the conclude method works only if the source object is referenced by name.

---

# uil-configure-grobj

Calls the appropriate configuration method for the graphical object.

## Synopsis

uil-configure-grobj  
(*grobj*: class item)

Argument	Description
<i>grobj</i>	The item to be configured.

## Description

The trashcan is not in the UIL hierarchy of items that uil-configure-grobj handles. As a result, if you put a trashcan on a tab page, when the tab page is hidden, the trashcan regions are made transparent, but when the tab page is brought to the top, no coloring action takes place and the trashcan remains transparent (invisible).

## uil-configure-grobj-method

Calls the appropriate configuration method for the specified graphical object. The object is configured by the procedure named in the object's `uil-configuration-method` attribute.

### Synopsis

uil-configure-grobj-method  
(*grobj*: class item)

Argument	Description
<i>grobj</i>	The item to be configured.

# uil-delete-grobj

Deletes the specified item (and its label and border, if they exist).

## Synopsis

uil-delete-grobj  
(*grobj*: class item)

Argument	Description
<i>grobj</i>	The item to be deleted.



## uil-delete-grobject-method

Deletes the specified item (and its label and border, if they exist). The object is deleted by the procedure named in the object's `uil-delete-method` attribute.

### Synopsis

`uil-delete-grobject-method`  
(*grobject*: class item)

Argument	Description
<i>grobject</i>	The item to be deleted.

# uil-delete-grobject-with-confirmation

Places a border around the object targeted for deletion and displays a message dialog asking if the item should be deleted. Deletes the object if the user clicks on Yes.

## Synopsis

uil-delete-grobject-with-confirmation

(*grobject*: class item, *window*: class g2-window)

-> *confirmation*: symbol

Argument	Description
<i>grobject</i>	The item to be deleted.
<i>window</i>	The window on which the item appears.

Return Value	Description
<u><i>confirmation</i></u>	One of the following symbols: unable-to-delete-border, unable-to-delete-configuration-class, yes, no, or cancel.

## uil-disable-grobj

Disables the graphical object and configures it to display its disabled state.

### Synopsis

uil-disable-grobj  
(*grobj*: class item)

Argument	Description
<i>grobj</i>	The item to be disabled.

## uil-disable-grobject-method

Calls the appropriate disable method for the specified graphical object. The object is disabled by the procedure named in the object's `uil-disable-method` attribute.

### Synopsis

`uil-disable-grobject-method`  
(*grobject*: class item)

Argument	Description
<i>grobject</i>	The item to be disabled.

# uil-drop-border-around-grobj

Creates a border and drops it around the extents of the specified object.

## Synopsis

uil-drop-border-around-grobj

(*grobj*: class item, *type*: symbol, *configuration*: class uil-configuration-class)

-> *border*: class uil-box-border

Argument	Description
<i>grobj</i>	The item to drop a border around.
<i>type</i>	The type of border to create. Specify one of the following border types: <b>wide</b> , <b>medium</b> , <b>narrow</b> , <b>thin</b> , <b>sunken-wide</b> , <b>sunken-medium</b> , <b>sunken-narrow</b> , or <b>sunken-thin</b> .
<i>configuration</i>	The configuration for the border.
Return Value	Description
<u><i>border</i></u>	The border that was created.

# uil-enable-grobj

Enables the graphical object and configures it to display its enabled state.

## Synopsis

uil-enable-grobj  
(*grobj*: class item)

Argument	Description
<i>grobj</i>	The item to be enabled.

## uil-enable-grobject-method

Calls the appropriate enable method for the specified graphical object. The object is enabled by the procedure named in the object's `uil-enable-method` attribute.

### Synopsis

`uil-enable-grobject-method`  
(*grobject*: class item)

Argument	Description
<i>grobject</i>	The item to be enabled.

## uil-initialize-grobj

Returns an object to its initial state.

### Synopsis

uil-initialize-grobj  
(*grobj*: item)

Argument	Description
<i>grobj</i>	The item to be initialized.

### Description

The initial state is defined differently for each class of object. For example, a push button is returned to its unselected state, a text object is updated from its `message-contents` attribute as though just cloned from the G2 GUIDE palette, and any current edit session on an edit box is cancelled.



## uil-initialize-grobj-method

Initializes the specified item by calling the procedure named by the object's uil-initialization-method attribute.

### Synopsis

uil-initialize-grobj-method(*grobj*: class item)

Argument	Description
<i>grobj</i>	The item to be initialized.

# uil-make-grobject-permanent

Makes the specified graphical object permanent.

## Synopsis

uil-make-grobject-permanent(*grobject*: class item)

Argument	Description
<i>grobject</i>	The item to be made permanent.

## Description

uil-make-grobject-permanent() makes a UIL control permanent, so that it remains in your KB after you reset G2, or save and reload your KB.

---

**Caution** Do not use the G2 action make permanent to make UIL controls permanent. Always use uil-make-grobject-permanent().

---

# uil-manage-grobj

Displays the graphical object at the specified coordinates.

## Synopsis

uil-manage-grobj (*grobj*: class item, *wksp-or-dlg*: item-or-value,  
*configuration*: symbol, *x*: integer, *y*: integer)  
 -> *graphical-object*: class item

Argument	Description
<i>grobj</i>	The item to be managed.
<i>wksp-or-dlg</i>	The workspace or dialog on which the item appears.
<i>configuration</i>	The configuration for the item.
<i>x</i>	The x coordinate at which the item is to be managed.
<i>y</i>	The y coordinate at which the item is to be managed.
Return Value	Description
<u><i>graphical-object</i></u>	The item that was managed.

# uil-manage-grobj-method

Displays the graphical object at the specified coordinates, by calling the procedure named by the Uil manage method attribute of the object.

## Synopsis

uil-manage-grobj-method (*grobj*: class item, *wksp*: item-or-value, *configuration*: symbol, *x*: integer, *y*: integer)  
-> *graphical-object*: class item

<b>Argument</b>	<b>Description</b>
<i>grobj</i>	The item to be managed.
<i>wksp</i>	The workspace on which the item appears.
<i>configuration</i>	The configuration for the item.
<i>x</i>	The x coordinate at which the item is to be managed.
<i>y</i>	The y coordinate at which the item is to be managed.

<b>Return Value</b>	<b>Description</b>
<u><i>graphical-object</i></u>	The item that was managed.

# uil-move-grobj

Moves an item to a new location, also checking label positioning and module consistency.

## Synopsis

uil-move-grobj  
(*grobj*: class item)  
-> *graphical-object*: item

Argument	Description
<i>grobj</i>	The item to be moved.

Return Value	Description
<u><i>graphical-object</i></u>	The object moved.

# uil-set-slider-value

Sets the value of a slider.

## Synopsis

uil-set-slider-value

(*dialog-or-workspace*: class item, *id*: text, *value*: quantity,  
*window*: class g2-window)

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace that contains the slider.
<i>id</i>	The ID of the slider.
<i>value</i>	The value to which the slider is set by this procedure.
<i>window</i>	The window where the slider appears.

## uil-size-of-grobj

Returns the width and height, in pixels, of the graphical object.

### Synopsis

uil-size-of-grobj

(*grobj*: class item)

-> *width*: integer, *height*: integer

Argument	Description
<i>grobj</i>	The item to return the size of.

Return Value	Description
<u><i>width</i></u> , <u><i>height</i></u>	The width and the height of the item, in pixels.

## uil-size-of-grobj-method

Returns the width and height, in pixels, of the graphical object by calling the procedure named by the graphical object's `uil-size-of-method` attribute.

### Synopsis

`uil-size-of-grobj-method`

(*grobj*: class item)

-> *width*: integer, *height*: integer

Argument	Description
<i>grobj</i>	The item to return the size of.

Return Value	Description
<i>width</i> , <i>height</i>	The width and the height of the item, in pixels.



## uil-unmanage-grobj

Removes the graphical object from the specified workspace but does not delete it.

### Synopsis

uil-unmanage-grobj

(*grobj*: class item, *drawable*: item-or-value)

Argument	Description
<i>grobj</i>	The item to be unmanaged.
<i>drawable</i>	The workspace on which the item appears.

# uil-update-grobj

Updates a graphical object using the value in the source attribute of the graphical object's source object.

## Synopsis

uil-update-grobj

(*grobj*: class item, *window*: class g2-window, *dlg-or-wksp*: class item, *button*: item-or-value, *action-queue*: item-or value)

Argument	Description
<i>grobj</i>	The object on which to run the update method.
<i>window</i>	The window on which the dialog or workspace containing the object is managed.
<i>dlg-or-wksp</i>	The dialog box or workspace containing the object.
<i>button</i>	The button that initiated the update action for the object (optional).
<i>action-queue</i>	The list of pending actions for the specified dialog initiated by button (optional).

## Description

The procedure updates the graphical object *grobj* using the value specified by the object's *uil-event-source-object* and *uil-event-source-attribute* attributes.

For information about how graphical objects are updated, see the *G2 GUIDE User's Guide*.

---

**Note** For objects residing on workspaces, the update method works only if the source object is referenced by name.

---

## Related Procedures

[uil-call-update-method-for-grobj](#)

[uil-update-grobj-method](#)

# uil-update-grobj-method

Updates the object using the value of object's source object and source attribute.

## Synopsis

uil-update-grobj-method

(*grobj*: class item, *window*: class g2-window, *dlg-or-wksp*: class item, *button*: item-or-value, *action-queue*: item-or-value)

Argument	Description
<i>grobj</i>	The object on which to run the update method.
<i>window</i>	The window on which the dialog or workspace containing the object is managed.
<i>dlg-or-wksp</i>	The dialog box or workspace containing the object.
<i>button</i>	The button that initiated the update action for the object (optional).
<i>action-queue</i>	The list of pending actions for the specified dialog initiated by button (optional).

## Description

This procedure updates the object's state using the value of the object's source attribute, as specified by the object's `uil-event-source-object` and `uil-event-source-attribute` attributes. The object is updated by the procedure named in the object's `uil-update-method` attribute.

---

**Note** For objects residing on workspaces, the update method works only if the source object is referenced by name.

---

## Related Procedures

[uil-call-update-method-for-grobj](#)

[uil-update-grobj](#)

## uil-utility-restore-grobj

Calls the corresponding restore utility for the class of object, if one exists. The utility iterates over all instances of the specified class.

### Synopsis

uil-utility-restore-grobj  
(*grobj-type*: symbol)

Argument	Description
<i>grobj-type</i>	The symbolic name of the class of object to run a utility restore procedure on.

## uil-validate-grobj

Validates the contents of the graphical object according to the format referred to by the object's `uil-format-specification` attribute. For information about the use of formats for validation, see the *G2 GUIDE User's Guide*.

### Synopsis

uil-validate-grobj

(*grobj*: class item, *window*: class g2-window, *dlg-or-wksp*: class item, *button*: item-or-value, *action-queue*: item-or value)

-> *validation*: text

Argument	Description
<i>grobj</i>	The object on which to run validation method.
<i>window</i>	The window on which the dialog or workspace containing the object is managed.
<i>dlg-or-wksp</i>	The dialog box or workspace containing the object.
<i>button</i>	The button that initiated the validation action for the object (optional).
<i>action-queue</i>	The list of pending actions for the specified dialog initiated by button (optional).
Return Value	Description
<u><i>validation</i></u>	OK, or an error message explaining why the validation failed.

# uil-validate-grobject-method

Validates the contents of the graphical object according to the `uil-format-specification-class` referred to by the object's `uil-format-specification` attribute. The object is validated by the procedure named in the object's `uil-validation-method` attribute.

## Synopsis

`uil-validate-grobject-method`

(*grobject*: class item, *window*: class g2-window, *dlg-or-wksp*: class item,  
*button*: item-or-value, *action-queue*: item-or-value)

-> *validation*: text

Argument	Description
<i>grobject</i>	The object on which to run the validation method.
<i>window</i>	The window on which the dialog or workspace containing the object is managed.
<i>dlg-or-wksp</i>	The dialog box or workspace containing the object.
<i>button</i>	The button that initiated the validation action for the object (optional).
<i>action-queue</i>	The list of pending actions for the specified dialog initiated by button (optional).
Return Value	Description
<u><i>validation</i></u>	OK, if the validation is successful. Otherwise, an error message explaining why the validation failed.

## Procedures that Operate on Sliders

This section describes UIL procedures that operate only on sliders.

- uil-create-slider
- uil-get-slider-value
- uil-rotate-slider

# uil-create-slider

Creates a slider.

## Synopsis

uil-create-slider

(*slider-class*: symbol, *low*: quantity, *high*: quantity)

-> *slider-or-none*: item-or-value

Argument	Description
<i>slider-class</i>	The class of slider to create. Specify: uil-slider, uil-slider-regular, uil-slider-thin, uil-slider-special, uil-slider-vertical, uil-slider-vertical-with-scale, or any user-defined slider subclass.
<i>low</i>	The smallest value that the slider can contain.
<i>high</i>	The largest value that the slider can contain.
Return Value	Description
<u><i>slider-or-none</i></u>	The slider created by this procedure, or the symbol none.



## uil-get-slider-value

Returns the current value of the slider.

### Synopsis

uil-get-slider-value

(*dialog-or-workspace*: class item, *id*: text, *window*: class g2-window)

-> *value*: quantity

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace that contains the slider.
<i>id</i>	The ID of the slider.
<i>window</i>	The window on which the slider appears.

Return Value	Description
<u><i>value</i></u>	The current value of the slider.

# uil-rotate-slider

Rotates a slider clockwise by 90 degrees.

## Synopsis

uil-rotate-slider

(*slider*: class uil-slider)

Argument	Description
<i>slider</i>	The slider to be rotated.

## Additional Procedures that Support Graphical Objects

[uil-add-grobj-to-dialog](#)  
[uil-get-grobj-from-id](#)  
[uil-get-grobj-from-id-on-dlg-or-wksp](#)  
[uil-restore-dialog-grobj-list](#)  
[uti-copy-object-attributes](#)  
[uti-lift-to-top](#)



# Message Objects

---

*Describes procedures that operate on message objects belonging to subclasses of the class `uil-message-object`.*

Introduction **310**

Procedures that Operate on Message Objects **317**

- `uil-add-message-to-list` **318**
- `uil-add-message-to-list-at-position` **319**
- `uil-append-item-to-message` **321**
- `uil-clear-messages` **322**
- `uil-copy-message-object-system-attributes` **323**
- `uil-copy-messages` **324**
- `uil-create-message` **325**
- `uil-create-message-with-appended-item` **327**
- `uil-delete-selected-messages` **328**
- `uil-display-message-list` **329**
- `uil-does-message-belong-to-scroll-area` **330**
- `uil-find-excess-message-and-remove` **331**
- `uil-generate-time-stamp-for-message` **332**
- `uil-get-first-message` **333**
- `uil-get-index-of-scroll-message` **334**
- `uil-get-item-appended-to-message` **335**
- `uil-get-last-message` **336**
- `uil-get-message-appended-to-this-item` **337**
- `uil-get-message-from-index` **338**
- `uil-get-message-object-state` **339**
- `uil-get-scroll-area-for-this-message` **340**
- `uil-get-selected-message` **341**
- `uil-get-selected-message-objects` **342**
- `uil-insert-message-in-ordered-list` **343**
- `uil-is-message-selected` **344**
- `uil-message-selection` **345**
- `uil-message-selection-method` **346**
- `uil-message-unselection` **347**
- `uil-message-unselection-method` **348**

- uil-move-message 349
- uil-move-messages 350
- uil-on-double-click 351
- uil-on-single-click 352
- uil-remove-message 353
- uil-remove-message-from-list 354
- uil-select-message 355
- uil-select-message-on-window 356
- uil-sort-scroll-area-messages 357
- uil-unselect-message 358
- uil-update-message-list-with-message 359

Additional Procedures that Operate on Message Objects 360



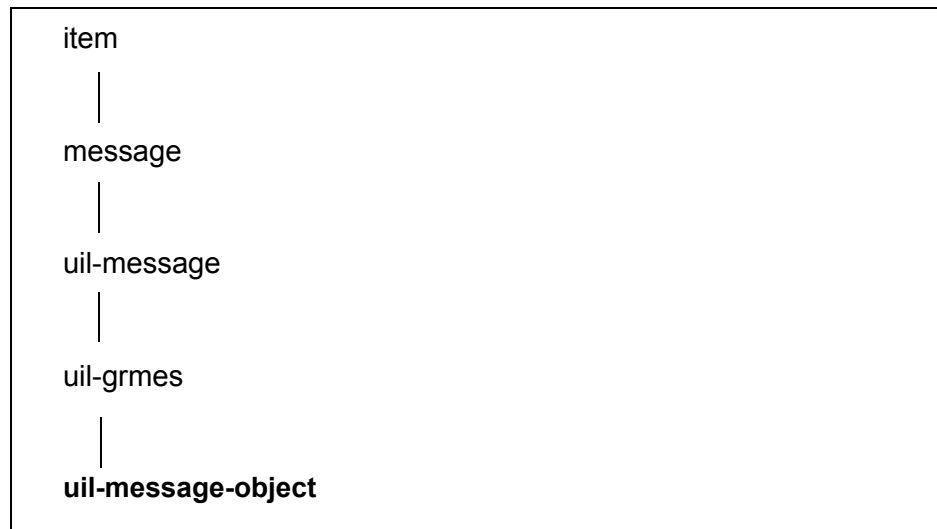
## Introduction

A scroll area manages a group of message objects. The scroll area can contain more message objects than it can display at one time. To see message objects that are not currently displayed, users can scroll the message objects up or down to bring the hidden message objects into view.

### Message Object Class Hierarchy

The following figure illustrates the UIL class hierarchy for message objects. All classes below it are message-definitions. The class `uil-message-object` has subclasses that are not shown in the diagram.

## Message Object Class Hierarchy



## Message Object Attributes

The following tables list attributes that are unique to message objects. See [Appendix B, Common Object Attributes](#) for information about the attributes that message objects have in common with other objects.

### Message Object Attributes

Attribute	Description
<b>configuration</b>	If you want to display this message object in a non-standard color, you can specify a customized configuration here to provide that color.
<i>Allowable values:</i>	Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .
<i>Default value:</i>	The configuration of the scroll area that contains this message object.

## Message Object Attributes

Attribute	Description
<b>uil-configuration-method</b>	Specifies the method that applies the configuration identified in the Configuration attribute.  <i>Allowable values:</i> Any valid configuration method.  <i>Default value:</i> uil-configure-scroll-message-method
<b>message-display-priority</b>	An integer value specifying the priority of message. Messages can be ordered in terms of priority.  <i>Allowable values:</i> 0 or any positive integer  <i>Default value:</i> 1
<b>uil-message-selection-method</b>	Names the procedure called when the message is selected.  <i>Allowable values:</i> Any valid procedure name  <i>Default value:</i> uil-message-selection-method
<b>uil-message-unselection-method</b>	Names the procedure called when the message becomes unselected.  <i>Allowable values:</i> Any valid procedure name  <i>Default value:</i> uil-message-unselection-method
<b>uil-format-specification</b>	Refers to a name of a uil-format-specification-class.  <i>Allowable values:</i> Name of an existing format (object of class uil-format-specification)  <i>Default value:</i> unspecified



## Message Object Attributes

Attribute	Description
<b>uil-edit-style</b>	Refers to a name of an edit style  <i>Allowable values:</i> Name of an existing edit style (object of class uil-field-edit-style)  <i>Default value:</i> uil-message-object-default-field-edit-style
<b>message-contents</b>	The current, unformatted value of the message object.  <i>Allowable values:</i> Any text value  <i>Default value:</i> " "
<b>uil-time-stamp</b>	The current subsecond real time at which the message was created. Used for the ordering methods, chronological and logbook.  <i>Allowable values:</i> not applicable  <i>Default value:</i> The subsecond real time when the message object was created. Example: 8.642e8  <i>Note:</i> Beginning in GUIDE/UIIL Version 4.1, Uil time stamp values are stored as integers that represent the creation times of messages, in seconds. In earlier versions of GUIDE/UIIL, Uil time stamp values were stored as text. If you need to see time stamps in the earlier format, you can convert them using the UIL procedure g2-unix-time-at-start().

## Operations Performed on Message Objects

The following table lists the common operations that application developers can perform on message objects.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations on Message Objects

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-message***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-grmes-as-symbol	n/a
	Get State	uil-get-message-object-state***	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a

### Operations on Message Objects

	Operation	Procedure	Method
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
✓	Message Selection	uil-message-selection***	uil-message-selection-method***
✓	Message Unselection	uil-message-unselection***	uil-message-unselection-method***
	Move	uil-move-grobject	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-select-message***	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	uil-unselect-message***	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Scroll Area and Message Object Relations

The following tables list the standard relations between scroll areas and message objects. You can reference these relations to access scroll areas and message objects programmatically.

**Scroll Area and Message Object Relations**

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-message-object	the-topmost-message-of	uil-scroll-area	one-to-one	none
uil-message-object	the-bottommost-message-of	uil-scroll-area	one-to-one	none
uil-message-object	a-message-belonging-to	uil-scroll-area	many-to-one	the-handler-for-the-message-of
uil-scroll-area	the-handler-for-the-message-of	uil-message-object	one-to-many	a-message-belonging-to
uil-message-object	a-selected-message-of	uil-scroll-area	many-to-one	none
item	the-item-appended-to-the-scroll-message	uil-message-object	one-to-one	the-uil-message-appended-to-the-item
uil-message-object	the-uil-message-appended-to-the-item	item	one-to-one	the-item-appended-to-the-scroll-message

## Procedures that Operate on Message Objects

This section describes procedures that you can use to manipulate `uil-message-object` objects.

### [uil-add-message-to-list](#)

- `uil-add-message-to-list-at-position`
- `uil-append-item-to-message`
- `uil-clear-messages`
- `uil-copy-message-object-system-attributes`
- `uil-copy-messages`
- `uil-create-message`
- `uil-create-message-with-appended-item`
- `uil-delete-selected-messages`
- `uil-display-message-list`
- `uil-does-message-belong-to-scroll-area`
- `uil-find-excess-message-and-remove`
- `uil-generate-time-stamp-for-message`
- `uil-get-first-message`
- `uil-get-index-of-scroll-message`
- `uil-get-item-appended-to-message`
- `uil-get-last-message`
- `uil-get-message-appended-to-this-item`
- `uil-get-message-from-index`
- `uil-get-message-object-state`
- `uil-get-scroll-area-for-this-message`
- `uil-get-selected-message`
- `uil-get-selected-message-objects`
- `uil-insert-message-in-ordered-list`
- `uil-is-message-selected`
- `uil-message-selection`
- `uil-message-selection-method`
- `uil-message-unselection`
- `uil-message-unselection-method`
- `uil-move-message`
- `uil-move-messages`
- `uil-on-double-click`
- `uil-on-single-click`
- `uil-remove-message`
- `uil-remove-message-from-list`
- `uil-select-message`
- `uil-select-message-on-window`
- `uil-sort-scroll-area-messages`
- `uil-unselect-message`
- `uil-update-message-list-with-message`

# uil-add-message-to-list

Adds the specified message object to the given scroll area's message-list.

## Synopsis

uil-add-message-to-list (*scroll-area*: class uil-scroll-area,  
*msg-object*: class uil-message-object)  
-> {*message-object*: class uil-message-object | none: symbol}, *status*: symbol

Argument	Description
<i>scroll-area</i>	The scroll area to which the message is added.
<i>msg-object</i>	The message object to add to the scroll area.

Return Values	Description
<u><i>message-object</i></u>	The message object added, if the message object is added successfully.  The symbol none, if the message object is not added successfully.
<u><i>status</i></u>	message-added, if the message object is added successfully.  message-not-added, if the message object is not added successfully.

## Description

uil-add-message-to-list() does not automatically sort the scroll area, nor does it redisplay the message list. You can call uil-display-message-list() after calling this procedure to update the scroll area display.

## uil-add-message-to-list-at-position

Adds a message object to a scroll area's message-list at a specified position.

### Synopsis

```
uil-add-message-to-list-at-position
(scroll-area: class uil-scroll-area,
 message-object: class uil-message-object, position: integer)
-> message: item-or-value, status: symbol
```

Argument	Description
<i>scroll-area</i>	The scroll area to which the message is added.
<i>msg-object</i>	The message object to add to scroll area.
<i>position</i>	<p>The position in the scroll area message list at which the message object is added. 0 represents the top of the list, 1 the position immediately below that, and so on.</p> <p>Any existing message object at the location specified by <i>position</i> is moved down in the list by one position, as are any other message objects below <i>position</i>.</p> <p>If <i>position</i> is larger than the number of messages in the list, the message is added to the end of the message-list and appears at the bottom of the scroll area.</p>
Return Values	Description
<u><i>message-object</i></u>	<p>The message object added, if the message object is added successfully.</p> <p>The symbol <code>none</code>, if the message object is not added successfully.</p>
<u><i>status</i></u>	<p><code>message-added</code>, if the message object is added successfully.</p> <p><code>message-not-added</code>, if the message object is not added successfully.</p>

## Description

`uil-add-message-to-list-at-position()` is the same as the procedure `uil-add-message-to-list()`, with the addition of a *position* argument that specifies the position in the scroll area message list at which the message object is inserted.

This procedure does not automatically sort the scroll area, nor does it redisplay the message list. You can call `uil-display-message-list()` after calling this procedure to update the scroll area display.

Use of this procedure is recommended for unordered scroll areas. If the scroll area has an ordering method other than `unordered` -- for example, `alphabetic` or `chronological` -- subsequent scroll-area operations that automatically sort lists may put the new message in some place other than where it was inserted with `uil-add-message-to-list-at-position()`.



# uil-append-item-to-message

Appends an item to a message object.

## Synopsis

uil-append-item-to-message

(*item-class*: symbol, *message-object*: item-or-value)

-> *appended-item*: item-or-value

Argument	Description
<i>item-class</i>	The class name of the item appended to the message object.
<i>message-object</i>	The message object to which the item is appended.
Return Values	Description
<u><i>appended-item</i></u>	The item that is appended to the message object. If <i>item-class</i> refers to a class that does not exist, returns the symbol class-does-not-exist.

# uil-clear-messages

Clears the messages displayed in the specified scroll area.

## Synopsis

uil-clear-messages

(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area whose messages are to be cleared.

# uil-copy-message-object-system-attributes

Copies the system attributes from one message object definition to another.

## Synopsis

uil-copy-message-object-system-attributes  
(*message-from*: class message-definition,  
*message-to*: class message-definition)

Argument	Description
<i>message-from</i>	The source message object for copying attributes from.
<i>message-to</i>	The destination message object for copying attributes to.

# uil-copy-messages

Copies the message objects from one scroll area to another.

## Synopsis

uil-copy-messages

(*scroll-area-from*: class uil-scroll-area, *scroll-area-to*: class uil-scroll-area)

Argument	Description
<i>scroll-area-from</i>	The scroll area to copy messages from.
<i>scroll-area-to</i>	The scroll area to copy messages to.

# uil-create-message

Creates a message and adds it to the specified scroll area.

## Synopsis

uil-create-message

```
(scroll-area: class uil-scroll-area, text: value, priority: quantity)
-> {message-object: class uil-message-object | none: symbol},
   scroll-area: class uil-scroll-area
```

Argument	Description
<i>scroll-area</i>	The scroll area to which the scroll message is to be added.
<i>text</i>	The text of the scroll message.
<i>priority</i>	An integer specifying the priority for the message object.
Return Values	Description
<i>message-object</i>	The created message object, or the symbol <code>none</code> if the call to <code>uil-create-message</code> does not succeed in adding the message. <b>Note:</b> In GUIDE 3.0, this return argument is a message object.
<i>scroll-area</i>	The scroll area to which the message was added.

## Description

This procedure returns both the message object and the scroll area.

A message is deleted immediately after being added if 1) `uil-create-message` adds a message to a scroll area that already contains the maximum number of messages allowed (as specified by the `maximum-allowed-messages` attribute of the scroll area) and, 2) in the process of removing excess messages, UIL chooses the new message as the oldest or last message and deletes it accordingly. `uil-create-message` then returns the symbol `none`.

For example, suppose that the scroll area contains two messages, that two is the maximum number allowed, and that the `Ordering` method of the scroll area is `Alphabetic`. The messages contain the values "A" and "B". The message added by

`uil-create-message` contains the value "C". Thus, the message added by `uil-create-message` is the last message, and UIL deletes it.

# uil-create-message-with-appended-item

Appends an item to a message object.

## Synopsis

```
uil-create-message-with-appended-item
(scroll-area: class uil-scroll-area, txt: value, priority: quantity,
 item-class: symbol)
-> message-object: item-or-value, scroll-area: class uil-scroll-area,
   appended-item: item-or-value
```

Argument	Description
<i>scroll-area</i>	The scroll area to which the scroll message is to be added.
<i>txt</i>	The text of the scroll message.
<i>priority</i>	An integer specifying the priority for the message object.
<i>item-class</i>	The class name of the item appended to the message object.

Return Values	Description
<u><i>message-object</i></u>	The created message object, or the symbol <code>none</code> if the call to <code>uil-create-message</code> does not succeed in adding the message.
<u><i>scroll-area</i></u>	The scroll area to which the message was added.
<u><i>appended-item</i></u>	The item appended to the message object. If no item is appended, the symbol <code>class-does-not-exist</code> .

## Description

`uil-create-message-with-appended-item()` is similar to `uil-create-message()`, except for its last argument and last return value, both of which refer to the item appended to the message object. The last argument is a symbol representing the class name of the appended item, and the last return value is either the appended item itself or the symbol `class-does-not-exist`.

# uil-delete-selected-messages

Deletes all the selected messages from the specified scroll area.

## Synopsis

uil-delete-selected-messages  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area from which to delete all selected messages.



## uil-display-message-list

Updates the scroll areas to display its messages starting with the message whose index matches starting-index.

### Synopsis

uil-display-message-list

(*scroll-area*: class uil-scroll-area, *starting-index*: integer)

Argument	Description
<i>scroll-area</i>	The scroll area whose messages are to be displayed.
<i>starting-index</i>	The index for the first message to display.

# uil-does-message-belong-to-scroll-area

Returns a truth-value indicating whether or not the specified message is managed by the specified scroll area.

## Synopsis

uil-does-message-belong-to-scroll-area  
(*message*: item-or-value, *scroll-area*: item-or-value)  
-> result: truth-value

<b>Argument</b>	<b>Description</b>
<i>message</i>	The uil-message-object to check.
<i>scroll-area</i>	The uil-scroll-area to check.

<b>Return Value</b>	<b>Description</b>
<u>result</u>	If the scroll area manages the specified message then TRUE, else FALSE.

## uil-find-excess-message-and-remove

This procedure removes messages in excess of the maximum allowed by the specified scroll area. The messages removed are the oldest or the last, depending on the ordering method of the scroll area.

### Synopsis

uil-find-excess-message-and-remove  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to remove all excess messages from.

# uil-generate-time-stamp-for-message

Generates a time-stamp for the specified message object using the current real time.

## Synopsis

uil-generate-time-stamp-for-message  
(*msg-object*: class uil-message-object)

Argument	Description
<i>msg-object</i>	The message object to generate a time-stamp for.

## Description

This procedure places the time stamp that it generates in the message object's uil-time-stamp attribute. This attribute is used for sorting messages by the logbook and chronological ordering methods.

The procedure uil-create-message calls uil-generate-time-stamp-for-message when it creates messages.

## uil-get-first-message

Returns the first message in the specified scroll area.

### Synopsis

```
uil-get-first-message  
  (scroll-area: item-or-value)  
  -> result: item-or-value
```

Argument	Description
<i>scroll-area</i>	The scroll area from which to return the first message.

Return Value	Description
<u>result</u>	The uil-message-object that is the first message managed by the specified scroll area. If no message is found, then the symbol <code>item-not-found</code> is returned.

# uil-get-index-of-scroll-message

Returns the index of the specified message-object in the given scroll area's members-list.

## Synopsis

uil-get-index-of-scroll-message

(*scroll-area*: class uil-scroll-area, *msg-object*: class uil-message-object)

-> *index*: integer

Argument	Description
<i>scroll-area</i>	The scroll area containing the message.
<i>msg-object</i>	The message to return the index of.

Return Value	Description
<u><i>index</i></u>	The index of the specified message object.

# uil-get-item-appended-to-message

Returns the item appended to the message object.

## Synopsis

uil-get-item-appended-to-this-message

(*message*: item-or-value)

-> *appended-item*: item-or-value

Argument	Description
<i>message</i>	The message object to check for an appended item.

Return Value	Description
<u><i>appended-item</i></u>	The item appended to the message object. If no item is found, the symbol <code>no-item-found</code> is returned.

# uil-get-last-message

Returns the last message in the specified scroll area.

## Synopsis

uil-get-last-message  
(*scroll-area*: item-or-value)  
-> result: item-or-value

Argument	Description
<i>scroll-area</i>	The scroll area from which to return the last message.

Return Value	Description
<u>result</u>	The uil-message-object that is the last message managed by the specified scroll area. If no message is found, then the symbol <code>item-not-found</code> is returned.



## uil-get-message-appended-to-this-item

Returns the message object that is appended to the specified item.

### Synopsis

uil-get-message-appended-to-this-item

(*appended-item*: item-or-value)

-> *message-object*: item-or-value

Argument	Description
<i>appended-item</i>	The item to check for an appended message object.

Return Value	Description
<u><i>message-object</i></u>	The message object appended to item. If no message object is found, the symbol <code>no-message-found</code> is returned.

# uil-get-message-from-index

Returns the message object whose index matches the specified index.

## Synopsis

uil-get-message-from-index

(*scroll-area*: class uil-scroll-area, *Indx*: integer)

-> {*message-object*: class uil-message-object | message-not-found: symbol}

Argument	Description
<i>scroll-area</i>	The scroll area containing the message.
<i>Indx</i>	The index of the message to return.

Return Value	Description
<u><i>message-object</i></u>	The whose index is <i>Indx</i> , or the symbol message-not-found.

## uil-get-message-object-state

Returns the selected state of the specified message object.

### Synopsis

```
uil-get-message-object-state  
  (msg-object: class uil-message-object)  
  -> state: symbol
```

Argument	Description
<i>msg-object</i>	The message object to get the state of.

Return Value	Description
<u>state</u>	The selected state of the message object. One of selected or unselected.

## uil-get-scroll-area-for-this-message

Returns the scroll area that manages the specified message object.

### Synopsis

uil-get-scroll-area-for-this-message

(*message*: item-or-value)

-> result: item-or-value

Argument	Description
<i>message</i>	The uil-message-object from which to return the scroll area.

Return Value	Description
<u>result</u>	The scroll area that manages the specified message object. If no scroll area is found, then the symbol item-not-found is returned.

# uil-get-selected-message

Returns values of all selected messages in scroll area.

## Synopsis

uil-get-selected-message  
 (*dialog-or-workspace*: class item, *id*: text)  
 -> *text-array*: class text-array

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace on which the scroll area containing messages appears.
<i>id</i>	The ID of the scroll area containing messages.
Return Value	Description
<u><i>text-array</i></u>	A text-array of the values of all selected messages.

## Description

This procedure returns a text-array of the values of all the selected messages in the scroll area identified by the given *id* on the dialog or workspace. The calling application should make sure to delete the text-array that is returned.

# uil-get-selected-message-objects

Returns an item-array consisting of all the selected message objects in the scroll area.

## Synopsis

uil-get-selected-message-objects  
(*dialog-or-workspace*: class item, *id*: text)  
-> *result*: class item-array

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace on which the scroll area containing messages appears.
<i>id</i>	The ID of the scroll area containing the message objects.

Return Value	Description
<i>result</i>	An item-array consisting of all the selected messages.

## Description

This procedure returns an item-array consisting of all the selected message objects in the scroll area identified by *id* on the dialog or workspace.

The calling application should make sure to delete the item-array that is returned.

## uil-insert-message-in-ordered-list

Inserts the specified message in the given list.

### Synopsis

uil-insert-message-in-ordered-list

(*msg-object*: class uil-message-object, *msg-list*: class item-list)

Argument	Description
<i>msg-object</i>	The message to insert in list.
<i>msg-list</i>	The list to insert message into.

# uil-is-message-selected

Returns a truth-value indicating whether or not the specified message is selected.

## Synopsis

uil-is-message-selected  
(*message*: item-or-value)  
-> result: truth-value

Argument	Description
<i>message</i>	The message object to check if selected.

Return Value	Description
<u>result</u>	If the specified message object is selected then TRUE, else FALSE.



# uil-message-selection

Selects a message in a scroll area.

## Synopsis

uil-message-selection

(*msg*: class uil-message-object, *scroll-area*: class uil-scroll-area,  
*window*: class g2-window)

Argument	Description
<i>msg</i>	The message that is selected by this call to uil-message-selection.
<i>scroll-area</i>	The scroll area that contains the message.
<i>window</i>	The window on which the scroll area is managed.

## Description

This procedure is called by the default message selection method `uil-message-selection-method`, when a user clicks on a message object. The behavior of `uil-message-selection` depends on the setting of the following attributes of the scroll area that manages the message object:

`allow-multiple-simultaneous-selections`  
`allow-unselect-on-selected-messages`

For example, if a message object is already selected and `allow-multiple-simultaneous-selections` is set to `true`, clicking on an unselected message object selects the message. If `allow-multiple-simultaneous-selections` is set to `false`, clicking on an unselected message object does not select the message. The method `uil-message-unselection-method` is called if a message is to be unselected.

# uil-message-selection-method

Selects a message object whenever a user clicks on the message object.

## Synopsis

uil-message-selection-method

(*msg*: class uil-message-object, *scroll-area*: class uil-scroll-area,  
*window*: class g2-window)

Argument	Description
<i>msg</i>	The message that was selected.
<i>scroll-area</i>	The scroll area that contains the message.
<i>window</i>	The window on which the scroll area is managed.

## Description

This procedure is called whenever the user clicks on a message object within a scroll area. The procedure named by the object's uil-message-selection-method attribute is called. The default value is uil-message-selection-method. In the default case, the procedure uil-message-selection is called.

# uil-message-unselection

Unselects the specified message. This is the procedure that uil-message-unselection-method calls by default.

## Synopsis

uil-message-unselection

(*msg*: class uil-message-object, *scroll-area*: class uil-scroll-area,  
*window*: class g2-window)

Argument	Description
<i>msg</i>	The message that was unselected.
<i>scroll-area</i>	The scroll area that contains the message.
<i>window</i>	The window on which the scroll area is managed.

# uil-message-unselection-method

Unselects a message object whenever a user selects another message object.

## Synopsis

uil-message-unselection-method

(*msg*: class uil-message-object, *scroll-area*: class uil-scroll-area,  
*window*: class g2-window)

Argument	Description
<i>msg</i>	The message that was unselected.
<i>scroll-area</i>	The scroll area that contains the message.
<i>window</i>	The window on which the scroll area is managed.

## Description

This procedure is called whenever a message becomes unselected as a result of the user either selecting another message within the scroll area, or selecting a message that is already selected when the `allow-unselect-on-selected-message` attribute of the scroll area is `true`.

The procedure named by the object's `uil-message-unselection-method` attribute is called. In the default case, the procedure `uil-message-unselection` is called.

## uil-move-message

Moves the specified message from its current scroll area to the specified destination scroll area.

### Synopsis

uil-move-message

(*msg-object*: class uil-message-object,  
*destination-scroll-area*: class uil-scroll-area)

Argument	Description
<i>msg-object</i>	The message to be moved.
<i>destination-scroll-area</i>	The scroll area to which the message is to be moved.

## uil-move-messages

Moves all message object on a specified scroll area to another specified scroll area.

### Synopsis

uil-move-messages (*scroll-area-from*: class uil-scroll-area,  
*scroll-area-to*: class uil-scroll-area)

Argument	Description
<i>scroll-area-from</i>	The scroll area from which message objects are moved.
<i>scroll-area-to</i>	The scroll area to which message objects are moved.

## uil-on-double-click

Method of an item appended to a message object that selects the message object when a user double-clicks on the item.

### Synopsis

uil-on-double-click

(*trigger-item*: class uil-appended-item, *window*: class g2-window)

Argument	Description
<i>trigger-item</i>	Specify the subclass of uil-appended-item that you want to append to the message object.
<i>window</i>	The G2 window where the scroll area with the message object is displayed.

### Description

When a user double-clicks on the message object, uil-on-double-click() calls uil-message-selection-method() to select the message object to which *trigger-item* is appended.

Specify this procedure as a method of the subclass of uil-appended-item that you create to represent the items that you want to append to message objects.

## uil-on-single-click

Method of an item appended to a message object that selects the message object when a user single-clicks on the item.

### Synopsis

uil-on-single-click

(*trigger-item*: class uil-appended-item, *window*: class g2-window)

Argument	Description
<i>trigger-item</i>	Specify the subclass of uil-appended-item that you want to append to the message object.
<i>window</i>	The G2 window where the scroll area with the message object is displayed.

### Description

When a user single-clicks on the message object, `uil-on-single-click()` calls `uil-message-selection-method()` to select the message object to which *trigger-item* is appended.

Specify this procedure as a method of the subclass of `uil-appended-item` that you create to represent the items that you want to append to message objects.



## uil-remove-message

Removes the specified message object from the scroll area that manages it.

### Synopsis

uil-remove-message  
(*msg-object*: class uil-message-object)

Argument	Description
<i>msg-object</i>	The message to remove.

# uil-remove-message-from-list

Removes a specified message object from the message-list of the scroll area that manages it.

## Synopsis

uil-remove-message-from-list  
(*msg-object*: class uil-message-object)

Argument	Description
<i>msg-object</i>	The message object to remove from the message list.

## uil-select-message

Selects a specified message object.

### Synopsis

uil-select-message  
(*msg-object*: class uil-message-object)

Argument	Description
<i>msg-object</i>	The message to select.

# uil-select-message-on-window

Selects a specified message object on a specified window.

## Synopsis

uil-select-message-on-window

(*msg-object*: class uil-message-object, *window*: class g2-window)

Argument	Description
<i>msg-object</i>	The message object that is selected.
<i>window</i>	The window on which the scroll area containing the message object appears.

## uil-sort-scroll-area-messages

Sorts the message objects managed by a specified scroll area.

### Synopsis

uil-sort-scroll-area-messages

(*scroll-area*: class uil-scroll-area, *key*: symbol)

Argument	Description
<i>scroll-area</i>	The scroll area whose messages are to be sorted.
<i>key</i>	The key by which the messages are sorted. Specify <i>Alphabetic</i> , <i>Chronological</i> , <i>Logbook</i> , <i>Priority</i> , or <i>Unordered</i> .

### Description

This procedure sorts messages in the order specified by the *key* argument. For descriptions of these sorting methods, see the *G2 GUIDE User's Guide*.

# uil-unselect-message

Unselects a message in a scroll area.

## Synopsis

uil-unselect-message

(*msg-object*: class uil-message-object)

Argument	Description
<i>msg-object</i>	The message to be unselected.

## uil-update-message-list-with-message

Adds a message object to the specified scroll area's message list.

### Synopsis

uil-update-message-list-with-message

(*scroll-area*: class uil-scroll-area, *msg-object*: class uil-message-object)

Argument	Description
<i>scroll-area</i>	The scroll area whose message list is to be updated.
<i>msg-object</i>	The message object to add.

## Additional Procedures that Operate on Message Objects

[uil-create-custom-text](#)  
[uil-update-text](#)  
[uil-update-unformatted-text](#)  
[uil-launch-edit-session](#)  
[uti-copy-object-attributes](#)  
[uti-lift-to-top](#)



# Navigation Buttons

---

*Describes procedures that operate on navigation buttons.*

Introduction **361**

Procedures that Operate on Navigation Buttons **368**

    uil-create-navigation-button **369**

    uil-handle-navigation-button-method **370**

    uil-handle-workspace-button-method **371**

    uil-select-navigation-button **372**

Additional Procedures that Support Navigation Buttons **373**

Superseded Procedures **374**

    uil-delete-navigation-button **375**



## Introduction

Navigation buttons enable users to navigate between workspaces.

You can use the GUIDE palette to add navigation buttons to a workspace. When only the uilroot module is loaded, you can create navigation buttons by selecting New Object from the KB Workspace menu.

---

**Note** It is not recommended practice to use a navigation button to display a dialog. A dialog that is displayed by a navigation button does not have full GUIDE support for field editing, automatic updating and concluding of values, and other essential features of the GUIDE dialog system.

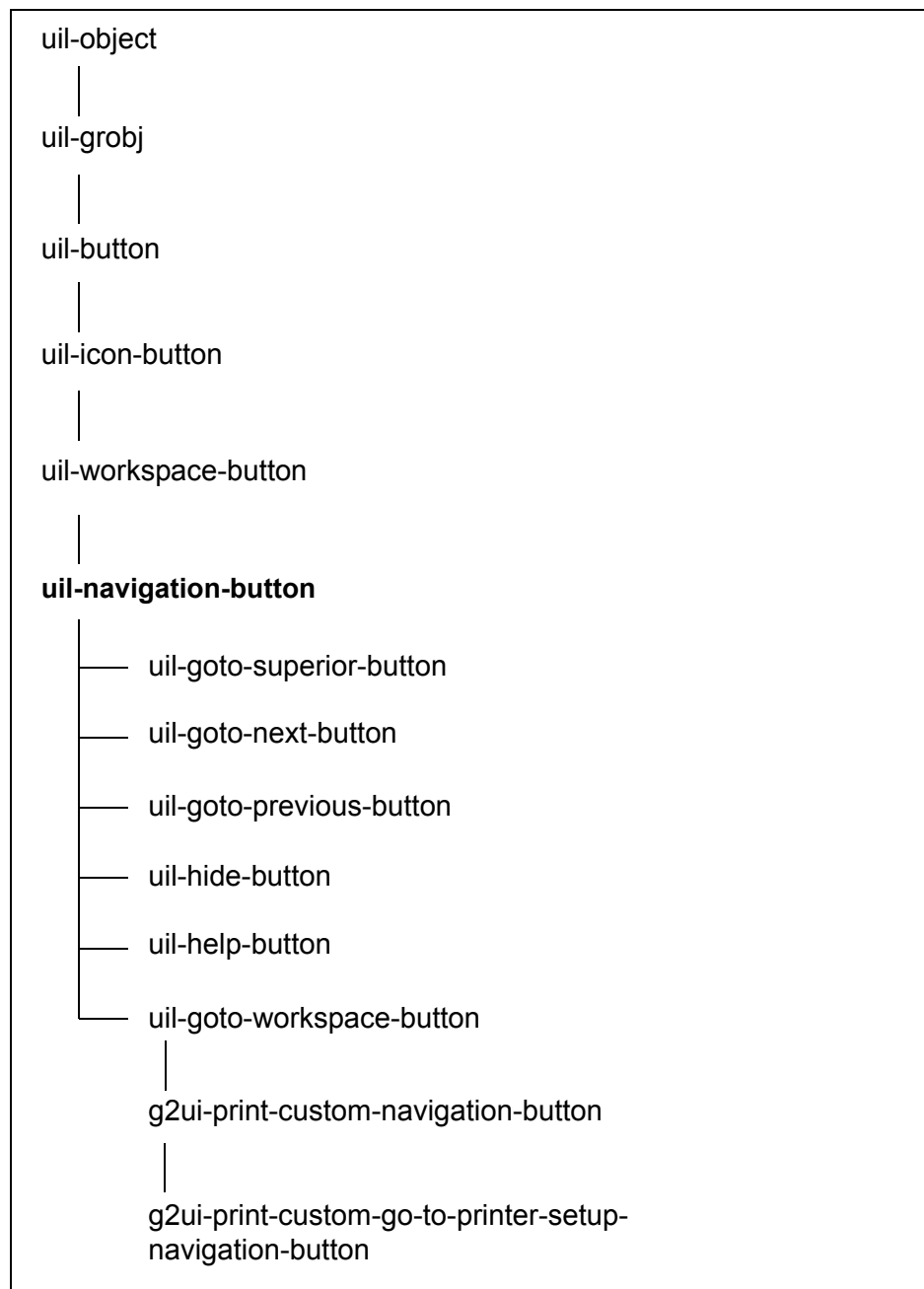
---

## Navigation Button Class Hierarchy

UIL supports six classes of navigation buttons: `uil-goto-workspace-button`, `uil-goto-superior-button`, `uil-goto-next-button`, `uil-goto-previous-button`, `uil-hide-button`, `uil-help-button`. All navigation buttons inherit from the class `uil-navigation-button`.

The following figure illustrates the UIL class hierarchy for navigation buttons. All classes shown are object-definitions. A class whose name is followed by dashes (for example, `uil-button --`) has subclasses that are not shown in the figure.

## Navigation Button Class Hierarchy



## Navigation Button Attributes

The following tables list attributes that are unique to navigation buttons. See [Appendix B, Common Object Attributes](#), for information about the attributes that navigation buttons have in common with other UIL objects.

### Navigation Button Attributes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this navigation button.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-navigation-button-configuration
<b>uil-handler-method</b>	Names the procedure called when the button is selected  <i>Allowable values:</i> valid procedure name  <i>Default value:</i> uil-handle-navigation-button-method
<b>uil-label-constant</b>	Stores a symbol used as a look-up constant that GUIDE/UIL uses to provide different natural language versions of text in text objects ( <b>uil-text</b> ) and buttons ( <b>uil-button</b> ). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i> .  <i>Allowable values:</i> Any symbol.  <i>Default value:</i> unspecified
<b>uil-translation-constraints</b>	Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.  <i>Allowable values:</i> left, right, or center  <i>Default value:</i> left

## Navigation Button Attributes

Attribute	Description
<b>label</b>	The text of a text instance used as a label for this object.  <i>Allowable values:</i> Any text value <i>Default value:</i> " "
<b>workspace-destination</b>	Refers to a name of a kb-workspace-to show.  <i>Allowable values:</i> Name of an existing workspace <i>Default value:</i> uil-dest
<b>uil-hide-this-workspace</b>	A predicate (true or false value) that if true hides this button's workspace when user navigates to another workspace using a navigation button.  <i>Allowable values:</i> true or false <i>Default value:</i> true for goto superior, goto next, and goto previous navigation buttons false for help buttons and goto workspace navigation buttons

## Operations Performed on Navigation Buttons

The following table lists the common operations that application developers can perform on navigation buttons.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---

### Operations on Navigation Buttons

	Operation	Procedure	Method
	Change Size	n/a	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobject	uil-clone-grobject-method
✓	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a
	Configure	uil-configure-grobject	uil-configure-grobject-method
	Create	uil-create-navigation-button ***	n/a
✓	Delete	uil-delete-grobject	uil-delete-grobject-method
✓	Disable	uil-disable-grobject	uil-disable-grobject-method
✓	Enable	uil-enable-grobject	uil-enable-grobject-method
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	uil-handle-button	uil-handle-navigation-button-method
	Highlight	uil-highlight-button	n/a
✓	Initialization	uil-initialize-grobject	uil-initialize-grobject-method
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
	Message Selection	n/a	n/a
	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	uil-reset-button	n/a
	Resize	n/a	n/a

### Operations on Navigation Buttons

	Operation	Procedure	Method
	Set/Select/Toggle	uil-select-button	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	n/a	n/a
	Update Wrapper	n/a	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

### Navigation Button Relations

The following table summarizes the standard relations between navigation buttons and their labels. You can reference these relations to access navigation buttons and their labels programmatically.

#### Relations Between Navigation Buttons and Other Objects

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UIL earlier than Version 5.0 and were not converted to have their label text in their icons.

# Procedures that Operate on Navigation Buttons

This section describes procedures that manipulate `uil-navigation-button` objects.

- `uil-create-navigation-button`
- `uil-handle-navigation-button-method`
- `uil-handle-workspace-button-method`
- `uil-select-navigation-button`



# uil-create-navigation-button

Creates and returns a navigation button.

## Synopsis

uil-create-navigation-button

(*id*: text, *kind*: symbol, *destination*: symbol)

-> *button*: class uil-grobj

Argument	Description
<i>id</i>	The ID for this button.
<i>kind</i>	The kind of navigation button to create. One of: goto-workspace, goto-superior, goto-next, goto-previous, hide, and help.
<i>destination</i>	The name of a top level workspace, the name of an item with a subworkspace, or the symbol <code>none</code> . The <i>destination</i> value sets the Workspace destination attribute of the navigation button.
Return Value	Description
<i>button</i>	The created navigation button.

# uil-handle-navigation-button-method

Handles selection of navigation buttons.

## Synopsis

uil-handle-navigation-button-method

(*button*: class uil-navigation-button, *window*: class item,  
*item-or-wksp*: class item)

Argument	Description
<i>button</i>	The button that was selected.
<i>window</i>	The window on which the dialog or workspace containing the button appears.
<i>item-or-wksp</i>	The dialog or workspace containing the button.

## Description

This procedure is called when the user clicks on the navigation button. It handles the operations associated with clicking on the navigation button. For information about handler methods, see the *G2 GUIDE User's Guide*.

# uil-handle-workspace-button-method

Handles selection of print workspace button.

## Synopsis

uil-handle-workspace-button-method

(*button*: class uil-navigation-button, *window*: class item,  
*item-or-wksp*: class item)

Argument	Description
<i>button</i>	The button that was selected.
<i>window</i>	The window on which the dialog or workspace containing the button appears.
<i>item-or-wksp</i>	The dialog or workspace containing the button.

## Description

This procedure is called when the user clicks on a print workspace button. It handles the operations associated with clicking on the button. For information about handler methods, see the *G2 GUIDE User's Guide*.

# uil-select-navigation-button

Calls the handler method for the navigation button when the button is selected.

## Synopsis

uil-select-navigation-button (*button*: class uil-navigation-button,  
*window*: class g2-window, *drawable*: class item)

Argument	Description
<i>button</i>	The button that was selected.
<i>window</i>	The window on which the dialog or workspace containing the button appears.
<i>drawable</i>	The dialog or workspace containing the button.

## Additional Procedures that Support Navigation Buttons

[uil-add-label-to-button](#)  
[uil-create-custom-button](#)  
[uil-create-text-for-button-from-label](#)  
[uil-disable-button](#)  
[uil-enable-button](#)  
[uil-handle-button](#)  
[uil-handle-button-method](#)  
[uil-highlight-button](#)  
[uil-reset-button](#)  
[uil-select-button](#)  
[uil-update-button-label](#)  
[utl-copy-object-attributes](#)  
[utl-lift-to-top](#)

## Superseded Procedures

The following procedure is supported, but it does not take advantage of the functionality introduced in GUIDE 4.0 and subsequent versions. Recommended procedure: `uil-delete-grobj-method`.

`uil-delete-navigation-button`

# uil-delete-navigation-button

Deletes the navigation button.

## Synopsis

uil-delete-navigation-button  
(*button*: class uil-navigation-button)

Argument	Description
<i>button</i>	The navigation button to delete.





# Push Buttons

---

*Describes procedures that operate on push buttons.*

Introduction **377**

Procedures that Operate on Push Buttons **386**

    uil-create-pushbutton-to-launch-dialog **387**

    uil-create-styled-pushbutton **388**

    uil-unselect-button **389**

Additional Procedures that Support Push Buttons **390**

Superseded Procedures **391**

    uil-create-pushbutton **392**

    uil-delete-pushbutton **393**

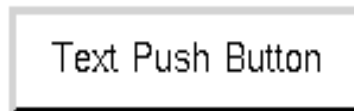


## Introduction

GUIDE supports two kinds of push buttons: icon push buttons and text push buttons. Icon and text push buttons differ from each other only in appearance. Labels of icon buttons are placed to the right of the buttons.



Icon Push Button

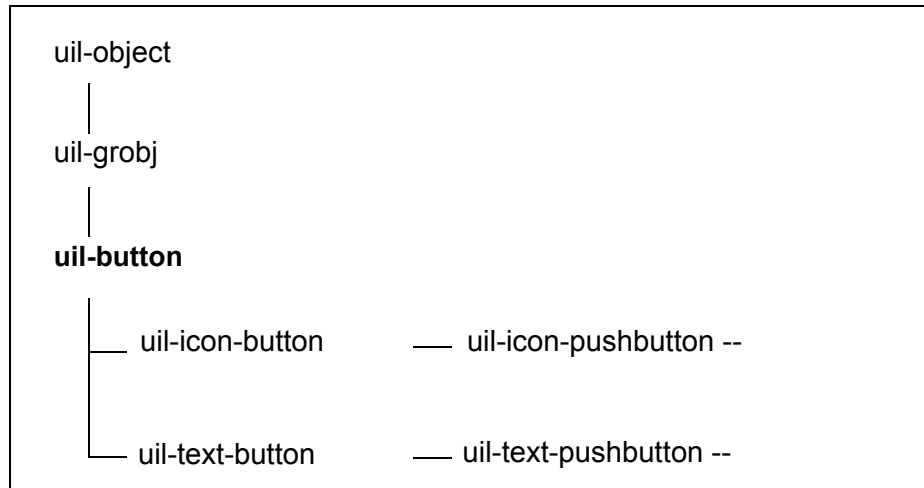


Selecting the label of either kind of push button has the same effect as selecting the push button itself.

## Push Button Class Hierarchy

The following figure illustrates the UIL class hierarchy for push buttons. All classes shown are object-definitions. A class whose name is followed by dashes (for example, `uil-icon-pushbutton --`) has subclasses that are not shown in the figure.

Push Button Class Hierarchy



## Push Button Attributes

The following tables list attributes that are unique to push buttons. See [Appendix B, Common Object Attributes](#), for information about the attributes that push buttons have in common with other UIL objects.

## Push Button Attributes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this push button. Text pushbuttons and icon pushbuttons have different default configurations.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-text-pushbutton-default-button-configuration or uil-icon-pushbutton-configuration
<b>state</b>	Indicates whether the push button is enabled or disabled, as set by the <b>Enable.</b> and <b>Disable.</b> menu choices, or by the uil-enable-grobj or uil-disable-grobj procedures.  <i>Allowable values:</i> enabled or disabled  <i>Default value:</i> enabled
<b>uil-handler-method</b>	Names the procedure called when the button is selected.  <i>Allowable values:</i> Valid procedure name  <i>Default value:</i> uil-handle-button-method
<b>uil-is-dialog-button</b>	Indicates whether or not the button is on a dialog.  <i>Allowable values:</i> true or false  <i>Default value:</i> true

## Push Button Attributes

Attribute	Description
<b>uil-label-constant</b>	Stores a symbol used as a look-up constant that GUIDE/UIIL uses to provide different natural language versions of text in text objects (uil-text) and buttons (uil-button). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i> .
	<i>Allowable values:</i> Any symbol.
	<i>Default value:</i> unspecified
<b>label</b>	The text of the label for this object.
	<i>Allowable values:</i> Any uil-text.
	<i>Default value:</i> "Start"
<b>uil-translation-constraints</b>	Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.
	<i>Allowable values:</i> left, right, or center
	<i>Default value:</i> left
<b>uil-resize-to-fit-label</b>	(for Motif-style uil-text-pushbutton only) If true, GUIDE/UIIL automatically resizes the push button to fit the label text that you specify for the button. Otherwise, GUIDE/UIIL does not resize the button.
	You can set this attribute using the Edit Pushbutton dialog. For information about how to use this dialog, see the <i>G2 GUIDE User's Guide</i> .
	<i>Allowable values:</i> true of false
	<i>Default value:</i> true

## Push Button Attributes

Attribute	Description
<b>uil-minimum-width</b>	(for Motif-style uil-text-pushbutton only) The minimum width to which this push button can be resized to fit the label. This attribute is ignored if Uil resize to fit label is false.  <i>Allowable values:</i> positive integers <i>Default value:</i> 0
<b>uil-minimum-height</b>	(for Motif-style uil-text-pushbutton only) The minimum height to which this push button can be resized to fit the label. This attribute is ignored if Uil resize to fit label is false.  <i>Allowable values:</i> positive integer <i>Default value:</i> 0
<b>callback</b>	Specifies the name of the callback procedure that is called by the button's handler method.  <i>Allowable values:</i> A valid procedure name. The procedure must have the correct arguments for callback procedures. <i>Default value:</i> uil-do-nothing
<b>uil-defaulted-button</b>	Indicates whether or not the button is the default button for the dialog.  <i>Allowable values:</i> true or false <i>Default value:</i> false

## Push Button Attributes

Attribute	Description
<b>uil-cancel-button</b>	Indicates whether or not this button is the cancel button for the dialog.
<i>Allowable values:</i>	true or false
<i>Default value:</i>	false
<b>uil-update-button</b>	Indicates whether or not this button is the update button for the dialog.
<i>Allowable values:</i>	true or false
<i>Default value:</i>	false
<b>uil-button-selection-status</b>	Indicates the button's current selection status.
<i>Allowable values:</i>	selected or unselected
<i>Default value:</i>	unselected
<b>uil-release-button-after-selection</b>	Indicates that the button is to be released from its selected state immediately following the execution of its actions or callback.
<i>Allowable values:</i>	true or false
<i>Default value:</i>	false

### Push Button Attributes

Attribute	Description
<b>uil-action-description</b>	Specifies the names of procedures to execute as actions for the button when the button is selected.

*Allowable values:* The names of any procedures that can be used as push button actions, represented by an array of symbolic values.

*Default value:* a symbol-array

## Operations Performed on Push Buttons

The following table lists the common operations that application developers can perform on push buttons.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations Performed on Push Buttons

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	uil-change-window-style-of-grobj	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-pushbutton ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method

## Operations Performed on Push Buttons

	Operation	Procedure	Method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-button-as-symbol	n/a
	Get State	n/a	n/a
✓	Handler	uil-handle-button	uil-handle-button-method
	Highlight	uil-highlight-button	n/a
✓	Initialize	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	uil-reset-button	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-select-button	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	uil-unselect-button	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a



## Push Button Relations

The following table summarizes the standard relations between push buttons and their labels. You can reference these labels to access the push buttons and their labels programmatically.

### Relations Between Push Buttons and Other Objects

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with earlier versions of GUIDE/UII than Version 5.0 and were not converted to have their label text in their icons.

# Procedures that Operate on Push Buttons

This section describes procedures that you can use to manipulate push buttons.

[uil-create-pushbutton-to-launch-dialog](#)

uil-create-styled-pushbutton

uil-unselect-button

## uil-create-pushbutton-to-launch-dialog

Creates a push button that users can press to launch a specified dialog.

### Synopsis

uil-create-pushbutton-to-launch-dialog

(*dialog*: class uil-dialog)

-> *push-button*: class uil-button

Argument	Description
<i>dialog</i>	The dialog that the push button created is to launch.

Return Value	Description
<u><i>push-button</i></u>	The push button created by this procedure to launch <i>dialog</i> .

# uil-create-styled-pushbutton

Creates and returns a push button.

## Synopsis

uil-create-styled-pushbutton

(*id*: text, *kind*: symbol, *size*: symbol, *label*: text,  
*callback*: symbol, *style*: item-or-value)

-> *push-button*: class uil-button

Argument	Description
<i>id</i>	The ID of the new button.
<i>kind</i>	The type of button. Specify <i>iconic</i> or <i>text</i> .
<i>size</i>	The size of the button.  For Motif style push buttons, specify <i>small</i> , <i>medium</i> , or <i>large</i> .  For Windows style push buttons, specify <i>small-msw</i> , <i>medium-msw</i> , or <i>large-msw</i>
<i>label</i>	The text operating as a label for the button.
<i>callback</i>	The name of the callback procedure executed when the button is selected.
<i>style</i>	Not used. Specify the symbol <i>none</i> .
Return Value	Description
<i>push-button</i>	The created push button.

# uil-unselect-button

Unselects a push button.

## Synopsis

uil-unselect-button  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The button to unselect.

## Description

This procedure sets the push button's `uil-button-selection-status` to unselected, and changes the colors of the button to reflect its unselected state.

## Related Procedure

[uil-select-button](#)

## Additional Procedures that Support Push Buttons

[uil-add-label-to-button](#)  
[uil-create-custom-button](#)  
[uil-create-text-for-button-from-label](#)  
[uil-disable-button](#)  
[uil-enable-button](#)  
[uil-handle-button](#)  
[uil-handle-button-method](#)  
[uil-highlight-button](#)  
[uil-reset-button](#)  
[uil-return-size-of-button-as-symbol](#)  
[uil-select-button](#)  
[uil-update-button-label](#)  
[uil-do-nothing](#)  
[utl-copy-object-attributes](#)  
[utl-lift-to-top](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UII, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

- uil-create-pushbutton
- uil-delete-pushbutton

# uil-create-pushbutton

Creates and returns a push button. Use instead: uil-create-styled-pushbutton.

## Synopsis

uil-create-pushbutton

(*id*: text, *kind*: symbol, *callback*: symbol, *size*: symbol, *label*: text)

-> push-button: class uil-button

Argument	Description
<i>id</i>	The ID for the button.
<i>kind</i>	The type of pushbutton. Specify the symbol <i>iconic</i> or <i>text</i> .
<i>callback</i>	A symbol specifying the name of the callback procedure executed when the button is selected.
<i>size</i>	The size and style of the button.  For Motif style push buttons, specify <i>small</i> , <i>medium</i> , or <i>large</i> .  For Windows style push buttons, specify <i>small-msw</i> , <i>medium-msw</i> , or <i>large-msw</i> .
<i>label</i>	The text used as the button's label.
Return Value	Description
<u>push-button</u>	The created push button.



## uil-delete-pushbutton

Deletes a push button. Recommended procedure: uil-delete-grobj-method.

### Synopsis

uil-delete-pushbutton  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The push button to delete.



# Radio Boxes

---

*Describes procedures that operate on radio boxes.*

Introduction **396**

Procedures that Operate on Radio Boxes **402**

    uil-create-radio-box **403**

    uil-get-radio-box-value **404**

    uil-reset-radio-box **405**

    uil-set-radio-box-value **406**

Additional Procedures that Support Radio Boxes **407**

Superseded Procedures **408**

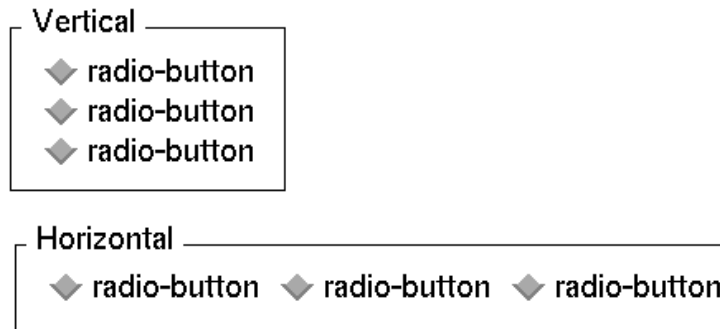
    uil-delete-radio-box **409**



# Introduction

A radio box manages a set of radio buttons. Only one radio button in a radio box can be selected at any one time.

Radio boxes contain three radio buttons by default. You can arrange the buttons horizontally or vertically within the box. The following figure illustrates the horizontal and vertical orientations of radio buttons in radio boxes.



## Radio Box Class Hierarchy

The following figure illustrates the UIL class hierarchy for radio boxes. All classes shown are object-definitions.

Radio Box Class Hierarchy



## Radio Box Attributes

The following tables list attributes that are unique to or have special meaning for radio boxes. See [Appendix B, Common Object Attributes](#), for information about the attributes that radio boxes have in common with other UIL objects.

<b>Radio Box Attributes</b>	
<b>Attribute</b>	<b>Description</b>
<b>configuration</b>	<p>Specifies the configuration that determines the color of this radio-box.</p> <p><i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Default value:</i> uil-radio-box-configuration</p>
<b>uil-button-orientation</b>	<p>Specifies whether buttons are arranged vertically or horizontally within the radio box. You can set this attribute using the <b>Edit Radio Box</b> dialog. For information about how to use this dialog, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> vertical or horizontal</p> <p><i>Default value:</i> vertical</p>
<b>members</b>	<p>Name of an array containing the names of the radio buttons managed by this radio box.</p> <p><i>Allowable values:</i> Any valid radio button name.</p> <p><i>Default value:</i> a text-array</p>
<b>members-index</b>	<p>The number of buttons in the radio box.</p> <p><i>Allowable values:</i> Integer &gt; 0</p> <p><i>Default value:</i> 3</p>

## Radio Box Attributes

Attribute	Description
<b>label</b>	Specifies the label that appears on the edit box. You can set the value of this attribute using the Edit Radio Box dialog. For information about how to use this dialog, see the <i>G2 GUIDE User's Guide</i> .
	<i>Allowable values:</i> Any valid text.
	<i>Default value:</i> ""
<b>radio-box-current-value</b>	The value of the currently selected radio button in the radio box.
	<i>Default value:</i> uil-none

## Operations Performed on Radio Boxes

The following table lists the common operations that application developers can perform on radio boxes.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations on Radio Boxes

	Operation	Procedure	Method
	Change Size	uil-change-size-of-gobj	n/a
	Change Style	uil-change-window-style-of-gobj	n/a
✓	Clone	uil-clone-gobj	uil-clone-gobj-method
✓	Conclude	uil-conclude-gobj	uil-conclude-gobj-method

### Operations on Radio Boxes

	Operation	Procedure	Method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-radio-box ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	n/a	n/a
	Get State	uil-get-radio-box-value ***	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-set-radio-box-value ***	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method

### Operations on Radio Boxes

	Operation	Procedure	Method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Radio Box and Radio Button Relations

The following table summarizes the standard relations between radio boxes and radio buttons. You can reference these relations to access radio boxes or radio buttons programmatically.

### Radio Box and Radio Button Relations

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-radio-box	the-group-master-of	uil-radio-button	one-to-many	grouped-by
uil-radio-button	grouped-by	uil-radio-box	many-to-one	the-group-master-of
uil-selection-box	the-group-master-of	uil-selection-button	one-to-many	grouped-by
uil-selection-button	grouped-by	uil-selection-box	many-to-one	the-group-master-of
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UII earlier than Version 5.0 and were not converted to have their label text in their icons.



For example, the radio box that manages a radio button is related to that radio button by the relation, **the-group-master-of**. Thus, you can access sibling radio buttons through the member-list of the radio box that is **the-group-master-of** the buttons.

## Procedures that Operate on Radio Boxes

This section describes UIL procedures that operate on `uil-radio-box` objects.

- `uil-create-radio-box`
- `uil-get-radio-box-value`
- `uil-reset-radio-box`
- `uil-set-radio-box-value`

# uil-create-radio-box

Creates and returns a radio-box.

## Synopsis

uil-create-radio-box

(*id*: text)

-> *radio-box*: class uil-grobj

Argument	Description
<i>id</i>	The ID for the radio box.

Return Value	Description
<u><i>radio-box</i></u>	The radio box created by this procedure.

# uil-get-radio-box-value

Returns the current value for the radio box, on a specified dialog or workspace, whose id matches a specified id.

## Synopsis

uil-get-radio-box-value

(*dlg-or-workspace*: class item, *id*: text)

-> {*current-value*: item-or-value | none}

Argument	Description
<i>dlg-or-workspace</i>	Dialog or workspace where radio box is displayed.
<i>id</i>	The ID of the radio box to get current value of.

Return Value	Description
<u><i>current-value</i></u>	The current value of the radio box, or the symbol none.

## uil-reset-radio-box

Iterates over all the radio-buttons managed by the specified radio box and sets them to their off state.

### Synopsis

uil-reset-radio-box  
(*radio-box*: class uil-radio-box)

Argument	Description
<i>radio-box</i>	The radio box to reset.

# uil-set-radio-box-value

Turns on one radio button in a radio box.

## Synopsis

uil-set-radio-box-value

(*dialog-or-workspace*: class item, *id*: text, *set-value*: value,  
*window*: class g2-window)

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace containing the radio box.
<i>id</i>	The ID of the radio box to set the value for.
<i>set-value</i>	The current value for the radio box. The radio button whose <b>On value</b> attribute matches <i>set-value</i> is toggled on.
<i>window</i>	The window on which the dialog or workspace containing the radio box is displayed.

## Additional Procedures that Support Radio Boxes

[uil-add-button-to-selection-box](#)

[uil-frame-selection-box](#)

[uil-get-selection-box-last-button](#)

[uil-get-selection-box-lead-button](#)

[uil-utility-check-for-empty-selection-box](#)

[utl-copy-object-attributes](#)

[utl-lift-to-top](#)

## Superseded Procedures

The following procedure is supported by this version of GUIDE but does not take advantage of the functionality introduced in GUIDE 4.0 and subsequent versions.

`uil-delete-radio-box`



## uil-delete-radio-box

Deletes the specified radio box and all of its radio buttons. Recommended procedure: `uil-delete-grobj-method`.

### Synopsis

`uil-delete-radio-box`  
(*radio-box*: class `uil-radio-box`)

Argument	Description
<i>radio-box</i>	The radio box to delete.



# Radio Buttons

---

*Describes procedures that operate on radio buttons.*

Introduction **411**

Procedures that Operate on Radio Buttons **417**

    uil-create-styled-radio-button **418**

    uil-create-styled-radio-button-with-different-sized-label **419**

Additional Procedures that Support Radio Buttons **421**

Superseded Procedures **422**

    uil-create-radio-button **423**

    uil-delete-radio-button **424**



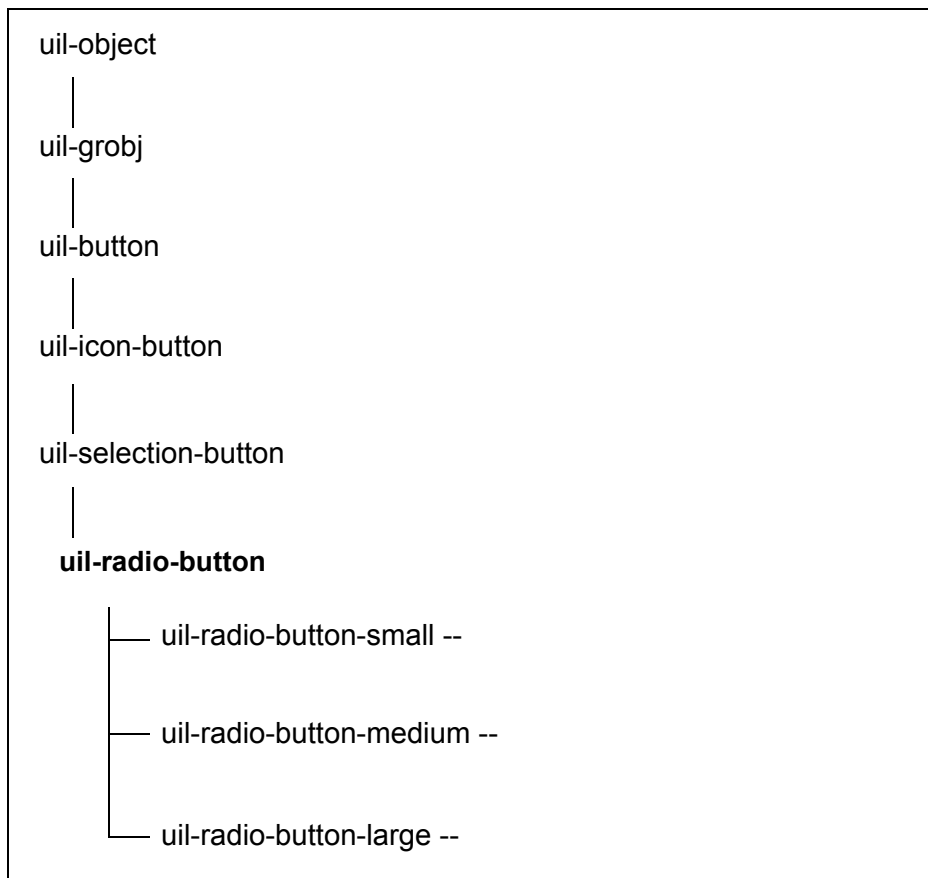
## Introduction

A radio box manages a set of radio buttons. Only one radio button in a radio box can be selected at any one time. All radio buttons inherit from the class `uil-radio-button`.

## Radio Button Class Hierarchy

The following figure shows the class hierarchy for radio buttons. All classes shown are object-definitions. A class whose name is followed by an em-dash (for example, `uil-radio-button-small --`) has subclasses that are not shown in the diagram.

## Radio Button Class Hierarchy



## Radio Button Attributes

The following table describes the attributes that are unique to radio buttons, or have special meaning for radio buttons. See [Appendix B, Common Object Attributes](#), for information about the attributes that radio buttons have in common with other UIL objects.

## Radio Button Attributes

Attribute	Description
<b>configuration</b>	<p>If you want to display this radio button in a non-standard color, you can specify a customized configuration here to provide that color.</p> <p><i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Default value:</i> The configuration of the radio box that contains this radio button.</p>
<b>uil-handler-method</b>	<p>Names the procedure called when the button is selected.</p> <p><i>Allowable values:</i> A valid procedure name. The procedure must have the correct arguments for handler methods.</p> <p><i>Default value:</i> uil-handle-button-method</p>
<b>uil-label-constant</b>	<p>Stores a symbol used as a look-up constant that GUIDE/UIL uses to provide different natural language versions of text in text objects (<b>uil-text</b>) and buttons (<b>uil-button</b>). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> Any symbol.</p> <p><i>Default value:</i> unspecified</p>
<b>uil-translation-constraints</b>	<p>Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.</p> <p><i>Allowable values:</i> left, right, or center</p> <p><i>Default value:</i> left</p>

## Radio Button Attributes

Attribute	Description
<b>label</b>	The text of a uil-text instance operating as a label for this object. <i>Allowable values:</i> any text value <i>Default value:</i> "radio button"
<b>on-value</b>	The value of the button when selected. <i>Allowable values:</i> any text or symbol value <i>Default value:</i> radio-button
<b>off-value</b>	The value of the button when not selected. <i>Allowable values:</i> any text or symbol value <i>Default value:</i> radio-button
<b>toggle-state</b>	Indicates the current state of the button, on or off. <i>Allowable values:</i> on or off <i>Default value:</i> off
<b>callback</b>	A symbol specifying the name of the callback procedure executed when the button is selected by the handler method. <i>Allowable values:</i> A valid procedure name. The procedure must have the correct arguments for callback procedures. <i>Default value:</i> uil-do-nothing

## Operations Performed on Radio Buttons

The following table lists the common operations that application developers can perform on radio buttons.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---

### Operations Performed on Radio Buttons

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	uil-change-window-style-of-grobj	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-radio-button ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-button-as-symbol	n/a
	Get State	n/a	n/a
✓	Handler	uil-handle-button	uil-handle-button-method
	Highlight	uil-highlight-button	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method

## Operations Performed on Radio Buttons

	Operation	Procedure	Method
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	uil-reset-button	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-select-button	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	uil-reset-button	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Radio Box and Radio Button Relations

For information about the relations between radio boxes and radio buttons, see [Radio Box and Radio Button Relations](#).



## Procedures that Operate on Radio Buttons

This section describes procedures that operate on radio buttons. Radio buttons are objects of classes that inherit from `uil-radio-button`.

`uil-create-styled-radio-button`

`uil-create-styled-radio-button-with-different-sized-label`

# uil-create-styled-radio-button

Creates and returns a radio button and adds the button to a specified radio box.

## Synopsis

uil-create-styled-radio-button

(*id*: text, *size*: symbol, *radio-box*: class uil-radio-box, *label*: text,  
*on-value*: value, *off-value*: value, *callback*: symbol, *style*: item-or-value)  
-> *radio-button*: class uil-button

Argument	Description
<i>id</i>	The ID for the radio button.
<i>size</i>	The size and style of button to create.  For Motif style buttons, specify small, medium, or large.  For Windows style buttons, specify small-msw, medium-msw, or large-msw.
<i>radio-box</i>	The radio box to add a button to.
<i>label</i>	The text to use as the label for the button.
<i>on-value</i>	The value, which when matched, sets button to be in on state.
<i>off-value</i>	The value, which when matched, sets button to be in off state.
<i>callback</i>	The name of a procedure which is called when the button is selected.
<i>style</i>	The symbol none (place holder for style object).
Return Value	Description
<u><i>radio-button</i></u>	The radio button added to the radio box.

## uil-create-styled-radio-button-with-different-sized-label

Creates and returns a radio button with a label of a specified size, and adds the button to a specified radio box.

### Synopsis

uil-create-styled-radio-button-with-different-sized-label

(*id*: text, *button-size*: symbol, *label-size*: symbol, *radio-box*: class uil-radio-box, *label*: text, *on-value*: value, *off-value*: value, *callback*: symbol, *style*: item-or-value)  
 -> *radio-button*: class uil-button

Argument	Description
<i>id</i>	The ID for the radio button.
<i>button-size</i>	The size of the button to create.  For Motif style buttons, specify small, medium, or large.  For Windows style buttons, specify small-msw, medium-msw, or large-msw.
<i>label-size</i>	The size of label to create for the button. Specify small, medium, or large.
<i>radio-box</i>	The radio box to add a button to.
<i>label</i>	The text to use as the label for the button.
<i>on-value</i>	The value, which when matched, sets button to be in on state.
<i>off-value</i>	The value, which when matched, sets button to be in off state.
<i>callback</i>	The name of a procedure which is called when the button is selected.
<i>style</i>	The symbol none (place holder for style object).

<b>Return Value</b>	<b>Description</b>
<u><i>radio-button</i></u>	The radio button added to the radio box.

## Additional Procedures that Support Radio Buttons

[uil-30r3-upgrade-radio-and-check-button-label-texts](#)  
[uil-utility-check-for-orphaned-selection-buttons](#)  
[util-copy-object-attributes](#)  
[util-lift-to-top](#)  
[uil-add-label-to-button](#)  
[uil-create-custom-button](#)  
[uil-create-text-for-button-from-label](#)  
[uil-disable-button](#)  
[uil-enable-button](#)  
[uil-handle-button](#)  
[uil-handle-button-method](#)  
[uil-highlight-button](#)  
[uil-reset-button](#)  
[uil-return-size-of-button-as-symbol](#)  
[uil-select-button](#)  
[uil-toggle-button](#)  
[uil-update-button-label](#)  
[uil-update-button-togglelabel](#)  
[uil-add-button-to-selection-box](#)  
[uil-get-selection-box-last-button](#)  
[uil-get-selection-box-lead-button](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UII, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

uil-create-radio-button  
uil-delete-radio-button

## uil-create-radio-button

Creates a radio button and adds it to a radio box. Recommended procedure: uil-create-styled-radio-button.

### Synopsis

uil-create-radio-button

(*id*: text, *radio-box*: class uil-radio-box, *callback*: symbol,  
*label*: text, *on-value*: item-or-value)

-> *radio-button*: class uil-gobj

Argument	Description
<i>id</i>	The ID for the radio button.
<i>radio-box</i>	The radio box to add button to. The radio box is the-group-master-of the created radio button, where the-group-master-of is a relation between a selection box and a selection button.
<i>callback</i>	Name of procedure run when button is selected.
<i>label</i>	The text to appear to the right of the radio button as its label.
<i>on-value</i>	The value of the button when it is selected.
Return Value	Description
<i>radio-button</i>	The created radio button.

## uil-delete-radio-button

Deletes a radio button. Recommended procedure: uil-delete-grobject-method.

### Synopsis

uil-delete-radio-button  
(*button*: class uil-radio-button)

Argument	Description
<i>button</i>	The radio button to be deleted.



# Scroll Areas

---

*Describes procedures that operate on scroll areas.*

Introduction **426**

Procedures that Operate on Scroll Areas **436**

- uil-add-resize-corner-to-scroll-area **437**
- uil-change-scroll-area-message-font-size **438**
- uil-create-customized-scroll-area **439**
- uil-create-scroll-area-from-array **440**
- uil-create-scroll-area-from-list **441**
- uil-create-sized-scroll-area-from-array **442**
- uil-create-sized-scroll-area-from-list **443**
- uil-create-styled-scroll-area **444**
- uil-get-related-scroll-area **445**
- uil-reset-scroll-area **446**
- uil-resize-scroll-area **447**
- uil-return-module-name-for-scroll-area-definition **448**
- uil-scroll-area-reflect **449**
- uil-scroll-decrement **450**
- uil-scroll-decrement-line **451**
- uil-scroll-decrement-line-method **452**
- uil-scroll-decrement-method **453**
- uil-scroll-increment **454**
- uil-scroll-increment-line **455**
- uil-scroll-increment-line-method **456**
- uil-scroll-increment-method **457**
- uil-scroll-to-position **458**
- uil-scroll-to-position-method **459**
- uil-select-matching-values-in-scroll-area **460**
- uil-update-scroll-area-from-array **461**
- uil-update-scroll-area-from-list **462**
- uil-update-scroll-area-message **463**
- uil-update-scroll-area-messages **464**

Additional Procedures that Support Scroll Areas **465**

Superseded Procedures **466**  
uil-create-scroll-area **467**  
uil-delete-scroll-area **468**



## Introduction

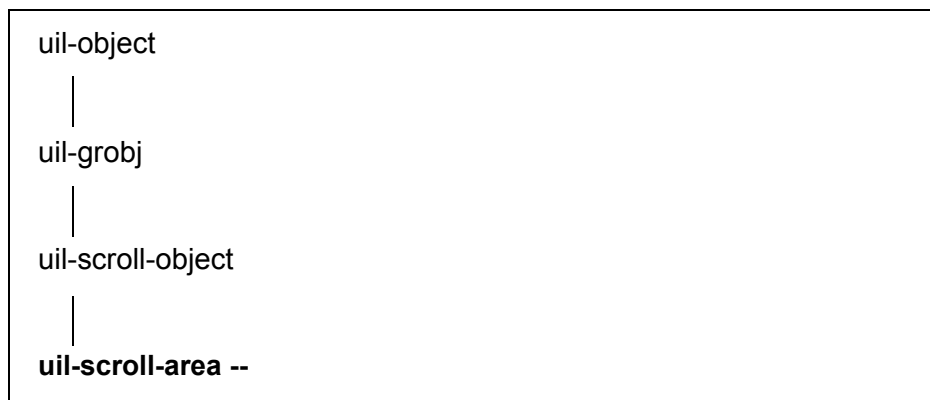
A scroll area manages a group of message objects. The scroll area can contain more message objects than it can display at one time. To see message objects that are not currently displayed, users can scroll up or down to bring the hidden message objects into view.

### Scroll Area Class Hierarchy

All scroll areas inherit from the class uil-scroll-area.

The following figure illustrates the UIL class hierarchy for scroll areas. All classes shown are object-definitions. A class whose name is followed by dashes (for example, uil-scroll-area -- ) has subclasses that are not shown in the diagram.

**Scroll Area Class Hierarchy**



### Scroll Area Attributes

The following tables list attributes that are unique to scroll areas. See [Appendix B, Common Object Attributes](#), for information about the attributes that scroll areas have in common with other objects.

## Scroll Area Attributes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this scroll area.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-default-scroll-configuration
<b>ordering-method</b>	 <i>Allowable values:</i> One of: unordered, alphabetic, chronological, logbook, priority  <i>Default value:</i> chronological
<b>maximum-allowed-messages</b>	Indicates the maximum number of messages the scroll area is allowed to manage.  <i>Allowable values:</i> 0 or any positive integer  <i>Default value:</i> 100
<b>uil-font-size-for-messages</b>	The size font used for the messages managed by scroll area.  <i>Allowable values:</i> small, medium, or large  <i>Default value:</i> medium
<b>uil-message-selection-method</b>	Names the procedure called whenever a message is selected.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for message selection methods.  <i>Default value:</i> uil-message-selection-method

## Scroll Area Attributes

Attribute	Description
<b>uil-message-unselection-method</b>	Names the procedure called whenever a message is unselected.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for message unselection methods.  <i>Default value:</i> uil-message-unselection-method
<b>uil-scroll-increment-method</b>	Names the procedure called whenever the user clicks on the scroll bar below the scroll thumb. The scroll area is incremented by one page.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for scroll increment methods.  <i>Default value:</i> uil-scroll-increment-method
<b>uil-scroll-increment-line-method</b>	Names the procedure called whenever the user clicks on the down-arrow of the scroll area's scroll-bar.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for scroll increment methods.  <i>Default value:</i> uil-scroll-increment-line-method
<b>uil-scroll-decrement-method</b>	Names the procedure called whenever the user clicks on the scroll bar above the scroll thumb. The scroll area is decremented by one page.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for scroll decrement methods.  <i>Default value:</i> uil-scroll-decrement-method

## Scroll Area Attributes

Attribute	Description
<b>uil-scroll-decrement-line-method</b>	<p>Names the procedure called whenever the user clicks on the up-arrow of the scroll area's scroll bar.</p> <p><i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for scroll increment methods.</p> <p><i>Default value:</i> uil-scroll-decrement-line-method</p>
<b>uil-scroll-to-position-method</b>	<p>Names the procedure called whenever the user clicks within the scroll area's scroll bar.</p> <p><i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for scroll to position methods.</p> <p><i>Default value:</i> uil-scroll-to-position-method</p>
<b>allow-multiple-simultaneous-selections</b>	<p>Specifies whether more than one message can be selected at the same time.</p> <p><i>Allowable values:</i> <b>true</b> (allow multiple selections) or <b>false</b> (do not allow multiple selections)</p> <p><i>Default value:</i> <b>false</b></p>
<b>allow-unselect-on-selected-message</b>	<p>Specifies whether or not users can unselect messages by clicking on them.</p> <p><i>Allowable values:</i> <b>true</b> (allow users to unselect selected messages) or <b>false</b> (do not allow users to unselect selected messages)</p> <p><i>Default value:</i> <b>false</b></p>

## Scroll Area Attributes

Attribute	Description
<b>uil-update-style</b>	
<i>Allowable values:</i>	clear-and-insert, select-matching-values, or reuse-messages.
<i>Default value:</i>	clear-and-insert
<b>uil-conclude-style</b>	
<i>Allowable values:</i>	conclude-selected, conclude-all, or conclude-all-including-disabled
<i>Default value:</i>	conclude-all
<b>uil-maximum-characters-to-display</b>	Indicates the maximum number of characters to display in scroll area's messages before clipping displayed value.
<i>Allowable values:</i>	An integer value, or the symbol unlimited.
<i>Default value:</i>	25
<b>uil-allow-multiple-lines</b>	A predicate (true or false value) that if true, wraps the text of a message to a new line instead of clipping its text.
<i>Allowable values:</i>	true or false
<i>Default value:</i>	false
<b>uil-allow-manual-message-move</b>	A predicate (true or false value) that if true, allows user to manually move messages within the scroll area, or between scroll areas.
<i>Allowable values:</i>	true or false
<i>Default value:</i>	true

### Scroll Area Attributes

Attribute	Description
<b>uil-format-specification</b>	Refers to the name of a format (object of class uil-format-specification-class).  <i>Allowable values:</i> The name of any existing format.  <i>Default value:</i> unspecified
<b>uil-display-disabled-messages</b>	Specifies whether disabled message objects in the scroll area displayed (although greyed out), or are invisible.  <i>Allowable values:</i> true or false  <i>Default value:</i> false
<b>number-of-visible-messages</b>	Indicates the number of messages visible in the scroll area. Changing this number causes the scroll area to resize.  <i>Allowable values:</i> Any integer  <i>Default value:</i> 4
<b>current-message-pointer</b>	The index of the first message object visible in the scroll area. The first message object has the index 0, the second message object has the index 1, and so on.  For example, if the third message object in the scroll area is the first message object displayed, the value of Current message pointer is 2.  <i>Allowable values:</i> 0 to (number of messages in scroll area - 1)  <i>Default value:</i> 0

## Scroll Area Attributes

Attribute	Description
<b>number-of-messages</b>	Indicates the total number of messages managed by the scroll area
<i>Allowable values:</i>	Integer (the number of messages currently in the scroll area)
<i>Default value:</i>	10
<b>message-list</b>	Lists the message objects managed by the scroll area.
<i>Default value:</i>	a uil-scroll-message-list
<b>orientation</b>	Indicates whether the scroll bar is along the right or left edge of the scroll area.
<i>Allowable values:</i>	0 (indicates left) or 180 (indicates right)
<i>Default value:</i>	180
<b>corner-state</b>	A value of off indicates that corner object should be removed. A value of on indicates corner object should be displayed.
<i>Allowable values:</i>	on or off
<i>Default value:</i>	on
<b>uil-last-enabled-message</b>	When Uil update style is reuse-message, this attribute contains the index value of the last enabled message in the scroll area. If Uil update style is set to a value other than reuse-message, the value of this attribute is ignored.
<i>Allowable values:</i>	0 to (number of messages in scroll area - 1)
<i>Default value:</i>	0



## Operations Performed on Scroll Areas

The following table lists the common operations that application developers can perform on scroll areas.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---

### Operations Performed on Scroll Areas

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-scroll-area ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialization	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a

## Operations Performed on Scroll Areas

	Operation	Procedure	Method
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	uil-message-selection	uil-message-selection-method
✓	Message Unselection	uil-message-unselection	uil-message-unselection-method
	Move	uil-move-grobj	n/a
	Reset	uil-reset-scroll-area ***	n/a
	Resize	uil-resize-scroll-area ***	n/a
✓	Scroll increment	uil-scroll-increment *** uil-scroll-increment-line***	uil-scroll-increment-method *** uil-scroll-increment-line-method ***
✓	Scroll decrement	uil-scroll-decrement *** uil-scroll-decrement-line ***	uil-scroll-decrement-method *** uil-scroll-decrement-line-method ***
✓	Scroll to position	uil-scroll-to-position ***	uil-scroll-to-position-method ***
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validate	n/a	n/a
	Validate Wrapper	n/a	n/a

## **Scroll Area and Message Object Relations**

For information about the relations between scroll areas and message objects, see [Scroll Area and Message Object Relations](#).

## Procedures that Operate on Scroll Areas

This section describes procedures that you can use to manipulate scroll objects.

- uil-add-resize-corner-to-scroll-area
- uil-change-scroll-area-message-font-size
- uil-create-customized-scroll-area
- uil-create-scroll-area-from-array
- uil-create-scroll-area-from-list
- uil-create-sized-scroll-area-from-array
- uil-create-sized-scroll-area-from-list
- uil-create-styled-scroll-area
- uil-get-related-scroll-area
- uil-reset-scroll-area
- uil-resize-scroll-area
- uil-return-module-name-for-scroll-area-definition
- uil-scroll-area-reflect
- uil-scroll-decrement
- uil-scroll-decrement-line
- uil-scroll-decrement-line-method
- uil-scroll-decrement-method
- uil-scroll-increment
- uil-scroll-increment-line
- uil-scroll-increment-line-method
- uil-scroll-increment-method
- uil-scroll-to-position
- uil-scroll-to-position-method
- uil-select-matching-values-in-scroll-area
- uil-update-scroll-area-from-array
- uil-update-scroll-area-from-list
- uil-update-scroll-area-message
- uil-update-scroll-area-messages

---

**Note** For descriptions of `uil-convert-uil-scroll-area-to-41r0` and other UIL procedures that enable you to upgrade your GUIDE applications, see [Upgrade and Utility Procedures](#).

---

## uil-add-resize-corner-to-scroll-area

Adds a resize corner to the scroll area. For information about how to resize scroll areas, see the *G2 GUIDE User's Guide*.

### Synopsis

uil-add-resize-corner-to-scroll-area  
(*scroll area*: class uil-scroll-object)

Argument	Description
<i>scroll area</i>	The scroll area to which the resize corner is added.

# uil-change-scroll-area-message-font-size

Changes the font size of messages in the scroll area. Does not change the size of the scroll area.

## Synopsis

uil-change-scroll-area-message-font-size  
(*scroll-area*: class: uil-scroll-area, *size*: symbol)  
-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>scroll-area</i>	The ID for the scroll area.
<i>size</i>	The font size for messages in the scroll area. Specify the symbol small, the symbol medium, or the symbol large.

Return Value	Description
<i>scroll-area</i>	The scroll area.

## uil-create-customized-scroll-area

Creates a scroll area from the specified information.

### Synopsis

uil-create-customized-scroll-area

(*id*: text, *height*: integer, *width*: integer, *font-size*: symbol,  
*scroll-area-definition*: item-or-value, *message-object-definition*: item-or-value,  
*module-name*: symbol)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>id</i>	The ID for the scroll area.
<i>height</i>	The height of the scroll area, in pixels.
<i>width</i>	The width of the scroll area, in pixels.
<i>font-size</i>	The font-size to use for all scroll messages.
<i>scroll-area-definition</i>	The name of an existing scroll area-definition.
<i>message-object-definition</i>	The name of an existing message object definition.
<i>module-name</i>	The module that the scroll area will reside in.
Return Value	Description
<u><i>scroll-area</i></u>	The created scroll area.

# uil-create-scroll-area-from-array

Creates a scroll area and populates it with message objects created from the elements of the given array.

## Synopsis

uil-create-scroll-area-from-array

(*id*: text, *array-size*: integer, *array*: class g2-array)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>id</i>	The ID for the scroll area.
<i>array-size</i>	The number of elements in the array.
<i>array</i>	An array containing values from which to create message objects.

Return Value	Description
<u><i>scroll-area</i></u>	The scroll area created.



## uil-create-scroll-area-from-list

Creates a scroll area and populates it with message objects created from the elements of the given list.

### Synopsis

uil-create-scroll-area-from-list

(*id*: text, *number*: integer, *list*: class g2-list)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>id</i>	The ID of the scroll area.
<i>number</i>	The number of elements in the list.
<i>list</i>	A list containing values to generate message objects for.

Return Value	Description
<u><i>scroll-area</i></u>	The scroll area created.

# uil-create-sized-scroll-area-from-array

Creates a scroll area in a specified height and width and populates it with message objects created from the elements of a specified array.

## Synopsis

uil-create-scroll-area-from-array

(*id*: text, *height*: integer, *width*: integer, *font-size*: symbol, *array-size*: integer, *array*: class g2-array)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>id</i>	The ID for the scroll area.
<i>height</i>	The height of the scroll area, in pixels.
<i>width</i>	The width of the scroll area, in pixels.
<i>font-size</i>	The size of the characters in the message objects. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>array-size</i>	The number of elements in the array.
<i>array</i>	An array containing values to create message objects for.

Return Value	Description
<u><i>scroll-area</i></u>	The scroll area created.

## uil-create-sized-scroll-area-from-list

Creates a scroll area and populates it with message objects created from the elements of the given list.

### Synopsis

uil-create-scroll-area-from-list

(*id*: text, *height*: integer, *width*:integer, *font-size*: symbol, *list-size*: integer,  
*list*: class g2-list)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>id</i>	The ID of the scroll area.
<i>height</i>	The height of the scroll area, in pixels.
<i>width</i>	The width of the scroll area, in pixels.
<i>font-size</i>	The size of the characters in the message objects. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>list-size</i>	The number of elements in the list.
<i>list</i>	A list containing values to generate message objects for.
Return Value	Description
<u><i>scroll-area</i></u>	The scroll area created.

# uil-create-styled-scroll-area

Creates and returns a scroll area from the specified information.

## Synopsis

uil-create-styled-scroll-area

(*id*: text, *height*: integer, *width*: integer, *font-size*: symbol, *style*: item-or-value)

-> *scroll-area*: class uil-scroll-area

<b>Argument</b>	<b>Description</b>
<i>id</i>	The ID for the scroll area.
<i>height</i>	The height of the scroll area, in pixels.
<i>width</i>	The width of the scroll area, in pixels.
<i>font-size</i>	The font-size to use for all scroll messages. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>style</i>	The symbol none (place holder for style object).

<b>Return Value</b>	<b>Description</b>
<u><i>scroll-area</i></u>	The created scroll area.

## uil-get-related-scroll-area

Returns the scroll area associated with a specified scroll-area component.

### Synopsis

uil-get-related-scroll-area  
(*scroll-component*: class uil-scroll-component)  
-> *scroll-area*: item-or-value

Argument	Description
<i>scroll-component</i>	Specify the scroll arrow, thumb, bar, or segment

Return Value	Description
<u><i>scroll-area</i></u>	The scroll area of which <i>scroll-component</i> is a part.

## uil-reset-scroll-area

Unselects any selected messages in the specified scroll area and displays messages starting with the first message in the message-list.

### Synopsis

uil-reset-scroll-area  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to be reset.

## uil-resize-scroll-area

Resizes the specified scroll area to the specified size.

### Synopsis

uil-resize-scroll-area

(*scroll-area*: class uil-scroll-area, *height*: integer, *width*: integer)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>scroll-area</i>	The scroll area to be resized.
<i>height</i>	The new height of the scroll area.
<i>width</i>	The new width of the scroll area.

Return Value	Description
<u><i>scroll-area</i></u>	The resized scroll area.

# uil-return-module-name-for-scroll-area-definition

This procedure returns the module name in which the scroll area definition resides.

## Synopsis

uil-return-module-name-for-scroll-area-definition  
(*scroll-definition*: class object-definition)  
-> *module-name*: symbol

Argument	Description
<i>scroll-definition</i>	The name of the scroll area object definition.

Return Value	Description
<u><i>module-name</i></u>	The name of the module where the scroll area definition resides.



## uil-scroll-area-reflect

Changes the position of the scroll bar on a scroll area from the left side to the right side, or from the right side to the left side.

### Synopsis

uil-scroll-area-reflect  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area whose scroll bar is to be reflected.

## uil-scroll-decrement

Moves message objects in the scroll area down one page when a user clicks on the scroll bar above the scroll thumb. This procedure is called by the system-defined method `uil-scroll-decrement-method`.

### Synopsis

`uil-scroll-decrement`

(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll downward.

## uil-scroll-decrement-line

Moves the message objects in the scroll area down one line when a user clicks on the scroll up arrow on the scroll bar. This procedure is called by the system-defined method `uil-scroll-decrement-line-method`.

### Synopsis

`uil-scroll-decrement-line`  
(*scroll-area*: class `uil-scroll-area`)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll downward.

# uil-scroll-decrement-line-method

Calls the scroll decrement line method for this scroll area.

## Synopsis

uil-scroll-decrement-line-method  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll downward.

## Description

uil-scroll-decrement-line-method is the default method named by the Uil scroll decrement line method scroll area attribute. When uil-scroll-decrement-line-method is named by this attribute, it calls the system-defined procedure uil-scroll-decrement-line whenever a user clicks on the scroll up arrow. uil-scroll-decrement-line then moves the message objects in the scroll area down one line.

You can create a user-defined scroll decrement line method to use in place of uil-scroll-decrement-line-method. You reference your user-defined method, rather than uil-scroll-decrement-line-method, from the Uil scroll decrement line method attribute. In this case, uil-scroll-decrement-line-method calls your user-defined method whenever a user clicks on the scroll up arrow. Your user-defined method should call uil-scroll-decrement-line, so that clicking on the scroll up arrow has the default behavior.

## uil-scroll-decrement-method

Calls the scroll decrement method for this scroll area.

### Synopsis

uil-scroll-decrement  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll downward.

### Description

uil-scroll-decrement-method is the default method named by the Uil scroll decrement method attribute of scroll areas. When uil-scroll-decrement-method is named by this attribute, it calls the system-defined procedure uil-scroll-decrement whenever a user clicks on the scroll bar above the scroll thumb. uil-scroll-decrement then moves the message objects in the scroll area down one page.

You can create a user-defined scroll decrement method to use in place of uil-scroll-decrement-method. You reference your user-defined method, rather than uil-scroll-decrement-method, from the Uil scroll decrement method attribute. In this case, uil-scroll-decrement-method calls your user-defined method whenever a user clicks on the scroll bar above the scroll thumb. Your user-defined method should call uil-scroll-decrement, so that clicking on the scroll bar above the scroll thumb has the expected default behavior.

## uil-scroll-increment

Moves the message objects in the scroll area up one page when a user clicks on the scroll bar below the scroll thumb. This procedure is called by the system-defined method `uil-scroll-increment-method`.

### Synopsis

`uil-scroll-increment`

(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll upward.

## uil-scroll-increment-line

Moves the message objects in the scroll area up one line when a user clicks on the scroll down arrow on the scroll bar. This procedure is called by the system-defined method `uil-scroll-increment-line-method`.

### Synopsis

`uil-scroll-increment-line`  
(*scroll-area*: class `uil-scroll-area`)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll upward.

# uil-scroll-increment-line-method

Calls the scroll increment line method for this scroll area.

## Synopsis

uil-scroll-increment-line-method  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll upward.

## Description

uil-scroll-increment-line-method is the default method named by the Uil scroll increment line method attribute of scroll areas. When uil-scroll-increment-line-method is named by this attribute, it calls the system-defined procedure uil-scroll-increment-line whenever a user clicks on the scroll down arrow. uil-scroll-increment-line then moves the message objects in the scroll area up one line.

You can create a user-defined scroll increment line method to use in place of uil-scroll-increment-line-method. You reference your user-defined method, rather than uil-scroll-increment-line-method, from the Uil scroll increment line method attribute. In this case, uil-scroll-increment-line-method calls your user-defined method whenever a user clicks on the scroll down arrow. Your user-defined method should call uil-scroll-increment-line, so that clicking on the scroll down arrow has the expected default behavior.



## uil-scroll-increment-method

Calls the scroll increment method for this scroll area.

### Synopsis

uil-scroll-increment-method  
 (*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to scroll upward.

### Description

uil-scroll-increment-method is the default method named by the Uil scroll increment method attribute of scroll areas. When uil-scroll-increment-method is named by this attribute, it calls the system-defined procedure uil-scroll-increment whenever a user clicks on the scroll bar below the scroll thumb. uil-scroll-increment then moves the message objects in the scroll area up one page.

You can create a user-defined scroll increment method to use in place of uil-scroll-increment-method. You reference your user-defined method, rather than uil-scroll-increment-method, from the Uil scroll increment method attribute. In this case, uil-scroll-increment-method calls your user-defined method whenever a user clicks on the scroll bar below the scroll thumb. Your user-defined method should call uil-scroll-increment, so that clicking on the scroll bar below the scroll thumb has the expected default behavior.

## uil-scroll-to-position

Scrolls the scroll area to a particular position when a user moves the scroll thumb. Dragging the scroll thumb up moves message objects down. Dragging the scroll thumb down moves message objects up. This procedure is called by the system-defined method `uil-scroll-to-position-method`.

### Synopsis

`uil-scroll-to-position`

(*scroll-segment*: class `uil-scroll-segment`)

Argument	Description
<i>scroll-segment</i>	Scroll segment in scroll bar that user clicks on.

## uil-scroll-to-position-method

Calls the scroll to position method for this scroll area.

### Synopsis

uil-scroll-to-position-method  
 (*scroll-segment*: class uil-scroll-segment)

Argument	Description
<i>scroll-segment</i>	Scroll segment in scroll bar that user clicks on.

### Description

uil-scroll-to-position-method is the default method named by the Uil scroll to position method attribute of scroll areas. When uil-scroll-to-position-method is named by this attribute, it calls the system-defined procedure uil-scroll-to-position whenever a user drags the scroll thumb up or down. uil-scroll-to-position then moves the message objects up if the scroll thumb is dragged down, or moves them down if the scroll thumb is dragged up.

You can create a user-defined scroll to position method to use in place of uil-scroll-to-position-method. You reference your user-defined method, rather than uil-scroll-to-position-method, from the Uil scroll to position method attribute. In this case, uil-scroll-to-position-method calls your user-defined method whenever a user drags the scroll thumb. Your user-defined method should call uil-scroll-to-position, so that dragging the scroll thumb has the expected default behavior.

# uil-select-matching-values-in-scroll-area

Selects message objects in a scroll area whose values match values in a G2 array or G2 list.

## Synopsis

uil-select-matching-values-in-scroll-area  
(*scroll-area*: class uil-scroll-area, *attr*: class item)

Argument	Description
<i>scroll-area</i>	The scroll area in which to select messages whose values match those in <i>attr</i> .
<i>attr</i>	A G2 array or G2 list containing values to match.

## Description

This procedure iterates through the elements of the array *attr*, selecting message objects managed by the specified scroll area if the values of the message objects match any element of the G2 array or G2 list.

## uil-update-scroll-area-from-array

Updates and displays messages for the scroll area based on the elements of the given array. This procedure is optimal for inserting multiple items. See the `uil-add-to-combo-box-list()` procedure for inserting an individual item.

### Synopsis

`uil-update-scroll-area-from-array`

(*scroll-area* class uil-scroll-area, *array*: class g2-array)

Argument	Description
<i>scroll-area</i>	The scroll area to be updated.
<i>array</i>	An array containing values to generate message objects for.

## uil-update-scroll-area-from-list

Updates and displays messages for the scroll area based on the elements of the given list. This procedure is optimal for inserting multiple items. See the `uil-add-to-combo-box-list()` procedure for inserting an individual item.

### Synopsis

`uil-update-scroll-area-from-list`

(*scroll area*: class `uil-scroll-area`, *list*: class `g2-list`)

Argument	Description
<i>scroll area</i>	The scroll area to be updated.
<i>list</i>	The list containing values to generate message objects for. You can specify any <code>g2-list</code> except <code>item-list</code> and its subclasses.

## uil-update-scroll-area-message

Updates a specified message object with a specified value.

### Synopsis

uil-update-scroll-area-message

(*scroll-area*: class uil-scroll-area, *msg-object*: class uil-message-object,  
*val*: value)

-> {*scroll-area*: class uil-scroll-area | *status*: symbol}

Argument	Description
<i>scroll-area</i>	The scroll area containing the message to be updated.
<i>msg-object</i>	The message object to be updated.
<i>val</i>	The new value for the message object.

Return Value	Description
<u><i>scroll-area</i></u>	The scroll area that was updated, or if the scroll area was not updated, either the symbol <code>not-found</code> , or the symbol <code>error</code> .

# uil-update-scroll-area-messages

Updates the messages in a specified scroll area with the values in a specified G2 array or G2 list.

## Synopsis

uil-update-scroll-area-messages  
(*scroll-area*: class uil-scroll-area, *vals*: class item)

Argument	Description
<i>scroll-area</i>	The scroll area to update.
<i>vals</i>	A G2 array or G2 list containing the values used to update the scroll area.

## Description

If *vals* references a G2 array, this procedure calls uil-update-scroll-area-from-array to update the scroll area. If *vals* references a G2 list, this procedure calls uil-update-scroll-area-from-list to update the scroll area.



## Additional Procedures that Support Scroll Areas

[uil-30r3-upgrade-scroll-area-keys](#)  
[utl-copy-object-attributes](#)  
[utl-lift-to-top](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UII, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

uil-create-scroll-area  
uil-delete-scroll-area

# uil-create-scroll-area

Creates and returns a scroll area. Recommended procedure: uil-create-styled-scroll-area.

## Synopsis

uil-create-scroll-area

(*id*: text, *height*: integer, *width*: integer)

-> *scroll-area*: class uil-scroll-area

Argument	Description
<i>id</i>	The ID of the new scroll area.
<i>height</i>	The height of the new scroll area.
<i>width</i>	The width of the new scroll area.

Return Value	Description
<u><i>scroll-area</i></u>	The new scroll area.

## uil-delete-scroll-area

Deletes the specified scroll area and all of the message objects that it manages.  
Recommended procedure: uil-delete-grobject-method.

### Synopsis

uil-delete-scroll-area  
(*scroll-area*: class uil-scroll-area)

Argument	Description
<i>scroll-area</i>	The scroll area to delete.

# Selection Boxes

---

*Describes procedures that operate on selection boxes.*

Introduction **469**

Procedures that Operate on Selection Boxes **475**

    uil-add-button-to-selection-box **476**

    uil-get-buttons-for-this-selection-box **477**

    uil-get-buttons-grouped-by-this-selection-box **478**

    uil-get-selection-box-for-button **479**

    uil-get-selection-box-last-button **480**

    uil-get-selection-box-lead-button **481**

    uil-utility-check-for-empty-selection-box **482**

Additional Procedures that Support Selection Boxes **483**

Superseded Procedures **484**

    uil-frame-selection-box **485**



## Introduction

This chapter describes procedures that operate on selection boxes. GUIDE supports two kinds of selection boxes: radio boxes and check boxes. A radio box manages a group of radio buttons. A check box manages a group of check buttons.

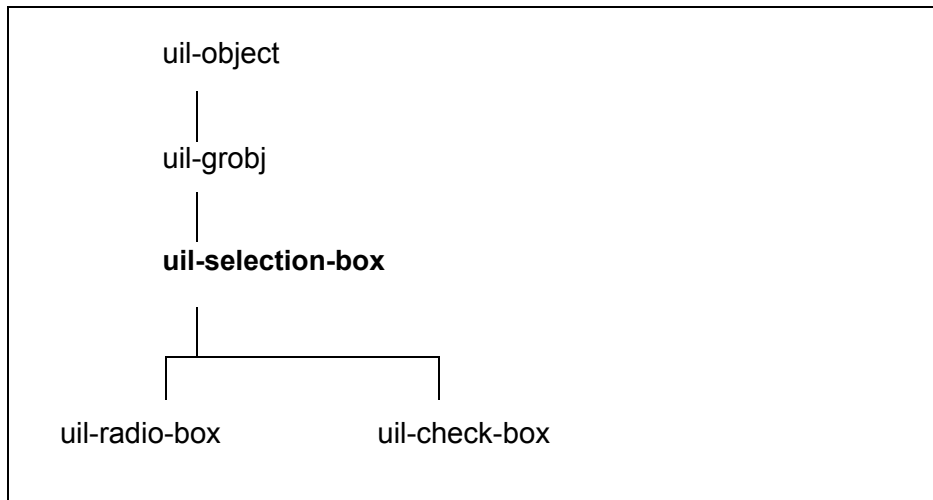
For information about procedures that operate on radio boxes, see [Radio Boxes](#).

For information about procedures that operate on check boxes, see [Check Boxes](#).

## Selection Box Class Hierarchy

All selection boxes inherit from the class `uil-selection-box`. The following figure illustrates the UIL class hierarchy for selection boxes. All classes shown are object definitions.

Selection Box Class Hierarchy



## Selection Box Attributes

See [Appendix B, Common Object Attributes](#), for information about the attributes that selection boxes have in common with other UIL objects.

### Selection Box Attributes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this selection box. Check boxes and radio boxes have different default configurations.
<i>Allowable values:</i>	Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .
<i>Default value:</i>	<code>uil-check-box-configuration</code> or <code>uil-radio-box-configuration</code>

## Selection Box Attributes

Attribute	Description
<b>members</b>	An array of the names of the buttons in the selection box.  <i>Allowable values:</i> Names of the selection buttons managed by this selection box.  <i>Default value:</i> a text-array
<b>member-index</b>	The number of buttons in the selection box.  <i>Allowable values:</i> The number of buttons managed by this selection box.  <i>Default value:</i> 3
<b>uil-button-orientation</b>	Specifies whether selection buttons are arranged vertically or horizontally within the select box that contains them.  <i>Allowable values:</i> vertical and horizontal  <i>Default value:</i> vertical
<b>label</b>	The text of the uil-text object that is the label of the selection box.  <i>Allowable values:</i> Any string, or "" (empty string)  <i>Default value:</i> ""

## Operations Performed on Selection Boxes

The following table lists the common operations performed on selection boxes.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

---

### Operations Performed on Selection Boxes

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	uil-change-window-style-of-grobj	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	uil-conclude-grobj	uil-conclude-grobj-method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	n/a	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialize	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a



### Operations Performed on Selection Boxes

	Operation	Procedure	Method
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

### Selection Box and Selection Button Relations

The following table summarizes standard relations between selection boxes and the buttons that they manage. You can reference these relations to access the selection boxes and selection buttons programmatically.

#### Selection Box and Selection Button Relations

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-selection-box	the-group-master-of	uil-selection-button	one-to-many	grouped-by
uil-selection-button	grouped-by	uil-selection-box	many-to-one	the-group-master-of

### Selection Box and Selection Button Relations

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The `the-uil-text-label-of` relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UIL earlier than Version 5.0 and were not converted to have their label text in their icons.

## Procedures that Operate on Selection Boxes

This section describes procedures that you can use to manipulate selection boxes.

- uil-add-button-to-selection-box
- uil-get-buttons-for-this-selection-box
- uil-get-buttons-grouped-by-this-selection-box
- uil-get-selection-box-for-button
- uil-get-selection-box-last-button
- uil-get-selection-box-lead-button
- uil-utility-check-for-empty-selection-box

## uil-add-button-to-selection-box

Adds a button to a selection box locating it at the given index in the member-list of the selection box.

### Synopsis

uil-add-button-to-selection-box

(*button*: class uil-selection-button, *box*: class uil-selection-box, *index*: integer, *perm*: truth-value)

Argument	Description
<i>button</i>	The button to add to selection box.
<i>box</i>	The selection box to add the button to.
<i>index</i>	The index identifying the location in the group of buttons managed by the selection box for the new button.
<i>perm</i>	If true, makes the selection box permanent.

## uil-get-buttons-for-this-selection-box

Gets a list of the buttons managed by a specified selection box.

### Synopsis

uil-get-buttons-for-this-selection-box  
(*selection-box*: item-or-value, *item-list*: class item-list)

Argument	Description
<i>selection-box</i>	A radio box or check box.
<i>item-list</i>	A list of the radio buttons in the radio box, or the check buttons in the check box.

# uil-get-buttons-grouped-by-this-selection-box

Populates an item-list with the selection buttons managed by the specified selection box.

## Synopsis

uil-get-buttons-grouped-by-this-selection-box  
(*box*: item-or-value, *buttons*: class item-list)

Argument	Description
<i>box</i>	The selection box from which to return the selection buttons.
<i>buttons</i>	An item list that is populated with the selection buttons managed by the specified selection box.

## uil-get-selection-box-for-button

Returns the uil-selection-box that manages the specified selection button.

### Synopsis

```
uil-get-selection-box-for-button
(button: item-or-value)
-> result: item-or-value
```

Argument	Description
<i>button</i>	The selection button from which to return the selection box.

Return Value	Description
<u>result</u>	The selection box that manages the specified selection button. If no selection box is found, the symbol no-selection-box-found is returned.

## uil-get-selection-box-last-button

Returns the last button (bottom-most) contained by the specified selection box.

### Synopsis

```
uil-get-selection-box-last-button
```

```
(box: class uil-selection-box)
```

```
-> {button: class uil-selection-button | not-found: symbol}
```

Argument	Description
<i>box</i>	The selection box that contains the button returned.

Return Value	Description
<u><i>button</i></u>	The last button, or the symbol not-found.



# uil-get-selection-box-lead-button

Returns the lead button (top-most) contained by the specified selection box.

## Synopsis

uil-get-selection-box-lead-button (*box*: class uil-selection-box)  
-> {*button*: class uil-selection-button | not-found: symbol}

Argument	Description
<i>box</i>	The selection box whose lead button is returned.

Return Value	Description
<u><i>button</i></u>	The lead button, or the symbol not-found.

# uil-utility-check-for-empty-selection-box

Checks to see if the specified selection box is managing selection buttons that actually exist. Check only if the number of members is greater than zero.

## Synopsis

uil-utility-check-for-empty-selection-box (*box*: class uil-selection-box)  
-> *box-is-empty*: truth-value

Argument	Description
<i>box</i>	The selection box checked.

Return Value	Description
<u><i>box-is-empty</i></u>	The symbol <code>false</code> , if the selection buttons exist, otherwise, <code>true</code> .

## Related Procedure

- [uil-utility-check-for-orphaned-selection-buttons](#)

## Additional Procedures that Support Selection Boxes

[uil-add-label-to-button](#)  
[uil-create-custom-button](#)  
[uil-create-text-for-button-from-label](#)  
[uil-disable-button](#)  
[uil-enable-button](#)  
[uil-handle-button](#)  
[uil-handle-button-method](#)  
[uil-highlight-button](#)  
[uil-reset-button](#)  
[uil-return-size-of-button-as-symbol](#)  
[uil-select-button](#)  
[uil-toggle-button](#)  
[uil-update-button-label](#)  
[uil-update-button-togglelabel](#)  
[uil-create-check-button](#)  
[uil-create-styled-check-button](#)  
[uil-create-styled-check-button-with-different-sized-label](#)  
[uil-delete-check-button](#)  
[uil-create-radio-button](#)  
[uil-create-styled-radio-button](#)  
[uil-create-styled-radio-button-with-different-sized-label](#)  
[uil-delete-radio-button](#)  
[uil-utility-check-for-orphaned-selection-buttons](#)  
[utl-copy-object-attributes](#)  
[utl-lift-to-top](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UII, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

`uil-frame-selection-box`

## uil-frame-selection-box

Creates a uil-box-object to frame all the selection buttons grouped by the selection box.

Beginning in GUIDE 4.0, selection boxes display a border by default. You can hide and show the border, using menu commands of the selection box.

### Synopsis

uil-frame-selection-box

(*grobj*: item)

-> *box-border*: uil-box-object | object-is-not-right-type: symbol}

Argument	Description
<i>grobj</i>	The selection box, or a selection button contained in selection box.

Return Value	Description
<u><i>box-border</i></u>	The box-border created to frame the selection box, or the symbol object-is-not-right-type



# Selection Buttons

---

*Describes procedures that operate on selection buttons.*

Introduction **487**

Procedures that Operate on Selection Buttons **493**

    uil-get-next-selection-button-y-position **494**

    uil-get-selection-button **495**

    uil-is-this-button-grouped-by-this-selection-box **496**

Additional Procedures that Support Selection Buttons **497**



## Introduction

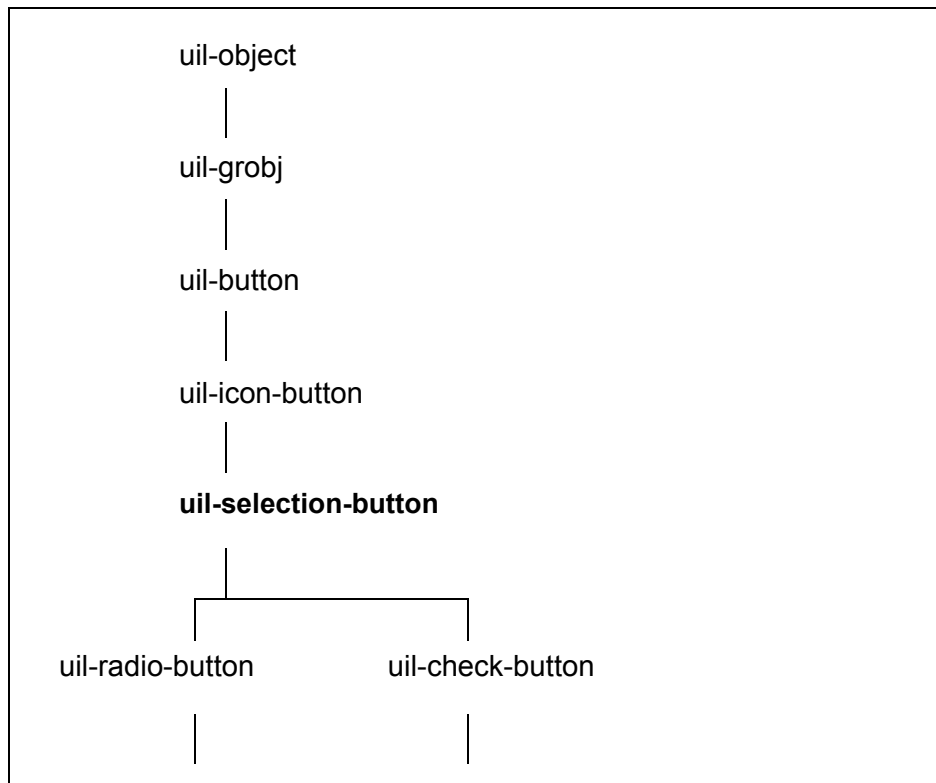
This chapter describes UIL procedures that operate on selection buttons. GUIDE supports two kinds of selection buttons: radio buttons and check buttons.

For information about procedures that operate only on radio buttons, see [Radio Buttons](#). For information about procedures that operation only on check buttons, see [Check Buttons](#).

## Selection Button Class Hierarchy

All selection buttons inherit from the class `uil-selection-button`. The following figure illustrates the UIL class hierarchy for radio buttons and check buttons. All classes shown are object definitions.

Selection Button Class Hierarchy



## Selection Button Attributes

The following table lists attributes that are unique to selection buttons. See [Appendix B, Common Object Attributes](#), for information about the attributes that selection boxes and selection buttons have in common with other UIL objects.



## Selection Button Attributes

Attribute	Description
<b>configuration</b>	<p>If you want to display this selection button in a non-standard color, you can specify a customized configuration here to provide that color.</p> <p><i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Default value:</i> The configuration of the selection box that contains this selection button.</p>
<b>uil-handler-method</b>	<p>Names the procedure called when the button is selected.</p> <p><i>Allowable values:</i> Valid procedure name. The procedure must have the require arguments for handler methods.</p> <p><i>Default value:</i> uil-handle-button-method</p>
<b>uil-label-constant</b>	<p>Stores a symbol used as a look-up constant that GUIDE/UII uses to provide different natural language versions of text in text objects (<b>uil-text</b>) and buttons (<b>uil-button</b>). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> Any symbol.</p> <p><i>Default value:</i> unspecified</p>
<b>uil-translation-constraints</b>	<p>Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.</p> <p><i>Allowable values:</i> left, right, or center</p> <p><i>Default value:</i> left</p>

## Selection Button Attributes

Attribute	Description
<b>label</b>	The text of a <code>uil-text</code> instance operating as a label for this object.  <i>Allowable values:</i> Any <code>uil-text</code> object <i>Default value:</i> "radio button" or "check button"
<b>on-value</b>	The value of the button when selected.  <i>Allowable values:</i> Any symbol <i>Default value:</i> radio-button
<b>off-value</b>	The value of the button when not selected.  <i>Allowable values:</i> Any symbol <i>Default value:</i> radio-button or check-button
<b>toggle-state</b>	Indicates the current state of the button, on or off.  <i>Allowable values:</i> on or off <i>Default value:</i> off
<b>callback</b>	A symbol specifying the name of the callback procedure executed when the button is selected.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for callback procedures.  <i>Default value:</i> uil-do-nothing

## Operations Performed on Selection Buttons

The following table lists the common operations that you can perform on selection buttons.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

---

### Operations Performed on Selection Boxes

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobject	n/a
	Change Style	uil-change-window-style-of-grobject	n/a
✓	Clone	uil-clone-grobject	uil-clone-grobject-method
✓	Conclude	uil-conclude-grobject	uil-conclude-grobject-method
	Conclude Wrapper	uil-call-conclude-method-for-grobject	n/a
✓	Configure	uil-configure-grobject	uil-configure-grobject-method
	Create	n/a	n/a
✓	Delete	uil-delete-grobject	uil-delete-grobject-method
✓	Disable	uil-disable-grobject	uil-disable-grobject-method
✓	Enable	uil-enable-grobject	uil-enable-grobject-method
	Get Size (As symbol)	uil-return-size-of-button-as-symbol	n/a
	Get State	n/a	n/a
✓	Handler	uil-handle-button	uil-handle-button-method
	Highlight	uil-highlight-button	n/a
✓	Initialization	uil-initialize-grobject	uil-initialize-grobject-method
	Make Permanent	uil-make-grobject-permanent	n/a
✓	Manage	uil-manage-grobject	uil-manage-grobject-method

## Operations Performed on Selection Boxes

	Operation	Procedure	Method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobject	n/a
	Reset	uil-reset-button	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-select-button	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobject	uil-size-of-grobject-method
	Unmanage	uil-unmanage-grobject	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobject	uil-update-grobject-method
	Update Wrapper	uil-call-update-method-for-grobject	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Selection Box and Selection Button Relations

For information about the relations between selection buttons and selection boxes, see [Selection Box and Selection Button Relations](#).

## Procedures that Operate on Selection Buttons

This section describes a UIL procedures that operates on selection buttons.

`uil-get-next-selection-button-y-position`

`uil-get-selection-button`

`uil-is-this-button-grouped-by-this-selection-box`

# uil-get-next-selection-button-y-position

Returns the recommended y-position of the next selection button.

## Synopsis

uil-get-next-selection-button-y-position  
(*button-label*: class uil-text)  
-> *y-position*: integer

Argument	Description
button-label	The uil-text object that contains the label for the button to use for determining the y-offset for the next button.

Return Value	Description
<i>y-position</i>	The y-position for the next selection button in the selection box.

## Description

uil-get-next-selection-button-y-position determines the position for the next (lower) selection button in the selection box with reference to the button label of the object specified by *button-label*. GUIDE uses this procedure internally to construct selection boxes.

# uil-get-selection-button

Returns a selection button whose ID or name matches the *id-or-name* argument.

## Synopsis

uil-get-selection-button

(*id-or-name*: text)

-> {*selection-button*: class uil-selection-button | not-found: symbol}

Argument	Description
<i>id-or-name</i>	The ID or name of the selection button to get.

Return Value	Description
<u><i>selection-button</i></u> :	The selection button, or the symbol not-found.

# uil-is-this-button-grouped-by-this-selection-box

Returns a truth-value indicating whether or not the specified selection button is managed by the specified selection box.

## Synopsis

uil-is-this-button-grouped-by-this-selection-box  
(*button*: item-or-value, *box*: item-or-value)  
-> *result*: truth-value

Argument	Description
<i>button</i>	The selection button to check.
<i>box</i>	The selection box to check.

Return Value	Description
<u><i>result</i></u>	If the selection button is managed by the selection box, then TRUE, else FALSE.



## Additional Procedures that Support Selection Buttons

[uil-add-label-to-button](#)  
[uil-create-custom-button](#)  
[uil-create-text-for-button-from-label](#)  
[uil-disable-button](#)  
[uil-enable-button](#)  
[uil-handle-button](#)  
[uil-handle-button-method](#)  
[uil-highlight-button](#)  
[uil-reset-button](#)  
[uil-return-size-of-button-as-symbol](#)  
[uil-select-button](#)  
[uil-toggle-button](#)  
[uil-update-button-label](#)  
[uil-update-button-togglelabel](#)  
[uil-create-check-button](#)  
[uil-create-styled-check-button](#)  
[uil-create-styled-check-button-with-different-sized-label](#)  
[uil-delete-check-button](#)  
[uil-create-radio-button](#)  
[uil-create-styled-radio-button](#)  
[uil-create-styled-radio-button-with-different-sized-label](#)  
[uil-delete-radio-button](#)  
[uil-utility-check-for-orphaned-selection-buttons](#)  
[util-copy-object-attributes](#)  
[util-lift-to-top](#)



# Separators

---

*Describes procedures that operate on separators.*

Introduction **499**

Procedures that Operate on Separators **504**

    uil-create-line-separator **505**

    uil-rotate-line-separator **506**

Additional Procedures that Support Separators **507**



## Introduction

Separators are lines that you can use to organize the contents of workspaces or dialog subworkspaces visually. Separators have two orientations, horizontal and vertical.

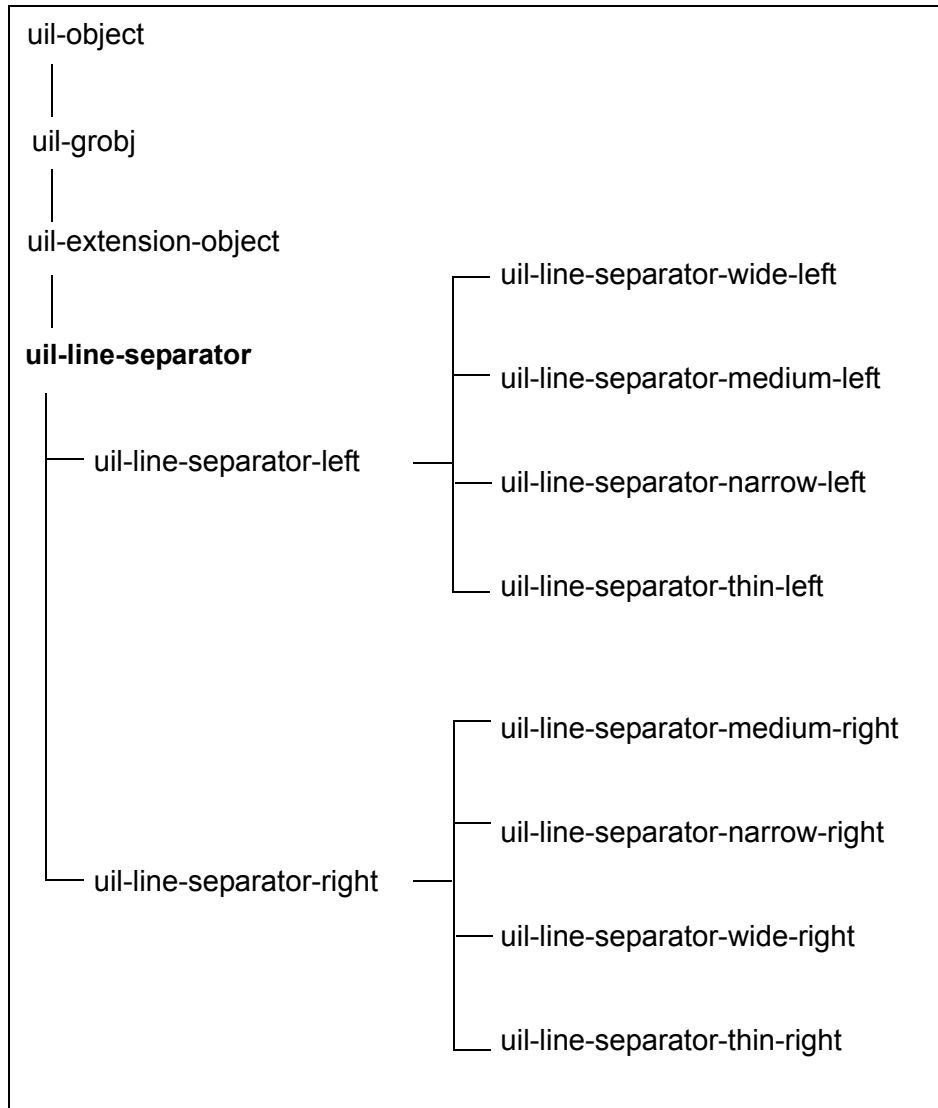
Separators do not have values or state.

### Separator Class Hierarchy

All separators inherit from the class uil-line-separator.

The following figure illustrates the UIL class hierarchy for separators. All classes shown are object-definitions.

## Separator Class Hierarchy



## Separator Attributes

The following table describes the attribute of separators that specifies the current separator configuration.

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this separator.
<i>Allowable values:</i>	Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .
<i>Default value:</i>	uil-line-and-border-configuration

For information about other attributes of separators, see [Appendix B, Common Object Attributes](#).

## Operations Performed on Separators

The following table lists the common operations that application developers can perform on uil-line-separator object.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations Performed on Separators

	Operation	Procedure	Method
	Change Size	n/a	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method

## Operations Performed on Separators

	Operation	Procedure	Method
	Create	uil-create-line-separator ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	n/a	n/a
✓	Enable	n/a	n/a
	Get Size (As symbol)	n/a	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialize	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	n/a	n/a
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	n/a	n/a
✓	Update	n/a	n/a
	Update Wrapper	n/a	n/a

**Operations Performed on Separators**

	<b>Operation</b>	<b>Procedure</b>	<b>Method</b>
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

# Procedures that Operate on Separators

This section describes procedures that operate on separators.

- uil-create-line-separator
- uil-rotate-line-separator



# uil-create-line-separator

Creates a line separator in one of four thicknesses, with a user-defined length (the distance between the two ends) and a choice of two orientations.

## Synopsis

uil-create-line-separator

(*line-size*: symbol, *length*: integer, *orientation*: symbol)

-> *line-separator*: class uil-line-separator

Argument	Description
<i>line-size</i>	Specify wide, medium, narrow, or thin.
<i>length</i>	The length of the line separator.
<i>orientation</i>	Specify left-to-right or top-to-bottom.
Return Value	Description
<u><i>line-separator</i></u>	The created line separator.

## Description

A line separator consists of three parts:

- a right end (a uil-line-separator-right),
- a left end (a uil-line-separator-left),
- a middle (a uil-line-separator-connector).

You can drag the left or right end of a separator to resize or move the separator.

# uil-rotate-line-separator

Rotates the specified line separator by 90 degrees.

## Synopsis

uil-rotate-line-separator

(*separator*: class uil-line-separator)

Argument	Description
<i>separator</i>	The line separator to rotate.

# Additional Procedures that Support Separators

[utl-copy-object-attributes](#)

[utl-lift-to-top](#)



# Text Objects

---

*Describes procedures that operate on text objects.*

Introduction **509**

Procedures that Operate on Text Objects **515**

- uil-create-styled-text **516**
- uil-get-gobj-for-label **517**
- uil-get-label-text **518**
- uil-select-text **519**
- uil-translate-text **520**

Additional Procedures that Support Text Objects **521**

Superseded Procedures **522**

- uil-create-text **523**
- uil-delete-text **524**



## Introduction

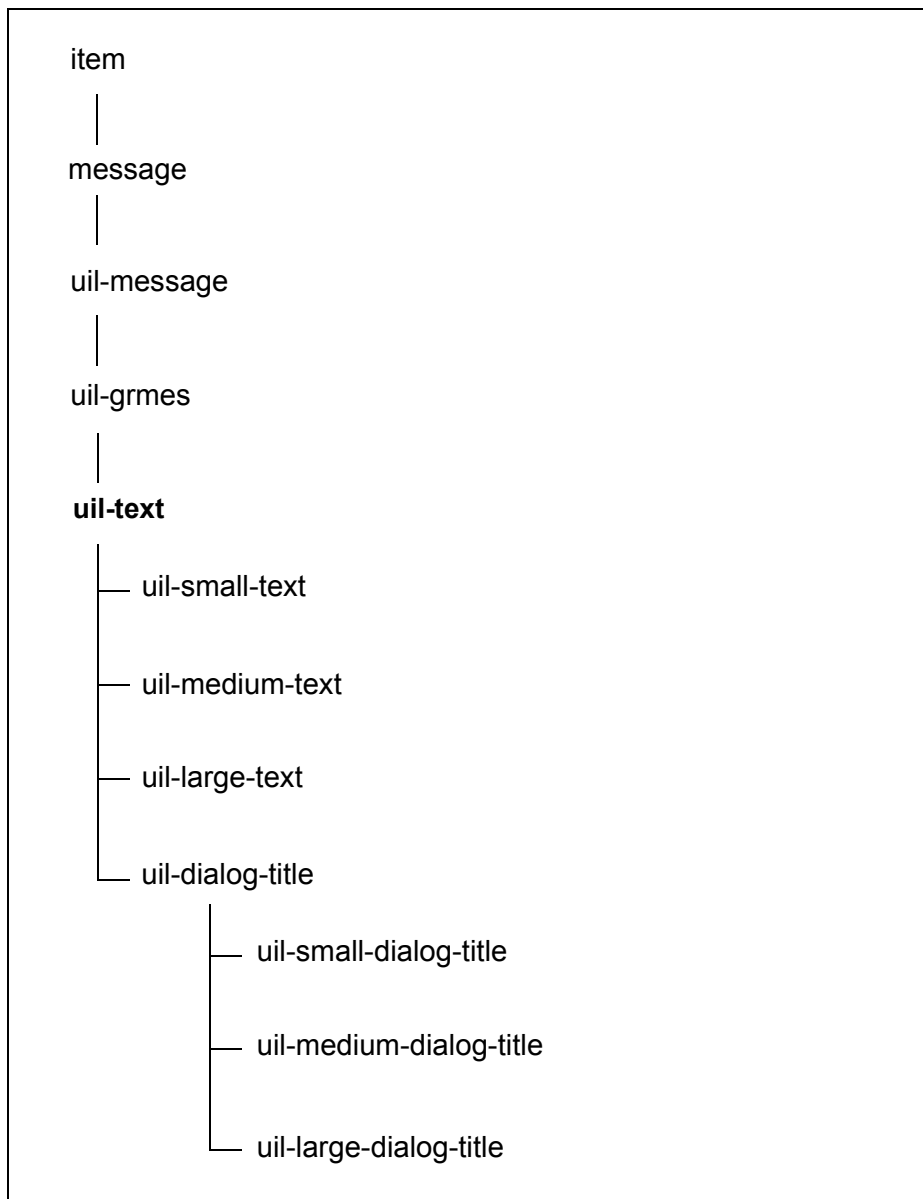
Text objects are fields that display text to the user. Users cannot edit the values in text objects. You can place borders around text objects.

### Text Object Class Hierarchy

All text objects inherit from the class `uil-text`.

The following figure illustrates the UIL class hierarchy for text objects. All classes below `message` are message-definitions.

## Text Object Class Hierarchy



## Text Object Attributes

The following tables list attributes that are unique to text objects. See [Appendix B, Common Object Attributes](#), for information about the attributes that text objects have in common with other UIL objects.

## Text Object Attributes

Attribute	Description
<b>configuration</b>	Specifies the configuration that determines the color of this text object.  <i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i> .  <i>Default value:</i> uil-text-configuration
<b>uil-maximum-characters-to-display</b>	Specifies the number of characters to display before clipping the text visible in the text object. The default value is the symbol unlimited.  <i>Allowable values:</i> Any integer  <i>Default value:</i> unlimited
<b>uil-label-constant</b>	Stores a symbol used as a look-up constant that GUIDE/UII uses to provide different natural language versions of text in text objects ( <i>uil-text</i> ) and buttons ( <i>uil-button</i> ). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i> .  <i>Allowable values:</i> Any symbol.  <i>Default value:</i> unspecified
<b>uil-translation-constraints</b>	Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.  <i>Allowable values:</i> left, right, or center  <i>Default value:</i> left

## Text Object Attributes

Attribute	Description
<b>uil-format-specification</b>	Refers to the name of a format (object of class uil-format-specification). <i>Allowable values:</i> The name of an existing format. <i>Default value:</i> unspecified
<b>uil-stand-alone-text</b>	Indicates whether the text object is stand-alone, or is related to a button, edit box, or any other object. <i>Allowable values:</i> <b>true</b> (text object is stand alone) or <b>false</b> (the text object is related to another object) <i>Default value:</i> <b>true</b>
<b>message-contents</b>	The current unformatted contents of the text object. <i>Allowable values:</i> Any text <i>Default value:</i> <b>text</b>

## Operations Performed on Text Objects

The following table lists the common operations that can be performed on text objects.

---

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

---



### Operations Performed on Text Objects

	Operation	Procedure	Method
	Change Size	uil-change-size-of-grobj	n/a
	Change Style	n/a	n/a
✓	Clone	uil-clone-grobj	uil-clone-grobj-method
✓	Conclude	n/a	n/a
	Conclude Wrapper	n/a	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-text ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size	uil-return-size-of-grmes- as-symbol	n/a
	Get State	n/a	n/a
✓	Handler	n/a	n/a
	Highlight	n/a	n/a
✓	Initialize	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	n/a	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-select-text ***	n/a

## Operations Performed on Text Objects

	Operation	Procedure	Method
	Set State	n/a	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validation Method	n/a	n/a
	Validate Wrapper	n/a	n/a

## Text Object Relations

The following table summarizes the standard relations between text objects and UIL objects that use the text objects as labels. You can reference these relations to access the text object and UIL objects programmatically.

### Text Object Relations

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label

**Note:** The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.

Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UIL earlier than Version 5.0 and were not converted to have their label text in their icons.

## Procedures that Operate on Text Objects

This section describes procedures that operate on text objects.

- uil-create-styled-text
- uil-get-grobj-for-label
- uil-get-label-text
- uil-select-text
- uil-translate-text

# uil-create-styled-text

Creates and returns a text object.

## Synopsis

uil-create-styled-text

(*id*: text, *size*: symbol, *initial-value*: item-or-value, *style*: item-or-value)

-> text-object: class uil-text)

Argument	Description
<i>id</i>	The ID for the text object.
<i>size</i>	The size of text to create. Specify one of the values: <b>small</b> , <b>medium</b> , or <b>large</b> .
<i>initial-value</i>	The value for the text object.
<i>style</i>	The symbol <b>none</b> (place holder for style object).

Return Value	Description
<u>text-object</u>	The created text object.

## uil-get-grobj-for-label

Returns the UIL object associated with the uil-text object containing the label text.

### Synopsis

```
uil-get-grobj-for-label
  (label: item-or-value )
  -> result: item-or-value
```

Argument	Description
<i>label</i>	The uil-text object that provides the text of another object's label

Return Value	Description
<u>result</u>	The UIL object that is associated with the uil-text object containing the label text.  If the associated item is not found, this procedure returns the symbol <code>item-not-found</code> .

# uil-get-label-text

Returns the text object that provides the label for a specified UIL control.

## Synopsis

uil-get-label-text  
(*obj*: class item )  
-> result: item-or-value

Argument	Description
<i>obj</i>	The object with the label.

Return Value	Description
<u>result</u>	The uil-text object that provides the text of the label. The text of the label is in the message-contents attribute of the object.  If no uil-text object is found, this procedure returns not-found.

## uil-select-text

Selects the specified text object.

### Synopsis

uil-select-text

(*txt*: class uil-text, *window*: class g2-window)

Argument	Description
<i>txt</i>	The text to be selected.
<i>window</i>	The window on which the dialog or workspace containing the text is managed.

### Description

If the text object is related to a uil-button object, then the button is also selected. The invokes the uil-handler-method for the button, which in turn calls the callback for the button if one is specified.

# uil-translate-text

Specifies a GFR Resource Group that provide text in different languages for a uil-text or uil-button.

## Synopsis

uil-translate-text

(*text-item*: item-or-value, *resource-group*: class gfr-text-resource-group, *gfr-version*: float, *gfr-revision*: integer, *window*: class g2-window)

Argument	Description
<i>text-item</i>	Any UIL control that inherits from uil-text or uil-button.
<i>resource-group</i>	A GFR text resource group. Text for <i>text-item</i> is provided by the GFR local text resource in <i>resource-group</i> that corresponds to the currently selected language in <i>window</i> .
<i>gfr-version</i>	The version of GFR used to create <i>resource-group</i> .
<i>gfr-revision</i>	The revision of GFR used to create <i>resource-group</i> .
<i>window</i>	The G2 window where the text is displayed.



## Additional Procedures that Support Text Objects

[uil-30r3-upgrade-button-label-texts](#)  
[uil-30r3-upgrade-radio-and-check-button-label-texts](#)  
[uil-utility-synchronize-text-fields-for-uil-text](#)  
[uti-copy-object-attributes](#)  
[uti-lift-to-top](#)  
[uil-create-text-for-button-from-label](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UII, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

uil-create-text  
uil-delete-text

## uil-create-text

Creates and returns a text object. Recommended procedure: uil-create-styled-text.

### Synopsis

uil-create-text

(*id*: text, *initial-value*: item-or-value, *size*: symbol)

-> text-object: class uil-text

Argument	Description
<i>id</i>	The ID for the text to be created.
<i>initial-value</i>	The initial value of the text.
<i>size</i>	The size of the text object. Specify <b>small</b> , <b>medium</b> , or <b>large</b> .

Return Value	Description
<u>text-object</u>	The created text.

## uil-delete-text

Deletes the specified text object. Recommended: uil-delete-grobj-method.

### Synopsis

uil-delete-text  
(*txt*: class uil-text)

Argument	Description
<i>txt</i>	The text object to delete.

# Toggle Buttons

---

*Describes procedures that operate on toggle buttons.*

Introduction **525**

Procedures that Operate on Toggle Buttons **533**

    uil-create-styled-toggle-button **534**

    uil-get-toggle-button-state **535**

    uil-set-toggle-button-state **536**

Additional Procedures that Support Toggle Buttons **537**

Superseded Procedures **538**

    uil-create-toggle-button **539**

    uil-delete-toggle-button **540**



## Introduction

Toggle buttons consist of two UIL classes, `uil-icon-toggle-button` and `uil-text-toggle-button`. The difference lies in the placement of the label that is associated with the toggle button. For icon toggle buttons, the label is placed to the right of the button. For text toggle buttons, the label is placed on top of the button. In both cases, selecting the label has the same effect as selecting the button itself.

---

**Note** In GUIDE/UIL, the labels of Motif-style toggle buttons are part of the button icons, and are not contained in separate text objects. However, the labels of Windows-style toggle buttons are contained in separate text objects, as are the labels of all buttons created with versions of GUIDE/UIL earlier than 5.0.

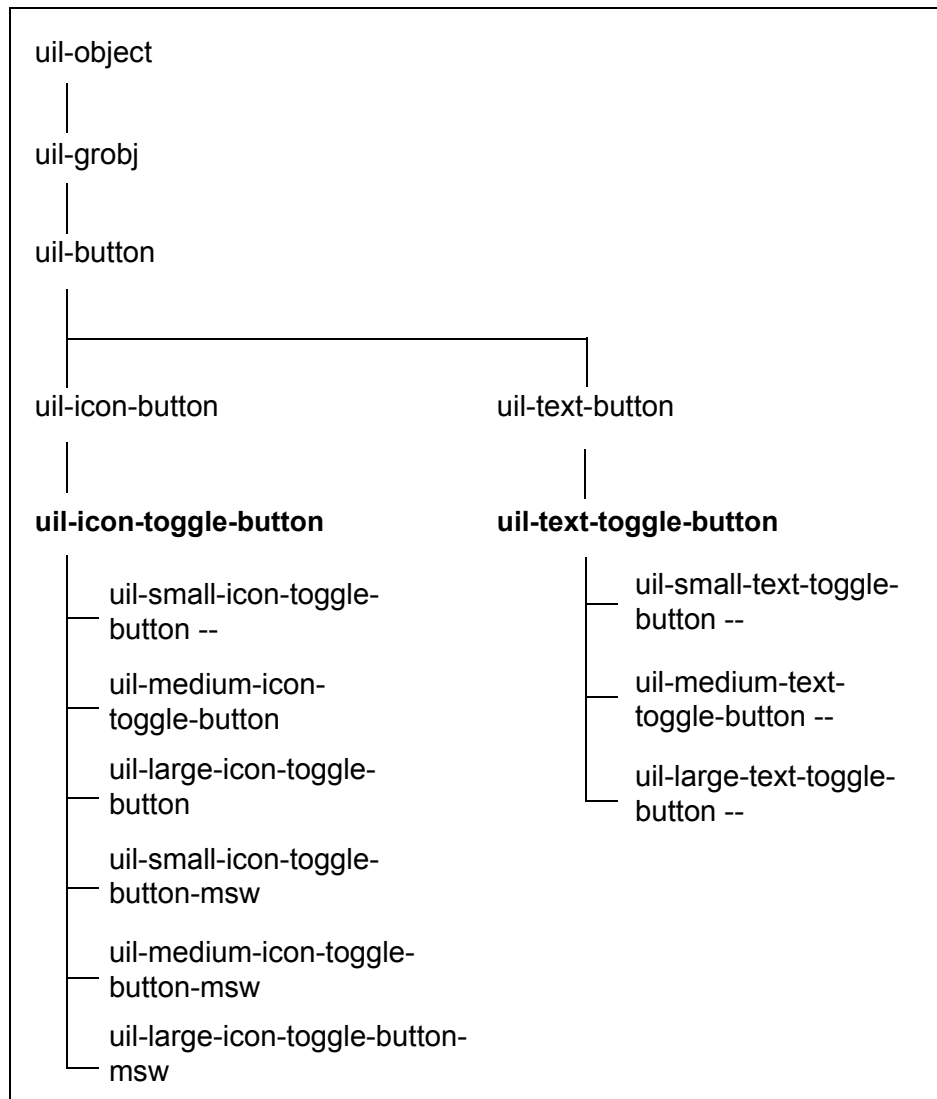
---

## Toggle Button Class Hierarchy

All toggle buttons inherit from the class `uil-button`.

The following figure illustrates the UIL class hierarchy for toggle buttons. All classes shown are object-definitions. A class whose name is followed by dashes (for example, `uil-large-text-toggle-button --`) has subclasses that are not shown in the figure.

Toggle Button Class Hierarchy



## Toggle Button Attributes

### Toggle Button Attributes

Attribute	Description
<b>configuration</b>	<p>Specifies the configuration that determines the color of this toggle button. Icon toggle buttons and text toggle buttons have different default configurations.</p> <p><i>Allowable values:</i> Any valid configuration. For information about configurations, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Default value:</i> <code>uil-icon-toggle-button-configuration</code> or <code>uil-text-toggle-button-configuration</code></p>
<b>uil-handler-method</b>	<p>Names a procedure called when the button is selected.</p> <p><i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for handler methods.</p> <p><i>Default value:</i> <code>uil-handle-button-method</code></p>
<b>uil-label-constant</b>	<p>Stores a symbol used as a look-up constant that GUIDE/UII uses to provide different natural language versions of text in text objects (<code>uil-text</code>) and buttons (<code>uil-button</code>). For information about how to internationalize dialogs in this way, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> Any symbol.</p> <p><i>Default value:</i> <code>unspecified</code></p>

## Toggle Button Attributes

Attribute	Description
<b>uil-translation-constraints</b>	Specifies that the left edge, right edge, or center of the label text be maintained if the label text is translated into a different natural language.  <i>Allowable values:</i> left, right, or center  <i>Default value:</i> left
<b>label</b>	The text of a <code>uil-text</code> instance operating as a label for this object.  <i>Allowable values:</i> text value  <i>Default value:</i> "on"
<b>callback</b>	Specifies the name of the callback procedure executed by the handler method when the button is selected.  <i>Allowable values:</i> Valid procedure name. The procedure must have the correct arguments for callback procedures.  <i>Default value:</i> uil-do-nothing
<b>toggle-state</b>	Indicates the current on or off state of the button.  <i>Allowable values:</i> on or off  <i>Default value:</i> on



### Toggle Button Attributes

Attribute	Description
<b>toggle-label</b>	The text value used as the button's label when toggle state is off.  <i>Allowable values:</i> Any text <i>Default value:</i> off
<b>on-value</b>	The value of the button in the on state.  <i>Allowable values:</i> symbol <i>Default value:</i> on
<b>off-value</b>	The value of the button in the off state.  <i>Allowable values:</i> symbol <i>Default value:</i> off
<b>uil-resize-to-fit-label</b>	(for Motif-style uil-text-toggle-button only) If true, GUIDE/UIIL automatically resizes the toggle button to fit the label text that you specify for the button. Otherwise, GUIDE/UIIL does not resize the button.  You can set this attribute using the Edit Toggle Button dialog. For information about how to use this dialog, see the <i>G2 GUIDE User's Guide</i> .  <i>Allowable values:</i> true of false <i>Default value:</i> true

## Toggle Button Attributes

Attribute	Description
<b>uil-minimum-width</b>	(for Motif-style uil-text-toggle-button only) The minimum width to which this toggle button can be resized to fit the label. This attribute is ignored if Uil resize to fit label is false.  <i>Allowable values:</i> positive integers  <i>Default value:</i> 0
<b>uil-minimum-height</b>	(for Motif-style uil-text-toggle-button only) The minimum height to which this toggle button can be resized to fit the label. This attribute is ignored if Uil resize to fit label is false.  <i>Allowable values:</i> positive integer  <i>Default value:</i> 0

## Operations Performed on Toggle Buttons

The following table lists common operations that can be performed on toggle buttons.

**Note** A check mark in the left column indicates that the method for performing the operation is referenced from the object's attribute table.

Procedures whose names are followed by asterisks (\*\*\*) are described in this chapter.

### Operations Performed on Toggle Buttons

	Operation	Procedure	Method
	Change Size	uil-change-size-of-gobj	n/a
	Change Style	uil-change-window-style-of-gobj	n/a
✓	Clone	uil-clone-gobj	uil-clone-gobj-method
✓	Conclude	uil-conclude-gobj	uil-conclude-gobj-method

### Operations Performed on Toggle Buttons

	Operation	Procedure	Method
	Conclude Wrapper	uil-call-conclude-method-for-grobj	n/a
✓	Configure	uil-configure-grobj	uil-configure-grobj-method
	Create	uil-create-styled-toggle-button ***	n/a
✓	Delete	uil-delete-grobj	uil-delete-grobj-method
✓	Disable	uil-disable-grobj	uil-disable-grobj-method
✓	Enable	uil-enable-grobj	uil-enable-grobj-method
	Get Size (As symbol)	uil-return-size-of-button-as-symbol	n/a
	Get State	uil-get-toggle-button-state ***	n/a
✓	Handler	uil-handle-button	uil-handle-button-method
	Highlight	uil-highlight-button	n/a
✓	Initialize	uil-initialize-grobj	uil-initialize-grobj-method
	Make Permanent	uil-make-grobj-permanent	n/a
✓	Manage	uil-manage-grobj	uil-manage-grobj-method
✓	Message Selection	n/a	n/a
✓	Message Unselection	n/a	n/a
	Move	uil-move-grobj	n/a
	Reset	uil-reset-button	n/a
	Resize	n/a	n/a
	Set/Select/Toggle	uil-toggle-button	n/a
	Set State	uil-set-toggle-button-state ***	n/a
✓	Size (W/H)	uil-size-of-grobj	uil-size-of-grobj-method
	Unmanage	uil-unmanage-grobj	n/a

## Operations Performed on Toggle Buttons

	Operation	Procedure	Method
	Unset/Unselect	n/a	n/a
✓	Update	uil-update-grobj	uil-update-grobj-method
	Update Wrapper	uil-call-update-method-for-grobj	n/a
✓	Validation	n/a	n/a
	Validate Wrapper	n/a	n/a

## Toggle Button Relations

The following table summarizes standard relations between toggle buttons and their labels. You can reference these relations to access the toggle buttons and their labels programmatically.

### Relations Between Toggle Buttons and Their Labels

Class of Subject item	Relation	Class of Object item	Type of Relation	Inverse Relation
uil-text	the-uil-text-label-of	item, with text label	one-to-one	the-uil-button-owning-this-text-label
<p><b>Note:</b> The the-uil-text-label-of relation applies only to objects whose labels are contained in separate text objects.</p> <p>Thus, it applies only to Windows-style buttons and to Motif-style buttons that were created with versions of GUIDE/UII earlier than Version 5.0 and were not converted to have their label text in their icons.</p>				

## Procedures that Operate on Toggle Buttons

This section describes procedures that operate on toggle buttons.

- uil-create-styled-toggle-button
- uil-get-toggle-button-state
- uil-set-toggle-button-state

# uil-create-styled-toggle-button

Creates and returns a toggle button.

## Synopsis

uil-create-styled-toggle-button

(*id*: text, *kind*: symbol, *size*: symbol, *label*: text, *toggle-label*: text,  
*on-value*: value, *off-value*: value, *callback*: symbol, *style*: item-or-value)  
-> toggle-button: class uil-button

Argument	Description
<i>id</i>	The ID for the toggle button.
<i>kind</i>	The type of toggle button. Specify <b>text</b> or <b>iconic</b> .
<i>size</i>	The size of the toggle button.  For Motif style buttons, specify <b>small</b> , <b>medium</b> , or <b>large</b> .  For Windows style buttons, specify <b>small-msw</b> , <b>medium-msw</b> , or <b>large-msw</b> .
<i>label</i>	The text to use as the label for the toggle button.
<i>toggle-label</i>	The text to use as the toggle label for the toggle button.
<i>on-value</i>	The value of the button when it is selected (toggle-state is on).
<i>off-value</i>	The value of the button when it is not selected (toggle-state is off).
<i>callback</i>	The name of a procedure that is called when the toggle button is selected.
<i>style</i>	The symbol <b>none</b> (place holder for style object).
Return Value	Description
<u>toggle-button</u>	The created toggle button.

## uil-get-toggle-button-state

Returns the current state of the specified toggle button.

### Synopsis

uil-get-toggle-button-state

(*dialog-or-workspace*: class item, *id*: text)

-> {*button-state*: symbol | uil-none: symbol}

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace containing the toggle button.
<i>id</i>	The ID of the toggle button to return the state of.
Return Value	Description
<u><i>button-state</i></u>	The state of the specified toggle button (the symbol <code>on</code> or the symbol <code>off</code> ), or the symbol <code>uil-none</code> if the button cannot be found.

## uil-set-toggle-button-state

Sets the state of the specified button to the state reflected by the state argument.

### Synopsis

uil-set-toggle-button-state

(*dialog-or-workspace*: class item, *id*: text, *state*: symbol)

Argument	Description
<i>dialog-or-workspace</i>	The dialog or workspace on which the button appears.
<i>id</i>	The ID of the button to set the state of.
<i>state</i>	The state to set button. Specify on or off.



## Additional Procedures that Support Toggle Buttons

[uil-add-label-to-button](#)  
[uil-create-custom-button](#)  
[uil-create-text-for-button-from-label](#)  
[uil-disable-button](#)  
[uil-enable-button](#)  
[uil-handle-button](#)  
[uil-handle-button-method](#)  
[uil-highlight-button](#)  
[uil-reset-button](#)  
[uil-return-size-of-button-as-symbol](#)  
[uil-select-button](#)  
[uil-toggle-button](#)  
[uil-update-button-label](#)  
[uil-update-button-togglelabel](#)  
[util-copy-object-attributes](#)

## Superseded Procedures

The following procedures are supported by this version of GUIDE/UII, but do not take advantage of features introduced in GUIDE 4.0 and subsequent versions.

uil-create-toggle-button  
uil-delete-toggle-button

# uil-create-toggle-button

Creates and returns a toggle button. Use instead: uil-create-styled-toggle-button.

## Synopsis

uil-create-toggle-button

(*id*: text, *kind*: symbol, *callback*: symbol,  
*size*: symbol, *label*: text, *toggle-label*: text)  
 -> *toggle-button*: class uil-button

Argument	Description
<i>id</i>	The ID for the toggle button.
<i>kind</i>	The type of toggle button. Specify <b>text</b> or <b>iconic</b> .
<i>callback</i>	The name of a procedure that is called when the toggle button is selected.
<i>size</i>	The size of the toggle button.  For Motif style buttons, specify <b>small</b> , <b>medium</b> , or <b>large</b> .  For Windows style buttons, specify <b>small-msw</b> , <b>medium-msw</b> , or <b>large-msw</b> .
<i>label</i>	The text to use as the label for the toggle button.
<i>toggle-label</i>	The text to use as the toggle-label for the toggle button.
Return Value	Description
<u><i>toggle-button</i></u>	The created toggle button

## uil-delete-toggle-button

Deletes the specified toggle button. Use instead: uil-delete-grobject-method.

### Synopsis

uil-delete-toggle-button  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The toggle button to delete.

# Items and Relations

---

*Summarizes UIL procedures for accessing UIL objects in relations, and describes procedures that access information about relations themselves.*

Procedures for Accessing UIL Objects in Relations **542**

Procedures that Access Information about Relations **545**

    uil-are-any-related-to **546**

    uil-is-related-to **547**

    uil-get-related-item **548**

    uil-get-related-items **549**



# Procedures for Accessing UIL Objects in Relations

The following table summarizes the UIL procedures that access UIL objects in relations.

**Procedures that Access Objects through Relations**

Relation/Inverse	Class	UIL Procedure
the-uil-text-label-of	uil-text	uil-get-label-text
the-uil-button-owning-this-text-label	item	uil-get-grobj-for-label
the-uil-boxed-message-of	uil-grmes	uil-get-message-for-border
none	uil-extension-object	—
the-bottommost-message-of	uil-message-object	uil-get-last-message
none	uil-scroll-area	—
the-topmost-message-of	uil-message-object	uil-get-first-message
none	uil-scroll-area	—
a-message-belonging-to	uil-message-object	uil-does-message-belong-to-scroll-area
the-handler-for-the-message-of	uil-scroll-area	uil-get-scroll-area-for-this-message
a-selected-message-of	uil-message-object	uil-is-message-selected
none	uil-scroll-area	uil-get-selected-message-objects
the-current-edit-field-of	message	uil-get-control-with-focus
none	dialog	—
the-uil-editor-field-mask-for	uil-grmes	<i>Avoid this completely</i>
none	uil-grmes	—

### Procedures that Access Objects through Relations

Relation/Inverse	Class	UIL Procedure
the-current-uil-message-being-edited-on	message	uil-get-control-with-focus-on-window
none	g2-window	—
the-object-receiving-dialog-activity-from	item	uil-find-destination-object
none	item	—
a-uil-dialog-presently-in-use-on-window	uil-dialog	uil-is-dialog-managed-on-window uil-get-window-on-which-dialog-is-managed
the-g2-window-now-showing-dialog	g2-window	uil-get-window-on-which-dialog-is-managed uil-get-dialogs-managed-on-window
the-uil-dialog-resulting-from-action	item	uil-get-dialog-launched-for-item uil-is-this-dialog-managed-for-this-item
the-object-initiating-dialog-activity-for	item	uil-find-initiating-object
a-uil-child-cascaded-dialog-of	uil-dialog	uil-is-this-a-child-dialog-of-this-dialog uil-get-child-dialogs
the-uil-parent-of-this-cascaded-dialog	uil-dialog	uil-is-this-the-parent-dialog uil-get-parent-dialog
the-uil-temporary-storage-object-of	item	uil-find-temporary-storage-object
a-uil-dialog-using-the-temporary-storage-object	uil-dialog	uil-get-dialog-for-this-temporary-storage-object

### Procedures that Access Objects through Relations

Relation/Inverse	Class	UIL Procedure
the-item-appended-to-the-uil-scroll-message	item	uil-get-item-appended-to-message
the-uil-message-appended-to-the-item	uil-message-object	uil-get-message-appended-to-this-item
the-group-master-of grouped-by	uil-selection-box uil-selection-button	uil-get-selection-box-for-button uil-get-buttons-grouped-by-this-selection-box uil-is-this-button-grouped-by-this-selection-box



## Procedures that Access Information about Relations

The section contains descriptions of procedures that access information about relations.

- uil-are-any-related-to
- uil-is-related-to
- uil-get-related-item
- uil-get-related-items

# uil-are-any-related-to

Returns true if there exist any items related to the target item by the specified relation.

## Synopsis

uil-are-any-related-to

(*TheRelationName*: symbol, *TheObject*: item-or-value)

-> *result*: truth-value

Argument	Description
<i>TheRelationName</i>	The name of a relation.
<i>TheObject</i>	An object to check against <i>TheRelationName</i> .

Return Value	Description
<u><i>result</i></u>	If there are any items related to <i>TheObject</i> by the specified relation, then TRUE, else FALSE.

## Example

```
TheObj: class item;  
AnyRelated: truth-value;  
TheObj = call uil-get-grobj-from-id-on-dlg-or-wksp ("my-edit-box", dialog);  
AnyRelated = call uil-are-any-related-to (the symbol the-uil-text-label-of,  
TheObj);
```

## uil-is-related-to

Returns true if the test item is related to the target item through the specified relation.

### Synopsis

uil-is-related-to

(*TheRelatedItem*: item-or-value, *TheRelationName*: symbol,  
*TheObject*: item-or-value)

-> *result*: truth-value

Argument	Description
<i>TheRelatedItem</i>	A test item.
<i>TheRelationName</i>	The name of a relation.
<i>TheObject</i>	An object to check against <i>TheRelationName</i> and <i>TheRelatedItem</i> .

Return Value	Description
<u><i>result</i></u>	If <i>TheRelatedItem</i> is in the relation specified by <i>TheRelationName</i> with <i>TheObject</i> then TRUE, else FALSE.

### Example

```
TheObj: class item;
Label: class uil-text;
IsRelated: truth-value;
TheObj = call uil-get-grobject-from-id-on-dlg-or-wksp ("my-edit-box", dialog);
Label = call uil-get-grobject-from-id-on-dlg-or-wksp ("my-label", dialog);
IsRelated = call uil-is-related-to (Label, the-symbol the-uil-text-label-of,
TheObj);
```

# uil-get-related-item

Returns the item related to the target item by the specified relation.

## Synopsis

uil-get-related-item

(*TheRelationName*: symbol, *TheObject*: item-or-value)

-> result: item-or-value

Argument	Description
<i>TheRelationName</i>	The name of a relation.
<i>TheObject</i>	An object to check against <i>TheRelationName</i> for a related item.

Return Value	Description
<u>result</u>	The item related to <i>TheObject</i> by the specified relation, or the symbol no-related-items.

## Example

```
TheObj: class item;  
TheRelatedObj: item-or-value;  
TheObj = call uil-get-grobj-from-id-on-dlg-or-wksp ("my-edit-box", dialog);  
TheRelatedObj = call uil-get-related-item (the symbol the-uil-text-label-of,  
TheObj);
```

## uil-get-related-items

Returns the item(s) related to the target item by the specified relation.

### Synopsis

uil-get-related-item

(*TheRelationName*: symbol, *TheObject*: item-or-value

-> result: item-or-value

Argument	Description
<i>TheRelationName</i>	The name of a relation.
<i>TheObject</i>	An object to check against <i>TheRelationName</i> for related items.

Return Value	Description
<u>result</u>	A g2-list containing the items related to the target item by the specified relation. If no items are related to the target item, then the symbol no-related-items is returned.

### Example

TheRelatedObjs: item-or-value;

TheRelatedObjs = call uil-get-related-items (the symbol a-uil-child-cascaded-dialog-of, dialog);



# Upgrade and Utility Procedures

---

*Describes procedures that upgrade elements of your GUIDE user interface.*

## Upgrade Procedures **553**

### Upgrade Procedures **554**

- uil-30r1-upgrades **555**
- uil-30r3-and-30r4-upgrades **556**
- uil-30r3-upgrade-button-label-texts **557**
- uil-30r3-upgrade-radio-and-check-button-label-texts **558**
- uil-30r3-upgrade-scroll-area-keys **559**
- uil-40r0-upgrades **560**
- uil-convert-button-from-4.1-to-5.0 **561**
- uil-convert-buttons-from-4.1-to-5.0 **562**
- uil-convert-dialog-subworkspaces-from-4.1-to-5.0 **563**
- uil-convert-scroll-area-to-41r0 **564**

### Utility Procedures **565**

- uil-add-prefix-to-user-criteria-prefixes **567**
- uil-create-uilib-module-setting **568**
- uil-date-verification **569**
- uil-determine-if-guide-is-loaded **570**
- uil-display-color-selector-dialog **571**
- uil-do-nothing **573**
- uil-generate-rules-for-attributes **574**
- uil-get-attribute **575**
- uil-get-configuration **576**
- uil-get-text-extent **577**
- uil-get-user-criteria-prefixes **578**
- uil-guidata-launch-dialog-for-item **579**
- uil-is-this-item-near-another **580**
- uil-remove-guide-application **581**
- uil-remove-prefix-from-user-criteria-prefixes **582**
- uil-reset-uilib-module-setting-to-defaults **583**
- uil-reset-user-criteria-prefixes **584**
- uil-return-bin-workspace-for **585**
- uil-return-configuration-for-color **586**

uil-set-attribute 587  
uil-time-verification 588  
uil-utility-build-symbol-list-of-values-for-class-attribute 589  
uil-utility-change-module 591  
uil-utility-check-for-orphaned-selection-buttons 592  
uil-utility-cleanup-dialog-bin 593  
uil-utility-convert-from-edit-box-to-integer 594  
uil-utility-convert-from-edit-box-to-quantity 595  
uil-utility-convert-from-edit-box-to-symbol 596  
uil-utility-convert-from-edit-box-to-value 597  
uil-utility-create-procedure-xref-listing 598  
uil-utility-return-current-real-time 599  
uil-utility-synchronize-text-fields-for-uil-text 600  
uil-utility-unique-id 601  
uil-utility-unique-name 602  
utl-clone-through-copying 603  
utl-convert-from-text-to-integer 604  
utl-convert-from-text-to-quantity 605  
utl-convert-from-text-to-symbol 606  
utl-convert-from-text-to-value 607  
utl-convert-list-to-text 608  
utl-copy-object-attributes 609  
utl-copy-symbol-array-to-initial-values 610  
utl-copy-symbol-array-to-symbol-array 611  
utl-copy-symbol-array-to-symbol-list 612  
utl-copy-symbol-list-to-symbol-array 613  
utl-copy-symbol-list-to-symbol-list 614  
utl-copy-text-array-to-initial-values 615  
utl-copy-text-list-to-text-list 616  
utl-create-instance-of-class 617  
utl-get-module-information 618  
utl-get-module-owning-this-workspace 619  
utl-get-top-level-workspace-of-this-workspace 620  
utl-goto-superior 621  
utl-goto-workspace 622  
utl-hide-workspace 623  
utl-insert-text-in-alphabetized-list 624  
utl-insert-text-in-reversed-alphabetized-list 625  
utl-instances-exist 626  
utl-lift-to-top 627  
utl-remove-hyphens-from-text 628  
utl-remove-module-from-hierarchy 629  
utl-return-clipped-string 630  
utl-show-superior-workspace 631  
utl-transfer-to-at 632

Additional Utility Procedures 633





## Upgrade Procedures

This section describes procedures that upgrade elements of your GUIDE user interface. GUIDE runs these procedures automatically when it is initialized.

# Upgrade Procedures

[uil-30r1-upgrades](#)

[uil-30r3-and-30r4-upgrades](#)

[uil-30r3-upgrade-button-label-texts](#)

[uil-30r3-upgrade-radio-and-check-button-label-texts](#)

[uil-30r3-upgrade-scroll-area-keys](#)

[uil-40r0-upgrades](#)

[uil-convert-button-from-4.1-to-5.0](#)

[uil-convert-buttons-from-4.1-to-5.0](#)

[uil-convert-dialog-subworkspaces-from-4.1-to-5.0](#)

[uil-convert-scroll-area-to-41r0](#)

# uil-30r1-upgrades

## Synopsis

uil-30r1-upgrades ()

## Description

No-op for now. Calls utl-restore-scroll-area (replaced uil-restore-scroll-area). Refer to the GUIDE Version 3 Release Notes.

# uil-30r3-and-30r4-upgrades

## Synopsis

uil-30r3-and-30r4-upgrades ()

## Description

Deletes scroll area pieces (scroll-bar, scroll-arrows, scroll-thumb, scroll-segments) and makes sure that the object definition for the scroll-bar exists. Refer to the GUIDE Version 3.3 Release Notes.

# uil-30r3-upgrade-button-label-texts

## Synopsis

uil-30r3-upgrade-button-label-texts ()

## Description

Deletes and regenerates the uil-text objects that are labels of buttons (all uil-button types except for uil-selection-button). Refer to the GUIDE Version 3.3 Release Notes.

# **uil-30r3-upgrade-radio-and-check-button-label-texts**

## **Synopsis**

uil-30r3-upgrade-radio-and-check-button-label-texts ()

## **Description**

Deletes and regenerates the uil-text objects that are labels of selection buttons. Refer to the GUIDE Version 3.3 Release Notes.

# uil-30r3-upgrade-scroll-area-keys

## Synopsis

uil-30r3-upgrade-scroll-area-keys ()

## Description

Generates a unique key for all scroll-areas. Refer to the [GUIDE Version 3.3 Release Notes](#).

# uil-40r0-upgrades

## Synopsis

uil-40r0-upgrades ()

## Description

Moves selection boxes from the bins where they were stored in earlier GUIDE versions to the workspaces where their selection buttons reside.



## uil-convert-button-from-4.1-to-5.0

Converts a specified button created with GUIDE/UIL Version 4.1 to a Version 5.0-style button.

### Synopsis

uil-convert-button-from-4.1-to-5.0  
(*button*: class uil-button)

Argument	Description
<i>button</i>	The 4.1 button that is converted to a 5.0 button.

## **uil-convert-buttons-from-4.1-to-5.0**

Converts all buttons (uil-button) created with GUIDE/UIL Version 4.1 to Version 5.0-style buttons.

### **Synopsis**

```
uil-convert-buttons-from-4.1-to-5.0  
()
```

# uil-convert-dialog-subworkspaces-from-4.1-to-5.0

Converts dialog subworkspaces. Required only if you intend to provide extended context-sensitive help for GUIDE/UIL.

## Synopsis

uil-convert-dialog-subworkspaces-from-4.1-to-5.0  
(*window*: class g2-window)

Argument	Description
<i>window</i>	The G2 window on which dialog workspaces are converted.

## Description

You must run `uil-convert-dialog-subworkspaces-from-4.1-to-5.0()` only if you intend to provide extended context-sensitive help for UIL dialogs or items on UIL dialog.

You can also convert dialog subworkspaces for this purpose by selecting the following GUIDE Menu Bar choice:

Tools > GUIDE 5.0 Migration Tools > Convert dialog subworkspaces to uil-dialog-subworkspace

This menu choice opens a dialog that enables you to convert dialog subworkspaces. If you convert subworkspaces in this way, you do not need to use `uil-convert-dialog-subworkspaces-from-4.1-to-5.0()`.

For information about how to provide extended context-sensitive help for GUIDE/UIL, see the *G2 GUIDE User's Guide*.

# uil-convert-scroll-area-to-41r0

Converts a scroll area to GUIDE/UII Version 4.1 Rev. 0.

## Synopsis

uil-convert-scroll-area-to-41r0

(*scroll-area*: symbol, *window*: class g2-window)

Argument	Description
<i>scroll-area</i>	The scroll area to convert.
<i>window</i>	The G2 window where the scroll area is displayed.

# Utility Procedures

This section describes procedures that perform utility operations.

[uil-add-prefix-to-user-criteria-prefixes](#)

uil-create-uilib-module-setting

uil-date-verification

uil-determine-if-guide-is-loaded

uil-display-color-selector-dialog

uil-do-nothing

uil-generate-rules-for-attributes

uil-get-attribute

uil-get-configuration

uil-get-text-extent

uil-get-user-criteria-prefixes

uil-guidata-launch-dialog-for-item

uil-is-this-item-near-another

uil-remove-guide-application

uil-remove-prefix-from-user-criteria-prefixes

uil-reset-uilib-module-setting-to-defaults

uil-reset-user-criteria-prefixes

uil-return-bin-workspace-for

uil-return-configuration-for-color

uil-set-attribute

uil-time-verification

uil-utility-build-symbol-list-of-values-for-class-attribute

uil-utility-change-module

uil-utility-check-for-orphaned-selection-buttons

uil-utility-cleanup-dialog-bin

uil-utility-convert-from-edit-box-to-integer

uil-utility-convert-from-edit-box-to-quantity

uil-utility-convert-from-edit-box-to-symbol

uil-utility-convert-from-edit-box-to-value

uil-utility-create-procedure-xref-listing

uil-utility-return-current-real-time

uil-utility-synchronize-text-fields-for-uil-text

uil-utility-unique-id

uil-utility-unique-name

utl-clone-through-copying

utl-convert-from-text-to-integer

utl-convert-from-text-to-quantity

utl-convert-from-text-to-symbol

utl-convert-from-text-to-value

utl-convert-list-to-text

utl-copy-object-attributes

utl-copy-symbol-array-to-initial-values

utl-copy-symbol-array-to-symbol-array

utl-copy-symbol-array-to-symbol-list  
utl-copy-symbol-list-to-symbol-array  
utl-copy-symbol-list-to-symbol-list  
utl-copy-text-array-to-initial-values  
utl-copy-text-list-to-text-list  
utl-create-instance-of-class  
utl-get-module-information  
utl-get-module-owning-this-workspace  
utl-get-top-level-workspace-of-this-workspace  
utl-goto-superior  
utl-goto-workspace  
utl-hide-workspace  
utl-insert-text-in-alphabetized-list  
utl-insert-text-in-reversed-alphabetized-list  
utl-instances-exist  
utl-lift-to-top  
utl-remove-hyphens-from-text  
utl-remove-module-from-hierarchy  
utl-return-clipped-string  
utl-show-superior-workspace  
utl-transfer-to-at

## uil-add-prefix-to-user-criteria-prefixes

Adds the specified prefix to the prefixes listed in the user-criteria array.

### Synopsis

uil-add-prefix-to-user-criteria-prefixes  
(*NewPrefix*: text)

Argument	Description
<i>NewPrefix</i>	The new prefix to add to the array of built-in prefixes

### Description

The user-criteria array is used to screen out procedures listed in Available User Procedures scroll area that is on the Customize Dialog Actions dialog. For information about this dialog, see the chapter on push buttons in the *G2 GUIDE User's Guide*.

# uil-create-uilib-module-setting

Creates a UILLIB module setting that you can use to cause the GUIDE/UIL initialization panel *not* to be displayed on Start or Restart.

## Synopsis

uil-create-uilib-module-setting()  
-> *module-setting*: class uilib-module-setting

Return Value	Description
<u><i>module-setting</i></u>	The UILLIB module setting created by this procedure.

## Description

A uilib-module-setting has one attribute, uil-display-initialization-panel, which is a truth-value. By default, this attribute is set to TRUE, which causes the GUIDE/UIL initialization panel to be displayed on Start or Restart.

To cause the GUIDE/UIL initialization panel not to be displayed, you can invoke uil-create-uilib-module-setting() to create a uilib-module-setting and set the attribute of this uilib-module-setting to FALSE. GFR handles the initialization and propagation of the values.



# uil-date-verification

Verifies that a specified date adheres to a specified format.

## Synopsis

uil-date-verification

(*date*: text, *field-delimiter*: text, *date-format* : symbol)

-> verification: text

Argument	Description
<i>date</i>	The date to be verified.
<i>field-delimiter</i>	A field delimiter for separating the components of the date, or blank text (no delimiter)
<i>date-format</i>	The format to which the date should adhere.

Return Value	Description
"ok"	The date adheres to the specified format.
"invalid length of date entered"	The date contains too many or too few characters.
"incorrect field delimiter specified"	Field delimiter used is not the one specified in the currently selected format.
"invalid year specified"	The year specified was not within the range allowed by the format.
"invalid month specified"	The month specified was not a valid month.
"invalid day specified"	The day specified was not a valid day.

## Description

This procedure validates a date using formatting criteria specified in an existing format. Formats are instances of `uil-format-specification-class`. For information about formats, see the chapter on validation in the *G2 GUIDE User's Guide*.

The *text* argument receives a string indicating whether or not the date adheres to the specified format. The following table lists possible values of *text*.

## **uil-determine-if-guide-is-loaded**

Checks to see if GUIDE is loaded. It sets the logical parameter g2-guide-system-is-loaded.

### **Synopsis**

uil-determine-if-guide-is-loaded ()

# uil-display-color-selector-dialog

Opens the color selection dialog for a specified object.

## Synopsis

uil-display-color-selector-dialog

(*initiating-object*: item-or-value, *dialog-title*: text,  
*selection-method*: item-or-value, *unselection-method*: item-or-value,  
*user-action*: symbol, *window*: class g2-window)

-> *return*: item-or-value

Argument	Description
<i>initiating-object</i>	The object for which the dialog is being displayed.
<i>dialog-title</i>	The text for the title of the color selection dialog.
<i>selection-method</i>	The selection method called when a user selects a message in the color scroll-area, or the symbol <code>none</code> .
<i>unselection-method</i>	The unselection method called when a message is deselected in the color scroll-area, or the symbol <code>none</code> .
<i>user-action</i>	A user defined update method to be inserted before <code>uil-call-update-method</code> in the dialog-actions during the launching of the dialog
<i>window</i>	The G2 window where <i>initiating-object</i> is displayed.

Return Value	Description
<i>return</i>	A text value containing the name of the selected color

## Description

`uil-display-color-selector-dialog()` opens the color selection dialog for the object specified as *initiating-object*. For information how specify the colors of UIL objects, see the *G2 GUIDE User's Guide*.

# uil-do-nothing

A template to use for button callbacks. It is the default value for the `Callback` attribute of all buttons.

## Synopsis

uil-do-nothing

(*button*: class uil-button, *win*: class item, *dialog-or-workspace*: class item)

Argument	Description
<i>button</i>	The button that was selected.
<i>win</i>	The window on which the dialog or workspace containing the button is managed
<i>dialog-or-workspace</i>	The dialog or workspace containing the button

# uil-generate-rules-for-attributes

Generates rules that update an automatically-generated dialog being used to edit an instance of a user-defined class.

## Synopsis

uil-generate-rules-for-attributes

(*user-class*: symbol, *update-if-moved*: truth-value, *attribute-list*: symbol-list)

-> *rules*: class item-list

Argument	Description
<i>user-class</i>	Represents a class name.
<i>update-if-moved</i>	If true, then moving an instance of the class named by <i>user-class</i> will cause any dialog currently editing that instance to be updated with the new attribute values of that instance.
<i>attribute-list</i>	A list of names of attributes of the class <i>user-class</i> .

Return Value	Description
<i>rules</i>	A list containing a rule that updates each attribute listed in <i>attribute-list</i> .

## Description

uil-generate-rules-for-attributes() generates a list of rules for specified attributes of the user-defined class *user-class*. The rules are invoked whenever the attribute of any instance of the class named by *user-class* receives a value, and optionally when an instance of the class is moved.

The rules cause any automatically-generated dialog currently editing that instance to be updated with the new attribute value of that instance. For information about how to generate dialogs automatically, see the *G2 GUIDE User's Guide*.

# uil-get-attribute

Returns an attribute of a specified object.

## Synopsis

uil-get-attribute

(*object*: item-or-value, *attribute*: symbol)

-> *value*: value, *truth*: truth-value

Argument	Description
<i>object</i>	The object whose attribute is returned.
<i>attribute</i>	The attribute of <i>object</i> that is returned.

Return Value	Description
<u><i>value</i></u>	
<u><i>truth</i></u>	

# uil-get-configuration

Returns a configuration for the specified graphical object. If none is specified for the object, then a default configuration for the type of object is returned.

## Synopsis

uil-get-configuration

(*gobj*: class item)

-> {*configuration*: uil-configuration-class | configuration-not-found: symbol}

Argument	Description
<i>gobj</i>	The item to return the configuration for

Return Value	Description
<u><i>configuration</i></u>	The uil-configuration-class for the object, or the symbol configuration-not-found.

## Description

If there is no default configuration, this procedure returns the symbol configuration-not-found.



# uil-get-text-extent

Returns the width and height of a string.

## Synopsis

```
uil-get-text-extent
(string: text, font-size: symbol)
-> width: integer, height: integer
```

Argument	Description
<i>string</i>	The string whose width and height are returned.
<i>font-size</i>	Specify one of: small, medium, or large.

Return Value	Description
<u><i>width</i></u>	The width of the text string <i>string</i> , in workspace units.
<u><i>height</i></u>	The height of the text string <i>string</i> , in workspace units.

## Description

The information returned by `uil-get-text-extent()` can help you decide how large to make a button whose label you have resized.

You can use the *width* and *height* values returned by `uil-get-text-extent()` to change the size of a button using conclude statements such as the following:

```
conclude that the width of the icon-variables of my-button = width
conclude that the height of the icon-variables of my-button = height
```

---

**Note** If the `uil-resize-to-fit-label` attribute of a text button is set to TRUE, the button is resized automatically when its label is changed.

Icon buttons are always resized automatically when their labels are changed.

---

# uil-get-user-criteria-prefixes

Returns a text string containing the prefixes listed in the user-criteria array.

## Synopsis

uil-get-user-criteria-prefixes ()  
-> *prefixes*: text, *number-of-prefixes*: integer

Return Value	Description
<u><i>prefixes</i></u>	A text string containing the prefixes of the user-criteria array
<u><i>number-of-prefixes</i></u>	The number of prefixes in the returned text string

## Description

The number of prefixes is also returned for parsing purposes. Each prefix is separated by a comma (for example: 'guide-, uil-').

## uil-guidata-launch-dialog-for-item

Launches the Edit List or Edit Array dialog for editing the list or array contents of a UIL control. This procedure supersedes the `guidata-launch-dialog-for-item` procedure, which is still supported by this version of GUIDE/UIL. These two procedures take the same arguments and have the same functionality.

### Synopsis

`uil-guidata-launch-dialog-for-item`  
 (*user-item*: item-or-value, *window*: class g2-window)

Argument	Description
<i>user-item</i>	The dialog to check if managed.
<i>windows</i>	An item-list that is populated with the <code>g2-windows</code> on which the dialog is managed.

### Description

`uil-guidata-launch-dialog-for-item` launches the Edit List or Edit Array dialog with *user-item* as the initiating object. These dialogs enable you to specify the initial contents of an array or list attribute.

For information about how to use the Edit List or Edit Array dialog, see the *G2 GUIDE User's Guide*.

# uil-is-this-item-near-another

Checks for the existence of any item that is near a specified item on a specified workspace, and returns this item, if found.

## Synopsis

uil-is-this-item-near-another

(*obj*: class uil-grobj, *x*: integer, *y*: integer, *workspace*: class kb-workspace)

-> *item-is-found*: truth-value, {*item*: item | not-found: symbol}

Argument	Description
<i>obj</i>	The object to check for.
<i>x</i>	The item-x-position of the specified object ( <i>obj</i> ).
<i>y</i>	The item-y-position of the specified object ( <i>obj</i> ).
<i>workspace</i>	The workspace on which the specified object appears.

Return Value	Description
<u><i>item-is-found</i></u>	true if the item is found, false otherwise.
<u><i>item</i></u>	The item, if found, otherwise the symbol not-found.

# uil-remove-guide-application

Removes the `guide` module from your application.

## Synopsis

`uil-remove-guide-application`  
(*window*: class `g2-window`)

Argument	Description
<i>window</i>	The G2 window where the GUIDE application is running.

## Description

`uil-remove-guide-application` removes the `guide` module, as well as any modules in the GUIDE application hierarchy that require the `guide` module, provided that no module outside the GUIDE application hierarchy requires the `guide` module or the modules that require `guide`.

# uil-remove-prefix-from-user-criteria-prefixes

Removes the specified prefix from the prefixes listed in the user-criteria array.

## Synopsis

uil-remove-prefix-from-user-criteria-prefixes  
(*prefix*: text)

Argument	Description
<i>prefix</i>	The prefix to remove from the array of built-in prefixes.

## Description

The user-criteria array is used to screen out procedures listed in Available User Procedures scroll-area that is on the Customize Dialog Actions dialog. For information about this dialog, see the chapter on push buttons in the *G2 GUIDE User's Guide*.

# uil-reset-uilib-module-setting-to-defaults

Resets a uilib module to its default settings.

## Synopsis

uil-reset-uilib-module-setting-to-defaults  
(*setting*: class uilib-module-setting)

Argument	Description
<i>setting</i>	The uilib module setting that this procedure resets to defaults.

# uil-reset-user-criteria-prefixes

Resets the user-criteria text array to its default state.

## Synopsis

```
uil-reset-user-criteria-prefixes()
```

## Description

`uil-reset-user-criteria-prefixes` resets the `user-criteria` text array to its default contents. By default, this array includes the prefixes: `guide-`, `uil-`, `upp-`, `utl-`, `utp-`, and `g2-`. This array is referenced by the Customize Dialog Actions dialog, which does not include procedures with these prefixes in the list of procedures that it displays in the Available User Actions scroll area.

You can invoke `uil-reset-user-criteria-prefixes` at any time if for any reason you need to reset `user-criteria` text array to its default contents.

For information about GFR module settings, see the *G2 Foundation Resources User's Guide*.



## uil-return-bin-workspace-for

Returns the name of the configurations bin for the module that the specified item resides in.

### Synopsis

uil-return-bin-workspace-for

(*itm*: class item)

-> workspace: class kb-workspace

Argument	Description
<i>itm</i>	The item to return the bin workspace for.

Return Value	Description
<u>workspace</u>	The workspace that is the bin for the item.

# uil-return-configuration-for-color

Returns a configuration in a specified color that you can apply to message objects.

## Synopsis

uil-return-configuration-for-color  
(*color*: text) -> configuration: item-or-value

Argument	Description
<i>color</i>	The name of the color for which the configuration is returned.

Return Value	Description
<u>configuration</u>	The configuration associated with the specified color.  If you do not specify a valid color name for <i>color</i> , this procedure returns the symbol <i>none</i> .

## Description

You can use `uil-return-configuration-for-color()` to obtain a configuration that you can apply to message objects (`uil-message-object`) in scroll areas. You specify the color of this configuration in the *color* argument of the call to `uil-return-configuration-for-color()`. You can then apply the configuration returned in configuration to any message object.

### To apply the configuration returned by `uil-return-configuration-for-color` to a message object:

- 1 Set the configuration attribute of the message object to the configuration returned by `uil-return-configuration-for-color()`, through a statement such as the following:

conclude that the configuration of *message-object* = *config*

where *message-object* is the message object to which you are applying the configuration, and *config* is the configuration returned by `uil-return-configuration-for-color`.

- 2 Call `uil-configure-grobj-method` on the message object to apply the configuration to that message object. For information about this procedure, see [uil-configure-grobj-method](#).

## uil-set-attribute

Sets the attribute with the given value for the specified object.

### Synopsis

uil-set-attribute

(*gobj*: class item, *attribute*: symbol, *value*: value)

Argument	Description
<i>gobj</i>	The object to set the attribute value of.
<i>attribute</i>	The attribute to modify.
<i>value</i>	the new value for the attribute.

# uil-time-verification

Verifies that a specified time adheres to the specified format.

## Synopsis

uil-time-verification

(*time*: text, *field-delimiter*: text, *time-format* : symbol, *24-hour*: truth-value)

-> *verification*: text

Argument	Description
<i>time</i>	The time to be verified.
<i>field-delimiter</i>	The field delimiter for separating the components of the time, or blank text (no delimiter).
<i>time-format</i>	The format that the time should adhere to. Specify HHMMSS or HHMM.
<i>24-hour</i>	true or false If true, time is validated based on a 24 hour clock.

Return Value	Description
<u><i>verification</i></u>	Contains a string indicating whether or not the time adheres to the specified format. One of the following: "invalid length of time entered" "incorrect field delimiter specified" "invalid hour specified" "invalid minute specified" "invalid second specified" "ok"

# uil-utility-build-symbol-list-of-values-for-class-attribute

Returns a list of symbols describing the possible values for that attribute.

## Synopsis

uil-utility-build-symbol-list-of-values  
 (*class-name*: symbol, *attribute-name*: symbol)  
 -> list: item-or-value

Argument	Description
<i>class-name</i>	The name of the class of object from which this procedure returns attribute values.
<i>attribute-name</i>	The attribute from which this procedure returns values.

Return Value	Description
<u>list</u>	<p>The list of values from the specified attribute of the specified class.</p> <p>If the attribute <i>attribute-name</i> is not defined in the class <i>class-name</i>, returns the symbol <code>attribute-is-not-defined-in-object-definition</code>.</p> <p>If the attribute <i>attribute-name</i> does not contain values, returns the symbol <code>attribute-does-not-contain-values</code>.</p>

## Description

When given a class name and an attribute name (both symbols, the attribute name being an attribute of the class), it returns a list of symbols describing the possible values for that attribute, assuming that the attribute in question has a limited set of possible values.

If the attribute of the class is defined as follows:

my-attribute is a symbol, has values red, blue, or green, initially is green

uil-utility-build-symbol-list-of-values-for-class-attribute returns a list containing the symbols red, blue, and green. If the attribute does not contain a limited set of

values in this way, the procedure returns the symbol `attribute-does-not-contain-values`.

# uil-utility-change-module

Changes the module assignment of all workspaces in a module.

## Synopsis

uil-utility-change-module

(*old-module-name*: symbol, *new-module-name*: symbol,  
*window*: class g2-window)

Argument	Description
<i>old-module-name</i>	The name of the module to move objects out of.
<i>new-module-name</i>	The name of the module to move objects to.
<i>window</i>	The G2 window on which a prompt appears, asking you whether you want to delete the module <i>old-module-name</i> . Specify this <i>window</i> if you call procedure from a user menu choice.

## Description

This procedure changes the module assignment of every workspace from *old-module-name* to *new-module-name*. It handles transferring UIL objects from one module to another. It also enables you to delete the old module, if you choose.

The appropriate bins are created and objects are checked to ensure that they reside in the correct module. Both modules must exist in order for this procedure to function. If not, a message is generated informing the user of the missing module.

# uil-utility-check-for-orphaned-selection-buttons

Prompts user to create selection boxes for orphaned selection buttons.

## Synopsis

uil-utility-check-for-orphaned-selection-buttons  
(*window*: class g2-window)

Argument	Description
<i>window</i>	A window associated with the G2 session.

## Description

This procedure finds selection buttons that are no longer managed by an existing selection box (uil-selection-box).

If an orphan is found, a new selection box is created of the appropriate type, and the orphan is added to it. The procedure searches the workspace where the orphan is found, looking for other orphan buttons to add to the selection box.



## uil-utility-cleanup-dialog-bin

Organizes the icons on the specified workspace, so that the icons do not overlap one another. This procedure is run when you select the Cleanup dialog bin option from the menu of a dialog bin.

### Synopsis

uil-utility-cleanup-dialog-bin  
(*bin*: class kb-workspace)

Argument	Description
<i>bin</i>	The workspace to organize.

# uil-utility-convert-from-edit-box-to-integer

Returns the integer value of the `text` attribute of the specified edit box. Calls the G2 quantity function, which converts numeric text characters to integers. Returns 0 if `text` contains no characters that can be converted.

## Synopsis

uil-utility-convert-from-edit-box-to-integer  
(*gobj*: class uil-edit-box)  
-> *converted-text*: integer, *status*: symbol

Argument	Description
<i>gobj</i>	The edit box whose value is to be converted.

Return Values	Description
<i>converted-text</i>	Any non-zero integer indicates that the text box contained numeric text characters that have been converted. The value 0 is returned if no text was converted.
<i>status</i>	The symbol <code>OK</code> if text was converted to an integer and the symbol <code>bad</code> otherwise.

## uil-utility-convert-from-edit-box-to-quantity

Returns the quantity value of the message-contents attribute of the specified edit box.

### Synopsis

```
uil-utility-convert-from-edit-box-to-quantity
(gobj: class uil-edit-box)
-> quantity: class value, symbol: symbol
```

Argument	Description
<i>gobj</i>	The edit box whose value is to be converted.
quantity	symbol
A quantity	ok
0	bad

# uil-utility-convert-from-edit-box-to-symbol

Returns the symbol value of the message-contents attribute of the specified edit-box.

## Synopsis

uil-utility-convert-from-edit-box-to-symbol

(*grobj*: class uil-edit-box)

-> *symbol1*: symbol, *symbol2*: symbol

Argument	Description
<i>grobj</i>	The edit box whose value is to be converted.
<b>symbol1</b>	<b>symbol2</b>
Any symbol	ok
bad	bad

# uil-utility-convert-from-edit-box-to-value

Returns the value and the type of value of the message-contents attribute of the specified edit box.

## Synopsis

uil-utility-convert-from-edit-box-to-value

(*grobj*: class item)

-> *value* : value, *symbol1*: symbol, *symbol2*: symbol

Argument	Description
<i>grobj</i>	The edit box whose value is to be converted.

value	symbol1	symbol2
A text value	text	ok
A quantity value	quantity	ok
A symbol value	symbol	ok
none	none	bad

# uil-utility-create-procedure-xref-listing

Lists all procedures that start with the given key.

## Synopsis

uil-utility-create-procedure-xref-listing

(*window*: class g2-window, *key*: text, *order*: symbol, *directory-path*: text, *formatting*: truth-value)

Argument	Description
<i>window</i>	The window on which to display status information
<i>key</i>	The 3-character prefix to check procedure's name against
<i>order</i>	If you specify <b>alphabetic</b> , procedures are sorted alphabetically. If you specify any other value, procedures are sorted in reverse alphabetic order.
<i>directory-path</i>	The directory path of the file to which the procedure listing is written. The file name will be "[key]-procs.txt".
<i>formatting</i>	If True, the output is "pretty printed". A title is generated, the total number of procedures recorded, and argument information is placed on a line below the procedure's name.

# uil-utility-return-current-real-time

Returns the current real (absolute) time.

## Synopsis

uil-utility-return-current-real-time()

-> *time*: quantity

Return Value	Description
<u><i>time</i></u>	The current time in seconds.

# uil-utility-synchronize-text-fields-for-uil-text

## Synopsis

```
uil-utility-synchronize-text-fields-for-uil-text  
()
```

## Description

This procedure iterates over all `uil-text` objects and calls `uil-update-text` to synchronize its text contents with its `message-contents`. This procedure was useful for bringing text objects in older GUIDE applications up to date with version 4.0.



# uil-utility-unique-id

Generates unique ids with the specified prefix. The current real time is incorporated into the ID.

## Synopsis

```
uil-utility-unique-id  
  (prefix: text)  
-> id: text
```

Argument	Description
<i>prefix</i>	The prefix added to the unique ID.

Return Value	Description
<u><i>id</i></u>	The unique ID generated.

# uil-utility-unique-name

Generates a unique name with a specified prefix. The current real time is incorporated into the name.

## Synopsis

uil-utility-unique-name  
(*prefix*: text)  
-> *name*: text)

Argument	Description
<i>prefix</i>	The prefix added to the unique name.

Return Value	Description
<i>name</i>	The unique name generated.

# utl-clone-through-copying

Creates a copy of the specified object.

## Synopsis

utl-clone-through-copying  
(*source-object*: class item)  
-> copy: class item

Argument	Description
<i>source-object</i>	The object to make a copy of.

Return Value	Description
<u>copy</u>	The copy of the item.

# utl-convert-from-text-to-integer

Converts the specified text value to an integer value.

## Synopsis

utl-convert-from-text-to-integer

(*txt*: text)

-> *converted-text*: integer, *status*: symbol

Argument	Description
<i>txt</i>	The text to convert to an integer.
<b>integer</b>	<b>symbol</b>
An integer value	ok
0	bad

# utl-convert-from-text-to-quantity

Converts the specified text value to a quantity value.

## Synopsis

utl-convert-from-text-to-quantity

(*txt*: text)

-> *converted-text*: quantity, *symbol*: symbol

Argument	Description
<i>txt</i>	The text to convert to a quantity.
<b>quantity</b>	<b>symbol</b>
A quantity	ok
0	bad

# utl-convert-from-text-to-symbol

Converts the specified text value to a symbol.

## Synopsis

utl-convert-from-text-to-symbol

(*txt*: text)

-> *converted-text*: value, *symbol*: symbol

Argument	Description
<i>txt</i>	The text to convert to a symbol.
<b>value</b>	<b>symbol</b>
A symbol	ok
txt	bad

# util-convert-from-text-to-value

Converts the specified text value to a value.

## Synopsis

util-convert-from-text-to-value

(*txt*: text)

-> *converted-text*: value, *text-type*: symbol, *status*: truth-value

Argument	Description	
<i>txt</i>	The text to convert to a value.	
value	symbol	truth-value
A value	The symbol text	true
A value	The symbol quantity	true
A value	The symbol symbol	true
The symbol none	The symbol none	false

# utl-convert-list-to-text

Generates a text string composed of the elements of the specified list.

## Synopsis

```
utl-convert-list-to-text  
  (list: class g2-list)  
  -> list: text
```

Argument	Description
<i>list</i>	The list to convert to text.

Return Value	Description
<u><i>list</i></u>	A text string representing the elements in the specified list, separated by commas.



# utl-copy-object-attributes

Copies the values of the attributes of source object to the attribute slots of the destination object.

## Synopsis

utl-copy-object-attributes  
(*source*: class item, *dest*: class item)

Argument	Description
<i>source</i>	The object to copy the attribute values from.
<i>dest</i>	The object to copy the attribute values to.

# util-copy-symbol-array-to-initial-values

Iterates over the elements of the specified array and sets the initial-values to reflect the array's contents.

## Synopsis

util-copy-symbol-array-to-initial-values  
(*s-array*: class symbol-array)

Argument	Description
<i>s-array</i>	The symbol array to set the initial-values of .

# utl-copy-symbol-array-to-symbol-array

Copies the elements of one symbol array to another.

## Synopsis

utl-copy-symbol-array-to-symbol-array

(*source-array*: class symbol-array, *dest-array*: class symbol-array)

Argument	Description
<i>source-array</i>	The array to copy elements from.
<i>dest-array</i>	The array to copy elements to.

# utl-copy-symbol-array-to-symbol-list

Copies the elements of a symbol array to a symbol list.

## Synopsis

utl-copy-symbol-array-to-symbol-list

(*source-array*: class symbol-array, *dest-list*: class symbol-list)

Argument	Description
<i>source-array</i>	The array to copy elements from.
<i>dest-list</i>	The list to copy elements to.

# utl-copy-symbol-list-to-symbol-array

Copies the elements of a symbol list to a symbol array.

## Synopsis

utl-copy-symbol-list-to-symbol-array

(*source-list*: class symbol-list, *dest-array*: class symbol-array)

Argument	Description
<i>source-list</i>	The list to copy elements from.
<i>dest-array</i>	The array to copy elements to.

# utl-copy-symbol-list-to-symbol-list

Copies the elements of one symbol list to another.

## Synopsis

utl-copy-symbol-list-to-symbol-list

(*source-list*: class symbol-list, *dest-list*: class symbol-list)

Argument	Description
<i>source-list</i>	The list to copy elements from.
<i>dest-list</i>	The list to copy elements to.

## utl-copy-text-array-to-initial-values

Iterates over the elements of the specified array and sets the initial-values to reflect the array's contents.

### Synopsis

utl-copy-text-array-to-initial-values  
(*t-array*: class text-array)

Argument	Description
<i>t-array</i>	The text array to set the initial-values of .

# utl-copy-text-list-to-text-list

Copies the elements of one text list to another.

## Synopsis

utl-copy-text-list-to-text-list

(*source-list*: class text-list, *dest-list*: class text-list)

Argument	Description
<i>source-list</i>	The list to copy elements from.
<i>dest-list</i>	The list to copy elements to.



# utl-create-instance-of-class

Creates and returns an instance of the class named by icon.

## Synopsis

utl-create-instance-of-class

(*icon*: symbol)

-> *instance*: class item | invalid-class: symbol}

Argument	Description
<i>icon</i>	A symbol naming the class to create an instance of.

Return Value	Description
<u><i>instance</i></u>	An instance of the class, or the symbol invalid-class.

# utl-get-module-information

Finds specified module information.

## Synopsis

utl-get-module-information

(*sym*: symbol)

-> {*module-info-object*: item-or-value | not-found: symbol}

Argument	Description
<i>sym</i>	The symbol value to compare the text of the top-level-modules to.

Return Value	Description
<i>module-info-object</i>	The module-information object for the top-level-module whose text matches the text value of the specified symbol, or the symbol not-found.

## Description

This procedure iterates over each module-information object and compares the text value of the specified symbol to the text of the each top-level-module. If a match is found, the module-information object is returned.

# utl-get-module-owning-this-workspace

Returns the module assignment for the specified workspace.

## Synopsis

```
utl-get-module-owning-this-workspace
  (ws: class kb-workspace)
  -> {module-assignment: symbol | unspecified}
```

Argument	Description
<i>ws</i>	The workspace to return the module of.

Return Value	Description
<u><i>module-assignment</i></u>	The module-assignment of the specified workspace, or the symbol <code>unspecified</code> .

# utl-get-top-level-workspace-of-this-workspace

Returns the top-level workspace for the specified workspace.

## Synopsis

utl-get-top-level-workspace-of-this-workspace

(*ws*: class kb-workspace)

-> {*top-ws*: class kb-workspace | unknown: symbol}

Argument	Description
<i>ws</i>	The workspace to return the top-level workspace of.

Return Value	Description
<u><i>top-ws</i></u>	The top-level workspace of the specified workspace, or the symbol unknown.

## utl-goto-superior

Hides workspace of a specified button and displays workspace of superior item.

### Synopsis

utl-goto-superior

(*button*: class uil-button, *window*: class item, *workspace*: class kb-workspace)

Argument	Description
<i>button</i>	The button to go to the superior workspace of.
<i>window</i>	The window on which button is managed, or the workspace of button.
<i>workspace</i>	The workspace of the button.

### Description

This procedure hides the workspace of the specified button if the attribute `uil-hide-this-workspace` is true, and then displays the workspace of the item superior to the specified button.

If the *window* argument is a `g2-window`, then the button's workspace is hidden on the specified window, and the workspace of the superior item is displayed only on the specified window.

# util-goto-workspace

Displays a specified workspace.

## Synopsis

util-goto-workspace

(*button*: class uil-button, *window*: class item, *workspace*: class kb-workspace)

Argument	Description
<i>button</i>	The button to go to the superior workspace of.
<i>window</i>	The window on which button is managed, or the workspace of button.
<i>workspace</i>	The workspace of the button.

## Description

This procedure displays the workspace named by the workspace-destination attribute if it exists, or displays the subworkspace of the specified button if it exists, or displays the workspace of the superior object if the button is a util-goto-previous-button. If the uil-hide-this-workspace attribute is true, the workspace of the specified button is hidden.

## utl-hide-workspace

Hides the workspace of a specified button.

### Synopsis

utl-hide-workspace

(*button*: class uil-button, *window*: class item, *workspace*: class kb-workspace)

Argument	Description
<i>button</i>	The button to hide the workspace of.
<i>window</i>	The window on which button is managed, or the workspace of button.
<i>workspace</i>	The workspace of the button.

### Description

This procedure hides the workspace of the specified button. If the window argument is a *g2-window*, then the workspace of the button is only hidden for the specified window.

# utl-insert-text-in-alphabetized-list

Inserts the given text into the specified list alphabetically.

## Synopsis

utl-insert-text-in-alphabetized-list

(*txt*: class message, *lst*: class item-list)

Argument	Description
<i>txt</i>	The text to insert in the specified list.
<i>lst</i>	The list to insert text in alphabetically.



# utl-insert-text-in-reversed-alphabetized-list

Inserts the given text into the specified list in reverse alphabetical order.

## Synopsis

utl-insert-text-in-reversed-alphabetized-list  
(*txt*: class message, *lst*: class item-list)

Argument	Description
<i>txt</i>	The text to insert in the specified list.
<i>lst</i>	The list to insert text in reverse alphabetical order.

# utl-instances-exist

Checks for instances of a specified class.

## Synopsis

utl-instances-exist

(*class-name*: symbol)

-> class-exists: truth-value

Argument	Description
<i>class-name</i>	The name of the class to check for instances of.

Return Value	Description
<u>class-exists</u>	If instances of the specified class exist then true, otherwise false.

## utl-lift-to-top

Shows the workspace or subworkspace of a specified item on a specified window.

### Synopsis

utl-lift-to-top

(*itm*: class item, *window*: class g2-window)

Argument	Description
<i>itm</i>	The item whose workspace or subworkspace is lifted to the top.
<i>window</i>	The window on which the item is managed.

### Description

This procedure shows the workspace of the item *itm* on the window *window*. If the specified item is a dialog (`uil-dialog`), it shows the subworkspace of the dialog.

This has the effect of bringing the workspace or subworkspace to the forefront.

# utl-remove-hyphens-from-text

Removes hyphens from the specified text, and replaces them with spaces.

## Synopsis

utl-remove-hypens-from-text

(*txt*: text)

-> *new-text*: text

Argument	Description
<i>txt</i>	The text to remove the hyphens from.

Return Value	Description
<u><i>new-text</i></u>	The specified text, minus any hyphens.

# utl-remove-module-from-hierarchy

Deletes the specified module from the hierarchy.

---

**Caution** You should know how removing a module from the hierarchy will affect your application before using this procedure. See Chapter 5, “Modules and Modularized KBs” in the *G2 Reference Manual* for further information.

---

## Synopsis

utl-remove-module-from-hierarchy  
 (*module*: symbol, *client*: item-or-value)

Argument	Description
<i>module</i>	The module to remove from the hierarchy.
<i>client</i>	A display window to which output, such as error messages, is written. This can be a G2 window, a client session, or some other user-defined client item.

# utl-return-clipped-string

Clips the specified text to the given length. The clipped text is appended with "...".

## Synopsis

utl-return-clipped-string  
(*txt*: text, *length*: integer)  
-> *clipped-text*: text

<b>Argument</b>	<b>Description</b>
<i>txt</i>	The text to be clipped.
<i>length</i>	The length to clip the text to.

<b>Return Value</b>	<b>Description</b>
<u><i>clipped-text</i></u>	The clipped text.

# utl-show-superior-workspace

Displays the workspace of the superior item of the specified item on the given window.

## Synopsis

utl-show-superior-workspace

(*itm*: class item, *window*: class g2-window)

Argument	Description
<i>itm</i>	The item to display the superior workspace of.
<i>window</i>	The window on which to display the workspace.

## utl-transfer-to-at

Transfers the specified item to the given workspace at the specified x and y location.

### Synopsis

utl-transfer-to-at

(*itm*: class item, *workspace*: class kb-workspace, *x*: integer, *y*: integer)

Argument	Description
<i>itm</i>	The item to transfer.
<i>workspace</i>	The workspace to transfer item to.
<i>x</i>	The x location to position item on specified workspace.
<i>y</i>	The y location to position item on specified workspace.



## Additional Utility Procedures

[uil-utility-restore-grobj](#)

[uil-utility-check-for-empty-selection-box](#)

[uil-utility-create-permanent-copies-for-module](#)

[uil-utility-create-permanent-copy-for-dialog](#)



# System-Defined Relations

---

*Describes the relations between UIL objects.*

Introduction **635**

Relations among UIL Objects **636**

Relations among Dialogs and Supporting Objects **638**



## Introduction

UIL automatically establishes relations among UIL objects when you perform actions such as linking dialogs together in a cascaded hierarchy, or adding a radio button to a group of radio buttons.

For example, UIL establishes the relation **the-group-master-of** to express the relation between a selection box and each button that it contains. Each button in turn is related to the selection box by the relation **grouped-by**. Thus, every radio button is **grouped-by** some radio box, and every radio box is **the-group-master-of** each radio button that it contains.

You can also use UIL procedures to establish relations between particular UIL objects, or to override the relations that UIL establishes automatically.

You can reference the relations in procedures to manipulate UIL objects.

For example, a procedure can use the **grouped-by** relation to perform the same operation on all the radio buttons in a particular radio box. The radio buttons are referenced as the ones that are **grouped-by** the radio box:

```

for B = each uil-radio-button that is grouped-by RBOX do
    < some user-specified activity >
end

```

The line of code above specifies that some user-specified operation is to be performed on each of the radio buttons in the radio box named RBOX.

Each kind of relation is defined by a separate relation class. GUIDE provides a rich set of relation classes between dialogs and other UIL objects.

## Relations among UIL Objects

The following table summarizes standard relations between UIL objects:

**UIL Object Relations**

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-dialog	a-uil-dialog-presently-in-use-on-window	g2-window	many-to-many	the-g2-window-now-showing-dialog
uil-dialog	a-uil-child-cascaded-dialog-of	uil-dialog	many-to-one	the-uil-parent-dialog-of-this-cascaded-dialog
uil-dialog	a-uil-dialog-using-the-temporary-storage-object	temporary storage object	one-to-one	the-uil-temporary-storage-object-of
item	the-object-receiving-dialog-activity-from	item	one-to-many	none
uil-message	the-current-edit-field	uil-dialog	one-to-one	none
uil-message	the-current-uil-message-being-edited-on	g2-window	one-to-one	none
uil-message-object	the-bottommost-message-of	uil-scroll-area	one-to-one	none
uil-message-object	the-topmost-message-of	uil-scroll-area	one-to-one	none

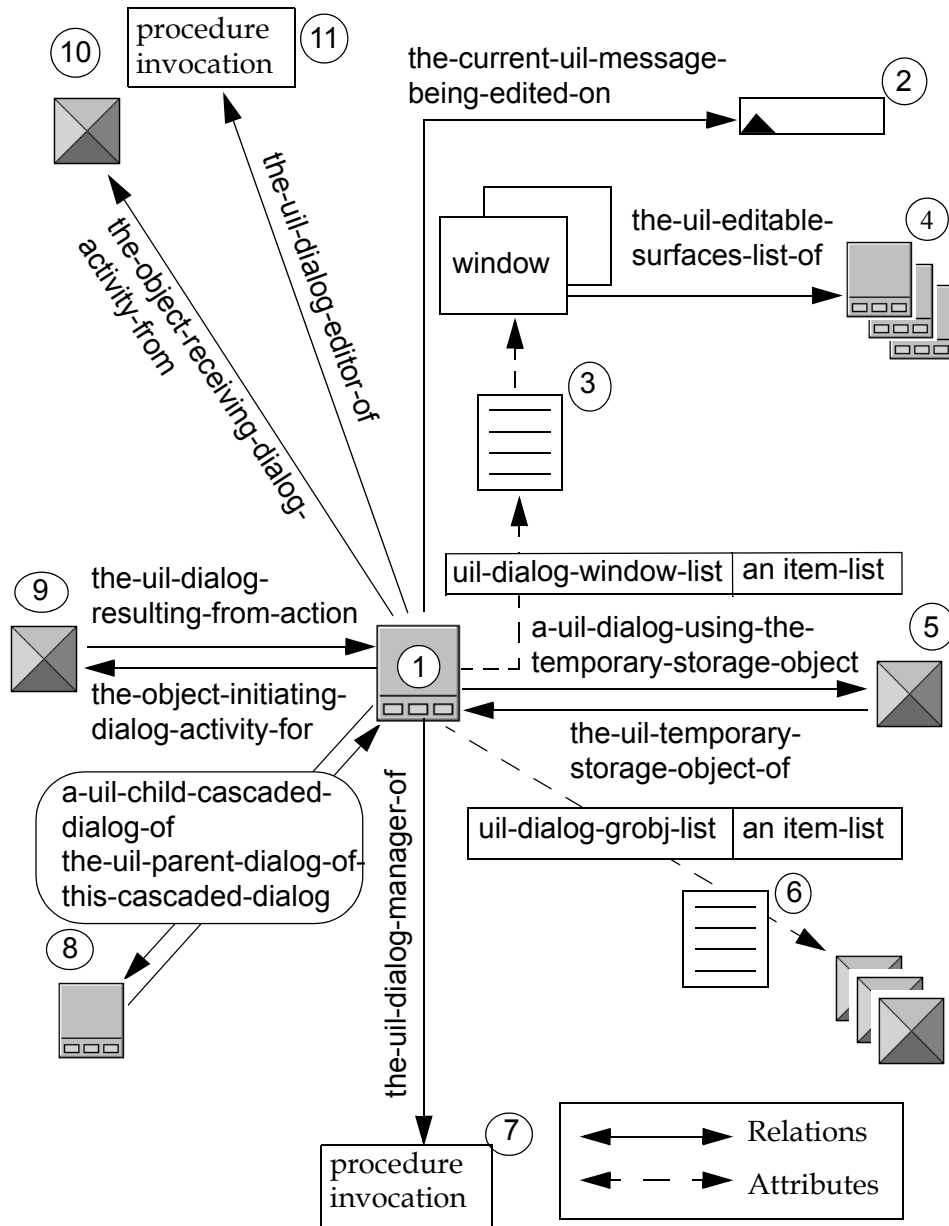
### UIL Object Relations

<b>Class of Subject item</b>	<b>Relation</b>	<b>Class of Object item</b>	<b>Type of Relation</b>	<b>Inverse Relation</b>
uil-message-object	a-message-belonging-to	uil-scroll-area	many-to-one	the-handler-for-the-message-of
uil-message-object	a-selected-message-of	uil-scroll-area	many-to-one	none
item	the-uil-dialog-resulting-from-action	item	many-to-one	the-object-initiating-dialog-activity-for
item	the-item-appended-to-the-scroll-message	uil-message-object	one-to-one	the-uil-message-appended-to-the-item
g2-window	the-uil-editable-surfaces-list-of	item list	one-to-one	none

# Relations among Dialogs and Supporting Objects

The following figure illustrates the relations that UIL establishes between a dialog and other UIL objects that support the dialog.

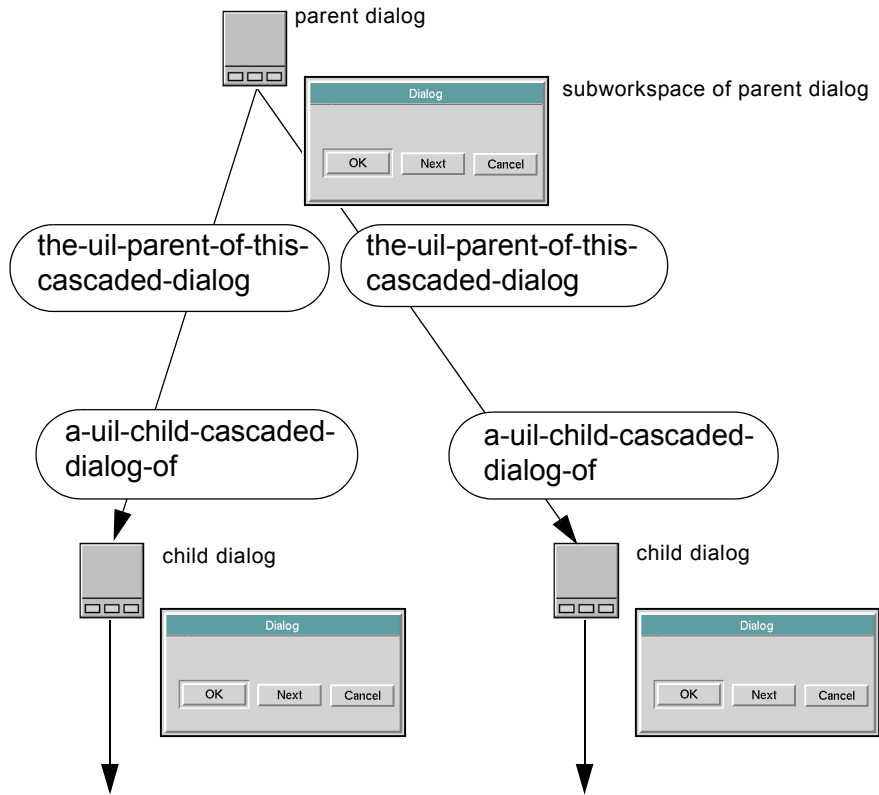
Relations Among Dialogs and Supporting Objects



The objects labeled on the figure above are explained here:

- 1** The dialog itself. Once a handle to the dialog exists, every other object can be located. The handle can be an argument or local variable of class `dialog`.
- 2** A G2 message containing the field currently being edited.
- 3** A list of the G2 windows on which the dialog is displayed.
- 4** A list of the currently displayed dialogs, with the most recently displayed dialog at the top.
- 5** A temporary storage object used with the dialog.
- 6** An item list of all the graphical objects (objects of class `uil-grobj`) and message objects (objects of class `uil-grmes`) on the dialog. Edit boxes appear in the list first, in the order in which they appear (top to bottom, left to right) on the dialog. User-defined objects are not included in this list.
- 7** A procedure invocation related to the dialog through the relation `the-uil-dialog-manager-of`. This procedure runs continuously while the dialog is in use, processing the events passed into the dialog. It remains in a wait state until events arrive.
- 8** A child dialog or dialogs of this dialog. A child dialog is related to its parent through the relation `the-uil-cascaded-dialog-of`.
- 9** The button or other object from which this dialog was initiated.
- 10** The relation `the-object-receiving-dialog-activity-from` is established between an object and a dialog by the UIL procedure `uil-set-destination-object-for-dialog`. The `uil-set-destination-object-for-dialog` procedure specifies the object as the target object for UIL controls on a dialog.
- 11** A procedure invocation related to the dialog through the relation `the-uil-dialog-editor-of`. For each dialog, this procedure controls the behavior of the editor state, which edit box is selected, which is next, how key and mouse gestures are handled, and so on.

The following figure illustrates a dialog that is the parent of two child dialogs, and the G2 relations that define the relationships between these dialogs.



You can reference relations in G2 procedures to specify the particular objects that you want to manipulate through the procedures.

For example, suppose that a selection box of radio buttons, named `my-radio-box`, contains three radio buttons, named `my-rb1`, `my-rb2`, and `my-rb3`. GUIDE defines the relationship between the radio box and the radio buttons using these relations:

- The radio box is `the-group-master-of` the radio buttons.
- Each radio button is `grouped-by` the radio box.



The following G2 procedure illustrates how you can reference these relations to specify the particular objects that you want to manipulate; in this example, to perform a set of operations iteratively on each of the radio buttons in the selection box named `my-radio-box`:

```
begin
RB: class uil-radio-button;
RBOX: class uil-radio-box = my-radio-box;
B: class uil-radio-button;
RB = my-rb-1;

RBOX = the uil-radio-box that is the-group-master-of RB;

for B = each uil-radio-button that is grouped-by RBOX do

* {specify operations to be performed on each radio button}
end:

end
```

In this procedure, the following line identifies the radio buttons on which the procedure is to perform operations, using their relation to the selection box `my-radio-box`:

```
for B = each uil-radio-button that is grouped-by RBOX do
```

Thus, this procedure operates on only those radio buttons that are included in (`grouped-by`) the selection box `my-radio-box`.

To obtain a summary of the relations between a UIL object and other UIL objects, choose `describe` from the menu of the UIL object.



# Common Object Attributes

---

*Summarizes the attributes that are common to all UIL objects, to all graphical message objects (uil-grmes), and to all UIL objects with values that can be updated or concluded.*



The following table summarizes the attributes that are common to all UIL objects. These attributes are not listed or described in the chapters that deal with particular classes of UIL objects.

## Attributes Common to All UIL Objects

Attribute	Description
<b>notes</b>	Displays information about problems with the UIL object, if any.  <i>Default value:</i> None.
<b>names</b>	The name or names of the UIL object.  <i>Default value:</i> None.

## Attributes Common to All UIL Objects

Attribute	Description
<b>uil-is-managed</b>	<p>Specifies whether or not the UIL object is visible (managed) on a window or workspace.</p> <p><i>Allowable values:</i> 1 (object is visible) or 0 (object is not visible)</p> <p><i>Default value:</i> 1</p>
<b>uil-is-permanent</b>	<p>Specifies whether or not the UIL object is a persistent G2 object.</p> <p><i>Allowable values:</i> <b>true</b> (object is persistent) or <b>false</b> (object is transient)</p> <p><i>Default value:</i> <b>true</b></p>
<b>id</b>	<p>A unique key identifying the UIL object.</p> <p><i>Allowable values:</i> The ID of a UIL object must be a text string that evaluates to a symbol.</p> <p>When you create a symbol value, the first character can consist of any Unicode character. If you begin a symbol value with a period (.) or a number (0-9), it must also include at least one alphabetic character or quote one of its numeric characters or any of the Unicode character set symbol, punctuation, or special characters, with the exception of the hyphen (-), underscore (_), period (.), and apostrophe ('). which don't need quoting. For example "123" and "my dialog" are not valid IDs, but "A23", "23A", "12@3", and "my-dialog" are valid IDs.</p> <p>For more information on working with characters in a symbol value, see Chapter 9, "Values and Types" in the <i>G2 Reference Manual, Volume 1</i>.</p> <p><i>Default value:</i> " "</p>

## Attributes Common to All UIL Objects

Attribute	Description
<b>configuration</b>	Specifies the configuration applied to this UIL object.  <i>Allowable values:</i> The name of an existing configuration. Configurations are instances of <code>uil-configuration-class</code> .  <i>Default value:</i> The default configuration for objects of this class.
<b>uil-delete-method</b>	The procedure that is called when you select the <b>Delete.</b> menu choice, or whenever a procedure calls <code>uil-delete-grobject-method</code> .  <i>Allowable values:</i> Name of an existing procedure with the correct arguments for a delete method.  <i>Default value:</i> <code>uil-delete-grobject-method</code>
<b>uil-configuration-method</b>	Names the procedure used to configure the UIL object.  <i>Allowable values:</i> Name of an existing procedure with the correct arguments for a configuration method.  <i>Default value:</i> <code>uil-configure-grobject-method</code>
<b>uil-initialization-method</b>	Names the procedure used to initialize the UIL object.  <i>Allowable values:</i> Name of an existing procedure with the correct arguments for an initialization method.  <i>Default value:</i> <code>uil-initialize-grobject-method</code>

## Attributes Common to All UIL Objects

Attribute	Description
<b>uil-manage-method-</b>	Specifies the method responsible for managing the object.  <i>Allowable values:</i> Name of an existing procedure with the correct arguments for a manage method.  <i>Default values:</i> uil-manage-grobj-method
<b>uil-size-of-method</b>	Names the procedure used to obtain and return the width and height of the UIL object.  <i>Allowable values:</i> Name of an existing procedure with the correct arguments for a size of method.  <i>Default value:</i> uil-size-of-grobj-method
<b>uil-size</b>	Specifies the size of the object.  <i>Allowable values:</i> Any one of the following symbols: <b>small</b> , <b>medium</b> , or <b>large</b> . <b>Note:</b> Do not edit this attribute value directly in the attribute table. Use the appropriate GUIDE graphical editor to set the size of UIL objects.  <i>Default value:</i> <b>medium</b>
<b>user-data</b>	An attribute to hold user data.  <i>Allowable values:</i> Any value  <i>Default value:</i> <b>none</b>

### Attributes Common to All UIL Objects

Attribute	Description
<b>uil-clone-method</b>	Names a procedure used to make a copy of the UIL object.  <i>Allowable values:</i> Name of an existing procedure with the correct arguments for a clone method.  <i>Default value:</i> uil-clone-groobj-method

The following table lists attributes that are common to graphical message (uil-grmes) objects. These include text objects, edit boxes, message objects, and tab pages.

### Attributes Common to uil-grmes

Attribute	Description
<b>uil-maximum- characters-to-display</b>	(uil-grmes only) The number of characters to display before clipping.  <i>Allowable values:</i> any integer, or the symbol unlimited  <i>Default values:</i> For edit boxes: 10  For text objects, message objects, dialog titles: unlimited
<b>uil-message- contents</b>	Contains the text of the title.  <i>Allowable values:</i> text value  <i>Default values:</i> For edit boxes and message objects: " "  For text objects: "Text", or appropriate label for icon button  For dialog titles: "Dialog Title"

The following table lists attributes that are common to UIL objects that have values that can be updated from or concluded to other objects.

### Attributes Common to UIL Objects that be Concluded or Updated

Attribute	Description
<b>uil-event-target-object</b>	<p>Specifies a target object used as a destination to which the conclude method directs the object's value when run.</p> <p><i>Allowable values:</i> An initiating object, destination object, temporary storage object, or object specified by name.</p> <p><b>Note:</b> For push buttons, the target object can be specified as "dialog-id" or the symbol <code>host-dialog</code>.</p> <p>For information about how to use embedded objects as target objects, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Default value:</i> unspecified</p>
<b>uil-event-target-attribute</b>	<p>Specifies the attribute used as the destination to which the conclude method directs the object's value when run.</p> <p><i>Allowable values:</i> Any class-specific attribute of a target object</p> <p><i>Default value:</i> unspecified</p>
<b>uil-event-source-object</b>	<p>Names an object or specifies an object used as a source from which the update method obtains the value used to set the object's value.</p> <p>For information about how to use embedded objects as source objects, see the <i>G2 GUIDE User's Guide</i>.</p> <p><i>Allowable values:</i> An initiating object, temporary storage object, or a named object.</p> <p><i>Default value:</i> unspecified</p>



## Attributes Common to UIL Objects that be Concluded or Updated

Attribute	Description
<b>uil-event-source-attribute</b>	Specifies the attribute used as a source from which the update method obtains the value used to set the object's value.  <i>Allowable values:</i> Any class-specific attribute of a source object  <i>Default value:</i> unspecified
<b>uil-conclude-value-immediately</b>	Indicates whether the conclude method is run immediately when the object loses focus.  <i>Allowable values:</i> <b>true</b> (run the conclude method as soon as the object loses focus) or <b>false</b>  <i>Default value:</i> <b>false</b>
<b>uil-update-value</b>	Indicates whether or not the update method is run on the UIL object.  <b>Note:</b> If the object is a dialog whose <b>uil-always-update-when-managed</b> attribute is set to <b>True</b> , the dialog is always updated when managed, regardless of the setting of this attribute.  <i>Allowable values:</i> <b>true</b> (run the update method on the object) or <b>false</b> (do not run the update method on the object)  <i>Default value:</i> <b>true</b>
<b>uil-conclude-method</b>	Names the procedure used to conclude the object's value into the target attribute of the target object specified for this UIL object.  <i>Allowable values:</i> A valid procedure name. The procedure must have the correct arguments for conclude methods.  <i>Default value:</i> <b>uil-conclude-grobject-method</b>

## Attributes Common to UIL Objects that be Concluded or Updated

Attribute	Description
<b>uil-update-method</b>	Names the procedure used to update the UIL object's value(s) based on the value(s) found in the source object and attribute.
<i>Allowable values:</i>	A valid procedure name. The procedure must have the correct arguments for update methods.
<i>Default value:</i>	uil-update-grobj-method

@ A B C D E F G H I J K L M  
# N O P Q R S T U V W X Y Z

---

**A**

## attributes

- of border objects
- of check boxes
- of check buttons
- of dialog titles
- of dialogs
- of edit boxes
- of message objects
- of navigation buttons
- of push buttons
- of radio boxes
- of radio buttons
- of scroll areas
- of selection boxes
- of selection buttons
- of separators
- of text objects
- of toggle buttons

**B**

## border object procedures

- uil-adjust-border-on-workspace
- uil-anchor-box-corner-to-workspace-edge
- uil-create-border-for-text
- uil-create-box-border
- uil-get-border
- uil-move-related-border
- uil-release-box-corner-from-workspace-edge

## border objects

- attributes of
- class hierarchy of
- operations performed on
- procedures that operate on

## button procedures

- uil-add-label-to-button
- uil-center-label-in-text-button
- uil-create-custom-button
- uil-create-text-for-button-from-label
- uil-handle-button

uil-handle-button-method

uil-highlight-button

uil-reset-button

uil-return-size-of-button-as-symbol

uil-select-button

uil-toggle-button

uil-update-button-label

uil-update-button-togglelabel

## buttons

- class hierarchy of
- procedures that operate on
- relations with other objects
- selecting, using uil-select-button
- toggling on and off, using uil-toggle-button
- unselecting, using uil-unselect-button

**C**

## check box procedures

- uil-create-check-box
- uil-get-check-box-values
- uil-set-check-box-values

## check boxes

- attributes of
- class hierarchy of
- creating, using uil-create-check-box
- getting values of, using uil-get-check-box-values
- operations on
- relations with other objects
- setting values of, using uil-set-check-box-values

## check button procedures

- uil-create-styled-check-button
- uil-create-styled-check-button-with-different-sized-label

## check buttons

- attributes of
- class hierarchy of
- operations on
- procedures that operate on
- relations with other objects

## classes

- for GUIDE tab dialogs
- combo box procedures
  - uil-add-to-combo-box-list
  - uil-clear-combo-box-list
  - uil-combo-box-message-selection
  - uil-combo-box-no-selection
  - uil-combo-box-show-list
  - uil-combo-box-unselection
  - uil-create-combo-box
  - uil-update-combo-box-list
- combo box relations
- concluding
  - graphical objects, using uil-call-conclude-method-for-grobj
- confirmation dialogs
  - launching with uil-post-generic-dialog
- customer support services

## D

- delay notification icon
  - launching with uil-post-delay-notification
  - removing with uil-remove-delay-notification-if-any
- destination objects
  - finding with uil-find-destination-object
  - setting with uil-set-destination-object-for-dialog
- dialog procedures
  - uil-add-apply-button-to-dialog
  - uil-add-cancel-button-to-dialog
  - uil-add-grobj-to-dialog
  - uil-add-grobj-to-tab-page
  - uil-add-item-to-dialog
  - uil-add-message-to-dialog
  - uil-add-ok-button-to-dialog
  - uil-add-standard-buttons-to-dialog
  - uil-add-tab-page-to-dialog
  - uil-cancel-edit-session-in-window-if-any
  - uil-check-module-consistency-for-dialog
  - uil-clear-transient-dialog-pool
  - uil-control-dialog-callback
  - uil-create-confirm-dialog
  - uil-create-dialog-border
  - uil-create-message-dialog
  - uil-create-query-dialog
  - uil-create-tab-dialog
  - uil-create-tab-page
  - uil-create-tab-pushbutton
  - uil-create-tailored-dialog

- uil-delete-permanent-copies-in-transient-dialog-pool
- uil-end-dialog-edit
- uil-find-destination-object
- uil-find-initiating-object
- uil-find-initiating-object-for-parent-dialog
- uil-find-source-object
- uil-find-target-object
- uil-find-temporary-storage-object
- uil-find-workspace-object-maxima
- uil-flush-dialog-event-queue
- uil-generate-customized-dialog
- uil-generate-default-dialog
- uil-get-active-dialog-on-window-if-one-exists
- uil-get-child-dialogs
- uil-get-dialog-for-this-temporary-storage-object
- uil-get-dialog-launched-for-item
- uil-get-dialogs-managed-on-window
- uil-get-grobj-from-id
- uil-get-grobj-from-id-on-dlg-or-wksp
- uil-get-master-dialog
- uil-get-parent-dialog
- uil-get-window-on-which-dialog-is-managed
- uil-get-windows-on-which-dialog-is-managed
- uil-hide-managed-dialog
- uil-insert-actions-into-dialog-event-queue
- uil-is-dialog-managed-on-window
- uil-is-this-a-child-dialog-of-this-dialog
- uil-is-this-dialog-managed-for-this-item
- uil-is-this-the-parent-dialog
- uil-locate-first-edit-box-on-dialog
- uil-locate-next-edit-box-on-dialog
- uil-move-grobj-to-tab-page-layer
- uil-override-initiating-object-for-dialog
- uil-permanent-copies-exist-in-transient-pool
- uil-post-delay-notification
- uil-post-generic-dialog
- uil-post-notification-dialog
- uil-quit-dialog-edit
- uil-raise-tab-page-to-top
- uil-raise-tab-page-to-top-callback
- uil-release-dialog
- uil-remove-all-remaining-actions
- uil-remove-delay-notification-if-any
- uil-remove-grobj-from-tab-page
- uil-remove-notification-dialog

- uil-remove-tab-page-from-dialog
- uil-reserve-dialog
- uil-restore-dialog-grobject-list
- uil-select-tab-page
- uil-set-destination-object-for-dialog
- uil-show-managed-dialog
- uil-shrink-wrap-dialog
- uil-simulate-play-mode
- uil-start-dialog
- uil-start-dialog-processing
- uil-start-or-refocus-dialog
- uil-store-dialog-in-home-bin
- uil-translate-dialog
- uil-unsimulate-play-mode
- uil-update-controls-for-specific-attribute
- uil-update-dialog-for-initiating-object
- uil-update-notification-text
- uil-update-wait-clock-face
- uil-utility-create-permanent-copies-for-module
- uil-utility-create-permanent-copy-for-dialog
- uil-wait-for-dialog
- dialog title procedures
  - uil-create-and-manage-dialog-title
  - uil-create-dialog-title
  - uil-create-styled-dialog-title
  - uil-justify-dialog-title
  - uil-title-button-callback
- dialog titles
  - attributes of
  - class hierarchy of
  - internationalization of
  - operations performed on
  - procedures that operate on
- dialogs
  - adding actions to event queues of, using uil-insert-actions-into-dialog-event-queue
  - attributes of
  - cancelling edit session for
  - class hierarchy of
  - finding initiating objects of, using uil-find-initiating-object
  - finding source objects of, using uil-find-source-object
  - finding target objects of, using uil-find-target-object
  - launching with uil-control-dialog-callback
  - launching with uil-start-dialog
  - launching with uil-start-dialog-processing
  - launching with uil-start-or-refocus-dialog
  - managing with uil-show-managed-dialog

- operations on
- procedures that operate on
- refocusing with uil-start-or-refocus-dialog
- relations with other objects
- relations with supporting UIL objects
- releasing with uil-release-dialog
- setting destination objects for, using uil-set-destination-object-for-dialog
- setting initiating objects of, using uil-override-initiating-object-for-dialog
- unmanaging with uil-hide-managed-dialog
- updating with uil-update-dialog-for-initiating-object
- disabling
  - graphical objects, using uil-disable-grobject-method

## E

- edit box procedures
  - uil-add-to-combo-box-list
  - uil-clear-combo-box-list
  - uil-combo-box-message-selection
  - uil-combo-box-no-selection
  - uil-combo-box-show-list
  - uil-combo-box-unselection
  - uil-commit-edit-session-result-to-edit-box
  - uil-create-combo-box
  - uil-create-spin-control
  - uil-create-styled-edit-box
  - uil-delete-spin-control
  - uil-edit-box-selection
  - uil-edit-box-selection-method
  - uil-edit-box-unselection
  - uil-edit-box-unselection-method
  - uil-get-control-with-focus
  - uil-get-control-with-focus-on-window
  - uil-get-edit-box-value
  - uil-launch-edit-session
  - uil-locate-next-rows-edit-box-on-dialog
  - uil-register-edit-session-results
  - uil-select-edit-box
  - uil-select-edit-box-on-dialog
  - uil-unselect-edit-box
  - uil-update-combo-box-list
  - uil-update-edit-box-label
- edit boxes
  - attributes of
  - class hierarchy of
  - operations performed on

- relations with other objects
- selecting, using `uil-select-edit-box`
- unselecting, using `uil-unselect-edit-box`
- updating with `uil-update-text`
- enabling
  - graphical objects, using `uil-enable-grobject-method`

## G

- graphical message procedures
  - `uil-create-custom-text`
  - `uil-get-message-for-border`
  - `uil-return-size-of-grmes-as-symbol`
  - `uil-update-text`
  - `uil-update-unformatted-text`
- graphical messages
  - class hierarchy of
  - operations performed on
  - procedures that operate on
- graphical object procedures
  - `uil-call-conclude-method-for-grobject`
  - `uil-call-update-method-for-grobject`
  - `uil-call-validate-method-for-grobject`
  - `uil-change-size-of-grobject`
  - `uil-change-window-style-of-grobject`
  - `uil-clone-grobject`
  - `uil-clone-grobject-method`
  - `uil-conclude-grobject`
  - `uil-conclude-grobject-method`
  - `uil-configure-grobject`
  - `uil-configure-grobject-method`
  - `uil-delete-grobject`
  - `uil-delete-grobject-method`
  - `uil-delete-grobject-with-confirmation`
  - `uil-disable-grobject`
  - `uil-disable-grobject-method`
  - `uil-drop-border-around-grobject`
  - `uil-enable-grobject`
  - `uil-enable-grobject-method`
  - `uil-initialize-grobject`
  - `uil-initialize-grobject-method`
  - `uil-make-grobject-permanent`
  - `uil-manage-grobject`
  - `uil-manage-grobject-method`
  - `uil-move-grobject`
  - `uil-rotate-slider`
  - `uil-set-slider-value`
  - `uil-size-of-grobject`
  - `uil-size-of-grobject-method`

- `uil-unmanage-grobject`
- `uil-update-grobject`
- `uil-update-grobject-method`
- `uil-utility-restore-grobject`
- `uil-validate-grobject`
- `uil-validate-grobject-method`
- graphical objects
  - class hierarchy of
  - concluding with `uil-call-conclude-method-for-grobject`
  - configuring with `uil-configure-grobject-method`
  - disabling with `uil-disable-grobject-method`
  - enabling with `uil-enable-grobject-method`
  - locating with `uil-get-grobject-from-id-on-dlg-or-wksp`
  - operations performed on
  - procedures that operate on
  - updating with `uil-call-update-method-for-grobject`
  - validating with `uil-call-validate-method-for-grobject`
- `guidata-launch-dialog-for-item`
- GUIDE
  - tab dialogs
  - classes for

## I

- initiating objects
  - finding with `uil-find-initiating-object`
- internationalization
  - of a `uil-text` or `uil-button`
  - of dialog titles
  - of text in dialogs

## L

- launching
  - confirmation dialogs, using `uil-post-generic-dialog`
  - message dialogs, using `uil-post-generic-dialog`
  - notification dialogs, using `uil-post-generic-dialog`
  - query dialogs, using `uil-post-generic-dialog`
- launching dialogs
  - with `uil-control-dialog-callback`
  - with `uil-start-dialog`

- with `uil-start-dialog-processing`
- with `uil-start-or-refocus-dialog`

## M

- make UIL controls permanent
- managing dialogs
  - with `uil-show-managed-dialog`
- message dialogs
  - launching with `uil-post-generic-dialog`
- message object procedure
  - `uil-append-item-to-message`
- message object procedures
  - `uil-add-message-to-list`
  - `uil-add-message-to-list-at-position`
  - `uil-clear-messages`
  - `uil-copy-message-object-system-attributes`
  - `uil-copy-messages`
  - `uil-create-message`
  - `uil-create-message-with-appended-item`
  - `uil-delete-selected-messages`
  - `uil-display-message-list`
  - `uil-does-message-belong-to-scroll-area`
  - `uil-find-excess-message-and-remove`
  - `uil-generate-time-stamp-for-message`
  - `uil-get-first-message`
  - `uil-get-index-of-scroll-message`
  - `uil-get-item-appended-to-message`
  - `uil-get-last-message`
  - `uil-get-message-appended-to-this-item`
  - `uil-get-message-from-index`
  - `uil-get-message-object-state`
  - `uil-get-scroll-area-for-this-message`
  - `uil-get-selected-message`
  - `uil-get-selected-message-objects`
  - `uil-insert-message-in-ordered-list`
  - `uil-is-message-selected`
  - `uil-message-selection`
  - `uil-message-selection-method`
  - `uil-message-unselection`
  - `uil-message-unselection-method`
  - `uil-move-message`
  - `uil-move-messages`
  - `uil-on-double-click`
  - `uil-on-single-click`
  - `uil-remove-message`
  - `uil-remove-message-from-list`
  - `uil-select-message`
  - `uil-select-message-on-window`
  - `uil-sort-scroll-area-messages`

- `uil-unselect-message`
- `uil-update-message-list-with-message`
- message objects
  - attributes of
  - class hierarchy of
  - creating, using `uil-create-message`
  - deleting, using `uil-delete-selected-messages`
  - displaying, using `uil-display-message-list`
  - getting from index, using `uil-get-message-from-index`
  - getting indexes of, using `uil-get-index-of-scroll-message`
  - operations performed on
  - procedure for appending items to
  - procedures that operate on
  - relations with scroll areas
  - selecting, using `uil-message-selection`
  - selecting, using `uil-select-message`
  - selecting, using `uil-unselect-message`
  - sorting, using `uil-sort-scroll-area-messages`
  - unselecting, using `uil-message-unselection`
  - updating with `uil-update-text`
- modules
  - GUIDE
  - `uieldg`

## N

- navigation button procedures
  - `uil-create-navigation-button`
  - `uil-handle-navigation-button-method`
  - `uil-handle-workspace-button-method`
  - `uil-select-navigation-button`
- navigation buttons
  - attributes of
  - class hierarchy of
  - operations performed on
  - relations with other objects
- notification dialogs
  - launching with `uil-post-generic-dialog`

## P

- procedure priority setting
- procedures
  - `uil-create-tab-dialog`
  - for GUIDE

- push button procedures
  - uil-create-pushbutton-to-launch-dialog
  - uil-create-styled-pushbutton
  - uil-unselect-button
- push buttons
  - attributes of
  - class hierarchy of
  - operations performed on
  - relations with other objects

## Q

- query dialogs
  - launching with uil-post-generic-dialog

## R

- radio box procedures
  - uil-create-radio-box
  - uil-get-radio-box-value
  - uil-reset-radio-box
  - uil-set-radio-box-value
- radio boxes
  - attributes of
  - class hierarchy of
  - operations performed on
  - procedures that operate on
  - relations with radio buttons
- radio button procedures
  - uil-create-styled-radio-button
  - uil-create-styled-radio-button-with-different-sized-label
- radio buttons
  - attributes of
  - class hierarchy of
  - operations performed on
  - relations with radio boxes
- refocusing dialogs
  - with uil-start-or-refocus-dialog
- relation procedures
  - uil-are-any-related-to
  - uil-get-related-item
  - uil-get-related-items
  - uil-is-related-to
- relations
  - among cascaded dialogs
  - between a dialog and supporting UIL objects
  - for tab pages
  - referencing in procedures

- releasing dialogs
  - with uil-release-dialog

## S

- scroll area procedures
  - uil-add-resize-corner-to-scroll-area
  - uil-change-scroll-area-message-font-size
  - uil-create-customized-scroll-area
  - uil-create-scroll-area-from-array
  - uil-create-scroll-area-from-list
  - uil-create-sized-scroll-area-from-array
  - uil-create-sized-scroll-area-from-list
  - uil-create-styled-scroll-area
  - uil-get-related-scroll-area
  - uil-reset-scroll-area
  - uil-resize-scroll-area
  - uil-return-module-name-for-scroll-area-definition
  - uil-scroll-area-reflect
  - uil-scroll-decrement
  - uil-scroll-decrement-line
  - uil-scroll-decrement-line-method
  - uil-scroll-decrement-method
  - uil-scroll-increment
  - uil-scroll-increment-line
  - uil-scroll-increment-line-method
  - uil-scroll-increment-method
  - uil-scroll-to-position
  - uil-scroll-to-position-method
  - uil-select-matching-values-in-scroll-area
  - uil-update-scroll-area-from-array
  - uil-update-scroll-area-from-list
  - uil-update-scroll-area-message
  - uil-update-scroll-area-messages
- scroll areas
  - attributes of
  - class hierarchy of
  - operations performed on
  - relations with message objects
- selecting
  - buttons, using uil-select-button
  - edit boxes, using uil-select-edit-box
  - message objects, using uil-message-selection
  - message objects, using uil-select-message
- selection box procedures
  - uil-add-button-to-selection-box
  - uil-get-buttons-for-this-selection-box



- uil-get-buttons-grouped-by-this-selection-box
- uil-get-selection-box-for-button
- uil-get-selection-box-last-button
- uil-get-selection-box-lead-button
- uil-utility-check-for-empty-selection-box
- selection boxes
  - attributes of
  - class hierarchy of
  - operations performed on
  - relations with selection buttons
- selection button procedures
  - uil-get-next-selection-button-y-position
  - uil-get-selection-button
  - uil-is-this-button-grouped-by-this-selection-box
- selection buttons
  - attributes of
  - class hierarchy of
  - operations performed on
  - procedures that operate on
  - relations with selection boxes
- separator procedures
  - uil-create-line-separator
  - uil-rotate-line-separator
- separators
  - attributes of
  - class hierarchy of
  - operations performed on
  - procedures that operate on
- slider procedures
  - uil-create-slider
  - uil-get-slider-value
  - uil-rotate-slider
- source objects
  - finding for dialogs, using uil-find-source-object
- spin control procedures
  - uil-create-spin-control
  - uil-delete-spin-control
- spin control relations
- superseded procedures
  - uil-apply-edit-dialog-callback
  - uil-clear-transient-dialog-pool-and-permanent-copies
  - uil-create-check-button
  - uil-create-edit-box
  - uil-create-pushbutton
  - uil-create-radio-button
  - uil-create-scroll-area
  - uil-create-sized-edit-box

- uil-create-text
- uil-create-toggle-button
- uil-delete-check-box
- uil-delete-check-button
- uil-delete-dialog
- uil-delete-edit-box
- uil-delete-navigation-button
- uil-delete-pushbutton
- uil-delete-radio-box
- uil-delete-radio-button
- uil-delete-scroll-area
- uil-delete-text
- uil-delete-toggle-button
- uil-disable-button
- uil-enable-button
- uil-frame-selection-box
- uil-get-query-dialog-value
- uil-manage-and-show-dialog
- uil-popdown-and-delete-dialog-callback
- uil-popdown-and-hide-managed-dialog-callback
- uil-popdown-and-keep-edit-dialog-callback
- uil-popdown-dialog-callback
- uil-update-wait-semaphore

## T

- tab dialogs
  - classes for
- tab pages
  - relations for
- target objects
  - finding with uil-find-target-object
- text object procedures
  - uil-create-styled-text
  - uil-get-grobject-for-label
  - uil-get-label-text
  - uil-select-text
  - uil-translate-dialog
  - uil-translate-text
- text objects
  - attributes of
  - class hierarchy of
  - operations performed on
  - procedures that operate on
  - relations with other objects
  - setting values of, using uil-update-text
- toggle button procedures
  - uil-create-styled-toggle-button
  - uil-get-toggle-button-state

- uil-set-toggle-button-state
- toggle buttons
  - attributes of
  - class hierarchy of
  - operations performed on
  - relations with other objects
- translation of text in dialogs

## U

- uil-30r1-upgrades
- uil-30r3-and-30r4-upgrades
- uil-30r3-upgrade-button-label-texts
- uil-30r3-upgrade-radio-and-check-button-label-texts
- uil-30r3-upgrade-scroll-area-keys
- uil-40r0-upgrades
- uil-add-apply-button-to-dialog
- uil-add-button-to-selection-box
- uil-add-cancel-button-to-dialog
- uil-add-grobject-to-dialog
- uil-add-grobject-to-tab-page
- uil-add-item-to-dialog
- uil-add-label-to-button
- uil-add-message-to-dialog
- uil-add-message-to-list
- uil-add-message-to-list-at-position
- uil-add-ok-button-to-dialog
- uil-add-prefix-to-user-criteria-prefixes
- uil-add-resize-corner-to-scroll-area
- uil-add-standard-buttons-to-dialog
- uil-add-tab-page-to-dialog
- uil-add-to-combo-box-list
- uil-adjust-border-on-workspace
- uil-anchor-box-corner-to-workspace-edge
- uil-append-item-to-message
- uil-apply-edit-dialog-callback
- uil-are-any-related-to
- uil-box-border class
  - See* border object procedures
- uil-button class
  - See* button procedures
- uil-call-conclude-method-for-grobject
- uil-call-update-method-for-grobject
- uil-call-validate-method-for-grobject
- uil-cancel-edit-session-in-window-if-any
- uil-center-label-in-text-button
- uil-change-scroll-area-message-font-size
- uil-change-size-of-grobject
- uil-change-window-style-of-grobject
- uil-check-box class
  - See* check box procedures
- uil-check-button class
  - See* check button procedures
- uil-check-module-consistency-for-dialog
- uil-clear-combo-box-list
- uil-clear-messages
- uil-clear-transient-dialog-pool
- uil-clear-transient-dialog-pool-and-permanent-copies
- uil-clone-grobject
- uil-clone-grobject-method
- uil-combo-box-message-selection
- uil-combo-box-no-selection
- uil-combo-box-show-list
- uil-combo-box-unselection
- uil-commit-edit-session-result-to-edit-box
- uil-conclude-grobject
- uil-conclude-grobject-method
- uil-configure-grobject
- uil-configure-grobject-method
- uil-control-dialog-callback
- uil-convert-button-from-4.1-to-5.0
- uil-convert-buttons-from-4.1-to-5.0
- uil-convert-dialog-subworkspaces-from-4.1-to-5.0
- uil-convert-scroll-area-to-41r0
- uil-copy-message-object-system-attributes
- uil-copy-messages
- uil-create-and-manage-dialog-title
- uil-create-border-for-text
- uil-create-box-border
- uil-create-check-box
- uil-create-check-button
- uil-create-combo-box
- uil-create-confirm-dialog
- uil-create-custom-button
- uil-create-customized-scroll-area
- uil-create-custom-text
- uil-create-dialog-border
- uil-create-dialog-title
- uil-create-edit-box
- uil-create-line-separator
- uil-create-message
- uil-create-message-dialog
- uil-create-message-with-appended-item
- uil-create-message-with-appended-item procedure
- uil-create-navigation-button
- uil-create-pushbutton
- uil-create-pushbutton-to-launch-dialog

- uil-create-query-dialog
- uil-create-radio-box
- uil-create-radio-button
- uil-create-scroll-area
- uil-create-scroll-area-from-array
- uil-create-scroll-area-from-list
- uil-create-sized-edit-box
- uil-create-sized-scroll-area-from-array
- uil-create-sized-scroll-area-from-list
- uil-create-slider
- uil-create-spin-control
- uil-create-styled-check-button
- uil-create-styled-check-button-with-different-sized-label
- uil-create-styled-dialog-title
- uil-create-styled-edit-box
- uil-create-styled-pushbutton
- uil-create-styled-radio-button
- uil-create-styled-radio-button-with-different-styled-label
- uil-create-styled-scroll-area
- uil-create-styled-text
- uil-create-styled-toggle-button
- uil-create-tab-dialog
- uil-create-tab-dialog procedure  
for GUIDE tab dialogs
- uil-create-tab-page
- uil-create-tab-pushbutton
- uil-create-tailored-dialog
- uil-create-text
- uil-create-text-for-button-from-label
- uil-create-toggle-button
- uil-create-uilib-module-setting
- uil-date-verification
- uil-delete-check-box
- uil-delete-check-button
- uil-delete-dialog
- uil-delete-edit-box
- uil-delete-grobj
- uil-delete-grobj-method
- uil-delete-grobj-with-confirmation
- uil-delete-navigation-button
- uil-delete-permanent-copies-in-transient-dialog-pool
- uil-delete-pushbutton
- uil-delete-radio-box
- uil-delete-radio-button
- uil-delete-scroll-area
- uil-delete-selected-messages
- uil-delete-spin-control
- uil-delete-text
- uil-delete-toggle-button
- uil-determine-if-guide-is-loaded
- uil-dialog class  
See dialog procedures
- uil-dialog-title class  
See dialog title procedures
- uil-disable-button
- uil-disable-grobj
- uil-disable-grobj-method
- uil-display-color-selector-dialog
- uil-display-message-list
- uildlg module  
for GUIDE
- uil-does-message-belong-to-scroll-area
- uil-do-nothing
- uil-drop-border-around-grobj
- uil-edit-box class  
See edit box procedures
- uil-edit-box-selection
- uil-edit-box-selection-method
- uil-edit-box-unselection
- uil-edit-box-unselection-method
- uil-enable-button
- uil-enable-grobj
- uil-enable-grobj-method
- uil-end-dialog-edit
- uil-find-destination-object
- uil-find-excess-message-and-remove
- uil-find-initiating-object
- uil-find-initiating-object-for-parent-dialog
- uil-find-source-object
- uil-find-target-object
- uil-find-temporary-storage-object
- uil-find-workspace-object-maxima
- uil-flush-dialog-event-queue
- uil-frame-selection-box
- uil-generate-customized-dialog
- uil-generate-default-dialog
- uil-generate-rules-for-attributes
- uil-generate-time-stamp-for-message
- uil-get-active-dialog-on-window-if-one-exists
- uil-get-attribute
- uil-get-border
- uil-get-buttons-for-this-selection-box
- uil-get-buttons-grouped-by-this-selection-box
- uil-get-check-box-values
- uil-get-child-dialogs
- uil-get-configuration
- uil-get-control-with-focus
- uil-get-control-with-focus-on-window
- uil-get-dialog-for-this-temporary-storage-object

- uil-get-dialog-launched-for-item
- uil-get-dialogs-managed-on-window
- uil-get-edit-box-value
- uil-get-first-message
- uil-get-grobject-for-label
- uil-get-grobject-from-id
- uil-get-grobject-from-id-on-dlg-or-wksp
- uil-get-index-of-scroll-message
- uil-get-item-appended-to-message
- uil-get-label-text
- uil-get-last-message
- uil-get-master-dialog
- uil-get-message-appended-to-this-item
- uil-get-message-for-border
- uil-get-message-from-index
- uil-get-message-object-state
- uil-get-next-selection-button-y-position
- uil-get-parent-dialog
- uil-get-query-dialog-value
- uil-get-radio-box-value
- uil-get-related-item
- uil-get-related-items
- uil-get-related-scroll-area
- uil-get-scroll-area-for-this-message
- uil-get-selected-message
- uil-get-selected-message-objects
- uil-get-selection-box-for-button
- uil-get-selection-box-last-button
- uil-get-selection-box-lead-button
- uil-get-selection-button
- uil-get-slider-value
- uil-get-text-extent
- uil-get-toggle-button-state
- uil-get-user-criteria-prefixes
- uil-get-window-on-which-dialog-is-managed
- uil-get-windows-on-which-dialog-is-managed
- uil-grmes class
  - See graphical message procedures*
- uil-grobject class
  - See graphical object procedures*
- uil-guidata-launch-dialog-for-item
- uil-handle-button
- uil-handle-button-method
- uil-handle-navigation-button-method
- uil-handle-workspace-button-method
- uil-hide-managed-dialog
- uil-highlight-button
- uil-icon-pushbutton class
  - See push button procedures*
- uil-icon-toggle-button class
  - See toggle button procedures*
- uil-initialize-grobject
- uil-initialize-grobject-method
- uil-insert-actions-into-dialog-event-queue
- uil-insert-message-in-ordered-list
- uil-is-dialog-managed-on-window
- uil-is-message-selected
- uil-is-related-to
- uil-is-this-a-child-dialog-of-this-dialog
- uil-is-this-button-grouped-by-this-selection-box
- uil-is-this-dialog-managed-for-this-item
- uil-is-this-item-near-another
- uil-is-this-the-parent-dialog
- uil-justify-dialog-title
- uil-launch-edit-session
- uil-line-separator class
  - See separator procedures*
- uil-locate-first-edit-box-on-dialog
- uil-locate-next-edit-box-on-dialog
- uil-locate-next-rows-edit-box-on-dialog
- uil-make-grobject-permanent
- uil-manage-and-show-dialog
- uil-manage-grobject
- uil-manage-grobject-method
- uil-message-object class
  - See message object procedures*
- uil-message-selection
- uil-message-selection-method
- uil-message-unselection
- uil-message-unselection-method
- uil-move-grobject
- uil-move-grobject-to-tab-page-layer
- uil-move-message
- uil-move-messages
- uil-move-related-border
- uil-navigation-button class
  - See navigation button procedures*
- uil-on-double-click
- uil-on-single-click
- uil-override-initiating-object-for-dialog
- uil-permanent-copies-exist-in-transient-dialog-pool
- uil-popdown-and-delete-dialog-callback
- uil-popdown-and-hide-managed-dialog-callback
- uil-popdown-and-keep-edit-dialog-callback
- uil-popdown-dialog-callback
- uil-post-delay-notification
- uil-post-generic-dialog
- uil-post-notification-dialog
- uil-quit-dialog-edit
- uil-radio-box class

*See* radio box procedures  
 uil-radio-button class  
     *See* radio button procedures  
 uil-raise-tab-page-to-top  
 uil-raise-tab-page-to-top-callback  
 uil-register-edit-session-results  
 uil-release-box-corner-from-workspace-edge  
 uil-release-dialog  
 uil-remove-all-remaining-actions  
 uil-remove-delay-notification-if-any  
 uil-remove-grobject-from-tab-page  
 uil-remove-guide-application  
 uil-remove-message  
 uil-remove-message-from-list  
 uil-remove-notification-dialog  
 uil-remove-prefix-from-user-criteria-prefixes  
 uil-remove-tab-page-from-dialog  
 uil-reserve-dialog  
 uil-reset-button  
 uil-reset-radio-box  
 uil-reset-scroll-area  
 uil-reset-uilib-module-setting-to-defaults  
 uil-reset-user-criteria-prefixes  
 uil-resize-scroll-area  
 uil-restore-dialog-grobject-list  
 uil-return-bin-workspace-for  
 uil-return-configuration-for-color  
 uil-return-module-name-for-scroll-area-  
     definition  
 uil-return-size-of-button-as-symbol  
 uil-return-size-of-grobject-as-symbol  
 uil-rotate-line-separator  
 uil-rotate-slider  
 uil-scroll-area class  
     *See* scroll area procedures  
 uil-scroll-area-reflect  
 uil-scroll-decrement  
 uil-scroll-decrement-line  
 uil-scroll-decrement-line-method  
 uil-scroll-decrement-method  
 uil-scroll-increment  
 uil-scroll-increment-line  
 uil-scroll-increment-line-method  
 uil-scroll-increment-method  
 uil-scroll-to-position  
 uil-scroll-to-position-method  
 uil-select-button  
 uil-select-edit-box  
 uil-select-edit-box-on-dialog  
 uil-selection-box class  
     *See* selection box procedures

uil-selection-button class  
     *See* selection button procedures  
 uil-select-matching-values-in-scroll-area  
 uil-select-message  
 uil-select-message-on-window  
 uil-select-navigation-button  
 uil-select-tab-page  
 uil-select-text  
 uil-set-attribute  
 uil-set-check-box-values  
 uil-set-destination-object-for-dialog  
 uil-set-radio-box-value  
 uil-set-slider-value  
 uil-set-toggle-button-state  
 uil-show-managed-dialog  
 uil-shrink-wrap-dialog  
 uil-simulate-play-mode  
 uil-size-of-grobject  
 uil-size-of-grobject-method  
 uil-sort-scroll-area-messages  
 uil-start-dialog  
 uil-start-dialog-processing  
 uil-start-or-refocus-dialog  
 uil-store-dialog-in-home-bin  
 uil-tab-dialog class  
 uil-tab-page class  
 uil-tab-pushbutton class  
 uil-text class  
     *See* text object procedures  
 uil-text-pushbutton class  
     *See* push button procedures  
 uil-text-toggle-button class  
     *See* toggle button procedures  
 uil-time-verification  
 uil-title-button-callback  
 uil-toggle-button  
 uil-translate-dialog  
 uil-translate-text  
 uil-unmanage-grobject  
 uil-unselect-button  
 uil-unselect-edit-box  
 uil-unselect-message  
 uil-unsimulate-play-mode  
 uil-update-button-label  
 uil-update-button-togglelabel  
 uil-update-combo-box-list  
 uil-update-controls-for-specific-attribute  
 uil-update-dialog-for-initiating-object  
 uil-update-edit-box-label  
 uil-update-grobject  
 uil-update-grobject-method

- uil-update-message-list-with-message
- uil-update-notification-text
- uil-update-scroll-area-from-array
- uil-update-scroll-area-from-list
- uil-update-scroll-area-message
- uil-update-scroll-area-messages
- uil-update-text
- uil-update-unformatted-text
- uil-update-wait-clock-face
- uil-update-wait-semicolon
- uil-utility-build-symbol-list-of-values-for-class-attribute
- uil-utility-change-module
- uil-utility-check-for-empty-selection-box
- uil-utility-check-for-orphaned-selection-buttons
- uil-utility-cleanup-dialog-bin
- uil-utility-convert-from-edit-box-to-integer
- uil-utility-convert-from-edit-box-to-quantity
- uil-utility-convert-from-edit-box-to-symbol
- uil-utility-convert-from-edit-box-to-value
- uil-utility-create-permanent-copies-for-module
- uil-utility-create-permanent-copy-for-dialog
- uil-utility-create-procedure-xref-listing
- uil-utility-restore-grobj
- uil-utility-return-current-real-time
- uil-utility-synchronize-text-fields-for-uil-text
- uil-utility-unique-id
- uil-utility-unique-name
- uil-validate-grobj
- uil-validate-grobj-method
- uil-wait-for-dialog
- unmanaging dialogs
  - with uil-hide-managed-dialog
- unselecting
  - edit boxes, using uil-unselect-edit-box
  - message objects, using uil-message-unselection
  - message objects, using uil-unselect-message
- updating
  - dialogs, using uil-update-dialog-for-initiating-object
  - edit boxes, using uil-update-text
  - graphical objects, using uil-call-update-method-for-grobj
  - message objects, using uil-update-text
  - scroll areas from arrays, using uil-update-scroll-area-from-array
  - scroll areas from lists, using uil-update-scroll-area-from-list
  - text objects, using uil-update-text
- upgrade procedures
  - uil-30r1-upgrades
  - uil-30r3-and-30r4-upgrades
  - uil-30r3-upgrade-button-label-texts
  - uil-30r3-upgrade-radio-and-check-button-label-texts
  - uil-30r3-upgrade-scroll-area-keys
  - uil-40r0-upgrades
  - uil-convert-button-from-4.1-to-5.0
  - uil-convert-buttons-from-4.1-to-5.0
  - uil-convert-dialog-subworkspaces-from-4.1-to-5.0
  - uil-convert-scroll-area-to-41r0
- utility procedures
  - guidata-launch-dialog-for-item
  - uil-add-prefix-to-user-criteria-prefixes
  - uil-create-uilib-module-setting
  - uil-date-verification
  - uil-determine-if-guide-is-loaded
  - uil-display-color-selector-dialog
  - uil-do-nothing
  - uil-generate-rules-for-attributes
  - uil-get-attribute
  - uil-get-configuration
  - uil-get-text-extent
  - uil-get-user-criteria-prefixes
  - uil-guidata-launch-dialog-for-item
  - uil-is-this-item-near-another
  - uil-remove-guide-application
  - uil-remove-prefix-from-user-criteria-prefixes
  - uil-reset-uilib-module-setting-to-defaults
  - uil-reset-user-criteria-prefixes
  - uil-return-bin-workspace-for
  - uil-return-configuration-for-color
  - uil-set-attribute
  - uil-time-verification
  - uil-utility-build-symbol-list-of-values-for-class-attribute
  - uil-utility-change-module
  - uil-utility-check-for-orphaned-selection-buttons
  - uil-utility-cleanup-dialog-bin
  - uil-utility-convert-from-edit-box-to-integer
  - uil-utility-convert-from-edit-box-to-quantity
  - uil-utility-convert-from-edit-box-to-symbol
  - uil-utility-convert-from-edit-box-to-value
  - uil-utility-create-procedure-xref-listing
  - uil-utility-return-current-real-time
  - uil-utility-synchronize-text-fields-for-uil-text
  - uil-utility-unique-id

- uil-utility-unique-name
- util-clone-through-copying
- util-convert-from-text-to-integer
- util-convert-from-text-to-quantity
- util-convert-from-text-to-symbol
- util-convert-from-text-to-value
- util-convert-list-to-text
- util-copy-object-attributes
- util-copy-symbol-array-to-initial-values
- util-copy-symbol-array-to-symbol-array
- util-copy-symbol-array-to-symbol-list
- util-copy-symbol-list-to-symbol-array
- util-copy-symbol-list-to-symbol-list
- util-copy-text-array-to-initial-values
- util-copy-text-list-to-text-list
- util-create-instance-of-class
- util-get-module-information
- util-get-module-owning-this-workspace
- util-get-top-level-workspace-of-this-workspace
- util-goto-superior
- util-goto-workspace
- util-hide-workspace
- util-insert-text-in-alphabetized-list
- util-insert-text-in-reversed-alphabetized-list
- util-instances-exist
- util-lift-to-top
- util-remove-hyphens-from-text
- util-remove-module-from-hierarchy
- util-return-clipped-string
- util-show-superior-workspace
- util-transfer-to-at

- util-clone-through-copying
- util-convert-from-text-to-integer
- util-convert-from-text-to-quantity
- util-convert-from-text-to-symbol
- util-convert-from-text-to-value
- util-convert-list-to-text
- util-copy-object-attributes
- util-copy-symbol-array-to-initial-values
- util-copy-symbol-array-to-symbol-array
- util-copy-symbol-array-to-symbol-list
- util-copy-symbol-list-to-symbol-array
- util-copy-symbol-list-to-symbol-list
- util-copy-text-array-to-initial-values
- util-copy-text-list-to-text-list
- util-create-instance-of-class
- util-get-module-information
- util-get-module-owning-this-workspace
- util-get-top-level-workspace-of-this-workspace
- util-goto-superior

- util-goto-workspace
- util-hide-workspace
- util-insert-text-in-alphabetized-list
- util-insert-text-in-reversed-alphabetized-list
- util-instances-exists
- util-lift-to-top
- util-remove-hyphens-from-text
- util-remove-module-from-hierarchy
- util-return-clipped-string
- util-show-superior-workspace
- util-transfer-to-at

## V

- validating
  - graphical objects, using uil-call-validate-method-for-grobj

