



Indesca / Infitran Utilities

Reference Guide

Indesca / Infitran

Version 4.1.0

Indesca / Infitran Utilities

Reference Guide

Indesca / Infitran 4.1.0

Document Name	Indesca / Infitran Utilities 4.1.0 Reference Guide				
Document ID	util-ref-4100				
Indesca / Infitran Utilities	z/OS	UNIX	Windows	OS/400	HP NonStop*
Universal Certificate	√	√	√		
Universal Control	√	√	√	√	√
Universal Copy		√	√	√	√
Universal Database Dump	√	√	√		
Universal Database Load	√	√	√		
Universal Display Log File				√	
Universal Encrypt	√	√	√	√	√
Universal Event Log Dump			√		
Universal Message Translator	√	√	√	√	√
Universal Query	√	√	√	√	√
Universal Spool List	√	√	√	√	
Universal Spool Remove	√	√	√	√	
Universal Submit Job				√	
Universal WTO	√				

* Utilities for Universal Products 2.1.1 is used on the HP NonStop operating system.

Stonebranch Documentation Policy

This document contains proprietary information that is protected by copyright. All rights reserved. No part of this publication may be reproduced, transmitted or translated in any form or language or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission, in writing, from the publisher. Requests for permission to make copies of any part of this publication should be mailed to:

Stonebranch, Inc.
950 North Point Parkway, Suite 200
Alpharetta, GA 30005 USA
Tel: (678) 366-7887
Fax: (678) 366-7717

Stonebranch, Inc.[®] makes no warranty, express or implied, of any kind whatsoever, including any warranty of merchantability or fitness for a particular purpose or use.

The information in this documentation is subject to change without notice.

Stonebranch shall not be liable for any errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this document.

All products mentioned herein are or may be trademarks of their respective owners.

© 2003-2010 by Stonebranch, Inc.

All rights reserved.



Summary of Changes

Changes for Indesca / Infitran Utilities 4.1.0 Reference Guide (util-ref-4100) February 10, 2010

Universal Submit Job 4.1.0.0

- Added Section [18.2.3 Universal Command Server Options Affecting USBMJOB](#).

Universal Database Dump 4.1.0.0

- Replaced RECOVER option with [DUMP_OPTIONS](#) option in [Chapter 8 Universal Database Dump](#).
- Modified [OVERWRITE](#) option in [Chapter 9 Universal Database Load](#).

Changes for Universal Products Utilities 3.2.0 Reference Guide (util-ref-3204) September 8, 2009

Universal Products 3.2.0.8

- Added the following code pages in Section [20.4 Character Code Pages](#):
 - IBM875
 - IBM4971

Changes for Universal Products Utilities 3.2.0 Reference Guide (util-ref-3203) July 29, 2009

Universal Products Utilities 3.2.0.1 for OS/400

- Modified document for upgrade from Universal Products Utilities 3.1.1 for OS/400 to Universal Products Utilities 3.2.0 for OS/400, including:
 - Changed the following OS/400 names throughout the document:
 - Universal Broker subsystem name from **UBROKER** to **UNVUBR320**.
 - Universal Broker user profile name from **UBROKER** to **UNVUBR320**.
 - Universal Products installation library name from **UNIVERSAL** to **UNVPRD320**.
 - Universal Products spool library name from **UNVSPPOOL** to **UNVSPL320**.
 - Universal Products temporary directory from **UNVTMP** to **UNVTMP320**.

Universal Control 3.2.0.1 for OS/400

- Specified the following configuration option for OS/400 in [Chapter 3 Universal Control Manager Configuration Options](#):
 - [ACTIVITY_MONITORING](#)
 - [CERTIFICATE_REVOCACTION_LIST](#)
 - [EVENT_GENERATION](#)
 - [OUTBOUND_IP](#)

Universal Query 3.2.0.1 for OS/400

- Specified the following configuration options for OS/400 in [Chapter 15 Universal Query](#):
 - [COMMAND_ID](#)
 - [COMPONENT_ID](#)
 - [MANAGERS](#)
 - [PLF_DIRECTORY](#)

Universal Spool 3.2.0.1 for OS/400

- Specified the following configuration options for OS/400 in [Chapter 16 Universal Spool List](#):
 - [COMPONENT](#)
 - [ID](#)
 - [LIST](#)
 - [MESSAGE_LEVEL](#)
 - [VERSION](#)
- Specified the following configuration options for OS/400 in [Chapter 17 Universal Spool Remove](#):
 - [COMPONENT](#)
 - [MESSAGE_LEVEL](#)
 - [VERSION](#)

Universal Submit Job 3.2.0.1 for OS/400

- Added the following SBMJOB Encapsulated configuration option in [Chapter 18 Universal Submit Job](#):
 - INLASPGRP

Changes for Universal Products Utilities 3.2.0 Reference Guide (util-ref-3202)**December 17, 2008**

- Changed the name of the environment variable for the Universal Control Manager [SYSTEM_ID](#) configuration option from UCTLSYSTEM to UCTLSYSTEMID.
- Changed the name of the environment variable for the Universal Query [SYSTEM_ID](#) configuration option from UQRYSYSTEM to UQRYSYSTEMID.

Changes for Universal Products Utilities 3.2.0 Reference Guide (util-ref-3201)**September 5, 2008**

- Added toll-free telephone number for North America in [Appendix A Customer Support](#).

Changes for Universal Products Utilities 3.2.0 Reference Guide (util-ref-320)**May 16, 2008****Universal Products 3.2.0.0**

- Added the following chapters:
 - [Chapter 8 Universal Database Dump](#)
 - [Chapter 9 Universal Database Load](#)
 - [Chapter 14 Universal Products Install Merge](#)
- Added the following configuration options in [Chapter 2 Universal Certificate](#):
 - [CERT_DB](#)
 - [CRL_FILE](#)
 - [CRL_FORMAT](#)
 - [NEXT_UPDATE_DAYS](#)
 - [NEXT_UPDATE_HOURS](#)
 - [REVOKE](#)
 - [REVOKE_REASON](#)
 - [STATE](#)
 - [VERIFY](#)
 - [TRANSPORT_FILE](#)

- [TRANSPORT_FILE_PWD](#)
- Added the following configuration options in [Chapter 12 Universal Event Log Dump](#):
 - [INSTALLATION_DIRECTORY](#)
 - [LOG_DIRECTORY](#)
 - [NLS_DIRECTORY](#)
- Added the following configuration options in [Chapter 15 Universal Query](#):
 - [BIF_DIRECTORY](#)
 - [COMMAND_ID](#)
 - [COMPONENT_ID](#)
 - [MANAGERS](#)
 - [NLS_DIRECTORY](#)
 - [PLF_DIRECTORY](#)
 - [SYSTEM_ID](#)

Contents

Summary of Changes	5
Contents	9
List of Tables	19
Preface	21
Document Structure	21
Cross-Reference Links	21
Conventions	22
Vendor References	23
Document Organization	24
Chapter 1 Overview	26
1.1 Configuration Options Information	27
Chapter 2 Universal Certificate	29
2.1 Overview	29
2.2 Configuration Options List	30
2.3 CA	32
2.4 CA_CERT_FILE	33
2.5 CA_CERT_FORMAT	34
2.6 CERT_DB	35
2.7 CERT_FILE	36
2.8 CERT_FORMAT	37

2.9	CODE_PAGE	38
2.10	COMMAND_FILE_ENCRYPTED	39
2.11	COMMAND_FILE_PLAIN	40
2.12	COMMON_NAME	41
2.13	COUNTRY	42
2.14	CREATE	44
2.15	CRL_FILE	45
2.16	CRL_FORMAT	46
2.17	DNS_NAME	47
2.18	EMAIL_ADDRESS	48
2.19	ENCRYPTION_KEY	49
2.20	HELP	50
2.21	IP_ADDRESS	51
2.22	KEY_SIZE	52
2.23	LOCALITY	53
2.24	MESSAGE_LEVEL	54
2.25	NEXT_UPDATE_DAYS	56
2.26	NEXT_UPDATE_HOURS	57
2.27	NLS_DIRECTORY	58
2.28	NOT_AFTER_DATE	59
2.29	NOT_BEFORE_DATE	60
2.30	ORGANIZATION	61
2.31	ORGANIZATIONAL_UNIT	62
2.32	PRINT	63
2.33	PRIVATE_KEY_FILE	64
2.34	PRIVATE_KEY_FORMAT	65
2.35	PRIVATE_KEY_PWD	66
2.36	REQUEST_FILE	67
2.37	REQUEST_FORMAT	68
2.38	REVOKE	69
2.39	REVOKE_REASON	70
2.40	SERIAL_NUMBER	71
2.41	STATE	72
2.42	TRANSPORT_FILE	73

2.43	TRANSPORT_FILE_PWD	74
2.44	VERIFY	75
2.45	VERSION	76
Chapter 3 Universal Control Manager Configuration Options		77
3.1	Overview	77
3.2	Configuration Options List	78
3.3	ACTIVITY_MONITORING	80
3.4	BIF_DIRECTORY	81
3.5	CA_CERTIFICATES	82
3.6	CERTIFICATE	83
3.7	CERTIFICATE_REVOCATION_LIST	84
3.8	CODE_PAGE	85
3.9	COMMAND_FILE_ENCRYPTED	86
3.10	COMMAND_FILE_PLAIN	87
3.11	COMMAND_ID	88
3.12	CTL_SSL_CIPHER_LIST	89
3.13	ENCRYPTION_KEY	90
3.14	EVENT_GENERATION	91
3.15	HELP	93
3.16	HOSTNAME_RETRY_COUNT	94
3.17	INSTALLATION_DIRECTORY	95
3.18	MESSAGE_LANGUAGE	96
3.19	MESSAGE_LEVEL	97
3.20	NETWORK_DELAY	99
3.21	NLS_DIRECTORY	100
3.22	OUTBOUND_IP	101
3.23	PLF_DIRECTORY	102
3.24	PRIVATE_KEY	103
3.25	PRIVATE_KEY_PWD	104
3.26	REFRESH_CMD	105
3.27	REMOTE_HOST	107
3.28	REMOTE_PORT	108
3.29	SSL_IMPLEMENTATION	109

3.30	START_CMD	110
3.31	STOP_CMD	111
3.32	SYSTEM_ID	112
3.33	USER_ID	113
3.34	USER_PASSWORD	114
3.35	VERIFY_HOST_NAME	115
3.36	VERIFY_SERIAL_NUMBER	117
3.37	VERSION	118
Chapter 4 Universal Control Server Configuration Options		119
4.1	Overview	119
4.2	Configuration Options List	120
4.3	ACTIVITY_MONITORING	121
4.4	CODE_PAGE	122
4.5	EVENT_GENERATION	123
4.6	INSTALLATION_DIRECTORY	125
4.7	LOGON_METHOD	126
4.8	MESSAGE_LEVEL	127
4.9	NLS_DIRECTORY	130
4.10	TMP_DIRECTORY	131
4.11	TRACE_DIRECTORY	132
4.12	USER_SECURITY	133
Chapter 5 Universal Control Component Definition Options		135
5.1	Overview	135
5.2	Component Definition Options Information	136
5.3	Component Definition Options	138
5.4	AUTOMATICALLY_START	139
5.5	COMPONENT_NAME	140
5.6	CONFIGURATION_FILE	141
5.7	RUNNING_MAXIMUM	142
5.8	START_COMMAND	143
5.9	WORKING_DIRECTORY	144

Chapter 6 Universal Control UACL Entries	145
6.1 Overview	145
6.2 UACL Entries Information	146
6.3 UACL Entries List	148
6.4 UCTL_ACCESS	149
6.5 UCTL_REQUEST	151
Chapter 7 Universal Copy	153
7.1 Overview	153
7.2 Configuration Options List	154
7.3 CPY_MODE	155
7.4 FRMFILE	156
7.5 FRMFILES	157
7.6 FRMMBR	158
7.7 HELP	159
7.8 INPUTMODE	160
7.9 MESSAGE_LEVEL	161
7.10 MODE	162
7.11 OUTPUT	163
7.12 OUTPUTMODE	164
7.13 REPLACE	165
7.14 TOFILE	166
7.15 TOMBR	167
7.16 TRANSACTIONAL	168
7.17 VERSION	169
Chapter 8 Universal Database Dump	170
8.1 Overview	170
8.2 Configuration Options List	171
8.3 DATABASE_FILE	172
8.4 DUMP_OPTIONS	173
Chapter 9 Universal Database Load	174
9.1 Overview	174
9.2 Configuration Options List	175

9.3 DATABASE_FILE	176
9.4 OVERWRITE	177
Chapter 10 Universal Display Log File	178
10.1 Overview	178
10.2 Configuration Options List	179
10.3 PRIMARY_FILE	180
10.4 PRIMARY_MEMBER	181
10.5 REMOVE_MEMBERS	182
10.6 SECONDARY_FILE	183
10.7 SECONDARY_MEMBER	184
Chapter 11 Universal Encrypt.....	185
11.1 Overview	185
11.2 Configuration Options List	186
11.3 AES	187
11.4 CODE_PAGE	188
11.5 ENCRYPTION_KEY	189
11.6 HELP	190
11.7 INPUT_FILE	191
11.8 NLS_DIRECTORY	192
11.9 OUTPUT_FILE	193
11.10 VERSION	194
Chapter 12 Universal Event Log Dump	195
12.1 Overview	195
12.2 Configuration Options List	196
12.3 BACKUP_LOG	197
12.4 CLEAR_LOG	198
12.5 END_TIME	199
12.6 HELP	200
12.7 INSTALLATION_DIRECTORY	201
12.8 LOG_DIRECTORY	202
12.9 LOG_TYPE	203
12.10 MESSAGE_DESTINATION	204

12.11 MESSAGE_LANGUAGE	205
12.12 MESSAGE_LEVEL	206
12.13 NLS_DIRECTORY	207
12.14 OUTPUT_FILE	208
12.15 PAGE_HEADERS	209
12.16 PAGE_LENGTH	210
12.17 REMOTE_SERVER	211
12.18 START_TIME	212
12.19 VERSION	213
Chapter 13 Universal Message Translator	214
13.1 Overview	214
13.2 Configuration Options List	215
13.3 HELP	216
13.4 IGNORE_CASE	217
13.5 MESSAGE_FILE	218
13.6 MESSAGE_LEVEL	219
13.7 TRANSLATION_TABLE	220
13.8 VERSION	221
Chapter 14 Universal Products Install Merge	222
14.1 Overview	222
14.2 Configuration Options List	223
14.3 BACKUP_DESTINATION	224
14.4 COMMAND_FILE_ENCRYPTED	226
14.5 COMMAND_FILE_PLAIN	227
14.6 COMPONENT_TYPE	228
14.7 CONFIGURATION_TYPE	230
14.8 DESTINATION_FILE	232
14.9 ENCRYPTION_KEY	233
14.10 HELP	234
14.11 INSTALLATION_DIRECTORY	235
14.12 KEEP_NOMATCH	236
14.13 MESSAGE_LEVEL	238
14.14 SOURCE_FILE	239

14.15 VERSION	241
Chapter 15 Universal Query	242
15.1 Overview	242
15.2 Configuration Options List	243
15.3 BIF_DIRECTORY	244
15.4 CODE_PAGE	245
15.5 COMMAND_ID	246
15.6 COMPONENT_ID	247
15.7 HELP	248
15.8 MANAGERS	249
15.9 MESSAGE_LANGUAGE	250
15.10 MESSAGE_LEVEL	251
15.11 NLS_DIRECTORY	253
15.12 OUTBOUND_IP	254
15.13 PING	255
15.14 PLF_DIRECTORY	256
15.15 REMOTE_HOST	257
15.16 REMOTE_PORT	258
15.17 REPORT	259
15.18 SYSTEM_ID	260
15.19 VERSION	261
Chapter 16 Universal Spool List	262
16.1 Overview	262
16.2 Configuration Options List	263
16.3 BROKER_SPOOL_DIR	264
16.4 COMPONENT	265
16.5 HELP	266
16.6 ID	267
16.7 LIST	268
16.8 MESSAGE_LEVEL	270
16.9 MOUNT_POINT	271
16.10 UCMD_SPOOL_DIR	272
16.11 VERSION	273

Chapter 17 Universal Spool Remove	274
17.1 Overview	274
17.2 Configuration Options List	275
17.3 BROKER_SPOOL_DIR	276
17.4 COMPONENT	277
17.5 HELP	278
17.6 MESSAGE_LEVEL	279
17.7 MOUNT_POINT	280
17.8 UCMD_SPOOL_DIR	281
17.9 UEM_SERIALNO	282
17.10 VERSION	283
Chapter 18 Universal Submit Job	284
18.1 Overview	284
18.2 Configuration Options List	285
18.2.1 USBMJOB-Specific Configuration Options	285
18.2.2 SBMJOB Encapsulated Configuration Options	286
18.2.3 Universal Command Server Options Affecting USBMJOB	287
18.3 Universal Submit Job (USBMJOB) Specific Options	288
18.3.1 COMMAND	288
18.3.2 COPY_SPOOL_FILES	288
18.3.3 ENCRYPTED_COMMAND_FILE	288
18.3.4 JOB_LOG_LIBRARY	289
18.3.5 JOB_STATUS_POLLING_INTERVAL	289
18.3.6 KEY	289
18.3.7 REMOTE_MESSAGE_PREFIX	290
18.3.8 REMOTE_REFRESH_INTERVAL	290
18.3.9 REMOTE_REPLY_COMMAND_PATH	290
18.3.10 REMOTE_REPLY_HOST	291
18.3.11 REMOTE_REPLY_PORT	291
18.3.12 REMOTE_REPLY_USER_ID	291
18.3.13 REMOTE_REPLY_USER_PWD	292
18.3.14 SPECIFY_PRINT_CONTROL_CHARS	292
18.3.15 TRACE	292
18.3.16 USE_REMOTE_REPLY_FACILITY	293
Chapter 19 Universal Write-to-Operator	294
19.1 Overview	294
19.2 Configuration Options List	295

19.3	CONSOLE_ID	296
19.4	CONSOLE_NAME	297
19.5	HELP	298
19.6	KEY	299
19.7	MESSAGE	300
19.8	MESSAGE_LEVEL	301
19.9	REPLY	302
19.10	TIMEOUT	303
19.11	VERSION	304
Chapter 20	Additional Information	305
20.1	Overview	305
20.2	DD Statements Used in JCL	306
20.3	SSL Cipher Suites	307
20.4	Character Code Pages	308
20.5	UTT Files	310
Appendix A	Customer Support	311

List of Tables

Chapter 2 Universal Certificate	29
Table 2.1 Universal Certificate Configuration Options	31
Table 2.2 Country Codes (Subset of ISO 3166)	43
Chapter 3 Universal Control Manager Configuration Options	77
Table 3.1 Universal Control Manager Configuration Options	79
Table 3.2 SSL Cipher Suites (for CTL_SSL_CIPHER_LIST)	89
Chapter 4 Universal Control Server Configuration Options	119
Table 4.1 Universal Control Server Configuration Options	120
Chapter 5 Universal Control Component Definition Options	135
Table 5.1 Universal Control Component Definition Options	138
Chapter 6 Universal Control UACL Entries	145
Table 6.1 Universal Control UACL Entries	148
Chapter 7 Universal Copy	153
Table 7.1 Universal Copy Configuration Options	154
Chapter 8 Universal Database Dump	170
Table 8.1 Universal Database Dump Configuration Options	171
Chapter 9 Universal Database Load	174
Table 9.1 Universal Database Load Configuration Options	175
Chapter 10 Universal Display Log File	178
Table 10.1 Universal Display Log File - Configuration Options	179

Chapter 11 Universal Encrypt	185
Table 11.1 Universal Encrypt Configuration Options	186
Chapter 12 Universal Event Log Dump	195
Table 12.1 Universal Event Log Dump - Configuration Options	196
Chapter 13 Universal Message Translator	214
Table 13.1 Universal Message Translator Configuration Options	215
Chapter 14 Universal Products Install Merge	222
Table 14.1 Universal Products Install Merge Configuration Options	223
Chapter 15 Universal Query	242
Table 15.1 Universal Query Configuration Options	243
Chapter 16 Universal Spool List	262
Table 16.1 Universal Spool List Configuration Options	263
Chapter 17 Universal Spool Remove	274
Table 17.1 Universal Spool Remove Configuration Options	275
Chapter 18 Universal Submit Job	284
Table 18.1 Universal Submit Job Configuration Options - USBMJOB-Specific	285
Table 18.2 Universal Submit Job - SBMJOB Encapsulated Configuration Options	286
Chapter 19 Universal Write-to-Operator	294
Table 19.1 Universal WTO Configuration Options	295
Chapter 20 Additional Information	305
Table 20.1 Universal Products Utilities - Additional Information	305
Table 20.2 Universal Control Manager for z/OS - Batch JCL DD Statements	306
Table 20.3 SSL Cipher Suites	307
Table 20.4 Character Code Pages	309
Table 20.5 UTT File Locations	310

Preface

Document Structure

This document is written using specific conventions for text formatting and according to a specific document structure in order to make it as useful as possible for the largest audience. The following sections describe the document formatting conventions and organization.

Cross-Reference Links

This document contains cross-reference links to information in its companion document, the Universal Products Utilities User Guide.

In order for the links to work correctly:

- Place the documents in the same folder.
- In Adobe Reader / Adobe Acrobat, de-select **Open cross-document link in same window** in the **General** category of your **Preferences** dialog (selected from the **Edit** menu).

Conventions

Specific text formatting conventions are used within this document to represent different information. The following conventions are used.

Typeface and Fonts

This document provides tables that identify how information is used. These tables identify values and/or rules that are either pre-defined or user-defined:

- *Italics* denotes user-supplied information.
- **Boldface** indicates pre-defined information.

Elsewhere in this document, **This Font** identifies specific names of different types of information, such as file names or directories (for example, `\abc\123\help.txt`).

Operating System-Specific Text

Most of this document describes the product in the context of all supported operating systems. At times, it is necessary to refer to operating system-specific information. This information is introduced with a special header, which is followed by the operating system-specific text in a different font size from the normal text.

z/OS

This text pertains specifically to the z/OS line of operating systems.

This text resumes the information pertaining to all operating systems.

Tips from the Stoneman



Look to the Stoneman for suggestions
or for any other information
that requires special attention.

Stoneman's Tip

Vendor References

References are made throughout this document to a variety of vendor operating systems. We attempt to use the most current product names when referencing vendor software.

The following names are used within this document:

- **z/OS** is synonymous with IBM z/OS and IBM OS/390 line of operating systems.
- **Windows** is synonymous with Microsoft's Windows 2000 / 2003 / 2008, Windows XP, Windows Vista, and Windows 7 lines of operating systems. Any differences between the different systems will be noted.
- **UNIX** is synonymous with operating systems based on AT&T and BSD origins and the Linux operating system.
- **OS/400** is synonymous with IBM OS/400, IBM i/5, and IBM i operating systems.
- **AS/400** is synonymous for IBM AS/400, IBM iSeries, and IBM System i systems.

Note: These names do not imply software support in any manner. For a detailed list of supported operating systems, see the Universal Products 4.1.0 Installation Guide.

Document Organization

The document is organized into the following chapters:

- [Overview](#) (Chapter 1)
Introduction to the reference information in this document.
- [Universal Certificate](#) (Chapter 2)
Detailed information on all Universal Certificate configuration options for all operating systems.
- [Universal Control Manager Configuration Options](#) (Chapter 3)
Detailed information on all Universal Control Manager configuration options for all operating systems.
- [Universal Control Server Configuration Options](#) (Chapter 4)
Detailed information on all Universal Control Server configuration options for all operating systems.
- [Universal Control Component Definition Options](#) (Chapter 5)
Detailed information on all Universal Control component definition options.
- [Universal Control UACL Entries](#) (Chapter 6)
Detailed information on all Universal Access Control List (UACL) entries.
- [Universal Copy](#) (Chapter 7)
Detailed information on all Universal Copy configuration options for all operating systems.
- [Universal Database Dump](#) (Chapter 8)
Detailed information on all Universal Database Dump configuration options for all operating systems.
- [Universal Database Load](#) (Chapter 9)
Detailed information on all Universal Database Load configuration options for all operating systems.
- [Universal Display Log File](#) (Chapter 10)
Detailed information on all Universal Display Log File configuration options for all operating systems.
- [Universal Encrypt](#) (Chapter 11)
Detailed information on all Universal Encrypt configuration options for all operating systems.
- [Universal Event Log Dump](#) (Chapter 12)
Detailed information on all Universal Event Log Dump configuration options for all operating systems.
- [Universal Message Translator](#) (Chapter 13)
Detailed information on all Universal Message Translator configuration options for all operating systems.
- [Universal Products Install Merge](#) (Chapter 14)
Detailed information on all Universal Products Install Merge configuration options for all operating systems.
- [Universal Query](#) (Chapter 15)
Detailed information on all Universal Query configuration options for all operating systems.

- [Universal Spool List](#) (Chapter 16)
Detailed information on all Universal Spool configuration options for all operating systems.
- [Universal Spool Remove](#) (Chapter 17)
Detailed information on all Universal Spool Remove configuration options for all operating systems.
- [Universal Submit Job](#) (Chapter 18)
Detailed information on all Universal Submit Job configuration options for all operating systems.
- [Universal Write-to-Operator](#) (Chapter 19)
Detailed information on all Universal Write-to-Operator configuration options for all operating systems.
- [Additional Information](#) (Chapter 20)
Additional information related to Universal Products Utilities.
- [Customer Support](#) (Appendix A)
Customer support contact information for Universal Products Utilities.

Chapter 1 Overview

The Universal Products Utilities 4.1.0 Reference Guide is a companion document to the Universal Products Utilities 4.1.0 User Guide.

It provides technical detail for the configuration options identified in that document.

The options are listed alphabetically, without regard to any specific operating system.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented on each option.

1.1 Configuration Options Information

For each configuration option, this chapter provides the following information.

Description

Describes the configuration option and how it is used.

Usage

Provides a table of one or more of the following types of information:

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<Format / Value>					
Command Line Option, Long Form	<Format / Value>					
Environment Variable	<Format / Value>					
Configuration File Keyword	<Format / Value>					
<OS/400> Parameter	<Format / Value>					

Method

Identifies the different methods used to specify Universal Products Utilities configuration options:

- Command Line Option, Short Form
- Command Line Option, Long Form
- Environment Variable
- Configuration File Keyword
- <OS/400> Parameter

Note: Each option can be specified using one or more methods.

Syntax

Identifies the syntax of each method that can be used to specify the option:

- **Format** Specific characters that identify the option.
- **Value** Type of value(s) to be supplied for this method.

Note: If a Method is not valid for specifying the option, the Syntax field contains n/a.

(Operating System)

Identifies (with a ✓) the operating systems for which each method of specifying the option is valid:

- OS/400
- HP NonStop
- UNIX
- Windows
- z/OS

Values

Identifies all possible values for the specified value type.

Defaults are identified in **[bracketed bold type]**.

<Additional Information>

Identifies any additional information specific to the option.

Chapter 2

Universal Certificate

2.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Certificate.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

2.2 Configuration Options List

Table 2.1, below, identifies the Universal Certificate configuration options.

Option Name	Description	Page
CA	Specification for whether or not the certificate should be marked as a Certificate Authority certificate	32
CA_CERT_FILE	File from which the CA certificate is read or to which the CA certificate is written	33
CA_CERT_FORMAT	Format of the CA certificate file specified by CA_CERT_FILE	34
CERT_DB	Certificate database name	35
CERT_FILE	File from which the certificate is read or to which the certificate is written	36
CERT_FORMAT	Format of the certificate file specified by CERT_FILE	37
CODE_PAGE	Character code page used to translate text data	38
COMMAND_FILE_ENCRYPTED	Name of an encrypted command file	39
COMMAND_FILE_PLAIN	Name of a plain text command file	40
COMMON_NAME	Common name of the subject field of a certificate	41
COUNTRY	Country name of the subject field of a certificate	42
CREATE	Specification for Universal Certificate to create a certificate, certificate request, certificate revocation list (CRL), or transport file	44
CRL_FILE	File name to which the Certificate Revocation List (CRL) is written	45
CRL_FORMAT	Format of the CRL file specified by CRL_FILE	46
DNS_NAME	Domain Name System (DNS) name of the computer system for which the certificate identifies	47
EMAIL_ADDRESS	Email address of the entity identified by the certificate	48
ENCRYPTION_KEY	Key used to encrypt the command file	49
HELP	Writes a description of the command options and their format	50
IP_ADDRESS	Internet Protocol (IP) address of the computer system for which the certificate identifies	51
KEY_SIZE	Key size of the RSA public / private keys	52
LOCALITY	Locality name of the subject field of a certificate	53
MESSAGE_LEVEL	Level of messages to write	54
NEXT_UPDATE_DAYS	Number of days to the next CRL update	56
NEXT_UPDATE_HOURS	Number of hours to the next CRL update	57
NLS_DIRECTORY	Directory name where the code page UTT files are located	58
NOT_AFTER_DATE	Last day for which the certificate is considered valid	59
NOT_BEFORE_DATE	First day for which the certificate is considered valid	60
ORGANIZATION	Organization name of the subject field of a certificate	61

Option Name	Description	Page
ORGANIZATIONAL_UNIT	Organizational unit name of the subject field of a certificate	62
PRINT	Specification that Universal Certificate is to print a certificate or certificate request	63
PRIVATE_KEY_FILE	File name from which the RSA private key is read or to which the RSA private key is written	64
PRIVATE_KEY_FORMAT	Format of the private key file specified by PRIVATE_KEY_FILE	65
PRIVATE_KEY_PWD	Password used to read and write the private key file specified by PRIVATE_KEY_FILE	66
REQUEST_FILE	File from which the certificate request is read or to which the certificate request is written	67
REQUEST_FORMAT	Format of the certificate request file specified by REQUEST_FILE	68
REVOKE	Specification that Universal Certificate is to revoke a certificate	69
REVOKE_REASON	Reason that a certificate is being revoked	70
SERIAL_NUMBER	Unique serial number to be assigned to the created certificate	71
STATE	State name of the subject field of a certificate	72
TRANSPORT_FILE	File containing certificate / private key information	73
TRANSPORT_FILE_PWD	Password used to protect the file specified by TRANSPORT_FILE	74
VERIFY	Specification that Universal Certificate is to verify a certificate	75
VERSION	Writes the program version and copyright information	76

Table 2.1 Universal Certificate Configuration Options

2.3 CA

Description

The CA option specifies whether or not the certificate should be marked as a Certificate Authority certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-ca <i>option</i>			✓	✓	✓
Environment Variable	UCRCA= <i>option</i>			✓	✓	✓

Values

option is the specification for whether or not the certificate is a CA certificate.

Valid values for *option* are:

- **yes**
Certificate is marked as a CA certificate. This is accomplished by setting the certificate `basicConstraint` extension `cA` to `true`.
- **no**
Certificate is not marked as a CA certificate.

[Default is no.]

2.4 CA_CERT_FILE

Description

The CA_CERT_FILE option specifies either:

- Name of the file from which the CA certificate is read.
- Name of the file to which the CA certificate is written.

(The CA certificate identifies the issuer of the certificate being created.)

The format of the file is specified by the [CA_CERT_FORMAT](#) option.

CA certificate information also can be imported from a transport file (specified via the [TRANSPORT_FILE](#) option). In this case, CA_CERT_FILE specifies the name of the file to which one or more CA certificates are written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-a <i>ddname</i> or <i>file</i>			✓	✓	✓
Command Line Option, Long Form	-ca_cert_file <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRCACERTFILE= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the file.

2.5 CA_CERT_FORMAT

Description

The CA_CERT_FORMAT option specifies the format of the CA certificate file specified by the [CA_CERT_FILE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-A <i>format</i>			✓	✓	✓
Command Line Option, Long Form	-ca_cert_format <i>format</i>			✓	✓	✓
Environment Variable	UCRCACERTFORMAT= <i>format</i>			✓	✓	✓

Values

format is the format of the CA certificate file.

Valid values for *format* are:

- **pem**
PEM-formatted file
- **der**
A DER-formatted file

[Default is pem.]

See Section [3.2.7 File Formats](#) in [Chapter 3 Universal Certificate](#) of the Universal Products Utilities 4.1.0 User Guide for details on file formats.

2.6 CERT_DB

Description

The CERT_DB option specifies the name of the certificate database.

UNIX and Windows

If the name of the certificate database is not specified, the certificate database is created in the current working directory with name **ucert.db**.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-cert_db <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRCERTDB= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the certificate database.

2.7 CERT_FILE

Description

The CERT_FILE option specifies either:

- Name of the file to which the certificate is written
- Name of the file from which the certificate is read

The format of the file is specified by the [CERT_FORMAT](#) option.

Certificate information also can be imported from a transport file (specified via the [TRANSPORT_FILE](#) option). In this case, CERT_FILE specifies the file name to which the certificate is written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-d <i>ddname</i> or <i>file</i>			✓	✓	✓
Command Line Option, Long Form	-cert_file <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRCERTFILE= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the file.

2.8 CERT_FORMAT

Description

The CERT_FORMAT option specifies the format of the certificate file specified by the [CERT_FILE](#) option.

Usage

Method	Method	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-D <i>format</i>			✓	✓	✓
Command Line Option, Long Form	-cert_format <i>format</i>			✓	✓	✓
Environment Variable	UCRCERTFORMAT= <i>format</i>			✓	✓	✓

Values

format is the format of the certificate file.

Valid values for *format* are:

- **pem**
PEM-formatted file
- **der**
A DER-formatted file

[Default is pem.]

See Section [3.2.7 File Formats](#) in [Chapter 3 Universal Certificate](#) of the Universal Products Utilities 4.1.0 User Guide for details on file formats.

2.9 CODE_PAGE

Description

The CODE_PAGE option specifies the character code page used to translate text data.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>codepage</i>			✓	✓	✓
Command Line Option, Long Form	-codepage <i>codepage</i>			✓	✓	✓
Environment Variable	UCRCODEPAGE= <i>codepage</i>			✓	✓	✓

Values

codepage is the character code page that is used to translate data.

codepage references a Universal Translate Table (UTT) file provided with the product (see Section [20.5 UTT Files](#) for information on UTT files). UTT files are used to translate between Unicode and the local single-byte code page. (All UTT files end with an extension of `.utt`.)

[Default

The default code page is different for different operating systems:

- **ISO8859-1 (8-bit ASCII) ASCII-based operating systems**
- **IBM1047 (EBCDIC) EBCDIC-based operating system]**

See Section [20.4 Character Code Pages](#) for a complete list of character code pages provided by Stonebranch Inc. for use with Universal Products.

2.10 COMMAND_FILE_ENCRYPTED

Description

The `COMMAND_FILE_ENCRYPTED` option specifies the name of an encrypted command file.

Command files specify an additional source of command line options. The options read from the file are processed exactly like options specified on the command line. Encrypted command files are an excellent place to store sensitive data such as passwords.

Universal Certificate is able to process command files that are either encrypted or plain text (see [COMMAND_FILE_PLAIN](#)). Use the Universal Encrypt utility to encrypt a plain text command file (see [Chapter 9 Universal Encrypt](#) in the Universal Products Utilities 4.1.0 User Guide).

Command files (encrypted or not) that contain sensitive data should be protected from unauthorized read access with file level security.

Note: If an encrypted file is specified in this option, a plain text file should not be specified additionally in the [COMMAND_FILE_PLAIN](#) option. If it is, the file specified in [COMMAND_FILE_PLAIN](#) will be used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-x <i>ddname</i> or <i>filename</i>			✓	✓	✓
Command Line Option, Long Form	-encryptedfile <i>ddname</i> or <i>filename</i>			✓	✓	✓
Environment Variable	n/a					

Values

ddname or *filename* is the name of the encrypted command file.

2.11 COMMAND_FILE_PLAIN

Description

The `COMMAND_FILE_PLAIN` option specifies the name of a plain text command file.

Command files specify an additional source of command line options. The options read from the file are processed exactly like options specified on the command line.

Universal Certificate is able to process command files that are either encrypted or plain text (see [COMMAND_FILE_ENCRYPTED](#)). Command files (encrypted or not) that contain sensitive data should be protected from unauthorized read access with file level security.

Note: If an encrypted file is specified in this option, a plain text file should not be specified additionally in the [COMMAND_FILE_ENCRYPTED](#) option. If it is, the file specified in this `COMMAND_FILE_PLAIN` option will be used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-f <i>ddname</i> or <i>filename</i>			✓	✓	✓
Command Line Option, Long Form	-file <i>ddname</i> or <i>filename</i>			✓	✓	✓
Environment Variable	n/a					

Values

ddname or *filename* is the name of the plain text command file.

2.12 COMMON_NAME

Description

The COMMON_NAME option specifies the common name of the **subject** field of a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-common_name <i>name</i>			✓	✓	✓
Environment Variable	UCRCOMMONNAME= <i>name</i>			✓	✓	✓

Values

name is the common name of the **subject** field.

More specifically, *name* is the certificate's **commonName** (CN) relative distinguished name (RDN) attribute of the **subject** distinguished name (DN).

2.13 COUNTRY

Description

The COUNTRY option specifies the country name of the **subject** field of a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-country <i>name</i>			✓	✓	✓
Environment Variable	UCRCOUNTRY= <i>name</i>			✓	✓	✓

Values

name is the country name of the **subject** field.

More specifically, *name* is the certificate's **countryName** (C) relative distinguished name (RDN) attribute of the **subject** distinguished name (DN); a two-character country code as defined by the ISO 3166 standard.

Country Codes

Table 2.2, below, identifies a subset of ISO 3166 country codes.

Code	Country
AU	Australia
BE	Belgium
CA	Canada
DK	Denmark
FR	France
DE	Germany
IT	Italy
NL	Netherlands
NO	Norway
PT	Portugal
ES	Spain
SE	Sweden
CH	Switzerland
GB	United Kingdom
US	United States

Table 2.2 Country Codes (Subset of ISO 3166)

2.14 CREATE

Description

The CREATE option specifies that Universal Certificate is to create a certificate, certificate request, certificate revocation list (CRL), or a transport file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-c <i>object</i></code>			✓	✓	✓
Command Line Option, Long Form	<code>-create <i>object</i></code>			✓	✓	✓
Environment Variable	<code>UCRCREATE=<i>object</i></code>			✓	✓	✓

Values

object is the specification for what Universal Certificate is to create.

Valid values for *object* are:

- **cert**
Create an X.509 certificate.
- **request**
Create a certificate request.
- **crl**
Create a Certificate Revocation List (CRL).
- **transport**
Create a PKCS#12-encoded transport file.

2.15 CRL_FILE

Description

The CRL_FILE option specifies the name of the file to which the Certificate Revocation List (CRL) is written.

The format of the file is specified by the [CRL_FORMAT](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-crl_file <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRCRLFILE= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the file to which the Certificate Revocation List (CRL) is written.

2.16 CRL_FORMAT

Description

The CRL_FORMAT option specifies the format of the CRL file specified by the [CRL_FILE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-crl_format <i>format</i>			✓	✓	✓
Environment Variable	UCRCRLFORMAT= <i>format</i>			✓	✓	✓

Values

format is the format of the CRL file.

Valid values for *format* are:

- **pem**
PEM-formatted file
- **der**
A DER-formatted file

[Default is pem.]

See Section [3.2.7 File Formats](#) in [Chapter 3 Universal Certificate](#) of the Universal Products Utilities 4.1.0 User Guide for details on file formats.

2.17 DNS_NAME

Description

The DNS_NAME option specifies the Domain Name System (DNS) name of the computer system identified by the certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-dns_name <i>name</i>			✓	✓	✓
Environment Variable	UCRDNSNAME= <i>name</i>			✓	✓	✓

Values

name is the Domain Name System (DNS) name of the computer system.

More specifically, *name* is the **dNSName** component of the **subjectAltName** extension.

Note: An IP address (for example, **10.20.30.40**) should not be used. IP address values are specified with the [IP_ADDRESS](#) option.

2.18 EMAIL_ADDRESS

Description

The EMAIL_ADDRESS option specifies the e-mail address of the entity identified by the certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-email_address <i>name</i>			✓	✓	✓
Environment Variable	UCREMAILADDRESS= <i>name</i>			✓	✓	✓

Values

name is the e-mail address of the entity.

More specifically, *name* is the **rfc822Name** component of the **subjectAltName** extension.

The format of *name* is defined by RFC 822. The name is of the form local-part@domain.

2.19 ENCRYPTION_KEY

Description

The ENCRYPTION_KEY option specifies key used to encrypt the command file (see [COMMAND_FILE_ENCRYPTED](#)).

This key acts much like a password for the encrypted command file in that can be used to protect the file from decryption by unauthorized users. If a key was used to encrypt a command file (when Universal Encrypt was run), that same key must be specified to decrypt the file; otherwise, the decryption will fail.

If no key is specified, the default value is used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-K <i>key</i>			✓	✓	✓
Command Line Option, Long Form	-key <i>key</i>			✓	✓	✓
Environment Variable	UCRKEY= <i>key</i>			✓	✓	✓

Values

key is the key used to encrypt the command file.

2.20 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h			✓	✓	✓
Command Line Option, Long Form	-help			✓	✓	✓
Environment Variable	n/a					

Values

(There are no values used with this option.)

2.21 IP_ADDRESS

Description

The IP_ADDRESS option specifies the Internet Protocol (IP) address of the computer system identified by the certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-ip_address <i>name</i>			✓	✓	✓
Environment Variable	UCRIPADDRESS= <i>name</i>			✓	✓	✓

Values

name is the Internet Protocol (IP) address of the computer system.

More specifically, *name* is the **iPAddress** component of the **subjectAltName** extension.

Note: DNS names (for example, **sysa.acme.com**) should not be used. DNS name values are specified with the [DNS_NAME](#) option.

2.22 KEY_SIZE

Description

The KEY_SIZE option specifies the key size of the RSA public / private keys.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-s <i>size</i>			✓	✓	✓
Command Line Option, Long Form	-key_size <i>size</i>			✓	✓	✓
Environment Variable	UCRKEYSIZE= <i>size</i>			✓	✓	✓

Values

size is the key size (number of bits) of the RSA public/private keys.

Valid values for *size* are:

- **512**
- **1024**
- **2048**

[Default is 1024.]

2.23 LOCALITY

Description

The LOCALITY option specifies the locality name of the **subject** field of a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-locality <i>name</i>			✓	✓	✓
Environment Variable	UCRLOCALITY= <i>name</i>			✓	✓	✓

Values

name is the locality name of the **subject** field.

More specifically, *name* is the certificate's `localityName` (L) relative distinguished name (RDN) attribute of the **subject** distinguished name (DN).

2.24 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>			✓	✓	✓
Command Line Option, Long Form	-level <i>level</i>			✓	✓	✓
Environment Variable	UCRLEVEL= <i>level</i>			✓	✓	✓

Values

level indicates either of the following level of messages:

- **trace**
Writes trace messages used for diagnostic purposes (see Section [Trace Files](#)).
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

z/OS

[Default is info.]

UNIX and Windows

[Default is warn.]

Trace Files

UNIX and Windows

Trace file name is **ucert . trc**. It is created in the current working directory.

z/OS

Trace file is written to ddname **UNVTRACE**.

2.25 NEXT_UPDATE_DAYS

Description

The NEXT_UPDATE_DAYS option specifies the number of days to the next CRL update.

The CRL `nextUpdate` value is set to the current date plus the number of days specified in this option.

CRL creation requires the use of either this option or [NEXT_UPDATE_HOURS](#).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-next_update_days <i>days</i>			✓	✓	✓
Environment Variable	UCRNEXTUPDATEDAYS= <i>days</i>			✓	✓	✓

Values

days is the number of days to the next CRL update.

[Default is 0.]

2.26 NEXT_UPDATE_HOURS

Description

The NEXT_UPDATE_HOURS option specifies the number of hours to the next CRL update.

The CRL `nextUpdate` value is set to the current date plus the number of hours specified in this option.

CRL creation requires the use of either this option or [NEXT_UPDATE_DAYS](#).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-next_update_hours <i>hours</i>			✓	✓	✓
Environment Variable	UCRNEXTUPDATEHOURS= <i>hours</i>			✓	✓	✓

Values

hours is the number of hours to the next CRL update.

[Default is 0.]

2.27 NLS_DIRECTORY

Description

The NLS_DIRECTORY option specifies the name of the directory where the code page UTT files are located.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-nls_directory <i>directory</i>			✓	✓	
Environment Variable	UCRNLSDIRECTORY= <i>directory</i>			✓	✓	

Values

directory is the name of the directory.

UNIX and Windows

Relative path names are relative to the current working directory.

2.28 NOT_AFTER_DATE

Description

The NOT_AFTER_DATE option specifies the last day for which the certificate is considered valid.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-not_after_date <i>date</i>			✓	✓	✓
Environment Variable	UCRNOTAFTERDATE= <i>date</i>			✓	✓	✓

Values

date is the last day for which the certificate is considered valid.

The format of date is either:

- *YYYY.MM.DD*
- *DAYS* (number of days after the current date)

[Default is 365.]

2.29 NOT_BEFORE_DATE

Description

The NOT_BEFORE_DATE option specifies the first day for which the certificate is considered valid.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-not_before_date <i>date</i>			✓	✓	✓
Environment Variable	UCRNOTBEFOREDATE= <i>date</i>			✓	✓	✓

Values

date is the first day for which the certificate is considered valid.

The format of *date* is *YYYY.MM.DD*.

[Default is the current date.]

2.30 ORGANIZATION

Description

The ORGANIZATION option specifies the organization name of the **subject** field of a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-organization <i>name</i>			✓	✓	✓
Environment Variable	UCRORGANIZATION= <i>name</i>			✓	✓	✓

Values

name is the organization name of the subject field of a certificate.

More specifically, *name* is the certificate's **organizationName** (O) relative distinguished name (RDN) attribute of the **subject** distinguished name (DN).

2.31 ORGANIZATIONAL_UNIT

Description

The ORGANIZATIONAL_UNIT option specifies the organizational unit name of the **subject** field of a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-organizational_unit <i>name</i>			✓	✓	✓
Environment Variable	UCRORGANIZATIONALUNIT= <i>name</i>			✓	✓	✓

Values

name is the organizational unit name of the **subject** field of a certificate.

More specifically, *name* is the certificate's `organizationalUnitName` (OU) relative distinguished name (RDN) attribute of the **subject** distinguished name (DN).

2.32 PRINT

Description

The PRINT option specifies that Universal Certificate is to print a certificate, certificate request, certificate revocation list (CRL), or transport file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-p <i>object</i>			✓	✓	✓
Command Line Option, Long Form	-print <i>object</i>			✓	✓	✓
Environment Variable	UCRPRINT= <i>object</i>			✓	✓	✓

Values

object is the specification for what to print.

Valid values for *object* are:

- **cert**
Print an X.509 certificate.
- **request**
Print a certificate request.
- **crl**
Print a Certificate Revocation List (CRL).
- **transport**
Print a PKCS#12-encoded transport file.

2.33 PRIVATE_KEY_FILE

Description

The PRIVATE_KEY_FILE option specifies either:

- File from which the RSA private key is read
- File to which the RSA private key is written

The key is encoded in a password encrypted PKCS #8 syntax.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-e <i>ddname</i> or <i>file</i>			✓	✓	✓
Command Line Option, Long Form	-private_key_file <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRPRIVATEKEYFILE= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the file.

The format of the file is specified by the [PRIVATE_KEY_FORMAT](#) option.

2.34 PRIVATE_KEY_FORMAT

Description

The PRIVATE_KEY_FORMAT option specifies the format of the private key file specified by the [PRIVATE_KEY_FILE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-E <i>format</i>			✓	✓	✓
Command Line Option, Long Form	-private_key_format <i>format</i>			✓	✓	✓
Environment Variable	UCRPRIVATEKEYFORMAT= <i>format</i>			✓	✓	✓

Values

format is the format of the private key file.

Valid values for *format* are:

- **pem**
PEM-formatted file
- **der**
A DER-formatted file

[Default is pem.]

See Section [3.2.7 File Formats](#) in [Chapter 3 Universal Certificate](#) of the Universal Products Utilities 4.1.0 User Guide for details on file formats.

2.35 PRIVATE_KEY_PWD

Description

The PRIVATE_KEY_PWD option specifies the private key password that is used to read and write the private key file specified by the [PRIVATE_KEY_FILE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<i>-w password</i>			✓	✓	✓
Command Line Option, Long Form	<i>-private_key_pwd password</i>			✓	✓	✓
Environment Variable	<i>UCRPRIVATEKEYPWD=password</i>			✓	✓	✓

Values

password is the private key password.

2.36 REQUEST_FILE

Description

The REQUEST_FILE option specifies either:

- Name of the file from which the certificate request is read.
- Name of the file to which the certificate request is written.

The request is encoded in PKCS #10 syntax.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-r <i>ddname</i> or <i>file</i>			✓	✓	✓
Command Line Option, Long Form	-request_file <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRREQUESTFILE= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the file.

The format of the file is specified by the [REQUEST_FORMAT](#) option.

2.37 REQUEST_FORMAT

Description

The REQUEST_FORMAT option specifies the format of the certificate request file specified by the [REQUEST_FILE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-R <i>format</i>			✓	✓	✓
Command Line Option, Long Form	-request_format <i>format</i>			✓	✓	✓
Environment Variable	UCRREQUESTFORMAT= <i>format</i>			✓	✓	✓

Values

format is the format of the certificate request file.

Valid values for *format* are:

- **pem**
PEM-formatted file
- **der**
DER-formatted file

[Default is pem.]

See Section [3.2.7 File Formats](#) in [Chapter 3 Universal Certificate](#) of the Universal Products Utilities 4.1.0 User Guide for details on file formats.

2.38 REVOKE

Description

The REVOKE option specifies that Universal Certificate is to revoke a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-revoke <i>object</i>			✓	✓	✓
Environment Variable	UCRREVOKE= <i>object</i>			✓	✓	✓

Values

object is the specification to revoke a certificate.

Valid values for *object* are:

- **cert**
Instructs Universal Certificate to revoke an X.509 certificate.

2.39 REVOKE_REASON

Description

The REVOKE_REASON option specifies the reason that a certificate is being revoked. (Valid reasons for certificate revocation are defined as part of RFC 3280.)

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-revoke_reason <i>reason</i>			✓	✓	✓
Environment Variable	UCRREVOKEREASON= <i>reason</i>			✓	✓	✓

Values

reason is the reason a certificate is being revoked.

Universal Certificate accepts that following valid values for *reason*:

- **unspecified**
No reason is given.
- **keyCompromise**
Subject's private key, or some other aspect of the subject, has been compromised.
- **caCompromised**
CA private key, or some other aspect of the subject, has been compromised.
- **affiliationChange**
Subject's name or other information in the certificate has changed. There is no reason to suspect the private key is compromised.
- **superseded**
Certificate has been superseded by another certificate. There is no reason to suspect the private key is compromised.
- **cessationOfOperation**
Certificate is no longer required for the purpose it was issued. There is no reason to suspect the private key is compromised.
- **privilegeWithdrawn**
Privilege contained within the certificate is withdrawn.

[Default is *unspecified*.]

2.40 SERIAL_NUMBER

Description

The SERIAL_NUMBER option specifies a unique serial number to be assigned to the created certificate.

If SERIAL_NUMBER is not used to specify a serial number, Universal Certificate automatically generates a random 8-byte serial number for the certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-serial_number <i>number</i>			✓	✓	✓
Environment Variable	UCRSERIALNUMBER= <i>number</i>			✓	✓	✓

Values

number is the serial number to be assigned to the certificate.

2.41 STATE

Description

The STATE option specifies the state name of the **subject** field of a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-state <i>name</i>			✓	✓	✓
Environment Variable	UCRSTATE= <i>name</i>			✓	✓	✓

Values

name is the state name of the **subject** field.

More specifically, *name* is the certificate's **stateName** (S) relative distinguished name (RDN) attribute of the **subject** distinguished name (DN).

2.42 TRANSPORT_FILE

Description

The TRANSPORT_FILE option specifies either:

- Name of the file from which the certificate and private key is read
- Name of the file to which the certificate and private key is written

It is a DER-formatted file encoded in PKCS#12 syntax.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-transport_file <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCRTRANFILE= <i>ddname</i> or <i>file</i>			✓	✓	✓

Values

ddname or *file* is the name of the file.

2.43 TRANSPORT_FILE_PWD

Description

The TRANSPORT_FILE_PWD option specifies the password used to protect the transport file (specified by the [TRANSPORT_FILE](#) option).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-transport_file_pwd <i>password</i>			✓	✓	✓
Environment Variable	UCRTRANFILEPWD= <i>password</i>			✓	✓	✓

Values

password is the password used to protect the transport file.

2.44 VERIFY

Description

The VERIFY option specifies that Universal Certificate is to verify a certificate.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-verify <i>object</i>			✓	✓	✓
Environment Variable	UCRVERIFY= <i>object</i>			✓	✓	✓

Values

object is the specification to verify a certificate.

Valid values for *object* are:

- **cert**
Instructs Universal Certificate to verify an X.509 certificate.

2.45 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v			✓	✓	✓
Command Line Option, Long Form	-version			✓	✓	✓
Environment Variable	n/a					

Values

(There are no values used with this option.)

Chapter 3

Universal Control Manager

Configuration Options

3.1 Overview

This chapter provides detailed information on the configuration options available for use with the Universal Control Manager.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Control 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented on each option.

3.2 Configuration Options List

Table 3.1 identifies all Universal Control configuration options.

Option	Description	Page
ACTIVITY_MONITORING	Specification for whether or not product activity monitoring events are generated.	80
BIF_DIRECTORY	Broker Interface Directory that specifies the location of the Universal Broker interface file.	81
CA_CERTIFICATES	ddname of the PEM formatted trusted CA X.509 certificates.	82
CERTIFICATE	ddname of Manager's PEM-formatted X.509 certificate.	83
CERTIFICATE_REVOCAION_LIST	Location of Manager's PEM-formatted CRL.	84
CODE_PAGE	Code page used to translate text data to and from the network.	85
COMMAND_FILE_ENCRYPTED	Encrypted command file.	86
COMMAND_FILE_PLAIN	Plain text command file.	87
COMMAND_ID	Identity of a started component.	88
CTL_SSL_CIPHER_LIST	SSL cipher list for the control session.	89
ENCRYPTION_KEY	Encryption key used to decrypt an encrypted command file specified by option COMMAND_FILE_ENCRYPTED.	90
EVENT_GENERATION	Events to be generated as persistent events.	91
HELP	Write configuration option help.	93
HOSTNAME_RETRY_COUNT	Number of host connection attempts before ending with an error.	94
INSTALLATION_DIRECTORY	Directory in which the product is installed.	95
MESSAGE_LANGUAGE	Language of messages formatted.	96
MESSAGE_LEVEL	Level of messages written.	97
NETWORK_DELAY	Maximum number of seconds considered acceptable to wait for data communications.	99
NLS_DIRECTORY	NLS directory.	100
OUTBOUND_IP	Host or IP address to use for all outgoing IP connections.	101
PLF_DIRECTORY	Program Lock File directory that specifies the location of the Universal Control Manager program lock file.	102
PRIVATE_KEY	ddname of Manager's PEM-formatted RSA private key.	103
PRIVATE_KEY_PWD	Password for the Manager's PRIVATE_KEY.	104
REFRESH_CMD	Instruct a Broker or component to refresh its configuration.	105
REMOTE_HOST	TCP/IP host name of the remote computer on which Universal Broker is running and accepting connections.	107
REMOTE_PORT	TCP/IP port number of the remote computer on which Universal Broker is running and accepting connections.	108

Option	Description	Page
SSL_IMPLEMENTATION	Secure Socket Layer (SSL) implementation to be used for network communications.	109
START_CMD	Instruction to a Universal Broker to start a component.	110
STOP_CMD	Instruction to stop a component being executed by a Broker.	111
SYSTEM_ID	Local Universal Broker with which the Universal Control Manager must register.	113
USER_ID	User ID or account with which to execute the Control command.	113
USER_PASSWORD	Password associated with USER_ID.	114
VERIFY_HOST	Specification that the Broker's X.509 certificate host name field must be verified.	115
VERIFY_SERIAL_NUMBER	Specification that the Broker's X.509 certificate serial number field must be verified.	117
VERSION	Write program version.	118

Table 3.1 Universal Control Manager Configuration Options

3.3 ACTIVITY_MONITORING

Description

The ACTIVITY_MONITORING option specifies whether or not product activity monitoring events are generated.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Environment Variable	n/a					
Configuration File Keyword	activity_monitoring <i>option</i>	√		√	√	√
STRUCT Parameter	n/a					

Values

option is the specification for whether or not product activity monitoring events are generated.

Valid values for *option* are:

- **yes**
Activate product activity monitoring events
- **no**
Deactivate product activity monitoring events

[Default is yes.]

3.4 BIF_DIRECTORY

Description

The BIF_DIRECTORY option specifies the Broker Interface File (BIF) directory where the Universal Broker interface file, `ubroker.bif`, is located.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	<code>-bif_directory directory</code>			✓		
Environment Variable	<code>UCTLBIFDIRECTORY=directory</code>			✓		
Configuration File Keyword	n/a					
STRUCT Parameter	n/a					

Values

directory is the name of the BIF directory.

[Default is `/var/opt/universal1`.]

3.5 CA_CERTIFICATES

Description

The CA_CERTIFICATES option specifies the location of the PEM-formatted trusted Certificate Authority (CA) X.509 certificates file.

Trust CA certificates are required if Universal Broker certificate authentication and verification is desired.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-ca_certs <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCTLCACERTS= <i>file</i>	✓		✓	✓	
Configuration File Keyword	ca_certificates <i>ddname</i> or <i>file</i>	✓		✓	✓	✓
STRUCT Parameter	CACERTS (<i>file</i> [<i>lib</i>])	✓				

Values

z/OS

ddname is the ddname of the X.509 certificates. The value is used only when the [SSL_IMPLEMENTATION](#) option is set to *OPENSSL*.

Allocated to the ddname must be either a sequential data set or a member of a PDS that has a variable record format.

UNIX and Windows

file is the path name of the X.509 certificates file. Relative paths are relative the current working directory.

OS/400

file is the qualified file name of the X.509 certificates file.

The file name also can be qualified by a library name (*lib*). If it is not, the library list ***LIBL** is searched for the first occurrence of the file name.

3.6 CERTIFICATE

Description

The CERTIFICATE option specifies the file / ddname name of the PEM-formatted X.509 certificate that identifies the Universal Control Manager.

A Universal Control Manager X.509 certificate is required if the Universal Broker requires client authentication.

Note: If the CERTIFICATE option is used, the [PRIVATE_KEY](#) option also is required.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-cert <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCTLCACERT= <i>file</i>	✓		✓	✓	
Configuration File Keyword	certificate <i>ddname</i> or <i>file</i>	✓		✓	✓	✓
STRUCT Parameter	CERT(<i>file</i> [<i>lib</i>])	✓				

Values

z/OS

ddname is the ddname of the X.509 certificate. The value is used only when the [SSL_IMPLEMENTATION](#) option is set to *OPENSSL*.

Allocated to the ddname must be either a sequential data set or a member of a PDS that has a variable record format.

UNIX and Windows

file is the path name of the X.509 certificate file. Relative paths are relative to the current working directory.

OS/400

file is the qualified file name of the X.509 certificate file. The file name can be qualified by a library name.

3.7 CERTIFICATE_REVOCATION_LIST

Description

The CERTIFICATE_REVOCATION_LIST option specifies the file / ddname of the PEM-formatted file containing the Certificate Revocation List (CRL) issued by the trusted Certificate Authority.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-crl <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCTLCRL= <i>file</i>	✓		✓	✓	
Configuration File Keyword	crl <i>ddname</i> or <i>file</i>	✓		✓	✓	✓
STRUCT Parameter	CRLFILE(<i>file</i> [<i>lib</i>]) [CRLMBR(<i>member</i>)]	✓				

Values

z/OS

ddname is the ddname of the file containing the CRL.

The value is used only when the [SSL_IMPLEMENTATION](#) option is set to **openssl**.

UNIX and Windows

file is the path name of the file containing the CRL. Relative paths are relative to the current working directory.

OS/400

file is the qualified file name of the CRL file. The file name can be qualified by a library name. If not, the library list *LIBL is searched for the first occurrence of the file name.

3.8 CODE_PAGE

Description

The CODE_PAGE option specifies the character code page that is used to translate text data received and transmitted over the network.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>codepage</i>		✓	✓	✓	✓
Command Line Option, Long Form	-codepage <i>codepage</i>		✓	✓	✓	✓
Environment Variable	UCTLCODEPAGE= <i>codepage</i>	✓	✓	✓	✓	
Configuration File Keyword	codepage <i>codepage</i>	✓	✓	✓	✓	✓
STRUCT Parameter	CODEPAGE(<i>codepage</i>)	✓				

Value

codepage is the character code page that is used to translate data.

codepage references a Universal Translate Table (UTT) file provided with the product (see Section [20.5 UTT Files](#) for information on UTT files). UTT files are used to translate between Unicode and the local single-byte code page. (All UTT files end with an extension of `.utt`.)

[Default

The default code page is different for different operating systems:

- **ISO8859-1 (8-bit ASCII) ASCII-based operating systems**
- **IBM1047 (EBCDIC) EBCDIC-based operating system]**

See Section [20.4 Character Code Pages](#) for a complete list of character code pages provided by Stonebranch Inc. for use with Universal Products.

3.9 COMMAND_FILE_ENCRYPTED

Description

The `COMMAND_FILE_ENCRYPTED` option specifies the data set (for z/OS) or file containing encrypted values for command line option parameters.

Command files specify an additional source of command line options. Storing options in a file can be used in situations where it is not desirable to explicitly specify them on the command line. The options read from the file are processed exactly like options specified on the command line. (See the Universal Control 4.1.0 User Guide for details on command files.) The options must be in their respective command line formats.

Universal Control Manager can process command files that are either encrypted or in plain text (see the [COMMAND_FILE_PLAIN](#) option). Encrypted command files are an excellent place to store sensitive data such as user IDs and passwords. Command files (encrypted or not) that contain sensitive data should be protected from unauthorized read access with a security system, such as RACF.

Use the Universal Encrypt utility provided with Universal Command to encrypt a plain text command file. (For information on Universal Encrypt, see the Universal Products Utilities 4.1.0 User Guide). If a key was used to encrypt the file, the same key must be supplied using the [ENCRYPTION_KEY](#) option.

Note: If a data set / file is specified in this option, it should not be specified additionally in the [COMMAND_FILE_PLAIN](#) option. If it is, the data set / file specified in [COMMAND_FILE_PLAIN](#) will be used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-x <i>ddname</i> or <i>filename</i>		✓	✓	✓	✓
Command Line Option, Long Form	-encryptedfile <i>ddname</i> or <i>filename</i>		✓	✓	✓	✓
Environment Variable	UCTLENCRYPTEDFILE= <i>filename</i>	✓	✓	✓	✓	
Configuration File Keyword	n/a					
STRUCT Parameter	ECMFILE(<i>filename</i>) [ECMMBR(<i>member</i>)]	✓				

Value

ddname or *filename* is the name of the data set or file, respectively, containing the encrypted command parameter values.

3.10 COMMAND_FILE_PLAIN

Description

The `COMMAND_FILE_PLAIN` option specifies the data set (for z/OS) or file containing plain text values for command line option parameters.

Command files specify an additional source of command line options. Storing options in a file can be used in situations where it is not desirable to explicitly specify them on the command line. The options read from the file are processed exactly like options specified on the command line. (See the Universal Command 4.1.0 User Guide for details on command files.) The options must be in their respective command line formats.

Universal Control Manager can process command files that are either in plain text or encrypted (see the `COMMAND_FILE_ENCRYPTED` option). It is strongly recommended that plain text files be further protected from unauthorized access using a native operating system security method, such as RACF.

Note: If a data set / file is specified in this option, it should not be specified additionally in the `COMMAND_FILE_ENCRYPTED` option. If it is, the data set / file specified in `COMMAND_FILE_PLAIN` will be used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-f <i>ddname</i> or <i>filename</i>		✓	✓	✓	✓
Command Line Option, Long Form	-file <i>ddname</i> or <i>filename</i>		✓	✓	✓	✓
Environment Variable	UCTLFILE= <i>filename</i>	✓	✓	✓	✓	
Configuration File Keyword	n/a					
STRUCT Parameter	CMDFILE(<i>filename</i>) [CMDMBR(<i>member</i>)]	✓				

Value

ddname (for z/OS) or *filename* (for OS/400 and UNIX) is the name of the data set or file name, respectively, containing the parameters and their values.

3.11 COMMAND_ID

Description

The COMMAND_ID option identifies a started Universal Products component. If the COMMAND_ID option is not specified, the component name is used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-C <i>id</i>			✓	✓	✓
Command Line Option, Long Form	-cmdid <i>id</i>			✓	✓	✓
Environment Variable	UCTLCMDID <i>id</i>	✓		✓	✓	
Configuration File Keyword	n/a					
STRUCT Parameter	CMDID(<i>id</i>)	✓				

Value

id is any value that identifies the component.

It is used primarily for recognition of components started by Universal Control.

OS/400

If *id* contains non-alphanumeric characters including spaces, it must be enclosed in single (') quotation marks. If a single (') quotation mark is part of the command, enter two single (') quotation marks to represent one.

3.12 CTL_SSL_CIPHER_LIST

Description

The CTL_SSL_CIPHER_LIST option specifies one or more SSL cipher suites that are acceptable to use for network communications on the control session, which is used for component internal communication.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-ctl_ssl_cipher_list <i>cipherlist</i>			✓	✓	✓
Environment Variable	UCTLCTLSSLCIPHERLIST= <i>cipherlist</i>	✓		✓	✓	
Configuration File Keyword	ctl_ssl_cipher_list <i>cipherlist</i>	✓		✓	✓	✓
STRUCT Parameter	CTLCPHRLST(<i>cipherlist</i>)	✓				

Values

cipherlist is a comma-separated list of SSL cipher suites. The list should be ordered with the most preferred suite first and the least preferred suite last.

[Table 3.2](#) identifies the list of SSL cipher suites supported for this option.

Cipher Suite	Description
<i>RC4-SHA</i>	128-bit RC4 encryption and SHA-1 message digest
<i>RC4-MD5</i>	128-bit RC4 encryption and MD5 message digest
<i>AES256-SHA</i>	256-bit AES encryption and SHA-1 message digest
<i>AES128-SHA</i>	128-bit AES encryption and SHA-1 message digest
<i>DES-CBC3-SHA</i>	128-bit Triple-DES encryption and SHA-1 message digest
<i>DES-CBC-SHA</i>	128-bit DES encryption and SHA-1 message digest

Table 3.2 SSL Cipher Suites (for CTL_SSL_CIPHER_LIST)

[Default is *RC4-SHA,RC4-MD5,AES256-SHA,AES128-SHA,DES-CBC3-SHA,DES-CBC-SHA.*]

3.13 ENCRYPTION_KEY

Description

The ENCRYPTION_KEY option specifies the key used to encrypt the command file.

This key acts much like a password for the encrypted command file in that it can be used to protect the file from decryption by unauthorized users.

If a key was used to encrypt a command file (when Universal Encrypt was run), that same key must be specified to decrypt the file, or the decryption will fail. If no key is specified, the default key is used.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-K key		✓	✓	✓	✓
Command Line Option, Long Form	-key key		✓	✓	✓	✓
Environment Variable	UCTLKEY= <i>key</i>					
Configuration File Keyword	n/a					
STRUCT Parameter	KEY(<i>key</i>)	✓				

Value

key is the key used to encrypt the command file.

3.14 EVENT_GENERATION

Description

The EVENT_GENERATION option specifies which events are to be generated and processed as persistent events.

A persistent event record is saved in a Universal Enterprise Controller (UEC) database for long-term storage.

(For a list of all event types for all Universal Products components, see the Universal Event Subsystem 4.1.0 Event Definitions Guide.)

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Environment Variable	n/a					
Configuration File Keyword	event_generation <i>types</i>	√		√	√	√
STRUCT Parameter	n/a					

Values

type specifies a comma-separated list of event types. It allows for all or a subset of all potential event message types to be selected.

Event type ranges can be specified by separating the lower and upper range values with a dash (-) character.

Event types can be selected for inclusion or exclusion:

- Inclusion operator is an asterisk (*).
- Exclusion operator is **X** or **x**.

Examples

- 100,101,102
Generate event types 100, 101, and 102.
- 100-102
Generate event types 100 through 102.
- 100-102,200
Generate event types 100 through 102 and 200.
- *
Generate all event types.
- *,X100
Generate all event types except for 100.
- x*
Generate no event types.
- *,X200-250,X300
Generate all event types except for 200 through 250 and 300.

[Default is X* (no event types).]

3.15 HELP

Description

The HELP option displays a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h		✓	✓	✓	✓
Command Line Option, Long Form	-help		✓	✓	✓	✓
Environment Variable	n/a					
Configuration File Keyword	n/a					
STRUCT Parameter	n/a					

Value

(There are no values for the HELP option.)

3.16 HOSTNAME_RETRY_COUNT

Description

The HOSTNAME_RETRY_COUNT option specifies the number of times that the Universal Control Manager will attempt to establish a connection with a specified Universal Broker before it ends with a connect error.

The Universal Control Manager will sleep for one second between connection attempts.

Connection errors occur for several reasons. A common reason is a failure to resolve the Universal Broker host name specified with the [REMOTE_HOST](#) option. This error can occur intermittently due to a temporary resource shortage or a temporary DNS problem. If your system is prone to host name resolution errors, it may help to have the Universal Control Manager retry the connection several times.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-hostname_retry_count <i>count</i>			✓	✓	✓
Environment Variable	UCTLHOSTNAMERETRYCOUNT= <i>count</i>	✓		✓	✓	
Configuration File Keyword	hostname_retry_count <i>count</i>	✓		✓	✓	✓
STRUCT Parameter	HSTNMRTYCT(<i>count</i>)	✓				

Value

count is the number of times that Universal Control will attempt to establish a connection.

[Default is 1.]

3.17 INSTALLATION_DIRECTORY

Description

The INSTALLATION_DIRECTORY option specifies the directory in which the Universal Control Manager is installed.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Environment Variable	n/a					
Configuration File Keyword	installation_directory <i>directory</i>		✓	✓	✓	
STRUCT Parameter	n/a					

Value

directory is the directory in which the Universal Control Manager is installed.

HP NonStop

[Default is \$SYSTEM.UNVBIN.]

UNIX

[Default is /opt/universal/uct1mgr.]

Windows

[Default is c:\Program Files\Universal\uct1mgr.]

3.18 MESSAGE_LANGUAGE

Description

The MESSAGE_LANGUAGE option specifies the Universal Message Catalog (UMC) that is used to format messages.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-L <i>language</i>		✓	✓	✓	✓
Command Line Option, Long Form	-lang <i>language</i>		✓	✓	✓	✓
Environment Variable	UCTLLANG= <i>language</i>	✓	✓	✓	✓	✓
Configuration File Keyword	language <i>language</i>	✓	✓	✓	✓	✓
STRUCT Parameter	MSGLANG(<i>language</i>)	✓				

Values

language is any UMC file provided by Stonebranch Inc.

There are different UMC files for different languages.

z/OS

The first three characters of the language name are used as a three-character suffix in the UMC member name. UMC files are read from the partitioned data set allocated on ddname UNVNLS. Universal Control message catalog member names start with UCTMC.

HP NonStop, UNIX, and Windows

The first three characters of the language name are used as a three-character suffix in the UMC file base name. All UMC files have a .UMC extension.

OS/400

The first three characters of the language name are used as a three-character suffix in the UMC member base name **UCMMC**. UMC files are located in the source physical file **UNVPRD410/UNVNLS**.

[Default is *ENGLISH*.]

3.19 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>		✓	✓	✓	✓
Command Line Option, Long Form	-level <i>level</i>		✓	✓	✓	✓
Environment Variable	UCTLLEVEL= <i>level</i>	✓	✓	✓	✓	✓
Configuration File Keyword	message_level <i>level</i>	✓	✓	✓	✓	✓
STRUCT Parameter	MSGLEVEL(<i>*level</i>)	✓				

Values

level indicates either of the following level of messages:

- **trace**
Writes trace messages used for diagnostic purposes (see Section [Trace Files](#)).
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

OS/400 and z/OS

[Default is info.]

HP NonStop, UNIX, and Windows

[Default is warn.]

Trace Files

OS/400

Trace file name is ***CURLIB/UNVTRCUCT (UCTn)**, where **n** is the job number of the job invoking Universal Control.

HP NonStop

Trace file name is **UCTLTRC**. It is created in the working subvolume of Universal Control Manager.

UNIX and Windows

Trace file name is **uct1 . trc**. It is created in the working directory of Universal Control Manager.

z/OS

Trace file is written to the data set referenced by the **UNVTRACE** ddname.

3.20 NETWORK_DELAY

Description

The NETWORK_DELAY option specifies the maximum acceptable delay in transmitting data over the network between the Universal Control Manager and Universal Control Server.

If a data transmission takes longer than this specified delay, the operation ends with a time-out error.

NETWORK_DELAY provides the ability to fine tune Universal Control's network protocol. When a data packet is sent over a TCP/IP network, the time it takes to reach the other end depends on many factors, such as network congestion and bandwidth. If the packet is lost before reaching the other end, the other end may wait indefinitely for the expected data. In order to prevent this situation, Universal Control times out waiting for a packet to arrive in the period of time specified by NETWORK_DELAY.

Note: An understanding of the TCP/IP protocol and the network configuration between the Universal Control Manager and Universal Control Server is required in order to determine the appropriate delay value.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-d <i>seconds</i>		✓	✓	✓	✓
Command Line Option, Long Form	-delay <i>seconds</i>		✓	✓	✓	✓
Environment Variable	UCTLNETWORKDELAY= <i>seconds</i>	✓	✓	✓	✓	✓
Configuration File Keyword	network_delay <i>seconds</i>	✓	✓	✓	✓	✓
STRUCT Parameter	DELAY(<i>seconds</i>)	✓				

Values

seconds is the number of seconds to delay before ending an operation with a time-out error.

[Default is 120.]

3.21 NLS_DIRECTORY

Description

The NLS_DIRECTORY option specifies the directory in which Universal Products NLS files are installed.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Environment Variable	n/a					
Configuration File Keyword	nls_directory <i>directory</i>			✓	✓	
STRUCT Parameter	n/a					

Values

directory is the directory in which NLS files are installed.

Defaults

UNIX

[Default is /opt/universal/nls.]

Windows

[Default is ..\nls.]

3.22 OUTBOUND_IP

Description

The OUTBOUND_IP option sets the host or IP address that Universal Control binds to when initiating outgoing connections.

Note: By default, the OUTBOUND_IP option is not set.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-outboundip <i>host</i>			✓	✓	✓
Environment Variable	UCTLOUTBOUNDIP= <i>host</i>	✓		✓	✓	
Configuration File Keyword	outboundip <i>host</i>	✓		✓	✓	✓
STRUCT Parameter	OUTBOUNDIP(<i>host</i>)	✓				

Values

host is the host or IP address.

3.23 PLF_DIRECTORY

Description

The PLF_DIRECTORY option specifies the Program Lock File (PLF) directory where the program lock files are located.

A program lock file is created and used by the Universal Control Manager process to store manager process termination information for the Universal Broker.

OS/400

Do not include this directory in any system or backup that requires an exclusive lock on the directory while Universal Control Manager is running.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-plf_directory <i>directory</i>			✓		
Environment Variable	UCTLPLFDIRECTORY= <i>directory</i>			✓		
Configuration File Keyword	n/a					
STRUCT Parameter	PLFDIR(<i>directory</i>)	✓				

Values

directory is the name of the PLF directory.

A full path name must be specified.

Defaults

UNIX

[Default is /var/opt/universa1/tmp.]

OS/400

[Default is /tmp.]

3.24 PRIVATE_KEY

Description

The PRIVATE_KEY option specifies the location of the PEM-formatted RSA private key that corresponds to the X.509 certificates specified by the CERTIFICATE option.

Note: PRIVATE_KEY is required only if a certificate is specified by CERTIFICATE.

z/OS

PRIVATE_KEY is used only when the SSL_IMPLEMENTATION option is set to OPENSSL.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-private_key <i>ddname</i> or <i>file</i>			✓	✓	✓
Environment Variable	UCTLPRIVATEKEY= <i>file</i>	✓		✓	✓	
Configuration File Keyword	private_key <i>ddname</i> or <i>file</i>	✓		✓	✓	✓
STRUCT Parameter	PVTKEYF(<i>file</i> [<i>lib</i>])	✓				

Values

z/OS

ddname is the ddname of the PEM formatted RSA private key that corresponds to the X.509 certificates. Allocated to the ddname must be either a sequential data set or a member of a PDS that has a variable record format.

UNIX and Windows

file is the path of the PEM-formatted RSA private key file that corresponds to the X.509 certificates.

OS/400

file is the qualified name of the PEM-formatted RSA private key file that corresponds to the X.509 certificates. The file name can be qualified by a library name (*lib*). If not, the library list *LIBL is searched for the first occurrence of the file name.

3.25 PRIVATE_KEY_PWD

Description

The PRIVATE_KEY_PWD option specifies the password or pass phrase for the PEM-formatted RSA private key specified with the [PRIVATE_KEY](#) option.

Note: Whether or not the password is required or not depends on whether or not it is required by the private key.

z/OS

PRIVATE_KEY_PWD is used only when the [SSL_IMPLEMENTATION](#) option is set to *OPENSSL*.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-private_key_pwd <i>password</i>			✓	✓	✓
Environment Variable	UCTLPRIVATEKEYPWD= <i>password</i>	✓		✓	✓	
Configuration File Keyword	privatekeypwd= <i>password</i>	✓		✓	✓	✓
STRUCT Parameter	PVTKEYPWD(<i>password</i>)	✓				

Values

password is the password for the private key.

3.26 REFRESH_CMD

Description

The REFRESH_CMD option directs Universal Broker to refresh the configuration data that it maintains for all components, including itself, or a single, specified component type.

A REFRESH_CMD option that does not specify a component tells Universal Broker to refresh the configuration data that it maintains for all components. The Broker will reread the configuration data and refresh its copy that it keeps in memory for all components.

A REFRESH_CMD option that specifies a component type tells Universal Broker to refresh its copy of the configuration data for that component type and forward the option to components of that type.

Currently, only Universal Event Monitor Server can be specified, as it is the only component that accepts a REFRESH_CMD option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-R [<i>compname</i>]		✓	✓	✓	✓
Command Line Option, Long Form	-refresh [<i>compname</i>]		✓	✓	✓	✓
Environment Variable	n/a					
Configuration File Keyword	n/a					
STRUCT Parameter	REFRESH(* <i>option</i>) [RFSHCMPNM(<i>compname</i>)]	✓				

Values

compname is the name of the component to which Universal Broker forwards this option.

Currently, only *uems* (Universal Event Monitor Server) is a valid value for *compname*.

If *compname* is not specified, Universal Broker refreshes the configuration data for all components (including itself).

OS/400

Valid values for *option* are:

- **yes**
Refresh the configuration data.
- **no**
Do not refresh the configuration data.

3.27 REMOTE_HOST

Description

The REMOTE_HOST option specifies the IP address or host name of the Universal Broker on the remote computer on which to run the command.

The remote computer must have a Universal Broker running and accepting connections.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-i <i>hostaddress</i>		✓	✓	✓	✓
Command Line Option, Long Form	-host <i>hostaddress</i>		✓	✓	✓	✓
Environment Variable	UCTLHOST <i>hostaddress</i>	✓	✓	✓	✓	✓
Configuration File Keyword	host <i>hostaddress</i>	✓	✓	✓	✓	✓
STRUCT Parameter	HOST(<i>hostaddress</i>)	✓				

Values

hostaddress is the IP address of the host computer.

The format of *hostaddress* can be either:

- IP address in dotted form (for example, 1.2.3.4)
- Host name (for example, *dallas*).

3.28 REMOTE_PORT

Description

The REMOTE_PORT option specifies the TCP port on the remote computer on which to send the command.

The remote computer must have a Universal Broker running and accepting connections on the port number.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-p <i>port</i>		✓	✓	✓	✓
Command Line Option, Long Form	-port <i>port</i>		✓	✓	✓	✓
Environment Variable	UCTLPORT <i>port</i>	✓	✓	✓	✓	
Configuration File Keyword	port <i>port</i>	✓	✓	✓	✓	✓
STRUCT Parameter	PORT(<i>port</i>)	✓				

Values

port is the TCP port on the remote computer.

The format of *port* can be either:

- Number (for example, 7887)
- Service name (for example, ubroker)

[Default is 7887.]

3.29 SSL_IMPLEMENTATION

Description

The SSL_IMPLEMENTATION option specifies the Secure Socket Layer (SSL) implementation to be used for network communications.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-ssl_implementation <i>option</i>					✓
Environment Variable	UCTLSSLIMPLEMENTATION= <i>option</i>					✓
Configuration File Keyword	ssl_implementation <i>option</i>					✓
STRUCT Parameter	n/a					

Values

option is the SSL implementation to be used.

Valid values for *option* are:

- **openssl**
OpenSSL SSL library is used for the SSL protocol.
- **system**
z/OS System SSL library is used for the SSL protocol. The z/OS System SSL library has installation and configuration prerequisites. (See the Universal Products 4.1.0 Installation Guide for a description of the prerequisites before using System SSL.)

[Default is openssl.]

3.30 START_CMD

Description

The START_CMD option specifies the name of the Universal Products component that is to be started on the machine specified by the REMOTE_HOST option.

Only those Universal Products server components that do not require interaction with a Manager application can be started with Universal Control. The Universal Broker checks the requested component's type against a list of component types that can be started by Universal Control. The Universal Broker will reject any attempt to start an ineligible component.

Note: In Universal Products 4.1.0, an event-driven UEM Server is the only component that can be started with Universal Control. An event-driven UEM Server component has a component type of `uems`.

If the Universal Control Server is configured with security enabled, a user identifier and password will be required to start a component. The component, once started, will run under the same security context of the Universal Broker.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-start <i>compname</i>			✓	✓	✓
Environment Variable	UCTLSTART= <i>compname</i>	✓		✓	✓	✓
Configuration File Keyword	n/a					
STRUCT Parameter	START(<i>compname</i>)	✓				

Values

compname is the name of the component to be started.

3.31 STOP_CMD

Description

The STOP_CMD option specifies the ID of the component that is to be terminated on the remote computer.

A component ID is assigned to a component when Universal Broker starts it. The Universal Query utility can be used to obtain a list of active components (along with their component IDs) that are managed by an instance of Universal Broker. (See the Universal Products Utilities 4.1.0 User Guide for information about Universal Query.)

STOP_CMD can require the user identifier and password of the user account with which the component to be stopped is executing. If so, the user identifier must be the same as the user identifier with which the component is executing; otherwise, STOP_CMD will fail. The user ID and password are specified with the [USER_ID](#) and [USER_PASSWORD](#) options.

Whether or not STOP_CMD requires a user ID and password depends on the Universal Control Server configuration.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-s <i>compID</i>		✓	✓	✓	✓
Command Line Option, Long Form	-stop <i>compID</i>		✓	✓	✓	✓
Environment Variable	UCTLSTOP= <i>compID</i>	✓	✓	✓	✓	✓
Configuration File Keyword	n/a					
STRUCT Parameter	STOP(<i>compID</i>)	✓				

Values

compID is the ID of the component to be terminated.

3.32 SYSTEM_ID

Description

The SYSTEM_ID option identifies the local Universal Broker with which the Universal Control Manager must register before the Manager performs any request.

Each Universal Broker running on a system is configured with a system identifier that uniquely identifies the Broker.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-system_id <i>ID</i>					✓
Environment Variable	UCTLSYSTEMID= <i>ID</i>					✓
Configuration File Keyword	n/a					
STRUCT Parameter	n/a					

Values

ID is the system identifier of the local Universal Broker.

(Refer to the local Universal Broker administrator for the appropriate system ID to use.)

3.33 USER_ID

Description

The USER_ID option specifies the user identifier that is used to sign on to the remote computer.

The Universal Control Server determines if this option is required.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-u <i>user</i>		✓	✓	✓	✓
Command Line Option, Long Form	-userid <i>user</i>		✓	✓	✓	✓
Environment Variable	UCTLUSERID= <i>user</i>	✓	✓	✓	✓	✓
Configuration File Keyword	userid <i>user</i>	✓	✓	✓	✓	✓
STRUCT Parameter	USER(<i>user</i>)	✓				

Values

user is the user identifier that is used to sign on to the remote computer.

Note: *user* must be a valid user identifier on the remote computer.

3.34 USER_PASSWORD

Description

The USER_PASSWORD option specifies the password for the user identifier that is specified in the [USER_ID](#) option.

The password is always encrypted, regardless of how encryption is configured on the Universal Control Manager and Universal Control Server.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-w password</code>		✓	✓	✓	✓
Command Line Option, Long Form	<code>-pwd password</code>		✓	✓	✓	✓
Environment Variable	<code>UCTLPWD=password</code>	✓	✓	✓	✓	✓
Configuration File Keyword	<code>password password</code>	✓	✓	✓	✓	✓
STRUCT Parameter	<code>PWD(password)</code>	✓				

Values

password is the password for the user identifier.

Note: *password* must be a valid password, on the remote computer, for the user identifier.

3.35 VERIFY_HOST_NAME

Description

The VERIFY_HOST_NAME option specifies whether or not the Universal Broker's X.509 certificate identity is verified.

Verification consists of verifying that the certificate is issued by a trusted CA. The [CA_CERTIFICATES](#) option specifies which CA certificates are considered trusted.

The identity is verified by matching the value specified by VERIFY_HOST_NAME to the Universal Broker's certificate host value.

The following certificate fields are matched in the order listed:

1. X.509 v3 **dNSName** field of the **subjectAltName** extension value
2. X.509 **commonName** attribute of the **subject** field's Distinguished Name (DN) value
3. X.509 v3 **iPAddress** field of the **subjectAltName** extension value

One of these fields must match for identification to be considered successful. If either verification or identification fails, the session is rejected and the Universal Control Manager terminates.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-verify_host_name <i>option</i>			✓	✓	✓
Environment Variable	UCTLVERIFYHOSTNAME= <i>option</i>	✓		✓	✓	
Configuration File Keyword	verify_host_name <i>option</i>	✓		✓	✓	✓
STRUCT Parameter	VFYHSTNM(<i>option</i>)	✓				

Values

option is the specification for whether or not the X.509 certificate identity is verified.

Valid values for *option* are:

- **yes**
Certificate identity is verified using the host name specified by the [REMOTE_HOST](#) option.
- **no**
Certificate identity is not verified.
- *hostname*
Certificate identity is verified using *hostname*. The value *hostname* can be a DNS host name or an IP address.

[Default is no.]

3.36 VERIFY_SERIAL_NUMBER

Description

The VERIFY_SERIAL_NUMBER option specifies a serial number which must be matched by the serial number of a verified Universal Broker X.509 certificate.

Certificate verification consists of verifying that the certificate is issued by a trusted CA. The [CA_CERTIFICATES](#) option specifies which CA certificates are considered trusted.

If either the certificate is not verified or the serial numbers do not match, the session is rejected and the Universal Control Manager terminates.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-verify_serial_number <i>number</i>			✓	✓	✓
Environment Variable	UCTLVERIFYSERIAL NUMBER= <i>number</i>	✓		✓	✓	
Configuration File Keyword	verify_serial_number <i>number</i>	✓		✓	✓	✓
STRUCT Parameter	VFYSERNUM(<i>number</i>)	✓				

Values

number is the serial number to be matched by the X.509 certificate serial number.

number can be specified in a hexadecimal format by prefixing it with *0x* or *0X*. For example, the value *0x016A392E7F* would be considered a hexadecimal format.

3.37 VERSION

Description

The VERSION option writes the program version information and copyright.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v		✓	✓	✓	✓
Command Line Option, Long Form	-version		✓	✓	✓	✓
Environment Variable	n/a					
Configuration File Keyword	n/a					
STRUCT Parameter	VERSION(* <i>option</i>)	✓				

Values

(There are no values to be specified for this option, except for OS/400.)

OS/400

Valid values for *option* are:

- YES
Write program version information and copyright.
- NO
Do not write program version information and copyright.

[Default is NO.]

Chapter 4

Universal Control Server

Configuration Options

4.1 Overview

This chapter provides detailed information on the configuration options available for use with the Universal Control Server.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Control 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented on each option.

4.2 Configuration Options List

Table 4.1 identifies all Universal Control Server configuration options.

Option	Description	Page
ACTIVITY_MONITORING	Specification for whether or not product activity monitoring events are generated.	121
CODE_PAGE	Code page used for text translation.	122
EVENT_GENERATION	Events to be generated as persistent events.	123
INSTALLATION_DIRECTORY	Base directory in which Universal Control Server is installed.	125
LOGON_METHOD	Method of how users are logged onto the system.	126
MESSAGE_LEVEL	Level of messages printed.	127
NLS_DIRECTORY	Location of UMC and UTT files.	130
TMP_DIRECTORY	Directory name used for temporary files.	131
TRACE_DIRECTORY	Location of trace files.	132
USER_SECURITY	Specification for whether or not user authentication is active.	132

Table 4.1 Universal Control Server Configuration Options

4.3 ACTIVITY_MONITORING

Description

The ACTIVIITY_MONITORING option specifies whether or not product activity monitoring events are generated.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	activity_monitoring <i>option</i>	✓		✓	✓	✓

Values

option is the specification for whether or not product activity monitoring events are generated.

Valid values for *option* are:

- **yes**
Activate monitoring events.
- **no**
Deactivate monitoring events.

[Default is no.]

4.4 CODE_PAGE

Description

The CODE_PAGE option specifies the character code page that is used to translate text data received and transmitted over the network.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	codepage <i>codepage</i>	✓	✓	✓	✓	✓

Value

codepage is the character code page that is used to translate data.

codepage references a Universal Translate Table (UTT) file provided with the product (see Section [20.5 UTT Files](#) for information on UTT files). UTT files are used to translate between Unicode and the local single-byte code page. (All UTT files end with an extension of `.utt`.)

[Default

The default code page is different for different operating systems:

- **ISO8859-1 (8-bit ASCII) ASCII-based operating systems**
- **IBM1047 (EBCDIC) EBCDIC-based operating system]**

See Section [20.4 Character Code Pages](#) for a complete list of character code pages provided by Stonebranch Inc. for use with Universal Products.

4.5 EVENT_GENERATION

Description

The EVENT_GENERATION option specifies which events are to be generated and processed as persistent events.

A persistent event record is saved in a Universal Enterprise Controller (UEC) database for long-term storage.

(For a list of all event types for all Universal Products components, see the Universal Event Subsystem 4.1.0 Event Definitions Guide.)

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	event_generation <i>types</i>	√		√	√	√

Values

type specifies a comma-separated list of event types. It allows for all or a subset of all potential event message types to be selected.

Event type ranges can be specified by separating the lower and upper range values with a dash (-) character.

Event types can be selected for inclusion or exclusion:

- Inclusion operator is an asterisk (*).
- Exclusion operator is **X** or **x**.

Examples

- 100,101,102
Generate event types 100, 101, and 102.
- 100-102
Generate event types 100 through 102.
- 100-102,200
Generate event types 100 through 102 and 200.
- *
Generate all event types.
- *,X100
Generate all event types except for 100.
- x*
Generate no event types.
- *,X200-250,X300
Generate all event types except for 200 through 250 and 300.

[Default is X* (no event types).]

4.6 INSTALLATION_DIRECTORY

Description

The INSTALLATION_DIRECTORY option specifies the location in which Universal Control Server is installed.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	installation_directory <i>directory</i>			✓	✓	

Values

directory is the location in which UCMD Server is installed.

Defaults

UNIX

[Default is /opt/universal/uct1srv.]

Windows

[Default is c:\Program Files\Universal\uct1srv.]

4.7 LOGON_METHOD

Description

The LOGON_METHOD option specifies the user's log on method.

If the UCMD Server is configured for user security (see the [USER_SECURITY](#) option), the log on method determines how the user is logged onto the Windows system.

If security is inactive, LOGON_METHOD is ignored.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	logon <i>option</i>				√	

Values

option is the user's log on method.

Valid values for *option* are:

- batch**
 Windows log on type is **batch**. A batch log on prevents the command from interacting with the desktop. The user ID logging on as a batch user requires the Windows User Right "Log on as a batch job." If the user does not have this right, the log on action will fail.
- interactive**
 Windows log on type is **interactive**. An interactive log on permits the command to interact with the desktop. No additional rights are required for a user to log on as an interactive user.

[Default is interactive.]

4.8 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	message_level <i>level</i> [/]	√	√	√	√	√

Values

level indicates either of the following level of messages:

- **trace**
Writes trace messages used for diagnostic purposes (see Section [Trace Files](#)).
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

[Default is info.]

Trace Files

OS/400

Trace file name is ***CURLIB/UNVTRCUCT (UCTn)**, where **n** is the Universal Server job number under which the Universal Control Server program is running. ***CURLIB** is the temporary library designated during the Universal Products installation process; the default temporary library is **UNVTMP410**.

HP NonStop

Trace file name is **uctXXXXX**, where:

- **XXXXX** is the last five decimal values of the component ID of the Universal Control Server.

The trace file is created in the **\$SYSTEM.UNVTRACE** subvolume.

UNIX and Windows

Trace file name is **uctsrv-N.trc**, where:

- **N** is the process ID of the Universal Control Server.

It is created in the working directory of the Universal Control Server.

z/OS

There are two possible destinations of the trace data:

1. If ddname **UNVTRMDL** is defined in the UBROKER started task procedure, a sequential data set is created using the data set allocated to **UNVTRMDL** as a model.

The dynamically allocated trace data set name is **#HLQ.UCT.Dyymmdd.Thhmmss.Cnnnnnnn**, where:

- **#HLQ** is the data set name allocated on the UNVTRMDL ddname.
- **yymmdd** is the year, month, and day.
- **hhmmss** is the hour, minute, second the data set was allocated.
- **nnnnnnn** is the last seven digits of the Server's component ID in hexadecimal format.

Each time that a server is restarted, its sequence number is incremented. If a server is restarted more than 15 times, tracing is disabled.

2. If ddname **UNVTRMDL** is not defined in the UBROKER started task procedure, member name **Tnnnnnn** is created in the PDS or PDS/E allocated to the **UNVTRACE** ddname, where:

- **nnnnnnn** is the last seven digits of the Server's component ID in hexadecimal format.
- **s** is the component ID's sequence number from 0 - F.

Each time that a server is restarted, its sequence number is incremented. If a server is restarted more than 15 times, tracing is disabled.

Depending on the error condition being diagnosed, it is possible that the member name of the **UNVTRACE** PDS or PDS/E is not created. If this occurs, the **UNVTRMDL** ddname must be used to create a sequential data set name. The records written to PDS and PDS/E members cannot be wrapped.

4.9 NLS_DIRECTORY

Description

The NLS_DIRECTORY option specifies the name of the directory where the Universal Control Server message catalog and code page tables are located.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	nls_directory <i>directory</i>			✓	✓	

Values

directory is the name of the directory where the catalog and tables are located.

Full path names are recommended.

Relative path names are relative to the `universal` installation directory.

Defaults

UNIX

[Default is `/opt/universal/nls.`]

Windows

[Default is `..\nls.`]

4.10 TMP_DIRECTORY

Description

The TMP_DIRECTORY option specifies the directory name that the Universal Control Server uses for temporary files.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	tmp_directory <i>directory</i>			✓	✓	✓

Values

directory is the name of the directory.
It should specify a fully qualified path name.

Defaults

UNIX

Default is `/var/opt/universal/tmp`.

Windows

[Default is `..tmp`].

z/OS

[Default is `/tmp`].

4.11 TRACE_DIRECTORY

Description

The TRACE_DIRECTORY option specifies the directory name that the Universal Control Server uses for its trace files.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	trace_directory <i>directory</i>			✓	✓	

Values

directory is the name of the trace file directory.

Relative path names are relative to the Universal Control Server installation directory. Full path names are recommended.

Defaults

Windows

[Default is *C:\Program Files\Universal\UCtSrv.*]

UNIX

[Default is */var/opt/universal/trace.*]

4.12 USER_SECURITY

Description

The USER_SECURITY option specifies whether or not to user security and, if so, the security method.

If user security is activated, the remote Universal Control Manager requesting command execution is required to supply a local user ID and password. The user's command is started as that user.

If user security is not activated, the user ID and password is not required from the remote user. The user's process is started with the same user ID as Universal Control Server ID.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Configuration File Keyword	security <i>option</i>	✓	✓	✓	✓	✓

Values

option is the specification (and method) for activating user security.

z/OS

- **default**
Use z/OS SAF user authentication method. The user ID must have a OMVS segment.
- **none**
No user security. Not recommended.

Windows

- **default**
User-supplied user ID and password is authenticated against the user profile.
- **none**
No user security. Not recommended.

UNIX

- **default**
Use UNIX default user authentication method, `/etc/passwd`.
- **trusted**
Use HP Trust Security authentication.
- **pam**
Use the Pluggable Authentication Modules (PAM) interface.
- **none**
No user security.

OS/400

- **default**
Security is activated and uses OS/400 authentication.
- **none**
Security is not activated.

HP NonStop

- **default**
Use HP NonStop default user authentication method, **SAFEGUARD**.
- **trusted**
Use HP Trust Security authentication.
- **pam**
Use the Pluggable Authentication Modules (PAM) interface.
- **none**
No user security.

Chapter 5

Universal Control

Component Definition Options

5.1 Overview

This chapter provides detailed information about the options that comprise Universal Control (UCTL) component definitions.

The options are listed alphabetically, without regard to any specific operating system.

Information on how component definitions are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [5.2 Component Definition Options Information](#) provides a guideline for understanding the information presented for each component definition option.

5.2 Component Definition Options Information

For each component definition option, this chapter provides the following information.

Description

Describes the option and how it is used.

Usage

Provides a table of the following information:

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	<Format / Value>					

Method

Identifies the method used for specifying a Universal Control component definition option:

- Component Definition Keyword

Syntax

Identifies the syntax of the method used to specify the option:

- **Format** Specific characters that identify the option.
- **Value** Type of value(s) to be supplied for this method.

(Operating System)

Identifies (with a ✓) the operating systems for which the method of specifying the option is valid:

- **O** OS/400
- **T** HP NonStop
- **U** UNIX
- **W** Windows
- **Z** z/OS

Values

Identifies all possible values for the specified value type.

Defaults are identified in **[bracketed bold type]**.

5.3 Component Definition Options

Table 5.1 identifies all of the options that can comprise a Universal Control component definition.

Component	Description	Page
AUTOMATICALLY_START	Specification for whether or not the UCTL Server starts automatically when Universal Broker is started.	139
COMPONENT_NAME	Name by which the clients know the UCTL Server.	140
CONFIGURATION_FILE *	Name of the UCTL Server's configuration file.	141
RUNNING_MAXIMUM	Maximum number of UCTL Servers that can run simultaneously.	142
START_COMMAND *	Program name of the UCTL Server.	143
WORKING_DIRECTORY *	Directory used as the working directory of the UCTL Server.	144
* These options are required in all component definitions.		

Table 5.1 Universal Control Component Definition Options

5.4 AUTOMATICALLY_START

Description

The AUTOMATICALLY_START option specifies whether or not the Universal Control Server starts automatically when Universal Broker is started.

Note: AUTOMATICALLY_START is optional in a component definition.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	auto_start <i>option</i>	✓	✓	✓	✓	✓

Values

option is the specification for how the Universal Control Server is started.

The only valid values for option is:

- **no**
Universal Control Server is not started automatically when Universal Broker is started. It is started only on demand.

5.5 COMPONENT_NAME

Description

The COMPONENT_NAME option specifies the name of the Universal Control Server. Component start requests refer to Universal Control Server by this name.

Note: COMPONENT_NAME is optional in a component definition. If it is not specified, the file name is used as the component name.

Usage

Specification Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	component_name <i>name</i>	√	√	√	√	√

Values

name is the name of the Universal Control Server.

There is only one valid value for *name*: **uct1**.

(This is the name of the Universal Control Server component definitions file / member.)

Note: This name should not be changed.

5.6 CONFIGURATION_FILE

Description

The CONFIGURATION_FILE option specifies the name of the Universal Control Server configuration file.

Note: CONFIGURATION_FILE is required in a component definition.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	configuration_file <i>member</i> or configuration_file <i>filename</i>	√	√	√	√	√

Values

member / filename is the name of the configuration member / file.

OS/400

Non-qualified file names are located in the library list *LIBL.

HP NonStop

Relative paths are relative to the component's working subvolume.

UNIX and Windows

Relative paths are relative to the component's working directory.

z/OS

Member names are located in the UNVCONF library allocated to the Universal Broker ddname UNVCONF. The installation default is UTSCFG00.

5.7 RUNNING_MAXIMUM

Description

The RUNNING_MAXIMUM option specifies the maximum number of UCTL Servers that can run simultaneously.

If this maximum number is reached, any command received to start a UCTL Server is rejected.

Note: RUNNING_MAXIMUM is optional in a component definition.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	running_max <i>maximum</i>	✓	✓	✓	✓	✓

Values

maximum is the maximum number of UCTL Servers that can run simultaneously.

[Default is 100.]

5.8 START_COMMAND

Description

The START_COMMAND option specifies the the full path name (member name for z/OS) of the Universal Control Server program.

Optionally, START_COMMAND also can specify command line options.

Note: START_COMMAND is required in a component definition.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	start_command <i>member</i> or start_command <i>name[options]</i>	√	√	√	√	√

Values

member / name is the program name of Universal Control Server.

options is the optional list of command line options.

z/OS

The program object must be in the Universal Broker's search order for loading program objects. The default location is the **SUNVLOAD** library allocated to the Broker's **STEPLIB** ddname.

HP NonStop, UNIX, Windows

name is the full path name of the Universal Control Server program.

OS/400

name is the Universal Command Server program. If the program name is non-qualified, the library list ***LIBL** is searched.

5.9 WORKING_DIRECTORY

Description

The WORKING_DIRECTORY option specifies the full path name of the directory used as the working directory of UCTL Server.

Note: WORKING_DIRECTORY is required in a component definition.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Component Definition Keyword	working_directory <i>directory</i>	✓	✓	✓	✓	✓

Values

directory is the full path name of the working directory.

[Default is (.).

HP NonStop, UNIX, Windows

directory is the full path name of the directory Universal Control Server uses as its working directory.

z/OS

directory is the HFS directory name that the Universal Control Server uses as its working directory.

OS/400

working_directory serves as a required placeholder only.

Note: Do not change this directory.

Chapter 6

Universal Control

UACL Entries

6.1 Overview

This chapter provides detailed information on the Universal Access Control List (UACL) entries available for use with Universal Control.

The UACL entries are listed alphabetically, without regard to any specific operating system.

Information on how these UACL entries are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [6.2 UACL Entries Information](#) provides a guideline for understanding the information presented for each UACL entry.

6.2 UACL Entries Information

For each UACL entry, this chapter provides the following information.

Description

Describes the UACL entry and how it is used.

Usage

Provides a table of the following information:

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UACL File Keyword	<Type / Rule>					

Method

Identifies the method used for specifying a UACL entry:

- UACL File Keyword

Syntax

Identifies the syntax of the method used for a UACL entry:

- **Type** Universal Products component to which the rule applies.
- **Rule** Client's identity, request to which the entry pertains, and security attributes that the entry enforces.

(Operating System)

Identifies (with a ✓) the operating systems for which the method of specifying the UACL entry is valid:

- **O** OS/400
- **T** HP NonStop
- **U** UNIX
- **W** Windows
- **Z** z/OS

Values

Identifies all possible values for the fields in a UACL entry rule.

Defaults are identified in **[bracketed bold type]**.

6.3 UACL Entries List

Table 6.1 identifies all Universal Control UACL Entries.

UACL Entry	Description	Page
UCTL_ACCESS	Allows or denies access to Universal Control Server services. There are two forms to this entry: <ul style="list-style-type: none">• uct1_access• uct1_cert_access	149
UCTL_REQUEST	Allows or denies access to Universal Control Server services based on client identification and request type.	151

Table 6.1 Universal Control UACL Entries

6.4 UCTL_ACCESS

Description

A UCTL_ACCESS UACL entry either allows or denies access to Universal Control Server services.

If access is permitted, UCTL_ACCESS also specifies whether or not user authentication is required.

There are two forms of the UCTL_ACCESS entry based on the client identification method:

- **uct1_access** form is for IP-based client identification.
- **uct1_cert_access** is for X.509 certificate-based client identification.

A **uct1_access** UACL entry is matched if all of the following occur:

- Request comes from an IP address identified by *host*.
- Remote end is executing as user *remote_user*.
- Remote user is requesting to execute a command as local user *local_user*.

A **uct1_cert_access** UACL entry is matched if both of the following occur:

- Request comes from a client with a certificate identifier of *certid*.
- Remote user is requesting to execute a command as local user *local_user*.

The first matching rule is used to control access.

See Section [2.6.2 UACL Entries](#) in the Universal Products Utilities 4.1.0 User Guide for details on *host*, *remote_user*, *local_user*, and *certid* specification syntax.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UACL File Keyword	uctl_access <i>host,remote_user,local_user,access,auth</i> uctl_cert_access * <i>certid,local_user,access,auth</i>	✓	✓	✓	✓	✓
* uctl_cert_access is not a valid form of UCTL_ACCESS for HP NonStop.						

Values

Valid values for *access* are:

- **deny**
Service is denied. A message is returned to the remote end. The connection is closed.
- **allow**
Service is accepted and processed.

Valid values for *auth* are:

- **auth**
Local user account must be authenticated. The Manager must provide a proper password for the account.
- **noauth**
User ID provided by the Manager does not have to match the user process being stopped.

Windows

To set **noauth** via the Universal Configuration Manager, de-select **Require matching local user account** when you are adding or editing an Access ACL (uctl_access) entry.

OS/400, HP NonStop, UNIX, z/OS

Additionally, **noauth** specifies that the local user account does not require user authentication. The Manager still must supply a password to satisfy command syntax rules, but it will not be verified. Any password value will suffice.

Note: **noauth** should be used with care. Turning off user authentication may violate your local security policies on the Server system.

6.5 UCTL_REQUEST

Description

A UCTL_REQUEST UACL entry allows or denies access to Universal Control Server services based on client identification and request type.

If access is permitted, the UCTL_REQUEST also specifies whether or not user authentication is required.

There are two forms of the UCTL_REQUEST entry based on the client identification method:

- **uctl_request** form is for IP-based client identification.
- **uctl_cert_request** is for X.509 certificate-based client identification.

A **uctl_request** UACL entry is matched if all of the following occur:

- Request comes from an IP address identified by *host*.
- Remote end is executing as user *remote_user*.
- Remote user is requesting to execute a command as local user *local_user*.

A **uctl_cert_request** UACL entry is matched if both of the following occur:

- Request comes from a client with a certificate identifier of *certid*.
- Remote user is requesting to execute a command as local user *local_user*.

The first matching rule is used to control access.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UACL File Keyword	uctl_request <i>host,remote_user,local_user,</i> <i>req_type,req_name,access,auth</i> uctl_cert_request <i>certid,local_user,req_type,req_name,</i> <i>access,auth</i>	✓		✓	✓	✓

Values

req_type specifies what type of request the Manager is requesting.

req_name further qualifies the request. The value of *req_name*, which depends on the value of *req_type*, is defined for each *req_type* below.

Valid values for *req_type* are:

- **refresh**
Manager request is for the refresh of an active component's configuration. *req_name* is a type of component, as specified in each component's definition. Not all component types can have their configurations refreshed from Universal Control.

Note: **refresh** does not include a Universal Broker REFRESH command, which is not processed by any UACL entry.
- **start**
Manager is requesting the start of the component.

req_name is the type of component which corresponds to an installed component definition. Not all components can be started from Universal Control.
- **stop**
Manager is requesting component termination.

req_name value is blank.

See Section [2.6.2 UACL Entries](#) in the Universal Control 4.1.0 User Guide for details on *host*, *remote_user*, *local_user*, and *certid* specification syntax.

Valid values for *access* are:

- **deny**
Service is denied. A message is returned to the remote end. The connection is closed.
- **allow**
Service is accepted and processed.

Valid values for *auth* are:

- **auth**
Local user account must be authenticated. The Manager must provide a proper password for the account.
- **noauth**
User ID provided by the Manager does not have to match the user process being stopped.

OS/400, HP NonStop, UNIX, z/OS

Additionally, **noauth** specifies that the local user account does not require user authentication. The Manager still must supply a password to satisfy command syntax rules, but it will not be verified. Any password value will suffice.

Note: **noauth** should be used with care. Turning off user authentication may violate your local security policies on the Server system.

Chapter 7

Universal Copy

7.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Copy.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

7.2 Configuration Options List

Table 7.1, below, identifies the Universal Copy configuration options.

Option Name	Description	Page
CPYMODE	Copy mode for reading and writing files	155
FRMFILE	Name of a file to copy	156
FRMFILES	Names of additional files to copy	157
FRMMBR	Name of a member in the file specified by FRMFILE	158
HELP	Writes a description of the command options and their format	159
INPUTMODE	Mode for opening the file for input	160
MESSAGE_LEVEL	Level of messages displayed by Universal Copy	161
MODE	Mode in which input files are read and output files are written	162
OUTPUT	File name that data is written to instead of standard output	163
OUTPUTMODE	Mode for opening the file for output	164
REPLACE	Specification for whether or not the file name specified with the OUTPUT option is replaced if it already exists	165
TOFILE	Output file that receives specified concatenated input files	166
TOMBR	Name of a member in the file specified by TOFILE	167
TRANSACTIONAL	Specification for whether or not the copy operation is performed in transactional mode	168
VERSION	Writes the program version and copyright information	169

Table 7.1 Universal Copy Configuration Options

7.3 CPY_MODE

Description

The CPY_MODE option sets the copy mode for reading and writing files.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	CPYMODE(*mode)	√				

Values

mode is the copy mode for reading and writing files.

Valid values for *mode* are:

- **binary**
Copy the data as binary data. The data is not translated in any manner.
- **text**
Copy the data as text data. All trailing blank characters in a record are ignored. A new-line character is inserted after the last non-blank character. The data is subjected to code page conversions.
- **savf**
Copy the data as a save file. This is required when working with save files. The data is not translated in any manner.

[Default is text.]

7.4 FRMFILE

Description

The FRMFILE option specifies the name of a file to copy.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	FRMFILE([<i>library</i> /] <i>file</i>)	√				

Values

file is the name of a file to copy.

Valid values for *file* are:

- ***stdin**
Data is copied from the job's standard input file.
 - If executed from an interactive job, standard input is allocated to the terminal. An ILE session manager screen is displayed in which the user enters the data to copy.
 - If executed from a batch job, file **QINLINE** is allocated to standard input.
- *file name*
Data is copied from the specified file.

[Default is *stdin.]

library is the name of a library with which *file* optionally can be qualified.

- ***libl**
File is located in the library list.
- ***curlib**
File is located in the current library.
- *library name*
File is located in the specified library.

7.5 FRMFILES

Description

The FRMFILES option specifies the names of additional files to copy.

Files are copied in the order listed, starting with the file specified by the [FRMFILE](#) option and continuing with the files specified in FRMFILES. The resulting output file is a concatenation of all input files. Up to 39 files can be specified in the list.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	FRMFILES([<i>library/</i>] <i>file</i> [<i>member</i>])...	√				

Values

file is the name of an additional file to copy.

library is the name of a library with which *file* optionally can be qualified.

Valid values for *library* are:

- ***libl**
File is located in the library list.
- ***curlib**
File is located in the current library.
- *library name*
File is located in the specified library.

member is the name of a member in the specified file to copy.

Valid values for *member* are:

- ***first**
First member in the file is processed.
- ***all**
All members in the file are processed as one.

[Default is *first.]

7.6 FRMMBR

Description

The FRMMBR option specifies the name of a member in the file specified by [FRMFILE](#).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	FRMMBR (<i>member</i>)	√				

Values

member is the name of a member in the file.

Valid values for *member* are:

- ***first**
First member in the file is processed.
- ***all**
All members in the file are processed as one.
- *member name*
Specified member name is processed.

[Default is *first.]

7.7 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h		✓	✓	✓	
Command Line Option, Long Form	-help		✓	✓	✓	
STRUCP Parameter	n/a					

Values

(There are no values for this option.)

7.8 INPUTMODE

Description

The INPUTMODE option specifies the mode for opening the file for input.

If this option is used, it overrides the [CPY_MODE](#) option for input files. See the `fopen()` function in the C Runtime Library manual (*ILE C/C++ for iSeries Run-Time Library Functions*) for information.

The default is to use the [CPY_MODE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	INPUTMODE('option')	√				

Values

'option' is the mode for opening the file.

7.9 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>		✓	✓	✓	
Command Line Option, Long Form	-level <i>level</i>		✓	✓	✓	
STRUCP Parameter	MSGLEVEL(* <i>level</i>)	✓				

Values

level indicates either of the following level of messages:

- **trace**
Writes trace messages used for diagnostic purposes.
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

OS/400

[Default is *info*.]

HP NonStop, UNIX, and Windows

[Default is *warn*.]

7.10 MODE

Description

The MODE option specifies the mode in which input files are read and output files are written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-m <i>mode</i>		✓	✓	✓	
Command Line Option, Long Form	-mode <i>mode</i>		✓	✓	✓	
STRUCP Parameter	n/a					

Values

mode is the mode in which the files are read and written.

Valid values for *mode* are:

- **binary**
Treats all data as binary data. No interpretation of end-of-line characters or end-of-file characters is performed.
- **text**
Treats all data as text data. End-of-line characters are interpreted.

[Default is binary.]

UNIX

There is no difference between binary and text.

HP NonStop

If the specified output file is within the Guardian file space, **TEXT** mode will generate an EDIT file with a file code of 101. If the specified output file is within the OSS file space, or the mode is set to **BINARY**, a C file with a file code of 180 will be generated.

7.11 OUTPUT

Description

The OUTPUT option specifies the name of a file to which data is written instead of standard output.

Note: See FILE in the Command Operands sections of [Chapter 5 Universal Copy](#) in the Universal Products Utilities 4.1.0 User Guide for operating system-specific file naming conventions.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-o <i>filename</i>		✓	✓	✓	
Command Line Option, Long Form	-output <i>filename</i>		✓	✓	✓	
STRUCP Parameter	n/a					

Values

filename is the name of the file to which data is written.

[Default is standard output.]

7.12 OUTPUTMODE

Description

The OUTPUTMODE option specifies the mode for opening the file for output.

If this option is used, it overrides the [CPY_MODE](#) option for output files. See the `fopen()` function in the C Runtime Library manual (*ILE C/C++ for iSeries Run-Time Library Functions*) for information.

The default is to use the [CPY_MODE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	OUTPUTMODE('option')	√				

Values

mode is the mode for opening the file.

7.13 REPLACE

Description

The REPLACE option specifies whether or not the file specified by the [OUTPUT](#) option is replaced (if it already exists).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-r <i>option</i>		✓	✓	✓	
Command Line Option, Long Form	-replace <i>option</i>		✓	✓	✓	
STRUCP Parameter	REPLACE(* <i>option</i>)	✓				

Values

option is the specification for whether or not to replace the file.

Valid values for *option* are:

- **yes**
File is replaced.
- **no**
File is not replaced.

[Default is yes.]

7.14 TOFILE

Description

The TOFILE option specifies name of the output file that receives the specified concatenated input files.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	TOFILE([<i>library/</i>] <i>file</i>)	√				

Values

file is the name of the output file receiving the input files.

Valid values for *file* are:

- ***stdout**
Output is written to standard output.
 - If executed from an interactive job, standard output is allocated to the terminal from which STRUCP is executed. The ILE session terminal is displayed to view the output.
 - If executed from a batch job, standard output is allocated to file **QPRINT**.
- *filename*
Output is written to the specified file name. If the file is not found, it is created as a physical source file with a record length of 266.

[Default is *stdout.]

library is optional name of a library with which *file* can be qualified.

Valid values for *library* are:

- ***libl**
File is located in the library list.
- ***curlib**
File is located in the current library.
- *library name*
File is located in the specified library.

7.15 TOMBR

Description

The TOMBR option specifies the name of a member in the file specified by the [TOFILE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUCP Parameter	TOMBR(<i>member</i>)	√				

Values

member is the name of a member in the file.

[Default is the file name.]

7.16 TRANSACTIONAL

Description

The TRANSACTIONAL option specifies whether or not the copy operation is performed in transactional mode.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>option</i>		✓	✓	✓	
Command Line Option, Long Form	-transactional <i>option</i>		✓	✓	✓	
STRUCP Parameter	n/a					

Values

option is the specification for whether or not the copy operation is performed in transactional mode.

Valid values for *option* are:

- **yes**
Data is copied in a transactional mode. The data first is copied to a temporary file on the same file system. When the copy operation completes successfully, the temporary file is renamed to the file name specified by the [OUTPUT](#) option.
- **no**
Data is not copied in a transactional mode.

[Default is no.]

7.17 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v		✓	✓	✓	
Command Line Option, Long Form	-version		✓	✓	✓	
STRUCP Parameter	VERSION (<i>option</i>)	✓				

Values

HP NonStop, UNIX, and Windows

There are no values for this option.

OS/400

Valid values for *option* are:

- **yes**
Write program version information and copyright.
- **no**
Do not write program version information and copyright.

[Default is no.]

Chapter 8

Universal Database Dump

8.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Database Dump (UDBDUMP). UDBDUMP is the Berkeley `db_dump` utility tailored specifically for Universal Products databases.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

8.2 Configuration Options List

Table 8.1, below, identifies the Universal Database Dump configuration options.

Option Name	Description	Page
DATABASE_FILE	Database file to be dumped.	172
DUMP_OPTIONS	Controls database dump behavior, including the format of the dump output and the extent to which data is recovered from a possibly corrupt database file.	173

Table 8.1 Universal Database Dump Configuration Options

8.3 DATABASE_FILE

Description

The DATABASE_FILE option specifies the database file to be dumped.

DATABASE_FILE is the last option specified on the command line.

z/OS

The database file must be located in the root directory of the HFS data set allocated on the UNVDB ddname. The HFS data set must be mounted prior to running UDBLOAD.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	<i>database</i>			✓	✓	✓

Values

database is the database file to be dumped.

8.4 DUMP_OPTIONS

Description

The DUMP_OPTIONS option controls database dump behavior, including the format of the dump output and the extent to which data is recovered from a possibly corrupt database file.

There are two forms of the DUMP_OPTIONS option:

- **-p** controls the format of the dump output, causing some printable characters to be dumped as text characters. This form of DUMP_OPTIONS is useful if you intend use standard text editors and tools to modify the contents of databases upon reload.
- **-r** controls data recovery behavior, and instructs the utility to recover as many records as possible.

Note: The **db_dump** utility also accepts a **-R** option. However, the upper case **-R** is not recommended. It specifies that aggressive recovery should be performed, which can result in potentially bad or deleted records being recovered.

When using DUMP_OPTIONS, only use lower case **-r**.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-p -r			✓	✓	✓
Command Line Option, Long Form	n/a					

Values

(There are no values used with this option.)

Chapter 9

Universal Database Load

9.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Database Load (UDBLOAD). UDBLOAD is the Berkeley `db_load` utility tailored specifically for Universal Product databases.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

9.2 Configuration Options List

Table 9.1, below, identifies the Universal Database Load configuration options.

Option Name	Description	Page
DATABASE_FILE	Database file to be dumped.	176
OVERWRITE	Specification to recover as many records as possible from a possibly corrupt database file.	177

Table 9.1 Universal Database Load Configuration Options

9.3 DATABASE_FILE

Description

The DATABASE_FILE option specifies the database file to be loaded.

DATABASE_FILE is the last option specified on the command line.

z/OS

The database file must be located in the root directory of the HFS data set allocated on the UNVDB ddname. The HFS data set must be mounted prior to running UDBLOAD.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	<i>database</i>			✓	✓	✓

Values

database is the database file to be loaded.

9.4 OVERWRITE

Description

The OVERWRITE option specifies that the database file is to be overwritten, not updated.

To simply update the database, omit this option.

OVERWRITE is specific to Universal Database Load, not to the Berkeley `db_load` utility.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-o			✓	✓	✓
Command Line Option, Long Form	n/a					

Values

(There are no values specified for this option.)

Chapter 10

Universal Display Log File

10.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Display Log File.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

10.2 Configuration Options List

Table 10.1, below, identifies the Universal Display Log File configuration options.

Option Name	Description	Page
PRIMARY_FILE	Name of the primary output file	180
PRIMARY_MEMBER	Name of the primary output file member	181
REMOVE_MEMBERS	Controls the deletion of job log output file members	182
SECONDARY_FILE	Name of the secondary output file	183
SECONDARY_MEMBER	Name of the secondary output file member	184

Table 10.1 Universal Display Log File - Configuration Options

10.3 PRIMARY_FILE

Description

The PRIMARY_FILE option specifies the name of the primary output file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UDSPJOGF Parameter	PRMRYFILE (<i>filename</i> [<i>library</i>])	√				

Values

filename is the name of the primary output file.

filename can be qualified by a *library* name.

10.4 PRIMARY_MEMBER

Description

The PRIMARY_MEMBER option specifies the name of the primary output file member.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UDSPJOGF Parameter	PRMRYMBR (<i>member</i>)	√				

Values

member is the name of the primary output file member.

10.5 REMOVE_MEMBERS

Description

The REMOVE_MEMBERS option controls the deletion of job log output file members.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UDSPJOGF Parameter	REMOVE (<i>*option</i>)	√				

Values

**option* is the specification for controlling the deletion of the job log output file members.

Valid values for **option* are:

- **yes**
Specified members will be deleted as a result of running this command.
- **no**
Specified job log output members will not be deleted as a result of running this command.

[Default value is *no*.]

10.6 SECONDARY_FILE

Description

The SECONDARY_FILE option specifies the name of the secondary output file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UDSPJOGF Parameter	SCNDRYFILE (<i>filename</i> [<i>library</i>])	√				

Values

filename is the name of the secondary output file.

filename can be qualified by a *library* name.

10.7 SECONDARY_MEMBER

Description

The SECONDARY_MEMBER option specifies the name of the secondary output file member.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
UDSPJOGF Parameter	SCNDRYMBR (<i>member</i>)	√				

Values

member is the name of the secondary output file member.

Chapter 11

Universal Encrypt

11.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Encrypt.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

11.2 Configuration Options List

Table 11.1, below, identifies the Universal Encrypt configuration options.

Option Name	Description	Page
AES	Specification for whether or not AES encryption is used	187
CODE_PAGE	Character code page used to translate text data received and encrypted	188
ENCRYPTION_KEY	Encryption key used by the encryption algorithm	189
HELP	Writes a description of the command options and their format	190
INPUT_FILE	Input file that is to be encrypted	191
NLS_DIRECTORY	Directory name where Universal Encrypt can find its code page tables	192
OUTPUT_FILE	File to which the encrypted input file is written	193
VERSION	Writes the program version and copyright information	194

Table 11.1 Universal Encrypt Configuration Options

11.3 AES

Description

The AES option specifies whether or not AES encryption is used.

Note: Although the default - for backward compatibility - is not to use AES encryption, the use of AES encryption is highly recommended.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-a <i>option</i>			✓	✓	✓
Command Line Option, Long Form	-aes <i>option</i>			✓	✓	✓
STRUEN Parameter	AES(<i>*option</i>)	✓				

Values

option is the specification for whether or not to use AES encryption.

Valid values for *option* are:

- **yes**
Use AES 256-bit encryption in CBC mode.
- **no**
Use DES 56-bit encryption.

[Default is NO.]

11.4 CODE_PAGE

Description

The CODE_PAGE option specifies the character code page used to translate text data.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>codepage</i>			✓	✓	✓
Command Line Option, Long Form	-codepage <i>codepage</i>			✓	✓	✓
STRUEN Parameter	CODEPAGE(<i>codepage</i>)	✓				

Values

codepage is the character code page that is used to translate data.

codepage references a Universal Translate Table (UTT) file provided with the product (see Section [20.5 UTT Files](#) for information on UTT files). UTT files are used to translate between Unicode and the local single-byte code page. (All UTT files end with an extension of `.utt`.)

[Default

The default code page is different for different operating systems:

- ISO8859-1 (8-bit ASCII) ASCII-based operating systems
- IBM1047 (EBCDIC) EBCDIC-based operating system]

See Section [20.4 Character Code Pages](#) for a complete list of character code pages provided by Stonebranch Inc. for use with Universal Products.

11.5 ENCRYPTION_KEY

Description

The ENCRYPTION_KEY option specifies the key used to encrypt the command file.

Note: The key specified by this ENCRYPTION_KEY option also must be provided to the Universal Product command for which the command file is intended.

If this option is not used, Universal Encrypt uses a default 8-byte key.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-K <i>key</i>		✓	✓	✓	✓
Command Line Option, Long Form	-key <i>key</i>		✓	✓	✓	✓
STRUEN Parameter	KEY(<i>key</i>)	✓				

Values

key is the key used to encrypt the command file.

key can contain from 1 to 32 characters long. However, it is recommended that *key* contain at least 8 characters.

HP NonStop

key contains from 1 to 8 characters. If less than eight characters are provided, *key* is padded on the right with null characters (binary zeros). It is recommended that all eight bytes be provided.

11.6 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h		✓	✓	✓	✓
Command Line Option, Long Form	-help		✓	✓	✓	✓
STRUEN Parameter	n/a					

Values

(There are no values used with this option.)

11.7 INPUT_FILE

Description

The INPUT_FILE option specifies the input file that is to be encrypted.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUEN Parameter	INFILE(<i>input_file</i>) [INMBR (<i>member</i>)]	√				

Values

input_file is the name of the file to be encrypted.

Valid values for *input_file* are:

- **STDIN*
Input file is read from standard input. Standard input is allocated to the workstation for interactive jobs and to file **QINLINE** for batch jobs. If executed as an interactive job, an ILE session terminal is displayed, from which input can be entered at the terminal.
- *file_name*
Input file is read from the specified file.

file_name can be qualified by a library name. Otherwise, library list ***LIBL** is searched for the first occurrence of the file name.

A member name can be used for further qualification by specifying the **INMBR** parameter.

11.8 NLS_DIRECTORY

Description

The NLS_DIRECTORY option specifies the name of the directory where the code page UTT files are located.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-nlsdir <i>directory</i>			✓	✓	
STRUEN Parameter	n/a					

Values

directory is the name of the directory.

Relative path names are relative to the installation directory. Full path names are recommended.

11.9 OUTPUT_FILE

Description

The OUTPUT_FILE option specifies the file to which the encrypted input file is written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
STRUEN Parameter	OUTFILE(<i>output_file</i>) [OUTMBR(<i>member</i>)]	√				

Values

output_file is the name of the file which the input file is written.

Valid values for *output_file* are:

- **STDOUT*
Encrypted input file is written to stdout. Standard output is allocated to the terminal if STRUEN is executed interactively. The ILE session terminal is displayed to view the output. Standard output is allocated to file **QPRINT** if STRUEN is executed in batch.
- *file_name*
Encrypted input file is written to the specified file.

file_name can be qualified by a library name. Otherwise, the library list ***LIBL** is searched for the first occurrence of the file name. If the file is not found, it is created as a physical source file with a record length of 266. If ***LIBL** is specified or implied, the file is created in **QGPL**.

A member name can be used for further qualification by specifying the **OUTMBR** parameter.

11.10 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v		✓	✓	✓	✓
Command Line Option, Long Form	-version		✓	✓	✓	✓
STRUEN Parameter	VERSION(<i>option</i>)	✓				

Values

HP NonStop, UNIX, Windows, and z/OS

There are no values for this option.

OS/400

Valid values for *option* are:

- **yes**
Write program version information and copyright.
- **no**
Do not write program version information and copyright.

[Default is no.]

Chapter 12

Universal Event Log Dump

12.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Event Log Dump.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

12.2 Configuration Options List

Table 12.1 identifies the Universal Event Log Dump configuration options.

Option Name	Description	Page
BACKUP_LOG	Causes the log to be backed up before it is cleared.	197
CLEAR_LOG	Causes the records in the log to be deleted from the log	198
END_TIME	Ending date and time	199
HELP	Writes a description of command options and their format	200
LOG_DIRECTORY	Log file directory	202
LOG_TYPE	Event log to be dumped	203
MESSAGE_DESTINATION	Location where messages are printed.	204
MESSAGE_LANGUAGE	Universal Message Catalog (UMC) file that will be used to write messages.	205
MESSAGE_LEVEL	Level of messages that will be displayed	206
INSTALLATION_DIRECTORY	Base directory where product is installed	201
NLS_DIRECTORY	UMC and UTT file directory	207
OUTPUT_FILE	Complete path to the file that will be used to store the selected event log records	208
PAGE_HEADER	Enables or disables the printing of page headers	209
PAGE_LENGTH	Number of lines that should be printed on each page.	210
REMOTE_SERVER	Name of a remote computer from which event log records should be retrieved	211
START_TIME	Starting date and time	212
VERSION	Writes the program version and copyright information	213

Table 12.1 Universal Event Log Dump - Configuration Options

12.3 BACKUP_LOG

Description

The BACKUP_LOG option causes the log (the System event log or the log specified by the LOG_TYPE option) to be backed up before it is cleared via the CLEAR_LOG option. BACKUP_LOG is valid only when used with CLEAR_LOG.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-b <i>filename</i>				√	
Command Line Option, Long Form	-backup <i>filename</i>				√	
Configuration File Keyword	n/a					

Values

filename is the name of the log to be backed up.

12.4 CLEAR_LOG

Description

The CLEAR_LOG option causes the records in the log (the System event log or the log specified by the LOG_TYPE option) to be deleted from the log.

When CLEAR_LOG used with other options (except the BACKUP_LOG option), the log first is dumped, then cleared.

Note: Administrator access is required to dump any of the event logs.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-c				√	
Command Line Option, Long Form	-clear				√	
Configuration File Keyword	n/a					

Values

(There are no values for this option.)

12.5 END_TIME

Description

The END_TIME option specifies the ending date and time of records to be dumped.

If END_TIME is not used, Universal Event Log Dump will select records up to and including the last (that is, the most recent) record in the log specified by the [LOG_TYPE](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-e <i>enddate</i> [, <i>endtime</i>]				√	
Command Line Option, Long Form	-etime <i>enddate</i> [, <i>endtime</i>]				√	
Configuration File Keyword	n/a					

Values

enddate is the ending date of records to be dumped.

Event log records for the current date can be selected by using an asterisk (*).

If Event Log Dump will run over consecutive days, a rolling date can be specified with an asterisk (*) followed by a negative value. For example, *-2 selects records that were generated prior to 2 days before the current date.

endtime, optionally, specifies the ending time of records to be dumped.

If *endtime* is omitted, a value of 23:59 is used.

Note: *enddate* and *endtime* must match the short date and short time styles, respectively, as specified in the Control Panel.

If a 12-hour time format is used, and *endtime* is specified, *enddate* and *endtime* together must be enclosed in double (") quotation marks (for example, "12/31/1999, 11:59 PM"). This ensures that the value will be read correctly from the command line.

12.6 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h				√	
Command Line Option, Long Form	-help				√	
Configuration File Keyword	n/a					

Values

(There are no values for this option.)

12.7 INSTALLATION_DIRECTORY

Description

The INSTALLATION_DIRECTORY option specifies the Universal Event Log Dump base installation directory

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Configuration File Keyword	installation_directory <i>directory</i>				√	

Value

directory is the name of the Universal Event Log Dump base installation directory.

Specify a full path name.

12.8 LOG_DIRECTORY

Description

The LOG_DIRECTORY option specifies the directory name where log files are created. Log file creation is specified by the [MESSAGE_DESTINATION](#) option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Configuration File Keyword	log_directory <i>directory</i>				√	

Value

directory is the directory where log files are created.

12.9 LOG_TYPE

Description

The LOG_TYPE option specifies the event log to be dumped.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>logtype</i>				√	
Command Line Option, Long Form	-logtype <i>logtype</i>				√	
Configuration File Keyword	<i>logtype logtype</i>				√	

Values

logtype is the event log to be dumped.

Valid values for *logtype* are:

- **system**
Dumps records from the System event log.
- **application**
Dumps records from the Application event log.
- **security**
Dumps records from the Security event log.
Note:Administrator access is required for this dump.

[Default is system.]

12.10 MESSAGE_DESTINATION

Description

The MESSAGE_DESTINATION option specifies the location where messages are to be written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-d <i>destination</i>				√	
Command Line Option, Long Form	-dest <i>destination</i>				√	
Configuration File Keyword	message_dest <i>destination</i>				√	

Values

destination is the location where messages are to be written.

Valid values for destination are:

- **stderr**
Writes the messages to the console.
- **logfile**
Write the messages to a log file. The log file location is located beneath the `ue1d` folder, in the log folder. The current log file name is `unv.1log`.
Past generation log files are named `unvNNNN.1log`, where `NNNN` equals the generation number. Currently, five generations are kept.

[Default value is stderr.]

12.11 MESSAGE_LANGUAGE

Description

The MESSAGE_LANGUAGE option specifies the Universal Message Catalog (UMC) file that will be used to write messages.

Each UMC file contains messages for a specific language.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-L <i>language</i>				√	
Command Line Option, Long Form	-lang <i>language</i>				√	
Configuration File Keyword	language <i>language</i>				√	

Values

language is the UMC file that will be used to write messages.

The first three characters of the language name are used as a three-character suffix in the UMC file base name (for example, `ue1mceng.umc`). All UMC files have a `.umc` extension.

[Default is `ue1mceng.umc`.]

12.12 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to be written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>				√	
Command Line Option, Long Form	-level <i>level</i>				√	
Configuration File Keyword	n/a					

Values

level is the level of messages to be written.

Valid values for *level* are:

- **trace**
Writes trace messages used for debugging. The trace file name is `ue1d.trc`. It is created in the directory where the `ue1d` program is located. Use only as directed by Stonebranch Technical Support.

Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning and error messages.
- **info**
Writes informational, warning and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

[Default is warn.]

12.13 NLS_DIRECTORY

Description

The NLS_DIRECTORY option specifies the directory name where the Universal Event Log Dump message catalog and code page tables are located.

Usage

Method	Syntax *	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Configuration File Keyword	nls_directory <i>directory</i>				√	

Values

directory is the name of the directory where the files are located.

12.14 OUTPUT_FILE

Description

The OUTPUT_FILE option specifies the complete path to the file that will be used to store the selected event log records.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-f <i>filename</i>				√	
Command Line Option, Long Form	-file <i>filename</i>				√	
Configuration File Keyword	n/a					

Values

filename is the complete path to the file that will be used to store the selected event log records.

12.15 PAGE_HEADERS

Description

The PAGE_HEADERS option enables or disables the writing of page headers.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-r <i>option</i>				√	
Command Line Option, Long Form	-header <i>option</i>				√	
Configuration File Keyword	header <i>option</i>				√	

Values

option is the specification for enabling or disabling the writing of page headers.

Valid values for *option* are:

- **yes**
Display column headings at the top of each page.
- **no**
Display report columns without headings.

{Default value is no.}

12.16 PAGE_LENGTH

Description

The PAGE_LENGTH option specifies the number of lines (records) to be written on each page.

If PAGE_LENGTH is not used, and [PAGE_HEADERS](#) is set to YES, a header will be written only at the top of the first page (since there is, in effect, only one page).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-n <i>pagelength</i>				√	
Command Line Option, Long Form	-length <i>pagelength</i>				√	
Configuration File Keyword	length <i>pagelength</i>				√	

Values

pagelength is the number of lines (records) to be written on each page.

12.17 REMOTE_SERVER

Description

The REMOTE_SERVER option specifies the name of a remote computer from which event log records should be retrieved.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-S <i>servername</i></code>				√	
Command Line Option, Long Form	<code>-server <i>servername</i></code>				√	
Configuration File Keyword	n/a					

Values

servername is the name of a remote computer from which event log records should be retrieved.

servername must be specified using the Universal Naming Convention (UNC) format, where the computer name is preceded by two back slashes (for example, `\\RMT1`).

12.18 START_TIME

Description

The START_TIME option specifies the starting date and time of records to be dumped.

If START_TIME is not used, Universal Event Log Dump will start with the first (that is, the earliest) record in the log specified by the LOG_TYPE option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-s <i>startdate</i> [, <i>starttime</i>]				√	
Command Line Option, Long Form	-stime <i>startdate</i> [, <i>starttime</i>]				√	
Configuration File Keyword	n/a					

Values

startdate is the starting date of records to be dumped.

Event log records for the current date can be selected by using an asterisk (*).

If Event Log Dump will run over consecutive days, a rolling date can be specified with an asterisk (*) followed by a negative value. For example, *-2 selects records that were generated in the previous 2 days.

starttime, optionally, specifies the starting time of records to be dumped.

If *starttime* is omitted, a value of 00:00 (midnight) is used.

Note: *enddate* and *endtime* must match the short date and short time styles, respectively, as specified in the Control Panel.

If a 12-hour time format is used, and *endtime* is specified, *enddate* and *endtime* together must be enclosed in double (") quotation marks (for example, "01/01/2000, 12:00 AM"). This ensures that the value will be read correctly from the command line.

12.19 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v				√	
Command Line Option, Long Form	-version				√	
Configuration File Keyword	n/a					

Values

(There are no values for this option.)

Chapter 13

Universal Message Translator

13.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Message Translator.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

13.2 Configuration Options List

Table 13.1 identifies the Universal Message Translator configuration options.

Option Name	Description	Page
HELP	Writes a description of the command options and their format	216
IGNORE_CASE	Specification that matching of message masks to the input file is not case sensitive	217
MESSAGE_FILE	Input message file name	218
MESSAGE_LEVEL	Level of messages that will be displayed	219
TRANSLATION_TABLE	Translation table file name	220
VERSION	Writes the program version and copyright information	220

Table 13.1 Universal Message Translator Configuration Options

13.3 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h		✓	✓	✓	✓
Command Line Option, Long Form	-help		✓	✓	✓	✓
STRUME Parameter	n/a					

Values

(There are no values for this option.)

13.4 IGNORE_CASE

Description

The IGNORE_CASE option specifies that the matching of message masks to the input file is not case sensitive.

(For example, if IGNORE_CASE is used, the word **Error** matches **ERROR**).

If this option is not used, the matching of message masks is case sensitive.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-i	✓	✓	✓	✓	✓
Command Line Option, Long Form	-ignorecase	✓	✓	✓	✓	✓
STRUME Parameter	IGNORECASE(* <i>option</i>)	✓				

Values

HP NonStop, UNIX, Windows, and z/OS

There are no values for this option.

OS/400

Valid values for *option* are:

- **yes**
Write program version information and copyright.
- **no**
Do not write program version information and copyright.

[Default is no.]

13.5 MESSAGE_FILE

Description

The MESSAGE_FILE option specifies the name of the input message file.

Note: If this option is not used, Universal Message Translator reads its input from standard input.

z/OS

If this option is not used, Universal Message Translator reads the input file from SYSIN ddname.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-f <i>messages</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-file <i>messages</i>	✓	✓	✓	✓	✓
STRUME Parameter	MSGFILE(<i>messages</i> [<i>library</i>]) [MSGMBR(<i>member</i>)]	✓				

Values

messages is the name of the input message file.

z/OS

messages is the ddname to which the input message file is allocated.

OS/400

Valid values for *messages* are:

- *stdin**
 Standard input is used to specify the messages. Standard input is allocated to the workstation for interactive jobs and to file **QINLINE** for batch jobs. If executed as an interactive job, an ILE session terminal is displayed from which input can be entered at the terminal.
 This value is not valid for the command line form. Simply do not specify the command line option to read from standard input.
- filename**
 Name of a file. The file name can be qualified by a library name. If not, the library list ***LIBL** is searched for the first occurrence of the file name.
 A member name can be used for further qualification by specifying the **MSGMBR** parameter.

13.6 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>	√	√	√	√	√
Command Line Option, Long Form	-level <i>level</i>	√	√	√	√	√
STRUME Parameter	MSGLEVEL (<i>*level</i>)	√		√	√	√

Values

level indicates either of the following level of messages:

- **verbose**
Writes messages that provide information on the message matching process. The messages are helpful for verifying or debugging a translation table. (Information, warning, and error messages also are printed.)
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

OS/400 and z/OS

[Default is info.]

HP NonStop, UNIX, and Windows

[Default is warn.]

13.7 TRANSLATION_TABLE

Description

The TRANSLATION_TABLE option specifies the name of the translation table file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>table</i>	√	√	√	√	√
Command Line Option, Long Form	-table <i>table</i>	√	√	√	√	√
STRUME Parameter	TBL(<i>table</i> [<i>library</i>]) [TBLMBR(<i>member</i>)]	√				

Values

table is the name of the translation table file.

z/OS

table is the ddname to which the translation table is allocated.

OS/400

Valid values for *table* are:

- **umetbl**
Default file name for the STRUME parameter.
The command line form does not default. An error is generated if no value is specified.
- *filename*
Translation table is read from a file. The file name can be qualified by a library name. Otherwise, the library list *LIBL is searched for the first occurrence of the file name.
A member name can be used for further qualification by specifying the TBLMBR parameter.

13.8 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v		✓	✓	✓	✓
Command Line Option, Long Form	-version		✓	✓	✓	✓
STRUME Parameter	n/a					

Values

(There are no values for this option.)

Chapter 14

Universal Products Install Merge

14.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Products Install Merge (UPIMERGE).

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

14.2 Configuration Options List

Table 14.1, below, identifies the Universal Products Install Merge configuration options.

Option Name	Description	Page
BACKUP_DESTINATION	Creates a copy of the original DESTINATION_FILE prior to the merge.	224
COMMAND_FILE_ENCRYPTED	Name of a file encrypted with Universal Encrypt that contains command options.	226
COMMAND_FILE_PLAIN	Name of a plain text file that contains command options.	227
COMPONENT_TYPE	Notifies UPIMERGE that the SOURCE_FILE is a component definition file that contains settings for the specified Universal server component. You cannot use this option with CONFIGURATION_TYPE . UPIMERGE ignores this option if INSTALLATION_DIRECTORY is omitted.	228
CONFIGURATION_TYPE	Notifies UPIMERGE that the SOURCE_FILE is a configuration file that contains settings for the specified Universal Products application. You cannot use this option with COMPONENT_TYPE . UPIMERGE ignores this option if INSTALLATION_DIRECTORY is omitted.	230
DESTINATION_FILE	Name of a file used to store the result of the merge.	232
ENCRYPTION_KEY	Key used to encrypt the file specified by COMMAND_FILE_ENCRYPTED .	233
HELP	Writes a description of the command options and their format.	234
INSTALLATION_DIRECTORY	Primary location in which the Universal Products server component identified with COMPONENT_TYPE or the Universal Products application identified by CONFIGURATION_TYPE resides.	235
KEEP_NOMATCH	Controls merge behavior when an option in DESTINATION_FILE has no match in SOURCE_FILE .	236
MESSAGE_LEVEL	Level of messages to write.	238
SOURCE_FILE	Name of a file used as input to the merge. If this parameter is omitted, UPIMERGE assumes input is redirected via stdin.	239
VERSION	Writes the program version and copyright information.	241

Table 14.1 Universal Products Install Merge Configuration Options

14.3 BACKUP_DESTINATION

Description

The BACKUP_DESTINATION option instructs UPIMERGE to create a copy of the original configuration or component definition file specified by DESTINATION_FILE before merging the contents of SOURCE_FILE into it.

UPIMERGE creates the backup file in the same directory as the original configuration or component definition file.

Windows

UPIMERGE generates a file name with a format of xxxnnnnn.tmp, where:

- xxx is a prefix based on the name of the Universal Products configuration or component definition file specified for DESTINATION_FILE. (UPIMERGE uses a default of uim if it does not recognize the file name.)
- nnnn is a unique alpha-numeric identifier.

UNIX

The format of the file name UPIMERGE generates is implementation-dependent.

On most systems, the format is xxxnnnnnn, where

- xxx is a prefix based on the name of the Universal Products configuration or component definition file specified for DESTINATION_FILE. (UPIMERGE uses a default of uim if it does not recognize the file name.)
- nnnnnn is a unique alpha-numeric identifier.

On some older UNIX systems, the format is undefined, but the name is still unique.

UPIMERGE reports the backup file it generates with message ID UNV4412I.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-bkup_dest option			✓	✓	
Environment Variable	UIMBKUPDEST=option			✓	✓	

Values

option specifies whether UPIMERGE creates a copy of the original configuration or component definition file.

Valid values for *option* are:

- **yes**
Create a backup of the file specified with `DESTINATION_FILE` before merging the contents of `SOURCE_FILE` into it.
- **no**
Merge the contents of `SOURCE_FILE` into `DESTINATION_FILE`, without saving a copy of the original `DESTINATION_FILE`.

[Default is no.]

14.4 COMMAND_FILE_ENCRYPTED

Description

The `COMMAND_FILE_ENCRYPTED` option specifies the file containing encrypted values for command line option parameters.

Command files specify an additional source of command line options. Storing options in a file is useful in situations where it is not desirable to specify them on the command line. The application reads the file and processes the options exactly like those specified on the command line. The options must be in their respective command line formats.

UPIMERGE can process encrypted or plain text command files (see the [COMMAND_FILE_PLAIN](#) option). Encrypted command files are an excellent place to store sensitive data that you may want to hide from the command line. As an added measure of security, Stonebranch, Inc. recommends that you configure the file system's security to protect any command file that contains sensitive data from unauthorized read access.

Use the Universal Encrypt utility provided with Universal Products to encrypt a plain text command file. (For information on Universal Encrypt, see the Universal Products Utilities 4.1.0 User Guide). If Universal Encrypt used a key to encrypt the file, use that same key to read the file using the [ENCRYPTION_KEY](#) option.

Note: If UPIMERGE encounters the `COMMAND_FILE_ENCRYPTED` and [COMMAND_FILE_PLAIN](#) options on its command line, it uses the file specified for [COMMAND_FILE_PLAIN](#).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-x <i>filename</i>			✓	✓	
Command Line Option, Long Form	-encryptedfile <i>filename</i>			✓	✓	
Environment Variable	n/a					

Values

filename is the name of the encrypted file that contains the command line options and their values.

14.5 COMMAND_FILE_PLAIN

Description

The `COMMAND_FILE_PLAIN` option specifies the file containing plain text values for command line option parameters.

Command files specify an additional source of command line options. Storing options in a file is useful in situations where it is not desirable to specify them on the command line. The application reads the file and processes the options exactly like those specified on the command line. The options must be in their respective command line formats.

UPIMERGE can process encrypted or plain text command files (see the [COMMAND_FILE_ENCRYPTED](#) option). Stonebranch, Inc. recommends that you configure the file system's security to protect command files from unauthorized access.

Note: If UPIMERGE encounters the [COMMAND_FILE_ENCRYPTED](#) and `COMMAND_FILE_PLAIN` options on its command line, it uses the file specified for `COMMAND_FILE_PLAIN`.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-f filename</code>			✓	✓	
Command Line Option, Long Form	<code>-file filename</code>			✓	✓	
Environment Variable	n/a					

Values

filename is the name of the file that contains the command line options and their values.

14.6 COMPONENT_TYPE

Description

The COMPONENT_TYPE option notifies UPIMERGE that [SOURCE_FILE](#) is a component definition file that contains settings for the specified Universal server component.

UPIMERGE uses this value together with the [INSTALLATION_DIRECTORY](#) value to set options in a component definition file that depend on the Universal Products server component's installed location.

Although COMPONENT_TYPE is not required to merge component definition files, using it ensures that UPIMERGE executes any custom merge logic necessary for a particular component's definition file.

Note: You cannot use COMPONENT_TYPE with the [CONFIGURATION_TYPE](#) option.

UPIMERGE ignores COMPONENT_TYPE if the [INSTALLATION_DIRECTORY](#) is omitted.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-comptype <i>type</i>			✓	✓	
Environment Variable	UIMCOMPTYPE= <i>type</i>			✓	✓	

Values

type identifies a Universal Products server component.

Valid values for *type*, and the Universal Products server component that each value represents, are:

- uac Universal Application Container Server
- ucmd Universal Command Server
- uctl Universal Control Server
- udm Universal Data Mover Server
- uemd Universal Event Monitor Server (demand-driven)
- uems Universal Event Monitor Server (event-driven)

[There is no default.]

14.7 CONFIGURATION_TYPE

Description

The CONFIGURATION_TYPE option notifies UPIMERGE that the file specified with the [SOURCE_FILE](#) option is a configuration file that contains settings for the specified Universal Products application.

UPIMERGE uses this value together with the [INSTALLATION_DIRECTORY](#) value to set options in a configuration file that depend on the Universal Product application's installed location.

Although CONFIGURATION_TYPE is not required to merge configuration files, using it ensures that UPIMERGE executes any custom merge logic necessary for a particular application's configuration file.

Note: You cannot use CONFIGURATION_TYPE with the [COMPONENT_TYPE](#) option.

UPIMERGE ignores CONFIGURATION_TYPE if the [INSTALLATION_DIRECTORY](#) option is omitted.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-cfgtype <i>type</i>			✓	✓	
Environment Variable	UIMCFGTYPE= <i>type</i>			✓	✓	

Values

type specifies the Universal Product configuration file to merge.

Valid values for *type*, and the Universal Product application that each value represents, are:

- `uac` Universal Access Control List
- `uacs` Universal Application Container Server
- `ubroker` Universal Broker
- `ucmd` Universal Command Manager
- `ucmds` Universal Command Server
- `uctl` Universal Control Manager
- `uctls` Universal Control Server
- `udm` Universal Data Mover Manager
- `udms` Universal Data Mover Server
- `uec` Universal Enterprise Controller
- `ueld` Universal Event Log Dump Utility
- `uem` Universal Event Monitor Manager
- `uems` Universal Event Monitor Server
- `uquery` Universal Query

14.8 DESTINATION_FILE

Description

The DESTINATION_FILE option identifies the configuration or component definition file that UPIMERGE uses to store the results of the merge with SOURCE_FILE.

During the merge process, UPIMERGE replaces all values in DESTINATION_FILE that have a match in SOURCE_FILE. If SOURCE_FILE contains configuration or component definition options not defined in DESTINATION_FILE, UPIMERGE adds those options to the end of the output file.

UPIMERGE does not attempt to sequence the entries it adds to DESTINATION_FILE, which means the order of options in the output file may not match the order of the same options specified in SOURCE_FILE. In addition, UPIMERGE does not update any comments in the output file. Finally, if the input file contains more than one entry for a given option, UPIMERGE adds every occurrence of that entry to DESTINATION_FILE. The application uses the value of the last entry that appears in the output file.

Options that reside only in DESTINATION_FILE are either commented out or left as-is, depending on the value of the KEEP_NOMATCH option.

The file name specified for DESTINATION_FILE must exist (even if it is empty) before executing UPIMERGE.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-dest <i>filename</i>			✓	✓	
Environment Variable	UIMDESTFILE= <i>filename</i>			✓	✓	

Values

filename is the name of the file that contains the results of the merge with SOURCE_FILE.

filename can contain a complete path to the output file or a path relative to the current directory.

14.9 ENCRYPTION_KEY

Description

The ENCRYPTION_KEY option specifies the key used to encrypt the file specified with the [COMMAND_FILE_ENCRYPTED](#) option.

This key acts much like a password, preventing unauthorized users from decrypting the encrypted command file.

If Universal Encrypt used a key to encrypt the file, UPIMERGE requires the same key to decrypt the file. (For information on Universal Encrypt, see the Universal Products Utilities 4.1.0 User Guide).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-K <i>key</i>			✓	✓	
Command Line Option, Long Form	-key <i>key</i>			✓	✓	
Environment Variable	UIMKEY= <i>key</i>			✓	✓	

Values

key is the key used to encrypt the command file.

14.10 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h			✓	✓	
Command Line Option, Long Form	-help			✓	✓	
Environment Variable	n/a					

Values

(There are no values used with this option.)

14.11 INSTALLATION_DIRECTORY

Description

The `INSTALLATION_DIRECTORY` option specifies the installed location of the Universal Product application or server component specified with the `CONFIGURATION_TYPE` or `COMPONENT_TYPE` option, respectively.

`INSTALLATION_DIRECTORY` is a command line-only parameter that `UPIMERGE` uses to set values for options in `DESTINATION_FILE` that depend upon the installed location of a specified Universal Products application or server component.

It does NOT specify the installed location of the `UPIMERGE` utility.

Note: `INSTALLATION_DIRECTORY` requires the `CONFIGURATION_TYPE` or the `COMPONENT_TYPE` option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-installdir <i>dirname</i>			✓	✓	
Environment Variable	UIMINSTALLDIR= <i>dirname</i>			✓	✓	

Values

dirname is the primary directory into which the Universal Products application or server component specified by `CONFIGURATION_TYPE` or `COMPONENT_TYPE` resides.

14.12 KEEP_NOMATCH

Description

The KEEP_NOMATCH option instructs UPIMERGE to comment out any option in [DESTINATION_FILE](#) that does not exist in [SOURCE_FILE](#).

The primary purpose of KEEP_NOMATCH is to prevent the introduction of new values for existing configuration options during a product upgrade.

For example, the typical sequence of events for an upgrade is as follows:

1. The install program archives a product's current configuration file to a well-known location.
2. The install program places a new configuration file from the distribution package into the well-known Universal Products configuration file directory.
3. The install program passes the name of the archive file to UPIMERGE as [SOURCE_FILE](#).
4. The install program passes the name of the newly-installed configuration file to UPIMERGE as [DESTINATION_FILE](#).

If the archive file contains no value for a given configuration option – [MESSAGE_LEVEL](#), for example – that means the corresponding program currently runs with the application-defined default. If the newly-installed configuration file were to contain an entry that sets [MESSAGE_LEVEL](#) to **audit**, and UPIMERGE kept this option by default, the result would be a change to the application's behavior (assuming the application-defined default for the option is something other than **audit**).

Note: As an additional precaution against a product upgrade changing an application's behavior, packaged configuration files have most options commented out. This approach allows product upgrades to "announce" the availability of new options while preserving an existing configuration.

On the other hand, a situation may arise after the install where it is desirable to introduce new options and/or values into a product's configuration. In this case, simply execute UPIMERGE with KEEP_NOMATCH set to **yes**.

While the discussion above focused on configuration files, UPIMERGE uses the same approach with component definition files.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-keep_nomatch <i>option</i>			✓	✓	
Environment Variable	UIMKEEPNOMATCH= <i>option</i>			✓	✓	

Values

option specifies whether UPIMERGE should "keep" or comment out any setting contained in [DESTINATION_FILE](#) that has no match in [SOURCE_FILE](#).

Valid values for *option* are:

- **yes**
Keep any option in [DESTINATION_FILE](#) without a match in [SOURCE_FILE](#).
- **no**
Comment out any option in [DESTINATION_FILE](#) with no match in [SOURCE_FILE](#).

[Default value is no.]

14.13 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>			✓	✓	
Command Line Option, Long Form	-level <i>level</i>			✓	✓	
Environment Variable	UIMLEVEL= <i>level</i>			✓	✓	

Values

level indicates either of the following level of messages:

- **trace**
Writes trace messages used for diagnostic purposes.
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

14.14 SOURCE_FILE

Description

The SOURCE_FILE option identifies the file that UPIMERGE merges into the configuration or component definition file specified by the DESTINATION_FILE option.

During the merge process, UPIMERGE replaces all values in DESTINATION_FILE that have a match in SOURCE_FILE.

If SOURCE_FILE contains configuration or component definition options not defined in DESTINATION_FILE, UPIMERGE adds those options to the end of the output file. If SOURCE_FILE contains more than one entry for a given option, UPIMERGE adds every occurrence of that entry to DESTINATION_FILE. The application uses the value of the last entry that appears in the output file.

Use of SOURCE_FILE is optional; it also is possible to provide input options and values to UPIMERGE using standard input (that is, stdin) redirection.

Figure 14.1, below, illustrates an example of stdin redirection:

```
upimerge -dest outfile.txt <infile.txt
```

Figure 14.1 Using stdin redirection with UPIMERGE

Note: It is possible to execute UPIMERGE without the redirected input file. However, the program may appear unresponsive. In this situation, UPIMERGE is actually waiting for an end-of-file indicator to signal the end of the redirected input.

Windows

Supply the end-of-file indicator by pressing <Ctrl+Z> <Enter>.

UNIX

Supply the end-of-file indicator by pressing <Ctrl+D>.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-source <i>filename</i>			✓	✓	
Environment Variable	UIMSOURCE= <i>filename</i>			✓	✓	

Values

filename is the name of the file that contains options and values that UPIMERGE merges into [DESTINATION_FILE](#).

filename can contain a complete path to the input file or a path relative to the current directory.

14.15 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v			✓	✓	
Command Line Option, Long Form	-version			✓	✓	
Environment Variable	n/a					

Values

(There are no values for this option.)

Chapter 15

Universal Query

15.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Query.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

15.2 Configuration Options List

Table 15.1 identifies the Universal Query configuration options.

Option Name	Description	Page
BIF_DIRECTORY	Broker Interface Directory that specifies the location of the Universal Broker interface file.	244
COMMAND_ID	Requests that Universal Query return information for all records that match the specified command ID.	245
COMPONENT_ID	Requests that Universal Query return information only for the specified component ID.	245
CODE_PAGE	Character code page used to translate text data received and transmitted over the network.	245
HELP	Writes a description of the command options and their format.	248
MANAGERS	Specification for whether or not Universal Query requests manager component information from the queried Broker.	249
MESSAGE_LANGUAGE	Universal Message Catalog (UMC) file used to write messages.	250
MESSAGE_LEVEL	Level of messages to write.	251
NLS_DIRECTORY	Directory where the Universal Query message catalog and code page tables are located.	253
OUTBOUND_IP	Sets the host or IP address that UQUERY binds to when initiating outgoing connections.	254
PING	Information Universal Query requests from Universal Broker.	255
PLF_DIRECTORY	Program Lock File directory that specifies the location of the Universal Query program lock file.	256
REMOTE_HOST	IP address of the remote computer.	257
REMOTE_PORT	TCP port number on the remote computer on which Universal Broker is accepting connections.	258
REPORT	Format in which Universal Broker information is printed.	259
SYSTEM_ID	Local Universal Broker with which the Universal Query must register.	260
VERSION	Writes the program version and copyright information.	259

Table 15.1 Universal Query Configuration Options

15.3 BIF_DIRECTORY

Description

The BIF_DIRECTORY option specifies the Broker Interface File (BIF) directory where the Universal Broker interface file, `ubroker.bif`, is located.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	<code>-bif_directory directory</code>			✓		
Environment Variable	<code>UQRYBIFDIRECTORY=directory</code>			✓		
Configuration File Keyword	n/a					
STRUQR Parameter	n/a					

Values

directory is the name of the BIF directory.

[Default is `/var/opt/universal1`.]

15.4 CODE_PAGE

Description

The CODE_PAGE option specifies the character code page used to translate text data.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-t <i>codepage</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-codepage <i>codepage</i>	✓	✓	✓	✓	✓
Environment Variable	UQRYCODEPAGE= <i>codepage</i>	✓	✓	✓	✓	
Configuration File Keyword	codepage <i>codepage</i>	✓	✓	✓	✓	✓
STRUQR Parameter	CODEPAGE(<i>codepage</i>)	✓				

Values

codepage is the character code page that is used to translate data.

codepage references a Universal Translate Table (UTT) file provided with the product . UTT files are used to translate between Unicode and the local single-byte code page. (All UTT files end with an extension of *.utt*.)

See Section [20.4 Character Code Pages](#) for a complete list of character code pages provided by Stonebranch Inc. for use with Universal Products.

See Section [20.5 UTT Files](#) for information on UTT files.

Default

The default code page is different for different operating systems:

- ISO8859-1 (8-bit ASCII) ASCII-based operating systems.
- IBM1047 (EBCDIC) EBCDIC-based operating system.

15.5 COMMAND_ID

Description

The COMMAND_ID option specifies a command ID that should be used by Universal Query when searching for component records.

When COMMAND_ID is used, Universal Query will return Broker-specific information and component information for all records that match the specified command ID.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-C <i>ID</i>			✓	✓	✓
Command Line Option, Long Form	-cmdid <i>ID</i>			✓	✓	✓
Environment Variable	UQRYCMDID= <i>ID</i>	✓		✓	✓	✓
Configuration File Keyword	n/a					
STRUQR Parameter	CMDID(<i>ID</i>)	✓				

Values

ID is the command ID used by Universal Query.

[There is no default.]

15.6 COMPONENT_ID

Description

The COMPONENT_ID option specifies a component ID that should be used by Universal Query when searching for component records.

When COMPONENT_ID is used, Universal Query will return Broker-specific information and component information only for the specified component ID.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-c <i>ID</i>			✓	✓	✓
Command Line Option, Long Form	-component <i>ID</i>			✓	✓	✓
Environment Variable	UQRYCOMPONENT= <i>ID</i>	✓		✓	✓	✓
Configuration File Keyword	n/a					
STRUQR Parameter	COMPID(<i>ID</i>)	✓				

Values

ID is the component ID used by Universal Query.

[There is no default.]

15.7 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h		✓	✓	✓	✓
Command Line Option, Long Form	-help		✓	✓	✓	✓
Environment Variable	n/a					
Configuration File Keyword	n/a					
STRUQR Parameter	n/a					

Values

(There are no values for this option.)

15.8 MANAGERS

Description

The MANAGERS option specifies whether or not Universal Query requests manager component information from the Broker being queried.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-managers <i>option</i>			✓	✓	✓
Environment Variable	UQRYMANAGERS= <i>option</i>	✓		✓	✓	✓
Configuration File Keyword	managers <i>option</i>	✓		✓	✓	✓
STRUQR Parameter	MANAGERS(* <i>option</i>)	✓				

Values

option is the specification for whether or not Universal Query requests manager component information.

Valid values for *option* are:

- **yes**
Request manager component information.
- **no**
Do not request manager component information.

[Default is yes.]

15.9 MESSAGE_LANGUAGE

Description

The MESSAGE_LANGUAGE option specifies the Universal Message Catalog (UMC) file that will be used to write messages.

Each UMC file contains messages for a specific language.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-L <i>language</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-lang <i>language</i>	✓	✓	✓	✓	✓
Environment Variable	UQRYLANG= <i>language</i>	✓	✓	✓	✓	
Configuration File Keyword	language <i>language</i>	✓	✓	✓	✓	✓
STRUQR Parameter	MSGLANG(<i>language</i>)	✓				

Values

language is the UMC file that will be used to write messages.

The first three characters of the language name are used as a three-character suffix in the UMC file base name (for example, `ue1mceng.umc`). All UMC files have a `.umc` extension.

z/OS

The first three characters of the language name are used as a three-character suffix in the UMC member name. UMC files are read from the partitioned data set allocated on ddname UNVNLS. Universal Query message catalog member names start with UQRMC.

OS/400

The first three characters of the language name are used as a three-character suffix in the UMC member base name `UCMMC`. UMC files are located in the source physical file `UNVPRD410/UNVNLS`.

[Default is `ue1mceng.umc`.]

15.10 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-level <i>level</i>	✓	✓	✓	✓	✓
Environment Variable	UQRYLEVEL= <i>level</i>	✓		✓	✓	
Configuration File Keyword	message_level <i>level</i>	✓		✓	✓	✓
STRUQR Parameter	MSGLEVEL(* <i>level</i>)	✓				

Values

level indicates either of the following level of messages:

- **trace**
Writes trace messages used for diagnostic purposes.
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

OS/400 and z/OS

[Default is info.]

HP NonStop, UNIX, and Windows

[Default is warn.]

Trace Files

UNIX, Windows, and z/OS

Trace file name is **uquery.trc**. It is created in the working directory of Universal Query.

HP NonStop

Trace file name is **UQRYTRC**. It is created in the working subvolume of Universal Query.

OS/400

Trace file name is ***CURLIB/UNVTRCUQR(UQRxxxxxx)**, where **xxxxxx** is the job number of the job invoking Universal Command.

15.11 NLS_DIRECTORY

Description

The NLS_DIRECTORY option specifies the name of the directory where the Universal Query message catalog and code page tables are located.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	n/a					
Environment Variable	n/a					
Configuration File Keyword	nls_directory <i>directory</i>			✓	✓	
STRUQR Parameter	n/a					

Values

directory is the name of the directory where the catalog and tables are located.

Full path names are recommended.

Relative path names are relative to the `universal` installation directory.

Defaults

UNIX

[Default is `/opt/universal/nls.`]

Windows

[Default is `.. \nls.`]

15.12 OUTBOUND_IP

Description

The OUTBOUND_IP option specifies the host or IP address that Universal Query binds to when initiating outgoing connections.

By default, no host or IP address is specified for this option.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a	✓				✓
Command Line Option, Long Form	-outboundip <i>host</i>	✓		✓	✓	✓
Environment Variable	UQRYOUTBOUNDIP= <i>host</i>	✓		✓	✓	
Configuration File Keyword	outboundip <i>host</i>	✓		✓	✓	✓
STRUQR Parameter	OUTBOUNDIP(<i>host</i>)	✓				

Values

host is the host or IP address that Universal Query binds to when initiating outgoing connections.

OS/400

Valid values for *host* are:

- *host*
Host or IP address that Universal Query binds to when initiating outgoing connections.
- (blank line)
No value is used.

15.13 PING

Description

The PING option specifies the type of information that Universal Query requests from Universal Broker.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-P <i>option</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-ping <i>option</i>	✓	✓	✓	✓	✓
Environment Variable	UQRYPING= <i>option</i>	✓	✓	✓	✓	
Configuration File Keyword	ping <i>option</i>	✓	✓	✓	✓	✓
STRUQR Parameter	PING(* <i>option</i>)	✓				

Values

option is the specification for the type of information that Universal Query requests.

Valid values for *option* are:

- **yes**
Universal Broker information only is returned.
- **no**
Universal Broker information and Universal Broker active component information is returned.

[Default is no.]

15.14 PLF_DIRECTORY

Description

The PLF_DIRECTORY option specifies the Program Lock File (PLF) directory where the program lock files are located.

A program lock file is created and used by the Universal Query process to store manager process termination information for the Universal Broker.

OS/400

Do not include this directory in any system or backup that requires an exclusive lock on the directory while Universal Query is running.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-plf_directory <i>directory</i>			✓		
Environment Variable	UQRYPLFDIRECTORY= <i>directory</i>	✓		✓		
Configuration File Keyword	n/a					
STRUQR Parameter	PLFDIR(<i>directory</i>)	✓				

Values

directory is the name of the PLF directory.

A full path name must be specified.

Defaults

UNIX

[Default is /var/opt/universa1/tmp.]

OS/400

[Default is /tmp.]

15.15 REMOTE_HOST

Description

The REMOTE_HOST option specifies the IP address of the remote computer.

Note: The remote computer must have a Universal Broker running and accepting connections.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-i <i>hostaddress</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-host <i>hostaddress</i>	✓	✓	✓	✓	✓
Environment Variable	UQUERYHOST= <i>hostaddress</i>	✓	✓	✓	✓	
Configuration File Keyword	host <i>hostaddress</i>	✓	✓	✓	✓	✓
STRUQR Parameter	HOST(<i>hostaddress</i>)	✓				

Values

hostaddress is the IP address of the remote computer.

The format of *hostaddress* can be either:

- IP address in dotted form (for example, 1.2.3.4)
- Host name (for example, *dallas*).

15.16 REMOTE_PORT

Description

The REMOTE_PORT option specifies the TCP port on the remote computer on which Universal Broker is accepting connections.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-p <i>port</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-port <i>port</i>	✓	✓	✓	✓	✓
Environment Variable	UQRYPORT= <i>port</i>	✓	✓	✓	✓	
Configuration File Keyword	port <i>port</i>	✓	✓	✓	✓	✓
STRUQR Parameter	PORT(<i>port</i>)	✓				

Values

port is the TCP port on which Universal Broker is accepting connections.

The format of *port* can be either:

- Number (for example, 7887)
- Service name (for example, *ubroker*).

[Default is 7887.]

15.17 REPORT

Description

The REPORT option specifies the format in which the Universal Broker information is written.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-R <i>format</i>	✓	✓	✓	✓	✓
Command Line Option, Long Form	-report <i>format</i>	✓	✓	✓	✓	✓
Environment Variable	UQRYREPORT= <i>format</i>	✓	✓	✓	✓	
Configuration File Keyword	report <i>format</i>	✓	✓	✓	✓	✓
STRUQR Parameter	REPORT(* <i>format</i>)	✓				

Values

format is the format in which the Universal Broker information is written.

Valid values for *report* are:

- **normal**
Universal Broker information is written one field per line.
- **fixed**
Universal Broker information is written in a table format with limited number of columns.

[Default is normal.]

15.18 SYSTEM_ID

Description

The SYSTEM_ID option identifies the local Universal Broker with which Universal Query must register before Universal Query performs any request.

Each Universal Broker running on a system is configured with a system identifier that uniquely identifies the Broker.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-system_id <i>ID</i>					√
Environment Variable	UQRYSYSTEMID= <i>ID</i>					√
Configuration File Keyword	n/a					
STRUQR Parameter	n/a					

Values

ID is the system identifier of the local Universal Broker.

(Refer to the local Universal Broker administrator for the appropriate system ID to use.)

15.19 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v	√	√	√	√	√
Command Line Option, Long Form	-version	√	√	√	√	√
Environment Variable	n/a					
Configuration File Keyword	n/a					
STRUQR Parameter	VERSION(* <i>option</i>)	√				

Values

HP NonStop, UNIX, Windows, and z/OS

There are no values for this option.

OS/400

Valid values for *option* are:

- **yes**
Write program version information and copyright.
- **no**
Do not write program version information and copyright.

[Default is no.]

Chapter 16

Universal Spool List

16.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Spool List.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

16.2 Configuration Options List

Table 16.1 identifies the Universal Spool List configuration options.

Option Name	Description	Page
BROKER_SPOOL_DIR	Directory location in which the Universal Broker Component database is located.	264
COMPONENT	Component identifier for which records will be selected to write.	265
HELP	Writes a description of command options and their format.	266
ID	Lists the contents of a specific record from the Universal Event Monitor event definition, event handler, or spool databases.	267
LIST	Type of database from which to select records to write.	268
MESSAGE_LEVEL	Level of messages to be written.	270
MOUNT_POINT	HFS directory in which the HFS databases allocated to ddnames UNVDB and UNVSPool are mounted.	271
UCMD_SPOOL_DIR	Directory location in which the Universal Server Component database is located.	272
VERSION	Writes the program version and copyright information.	272

Table 16.1 Universal Spool List Configuration Options

16.3 BROKER_SPOOL_DIR

Description

The BROKER_SPOOL_DIR option specifies the directory in which the Universal Broker component database (`bcomponent.db`) is located.

If this option is not used to specify the directory, the directory is read from the Universal Broker configuration file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-b <i>directory</i></code>			✓	✓	
Command Line Option, Long Form	<code>-brokerspooldir <i>directory</i></code>			✓	✓	
ULSTSE Parameter	n/a					

Values

directory is the directory in which the Universal Broker component database is located.

16.4 COMPONENT

Description

The COMPONENT option specifies the ID of a single component (job) for which records will be selected to be written.

The [LIST](#) option identifies the database of the component.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-c <i>cid</i>			✓	✓	✓
Command Line Option, Long Form	-component <i>cid</i>			✓	✓	✓
ULSTSE Parameter	COMPONENT(<i>cid</i>)	✓				

Values

cid is the component ID.

16.5 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h			✓	✓	✓
Command Line Option, Long Form	-help			✓	✓	✓
ULSTSE Parameter	n/a					

Values

(There are no values for this option.)

16.6 ID

Description

The ID option specifies the ID of a single record (from the Universal Event Monitor event definition, event handler, or spool databases) to be written.

The [LIST](#) option specifies the spool database.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-i <i>id</i>			✓	✓	
Command Line Option, Long Form	-id <i>id</i>			✓	✓	
ULSTSE Parameter	ID(<i>id</i>)	✓				

Values

id is the ID of a specific record.

- For an event definition or event handler record, *id* is the event ID or event handler ID, respectively.
- For a spool database record, *id* is the sequential serial number automatically assigned to the record.

16.7 LIST

Description

The LIST option specifies the database from which to select records to write.

- Universal Broker
- Universal Command Server
- Universal Event Monitor
- Spool

The **COMPONENT** option is used to select individual component (job) records from the database.

The **ID** option is used to select a single record from a Universal Event Monitor database.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a			✓	✓	✓
Command Line Option, Long Form	-list <i>option</i>			✓	✓	✓
ULSTSE Parameter	LIST(<i>*option</i>)	✓				

Values

option is the database from which to select records.

Valid values for *option* are:

- **ubroker**
List the contents of the Universal Broker Component database. A summary of all records is written.
- **ucmd**
List the contents of the Universal Command Server Component database. A summary of all records is written.
- **ueme** (UNIX and Windows only)
List the contents of the Universal Event Monitor Event Definition database. A summary of all records is written.
- **uemh** (UNIX and Windows only)
List the contents of the Universal Event Monitor Event Handler database. A summary of all records is written. Use the ID option to list the complete contents of a single record.
- **uems** (UNIX and Windows only)
List the contents of the Universal Event Monitor Spool database. A summary of all records is written. Use the ID option to list the complete contents of a single record.
- **urmtcfg**
List the contents of the I-Management Console Remote Configuration database. A summary of all records is written. Use the ID option to list the complete contents of a single record.
- **stdin**
List the standard input spool file for a specified component.
- **stderr**
List the standard error spool file for a specified component.
- **stdout**
List the standard output spool file for a specified component.

[Default is ubroker.]

16.8 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>			✓	✓	✓
Command Line Option, Long Form	-level <i>level</i>			✓	✓	✓
ULSTSE Parameter	LEVEL(* <i>level</i>)	✓				

Values

level indicates either of the following level of messages:

- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

z/OS and OS/400

[Default is info.]

UNIX and Windows

[Default is warn.]

16.9 MOUNT_POINT

Description

The MOUNT_POINT option specifies the HFS directory in which the HFS database allocated to ddnames UNVDB and UNVSPOOL are mounted.

The actual mount points will be subdirectories named after the HFS data set names being mounted.

If the mount points do not exist, they are created by Universal Spool List.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-mount_point <i>directory</i>					√
ULSTSE Parameter	n/a					

Values

directory is the HFS directory in which the HFS databases are mounted.

[Default is /tmp.]

16.10 UCMD_SPOOL_DIR

Description

The UCMD_SPOOL_DIR option specifies the directory in which the Universal Server component database (`scomponent.db`) is located.

If this option is not used to specify the directory, the directory of the Universal Broker component database is used (see the [BROKER_SPOOL_DIR](#) option).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-u <i>directory</i></code>			✓	✓	
Command Line Option, Long Form	<code>-ucmdspooldir <i>directory</i></code>			✓	✓	
ULSTSE Parameter	<code>n/a</code>					

Values

directory is the directory of the Universal Server component database.

16.11 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v			✓	✓	✓
Command Line Option, Long Form	-version			✓	✓	✓
ULSTSE Parameter	VERSION(* <i>option</i>)	✓				

Values

UNIX, Windows, z/OS

(There are no values for this option.)

OS/400

Valid values for *option* are:

- **yes**
Write the program version information and copyright.
- **no**
Do not write the program version information and copyright.

Chapter 17

Universal Spool Remove

17.1 Overview

This chapter provides detailed information on the configuration options available for use with the Universal Spool Remove utility.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

17.2 Configuration Options List

Table 17.1, below, identifies the Universal Spool Remove configuration options.

Option Name	Description	Page
BROKER_SPOOL_DIR	Directory location in which the Universal Broker Component database is located.	276
COMPONENT	Component identifier for which records will be removed.	277
HELP	Writes a description of the command options and their format.	278
MESSAGE_LEVEL	Level of messages that will be written.	279
MOUNT_POINT	HFS directory in which the HFS databases allocated to ddnames UNVDB and UNVSPPOOL are mounted.	280
UCMD_SPOOL_DIR	Directory location in which the Universal Server Component database is located.	281
UEM_SERIALNO	Serial number of the Universal Event Monitor spool database record to remove.	282
VERSION	Writes the program version and copyright information.	283

Table 17.1 Universal Spool Remove Configuration Options

17.3 BROKER_SPOOL_DIR

Description

The BROKER_SPOOL_DIR option specifies the directory in which the Universal Broker component database (`bcomponent.db`) is located.

If this option is not used to specify the directory, the directory is read from the Universal Broker configuration file.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-b <i>directory</i></code>			✓	✓	
Command Line Option, Long Form	<code>-brokerspooldir <i>directory</i></code>			✓	✓	
URMVSE Parameter	<code>n/a</code>					

Values

directory is the directory in which the Universal Broker component database is located.

17.4 COMPONENT

Description

The COMPONENT option specifies the ID of a component for which records will be removed from all databases.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-c cid</code>			✓	✓	✓
Command Line Option, Long Form	<code>-component cid</code>			✓	✓	✓
URMVSE Parameter	COMPONENT (<i>cid</i>)	✓				

Values

cid is the component ID.

17.5 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h			✓	✓	✓
Command Line Option, Long Form	-help			✓	✓	✓
URMVSE Parameter	n/a					

Values

(There are no values for this option.)

17.6 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>			✓	✓	✓
Command Line Option, Long Form	-level <i>level</i>			✓	✓	✓
URMVSE Parameter	LEVEL (<i>*level</i>)	✓				

Values

level indicates either of the following level of messages:

- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

z/OS and OS/400

[Default is info.]

UNIX and Windows

[Default is warn.]

17.7 MOUNT_POINT

Description

The MOUNT_POINT option specifies the HFS directory in which the HFS database allocated to ddnames UNVDB and UNVSPOOL are mounted.

The actual mount points will be subdirectories named after the HFS data set names being mounted.

If the mount points do not exist, they are created by Universal Spool List.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-mount_point <i>dir</i>					√
URMVSE Parameter	n/a					

Values

dir is the HFS directory in which the HFS databases are mounted.

[Default is /tmp.]

17.8 UCMD_SPOOL_DIR

Description

The UCMD_SPOOL_DIR option specifies the directory in which the Universal Server component database (`scomponent.db`) is located.

If this option is not used to specify the directory, the directory of the Universal Broker component database is used (see the [BROKER_SPOOL_DIR](#) option).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	<code>-u dir</code>			✓	✓	
Command Line Option, Long Form	<code>-ucmdspooldir dir</code>			✓	✓	
URMVSE Parameter	<code>n/a</code>					

Values

dir is the directory of the Universal Server component database.

17.9 UEM_SERIALNO

Description

The UEM_SERIALNO option specifies the serial number of the Universal Event Monitor spool database record to remove.

Use the Universal Spool List utility to generate a complete list of all UEM spool records (see [Chapter 16 Universal Spool List](#) in the Universal Products Utilities 4.1.0 User Guide).

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-s <i>serno</i>			✓	✓	
Command Line Option, Long Form	-uem_serialno <i>serno</i>			✓	✓	
URMVSE Parameter	n/a					

Values

serno is the serial number of the database record to remove.

17.10 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v			√	√	√
Command Line Option, Long Form	-version			√	√	√
URMVSE Parameter	VERSION(*option)	√				

Values

UNIX, Windows, z/OS

(There are no values for this option.)

OS/400

Valid values for *option* are:

- **yes**
Write the program version information and copyright.
- **no**
Do not write the program version information and copyright.

Chapter 18

Universal Submit Job

18.1 Overview

This chapter provides detailed information on the configuration options available for use with the Universal Submit Job (USBMJOB) utility.

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

18.2 Configuration Options List

The Universal Submit Job configuration options are separated into two categories:

1. [USBMJOB-Specific Configuration Options](#)
2. [SBMJOB Encapsulated Configuration Options](#)

18.2.1 USBMJOB-Specific Configuration Options

Table 18.1 identifies the USBMJOB-specific configuration options.

Option Name	Description	Page
COMMAND	Command that runs in the submitted batch job.	288
CPYSPLF	Specification for whether or not spooled output files generated by the submitted job be copied to standard output.	288
ENCRYPTED_COMMAND_FILE	Name of an encrypted command file.	288
JOB_LOG_LIBRARY	Library into which the job log will be placed.	289
JOB_STATUS_POLLING_INTERVAL	Number of seconds that USBMJOB will sleep between calls to check the status of the submitted job.	289
REMOTE_MESSAGE_PREFIX	Text string that will prefix any remote messages sent by USBMJOB.	290
REMOTE_REFRESH_INTERVAL	Time that a remote reply message will remain on a remote console without being replied to before it times out.	290
REMOTE_REPLY_COMMAND_PATH	Path (including the executable name) to the remote message handler (uwto).	290
REMOTE_REPLY_HOST	Host name of the remote system on which the uwto command is executed.	291
REMOTE_REPLY_PORT	Port of the Universal Broker on the remote system on which the Universal WTO command is executed.	291
REMOTE_REPLY_USER_ID	User ID for the remote system where the uwto command resides.	291
REMOTE_REPLY_USER_PWD	Password for the user on the remote system where the uwto command resides.	292
SPECIFY_PRINT_CONTROL_CHARS	Print control characters (if any) that are to replace the spooled file's internal print control characters.	292
TRACE	Specification for whether or not trace information will be printed to standard error.	292
USE_REMOTE_REPLY_FACILITY	Specification for whether or not USBMJOB will use the remote reply facility.	293

Table 18.1 Universal Submit Job Configuration Options - USBMJOB-Specific

18.2.2 SBMJOB Encapsulated Configuration Options

The SBMJOB encapsulated configuration options (Table 18.2, below) have a one-to-one relationship with the IBM SBMJOB command parameters of the same name.

Option Name	USBMJOB Parameter
ALLOW DISPLAY BY WRKSBMJOB	DSPSBMJOB({ *yes *no })
CODED CHARACTER SET ID	CCSID({ *current *sysval *usrprf *hex <i>coded_character_set_identifier</i> })
COPY ENVIRONMENT VARIABLES	CPYENVVAR({ *yes *no })
COUNTRY ID	CNTRYID({ *current *sysval *usrprf <i>country_id</i> })
CURRENT LIBRARY	CURLIB({ *current *usrprf *crtdft <i>current_library_name</i> })
HOLD ON JOB QUEUE	HOLD({ *jobd *no *yes })
INITIAL LIBRARY LIST	INLLIBL({ *current *jobd *sysval *none <i>library_name...</i> })
INLASPGRP	INLASPGRP({ *current *jobd *none })
INQUIRY MESSAGE REPLY	INQMSGRPY({ *jobd *rqd *dft *sysrpyl })
JOB DATE	DATE({ *jobd *sysval <i>job_date</i> })
JOB DESCRIPTION	JOB({ *usrprf [<i>library/</i>] <i>job_description</i> })
JOB MESSAGE QUEUE FULL ACTION	JOBMSGQFL({ *jobd *sysval *nowrap *wrap *prtwrap })
JOB MESSAGE QUEUE MAXIMUM SIZE	JOBMSGQMX({ *jobd *sysval <i>maximum_size_of_job_message_queue</i> })
JOB NAME	JOB({ *jobd <i>job_name</i> })
JOB PRIORITY	JOBPTY(<i>priority</i>)
JOB QUEUE	JOBQ({ *jobd [<i>library/</i>] <i>job_queue</i> })
JOB SWITCHES	SWS({ *jobd <i>switch_settings</i> })
LANGUAGE ID	LANGID({ *current *sysval *usrprf <i>language_id</i> })
LOG CL PROGRAM COMMANDS	LOGCLPGM({ *jobd *no *yes })
OUTPUT PRIORITY	OUTPTY(<i>priority</i>)
OUTPUT QUEUE	OUTQ({ *current *usrprf *dev *jobd [<i>library/</i>] <i>output_queue</i> })
PRINT DEVICE	PRTDEV({ *current *usrprf *sysval *jobd <i>printer_device_name</i> })
PRINT TEXT	PRTTXT(<i>text</i>)
SORT SEQUENCE	SRTSEQ({ *current *sysval *usrprf *hex *langidunq *langidshr [{ *libl *curlib <i>library_name</i> } /] <i>table_name</i> })
SUBMITTED FOR	SBMFOR(<i>job_number / user / job_name</i>)
SYSTEM LIBRARY LIST	SYSLIBL({ *current *sysval })
USER	USER({ *current *jobd <i>user_name</i> })

Table 18.2 Universal Submit Job - SBMJOB Encapsulated Configuration Options

18.2.3 Universal Command Server Options Affecting USBMJOB

The Universal Command Server `JOBLOG_COPY_KEEP` configuration option controls the keeping a copy of the job log returned via standard output on the local iSeries system.

See Chapter 3 Universal Command Server Configuration Options in the Universal Command 4.1.0 Reference Guide for specific information about `JOBLOG_COPY_KEEP`.

18.3 Universal Submit Job (USBMJOB) Specific Options

18.3.1 COMMAND

The COMMAND option specifies a command that runs in the submitted batch job.

The command can be a maximum of 3000 characters.

USBMJOB Parameter: `CMD(command)`

18.3.2 COPY_SPOOL_FILES

The CPYSPLF option specifies whether or not spooled output files generated by the submitted job are copied to standard output.

USBMJOB Parameter: `CPYSPLF({*yes | *no })`

- ***YES** will cause spooled output files generated by the submitted job to be written to standard output.
- ***NO** will prevent spooled output files generated by the submitted job from being written to standard output.

18.3.3 ENCRYPTED_COMMAND_FILE

The ENCRYPTED COMMAND FILE option specifies the name of an encrypted command file.

The Encrypted Command File option is used when the Remote Reply Facility is used. In order to issue inquiry commands remotely, Universal Submit Job must log on to the remote system. This is accomplished by internally issuing a command to the Universal Command Manager. Therefore, the Encrypted Command File for Universal Submit Job serves the same security purposes as the Encrypted Command File for Universal Command Manager. However, encrypted command files for Universal Submit Job should not contain commands. The command will be issued by Universal Submit Job.

Use the Universal Encrypt utility provided with Universal Command to encrypt a plain text command file. If a key was used to encrypt the file, the same key must be supplied using the KEY option.

Command files (encrypted or not) that contain sensitive data should be protected from unauthorized read access with file level security.

USBMJOB Parameter: ECMFILE(*cmd_file*) [ECMMBR(*member*)]

18.3.4 JOB_LOG_LIBRARY

The JOB LOG LIBRARY option specifies a library into which the job log will be placed.

The job log will be sent to files **USJPnnnnnn** and **USJSnnnnnn** (nnnnnn is the job number):

- **USJPnnnnnn** contains the primary messages of the job log.
- **USJSnnnnnn** contains the secondary messages of the job log.

The member name for both primary and secondary joblog files is of the form **Cnnnnnn**, where nnnnnn is the job number of the job that USBMJOB is running under.

USBMJOB Parameter: JOBLOGLIB(*library*)

18.3.5 JOB_STATUS_POLLING_INTERVAL

The JOB STATUS POLLING INTERVAL option specifies the number of seconds that USBMJOB will sleep between calls to check the status of the submitted job.

USBMJOB Parameter: POLL(*seconds*)

18.3.6 KEY

The KEY option specifies the encryption key used to encrypt the encrypted command file specified by [ENCRYPTED_COMMAND_FILE](#). If no encryption key is specified, a default key is used.

USBMJOB Parameter: KEY(*key*)

18.3.7 REMOTE_MESSAGE_PREFIX

The REMOTE MESSAGE PREFIX option allows the user to specify a text string up to 12 characters in length that will prefix any remote messages sent by USBMJOB.

This prefix can make it easier to relate a remote message to its associated job.

USBMJOB Parameter: RMTMSGPRFX(*prefix*)

18.3.8 REMOTE_REFRESH_INTERVAL

The REMOTE REFRESH INTERVAL option specifies a time interval (in seconds) that controls how long a remote reply message will remain on a remote console without being replied to before it will time out.

If the remote reply message times out, the message will be removed from the remote console. Universal Submit Job then will determine if the user job still is waiting for a reply. If it is, the remote reply message will be re-sent to the remote console.

USBMJOB Parameter: RMTREFRESH(*seconds*)

[Default is 0 (wait indefinitely for a reply).]

18.3.9 REMOTE_REPLY_COMMAND_PATH

The REMOTE REPLY COMMAND PATH option specifies the path (including the executable name) to the remote message handler (*uwto*). (See [Chapter 18 Universal Write-to-Operator](#) in the Universal Products Utilities 4.1.0 User Guide for information on the Universal WTO utility.)

USBMJOB Parameter: MSGCMDPATH(*command_path*)

18.3.10 REMOTE_REPLY_HOST

The REMOTE_REPLY_HOST option specifies the host name of the Universal Broker on the remote system on which the Universal WTO command is executed. (See [Chapter 18 Universal Write-to-Operator](#) in the Universal Products Utilities 4.1.0 User Guide for information on the Universal WTO utility.)

USBMJOB Parameter: RMTHOST(*host*)

The format of *host* is either:

- Host name (for example, *homer*)
- Numeric address (for example, *10.20.30.40*)

18.3.11 REMOTE_REPLY_PORT

The REMOTE_REPLY_PORT option specifies the port of the Universal Broker on the remote system on which the Universal WTO command is executed (see Section [18.3.10 REMOTE_REPLY_HOST](#)). (See [Chapter 18 Universal Write-to-Operator](#) in the Universal Products Utilities 4.1.0 User Guide for information on the Universal WTO utility.)

USBMJOB Parameter: RMTPORT(*port*)

Valid values for *port* are:

- Port number
- Service name

[Default is 7887.]

18.3.12 REMOTE_REPLY_USER_ID

The REMOTE_REPLY_USER_ID option specifies the user ID for the remote system where the Universal WTO command resides. (See [Chapter 18 Universal Write-to-Operator](#) in the Universal Products Utilities 4.1.0 User Guide for information on the Universal WTO utility.)

USBMJOB Parameter: RMTUSER(*userid*)

18.3.13 REMOTE_REPLY_USER_PWD

The REMOTE REPLY USER PASSWORD option specifies the password for the user on the remote system where the Universal WTO command resides. (See [Chapter 18 Universal Write-to-Operator](#) in the Universal Products Utilities 4.1.0 User Guide for information on the Universal WTO utility.)

USBMJOB Parameter: RMTPWD(*password*)

18.3.14 SPECIFY_PRINT_CONTROL_CHARS

Specifies which print control characters (if any) are to replace the spooled file's internal print control characters.

USBMJOB Parameter: SPLFCTLCHR(**characters*)

Valid values for *characters* are:

- *NONE*
- *FCFC*
- *PRTCTL*
- *S36FMT*

See the CL Reference, SC41-5722 for more information on these values.

18.3.15 TRACE

The TRACE option specifies whether or not trace information will be written to standard error.

USBMJOB Parameter: TRACE(**{*yes|*no}**)

- **YES* will cause trace information to be written to standard error.
- **NO* will prevent trace information from being written to standard error.

Note: Use TRACE only as directed by Stonebranch, Inc. [Customer Support](#).

18.3.16 USE_REMOTE_REPLY_FACILITY

The USER REMOTE REPLY FACILITY option specifies whether or not Universal Submit Job will use the remote reply facility.

The remote reply facility will detect messages, issued by the submitted job, that require a reply. The message then will be passed on to a remote system for a reply. When the reply is received, the reply will be sent to the message queue that is waiting for the reply.

USBMJOB Parameter: RMTRPY({ ***yes** | ***no** })

- ***YES** will cause USBMJOB to use the remote reply facility.
- ***NO** will cause USBMJOB to ignore message wait conditions for the submitted job.

[Default is *NO.]

Chapter 19

Universal Write-to-Operator

19.1 Overview

This chapter provides detailed information on the configuration options available for use with Universal Write-to-Operator (Universal WTO).

The options are listed alphabetically, without regard to any specific operating system.

Information on how these options are used is documented in the Universal Products Utilities 4.1.0 User Guide.

Section [1.1 Configuration Options Information](#) provides a guideline for understanding the information presented for each option.

19.2 Configuration Options List

Table 19.1, below, identifies the Universal WTO configuration options.

Option Name	Description	Page
CONSOLE_ID	ID of the console to which the message is routed	296
CONSOLE_NAME	Name of the console to which the message is routed	297
HELP	Writes a description of the command options and their format	298
KEY	Key to associate with the message	299
MESSAGE	Text to write to the z/OS operator console	300
MESSAGE_LEVEL	Level of messages to write	301
REPLY	Directs UWTO to issue a WTOR message and wait for an operator reply to the message	302
TIMEOUT	Number of seconds to wait for a WTOR operator reply	303
VERSION	Writes the program version and copyright information	304

Table 19.1 Universal WTO Configuration Options

19.3 CONSOLE_ID

Description

The CONSOLE_ID option specifies the console ID which the message is routed.

If CONSOLE_ID specifies an invalid console ID, the message is written to the default consoles. Universal WTO will write a warning message to standard error and end with exit code 1.

CONSOLE_ID and [CONSOLE_NAME](#) are mutually exclusive.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-consoleid <i>id</i>					✓
Environment Variable	UWTOCONSOLEID= <i>id</i>					✓

Values

id is the ID of the console to which the messages is routed.

Valid values for *id* are numeric values.

19.4 CONSOLE_NAME

Description

The CONSOLE_NAME option specifies the console name which the message is routed.

It provides a method to route messages based on console name.

If an invalid console name is specified, the message is written to the default consoles. Universal WTO will write a warning message to standard error and end with exit code 1.

CONSOLE_NAME and [CONSOLE_ID](#) are mutually exclusive.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-consolename <i>name</i>					√
Environment Variable	UWTOCONSOLENAME= <i>name</i>					√

Values

name specifies the name of the console to which the message is routed.

The format of *name* is 1 to 8 characters.

19.5 HELP

Description

The HELP option writes a description of the command options and their format.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-h					√
Command Line Option, Long Form	-help					√
Environment Variable	n/a					

Values

(There are no values for this option.)

19.6 KEY

Description

The KEY option specifies a key to associate with the message.

The key provides a way to identify messages to operators. The z/OS DISPLAY console commands can list messages key values.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-key <i>keyname</i>					
Environment Variable	UWTOKEY= <i>keyname</i>					√

Values

keyname is the message key.

The format of *keyname* is 1 to 8 characters.

19.7 MESSAGE

Description

The MESSAGE option specifies the text to write to the z/OS operator console.

The text is written as a WTO or WTOR message, as specified by the [REPLY](#) option.

Note: Even though Universal WTO executes in the z/OS Unix System Services environment, not all USS supported characters are supported in the console character set. Refer to IBM MCS Console documentation for character set support. Unsupported characters are translated to blanks when written to the console.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-m <i>message</i>					✓
Command Line Option, Long Form	-msg <i>message</i>					✓
Environment Variable	UWTOMSG= <i>message</i>					✓

Values

message is the text message to write to the z/OS operator console.

The maximum length of *message* depends on the type of message specified by [REPLY](#):

- WTO 770 characters
- WTOR 121 characters

Note: A WTO message that exceeds 125 characters is written as a multi-line WTO.

19.8 MESSAGE_LEVEL

Description

The MESSAGE_LEVEL option specifies the level of messages to write.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-l <i>level</i>					✓
Command Line Option, Long Form	-level <i>level</i>					✓
Environment Variable	UWTOLEVEL= <i>level</i>					✓

Values

level indicates either of the following level of messages:

- **trace**
Writes traces messages used for debugging. The trace file, named `uwto.trc`, is created in the working directory of Universal WTO.
Note: Use **trace** only as directed by Stonebranch, Inc. [Customer Support](#).
- **audit**
Writes audit, informational, warning, and error messages.
- **info**
Writes informational, warning, and error messages.
- **warn**
Writes warning and error messages.
- **error**
Writes error messages only.

[Default is warn.]

19.9 REPLY

Description

The REPLY option specifies the type of message to be issued by Universal WTO:

- WTO (do not request message reply)
- WTOR (request and wait for message reply)

For a WTOR message, the length of time to wait for a reply can be limited with the [TIMEOUT](#) option. The maximum reply length is 119 characters. The reply is written to Universal WTO's standard output file.

Note: A valid operator reply to a WTOR message can be zero characters. In this case, nothing is written to standard output.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-reply <i>option</i>					√
Environment Variable	UWTREPLY= <i>option</i>					√

Values

option specifies the type of message to be issued.

Valid values for option are:

- **no**
Issue a WTO message.
- **yes**
Issues a WTOR message.

[Default is no.]

19.10 TIMEOUT

Description

The TIMEOUT option specifies the number of seconds to wait for a reply to a WTOR message (see the [REPLY](#) option).

If a reply is not received within the specified time, the WTOR message is deleted and Universal WTO ends with exit code 2.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	n/a					
Command Line Option, Long Form	-timeout <i>seconds</i>					✓
Environment Variable	UWTOTIMEOUT= <i>seconds</i>					✓

Values

seconds is the length of time to wait for a reply.

[Default is 0 seconds (wait indefinitely for a reply.)]

19.11 VERSION

Description

The VERSION option writes the program version and copyright information.

Usage

Method	Syntax	OS/400	NonStop	UNIX	Windows	z/OS
Command Line Option, Short Form	-v			✓	✓	✓
Command Line Option, Long Form	-version			✓	✓	✓
Environment Variable	n/a					

Values

(There are no values for this option.)

Chapter 20

Additional Information

20.1 Overview

This chapter provides additional information related to Universal Products Utilities.

[Table 20.1](#), below, identifies this information and provides a link to its location in this document.

Information	Description	Page
DD statements	DD statements for use in the Universal Control Manager for z/OS batch JCL.	306
SSL cipher suites	SSL cipher suites for use with Universal Command.	307
Character Code Pages	Character code pages provided by Stonebranch Inc. for use with Universal Products on each supported operating system.	308
UTT Files	Universal Translate Table (UTT) files are used to translate between Unicode and the local single-byte code page	310

Table 20.1 Universal Products Utilities - Additional Information

20.2 DD Statements Used in JCL

Table 20.2 describes the DD statements used in the Universal Control Manager for z/OS batch JCL.

DD name	DCB Attributes *	Mode	Description
STEPLIB	DSORG=PO, RECFM=U	Input	Load library containing the program being executed.
UNVCONF	DSORG=PO, RECFM=(F, FB, V, VB)	Input	Universal Control configuration library.
UNVNLS	DSORG=PO, RECFM=(F, FB, V, VB)	Input	Universal Control national language support library. Contains message catalogs and code page translation tables.
SYSPRINT	DSORG=PS, RECFM=(F, FB, V, VB)	Output	stdout file for the UCTL program. UCTL does not write any messages to SYSPRINT.
SYSOUT	DSORG=PS, RECFM=(F, FB, V, VB)	Output	stderr file for the UCTL program. UCTL writes its messages to SYSOUT.
SYSIN	DSORG=PS, RECFM=(F, FB, V, VB)	Input	stdin file for the UCTL program. UCTL reads its command options from SYSIN.
* The C runtime library determines the default DCB attributes. Refer to the IBM manual <i>OS/390 C/C++ Programming Guide</i> for details on default DCB attributes for stream I/O			

Table 20.2 Universal Control Manager for z/OS - Batch JCL DD Statements

20.3 SSL Cipher Suites

Table 20.3 identifies all of SSL cipher suites provided by Stonebranch, Inc. for use with Universal Command.

Cipher Suite	Description
RC4-SHA	128-bit RC4 encryption and SHA-1 message digest.
RC4-MD5	128-bit RC4 encryption and MD5 message digest.
AES256-SHA	256-bit AES encryption and SHA-1 message digest.
AES128-SHA	128-bit AES encryption and SHA-1 message digest.
DES-CBC3-SHA	128-bit Triple-DES encryption and SHA-1 message digest.
DES-CBC-SHA	128-bit DES encryption and SHA-1 message digest.
NULL-SHA	No encryption and SHA-1 message digest.
NULL-MD5	No encryption and MD5 message digest.
NULL-NULL	No encryption, no data authentication, SSL is not used.

Table 20.3 SSL Cipher Suites

20.4 Character Code Pages

Table 20.4 identifies the character code pages provided by Stonebranch Inc. for use with Universal Products on each supported operating system.

Code Page	CCSID	z/OS	UNIX	Windows	OS/400		HP NonStop
					HFS	LIB	
IBM037	037	✓			✓	✓	
IBM273	273	✓			✓	✓	
IBM277	277	✓			✓	✓	
IBM278	278	✓			✓	✓	
IBM280	280	✓			✓	✓	
IBM284	284	✓			✓	✓	
IBM500	500	✓			✓	✓	
IBM875	875	✓					
IBM1047							
IBM1140	1140	✓			✓	✓	
IBM1141	1141	✓			✓	✓	
IBM1142	1142	✓			✓	✓	
IBM1143	1143	✓			✓	✓	
IBM1144	1144	✓			✓	✓	
IBM1145	1145	✓			✓	✓	
IBM1146	1146	✓			✓	✓	
IBM1147	1147	✓			✓	✓	
IBM1148	1148	✓			✓	✓	
IBM4971	4971	✓					
ISO8859-1	819		✓	✓	✓		✓
ISO8859-2	912		✓	✓	✓		✓
ISO8859-3	913		✓	✓	✓		✓
ISO8859-4	914		✓	✓	✓		✓
ISO8859-5	915		✓	✓	✓		✓
ISO8859-6	1089		✓	✓	✓		✓
ISO8859-7	813		✓	✓	✓		✓
ISO8859-8	916		✓	✓	✓		✓
ISO8859-9	920		✓	✓	✓		✓
ISO8859-10			✓	✓	✓		✓
ISO8859-13	921		✓	✓	✓		✓
ISO8859-14			✓	✓	✓		✓
ISO8859-15	923		✓	✓	✓		✓
PC437	437			✓	✓		

Code Page	CCSID	z/OS	UNIX	Windows	OS/400		HP NonStop
					HFS	LIB	
PC737	737			√	√		
PC775	775			√	√		
PC850	850			√	√		
PC852	852			√	√		
PC855	855			√	√		
PC857	857			√	√		
PC860	860			√	√		
PC861	861			√	√		
PC862	862			√	√		
PC863	863			√	√		
PC864	864			√	√		
PC865	865			√	√		
PC866	866			√	√		
PC869	869			√	√		
PC874	874			√	√		
WIN1250	1250			√	√		
WIN1251	1251			√	√		
WIN1252	1252			√	√		
WIN1253	1253			√	√		
WIN1254	1254			√	√		
WIN1255	1255			√	√		
WIN1256	1256			√	√		
WIN1257	1257			√	√		
WIN1258	1258			√	√		

Table 20.4 Character Code Pages

20.5 UTT Files

Table 20.5 identifies the Universal Translate Table (UTT) files that are used to translate between Unicode and the local single-byte code page.

Operating System	UTT File Location
OS/400	UTT files are located in the source physical file UNVPRD410 / UNVNLS . <i>codepage</i> is the member name of the UTT file.
z/OS	UTT files are located in the library allocated to the UNVNLS ddname. <i>codepage</i> is the member name of the UTT file.
UNIX	UTT files are located in the nls subdirectory of the installation directory. <i>codepage</i> is the base file name of the UTT file. All UTT files end with an extension of .utt .
Windows	UTT files are located in the NLS subdirectory of the installation directory. <i>codepage</i> is the base file name of the UTT file. All UTT files end with an extension of .utt .
HP NonStop	UTT files are located in the \$SYSTEM . UNVNLS subvolume. <i>codepage</i> is the base file name of the UTT file.

Table 20.5 UTT File Locations

Appendix A

Customer Support

Stonebranch, Inc. provides customer support, via telephone and e-mail, for Indesca / Infitran Utilities and all Indesca / Infitran components.

E-MAIL

All Locations

support@stonebranch.com

Customer support contact via e-mail also can be made via the Stonebranch website:

www.stonebranch.com

TELEPHONE

Customer support via telephone is available 24 hours per day, 7 days per week.

North America

(+1) 678 366-7887, extension 6

(+1) 877 366-7887, extension 6 [toll-free]

Europe

+49 (0) 700 5566 7887



**950 North Point Parkway, Suite 200
Alpharetta, Georgia 30005
U.S.A.**

