Oracle® Calendar Server

Reference Manual

Release 5.5

August, 2002

Part No. B10094-01

This book is a reference volume containing specific information on calendar server configuration parameters and administration utilities.

ORACLE®

Oracle Calendar Server Reference Manual, Release 5.5

Part No. B10094-01

Contents

C   Utilities

D   Time Zone Table

E   Extensions to Directory Server Schema

Index

List of Tables

Send Us Your Comments

Oracle Calendar Server Reference Manual, Release 5.5

Part No. B10094-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: infodev_us@oracle.com
- FAX: (650) 633-3836   Attn: Oracle Collaboration Suite Documentation Manager
- Postal service:
  Oracle Corporation
  Oracle Collaboration Suite Documentation Manager
  500 Oracle Parkway, Mailstop 2op5
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Oracle Calendar is scalable calendaring software, based on open standards, for efficiently scheduling people and resources. Among other features, it offers a dedicated database, real-time lookups and free-time searches, multiple time zone support and UTF-8 encoding to support international deployments, e-mail and wireless alerts, multi-platform support and an extensible Authentication, Compression and Encryption (ACE) framework for enhanced security.

The Oracle Calendar server is the back end to an integrated suite of calendaring and scheduling products. Networked users can use a desktop client (Windows, Macintosh, Motif), Web client or Microsoft Outlook to manage their calendars. Mobile users can synchronize their agendas with a variety of PDAs or, with the addition of Oracle's wireless technology, can send and receive calendar entries using a mobile phone.

Oracle Calendar is part of Oracle Collaboration Suite, offering integrated e-mail, voice mail, calendaring and wireless services. For more information on the other components of Oracle Collaboration Suite, please see Oracle's Web site or consult the relevant product documentation.

Intended Audience

This *Reference Manual* documents the configuration parameters and utilities included with your calendar server. This guide is directed at any administrator whose task is the installation, configuration, use and maintenance of Oracle Calendar in general and the Oracle Calendar server in particular. It is a companion volume to the *Oracle Calendar Server Administrator's Guide*, which documents deployment, configuration and maintenance procedures for your calendar server.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation   JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation   This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

This manual contains five reference appendices:

Appendix A
This appendix contains information on configuration parameters that set default values for user and resource attributes.

Appendix B
This appendix contains information on configuration parameters that control the behaviour of your calendar server.

Appendix C
This appendix contains information on the use and syntax of the command-line utilities provided for administering your calendar server.

Appendix D

This appendix provides a mapping between regions and their associated time zone classifications.

Appendix E

This appendix contains detailed information on the calendar-specific information stored in the Oracle Internet Directory.

Related Documents

For more information, see the following manuals in the Oracle Calendar documentation set:

- *Oracle Calendar Server Administrator's Guide*
- *Oracle Calendar Server Release Notes*

Conventions

In this manual, Windows and NT are both used to refer to the Windows95, Windows98, and Windows NT operating systems.

In examples, an implied carriage return occurs at the end of each line, unless otherwise noted. You must press the Return key at the end of a line of input.

The following conventions are also used in this manual:

Convention	Meaning
. . .	Vertical ellipsis points in an example mean that information not directly related to the example has been omitted.
. . .	Horizontal ellipsis points in statements or commands mean that parts of the statement orcommand not directly related to the example have been omitted
boldface text	Boldface type in text indicates a term defined in the text, the glossary, or in both locations.
monospaced font	This typeface is used for any text that appears on the computer screen or text that you should type. It is also used for file and path names and functions.
Cmd line	Refers to a procedure executed on the command line (UNIX or NT) using a calendar server utility.

Convention	Meaning
Web GUI	Refers to a procedure executed using the Calendar Administrator, an on-line administrative tool. Consult the calendar server *Administrator's Guide*, Appendix F, "The Calendar Administrator," for more information
/	Forward-slashes are used to separate directories in a path name, following UNIX syntax. For Windows operating systems, substitute back-slashes "\" for all forward-slashes unless otherwise instructed.
< >	Angle brackets enclose user-supplied names and variables.
[ ]	Brackets enclose optional clauses from which you can choose one or none.

A

User and Resource Parameters

This appendix details the parameters available to configure default user and resource profiles in the `/users/unison/misc/user.ini` and `/users/unison/misc/resource.ini` files respectively. For details on how to implement user and resource profiles, see your calendar server's *Administrator's Guide*.

Each parameter's stated default value is used if that parameter is omitted from its configuration file. These defaults are not necessarily the optimal settings for your installation. The initialization files supplied with the software contain settings that provide a good starting point for further configuration. It is strongly recommended that for reference purposes you keep a copy, in either printed or electronic format, of these files before modification.

Parameters for default user and resource profiles

The information that can be specified includes:

- Display preferences

- Refresh frequency, notification and reminder preferences

- Access rights for viewing and scheduling granted to other users

- Time zone, if different from that of the node

- List of public and administrative groups in which to include new users and resources

- List of designates for the user or resource

- Administrative rights for groups, holidays and resources (users only)

- Default directory address fields (users only)

Table A–1   User and resource profile

Parameter	Possible values	Default value	Description
ShowSunday	TRUE, FALSE	TRUE	Shows and hides Sundays
ShowSaturday	TRUE, FALSE	TRUE	Shows and hides Saturdays
TimeFormat	1 (24 hour), 2 (AM/PM)	2 (AM/PM)	sets time display format
StartDay	00h00 to 24h00	08h00	sets agenda start time for display
EndDay	00h00 to 24h00	18h00	sets agenda stop time for display
TimeInc	5, 10, 15, 20, 30, 60 (minutes)	30 minutes	defines time increment for day and week views
RefreshFrequency	0 ... 65536 (minutes)	15	sets refresh frequency of client
MailNotification	TRUE, FALSE	FALSE	specifies if the user can receive mail notification
Language	en-US fr-FR it-IT es-ES fi-FI de-DE pt-PT ja-JP zh-CN ko-KR sv-SE pt-BR nl-NL	en-US	Determines the language used for notification messages.
DefaultReminder	0 (disabled), 1 (enabled)	0	controls use of Pop-up Reminders

Table A–1   User and resource profile

Parameter	Possible values	Default value	Description
TimeBeforeReminder	0, 2, 5, 10, 60, 120, 240 (minutes) 12, 24, 48, 96 (hours) 7, 14, 31 (days)	0	sets reminder time for Default Reminder
SMSServiceEnable	TRUE, FALSE	TRUE	Enables wireless reminders and notification
ViewNormalEvent	YES, NO, TIME	NO	default security given to other users
ViewPersonalEvent	YES, NO, TIME	NO	see above
ViewConfidentialEvent	YES, NO, TIME	NO	see above
CanBookMe	TRUE, FALSE	FALSE	specifies if user can be invited
ViewNormalTask	YES, NO	NO	default security given to other users
ViewPersonalTask	YES, NO	NO	see above
ViewConfidentialTask	YES, NO	NO	see above
CreatePublicGroups	TRUE, FALSE	FALSE	controls ability to create Public groups
ManageAdmGroups	TRUE, FALSE	FALSE	controls ability to create Admin groups
ManageHolidays	TRUE, FALSE	FALSE	controls ability to manage holidays
OU1	<Organizational Unit 1>	n/a	value for directory address field
OU2	<Organizational Unit 2>	n/a	see above
OU3	<Organizational Unit 3>	n/a	see above
OU4	<Organizational Unit 4>	n/a	see above
O	<Organization>	n/a	see above
C	<Country>	n/a	see above
A	<Administrative Domain>	n/a	see above

Table A–1   User and resource profile

Parameter	Possible values	Default value	Description
P	<Private Domain>	n/a	see above
TimeZone	<Time zone>	value defined in `unison.ini`	defines a time zone specifically for the user or resource
Group0 ... Group9	<Admin or public group name>	n/a	specifies groups in which to include the user or resource
Designate0 ... Designate9	<User name>	n/a	defines users who may act as designates for the new user or resource

Display preferences

```
ShowSunday = TRUE/FALSE
ShowSaturday = TRUE/FALSE
```
These parameters determine whether or not these days will be part of the week view on the client. The default is TRUE.

```
TimeFormat = 1/2
```
This parameter determines whether or not time is displayed in military (24h) or standard (AM/PM) time. The default is 2 -- AM/PM.

```
StartDay = <time of day>
```
This parameter determines the first time slot displayed in the user's agenda (day & week view only). Earlier time slots can still be viewed by using the vertical scroll bar. This does not affect the regular business hours of the user. The default is 08h00.

```
EndDay = <time of day>
```
This parameter is used to define the last time slot displayed in a user's agenda (day & week view only), although it has little effect given that other settings, such as StartDay, time slot increments and spacing height, also affect how little or how much of the day is displayed. Later time slots can still be viewed by using the vertical scroll bar. This does not affect the regular business hours of the user. The default is 18h00.

```
TimeInc = <time_in_minutes>
```
This parameter defines the time slot increment for the day & week views. Adjusting the value of this parameter affects how much of your day is displayed on the screen. Only the following values can be specified: 5, 10, 15, 20, 30, 60 (minutes). The default is 15 minutes.

Refresh, notification & reminder preferences

RefreshFrequency = <time_in_minutes>
This parameter sets the refresh frequency of the client in minutes. A value of 0 would effectively disable the refresh. The default is 15 minutes.

MailNotification = TRUE/FALSE
This parameter specifies whether or not the user can receive mail notification. Note, this setting has no effect on the users' ability to send mail notification. The default is FALSE.

```
Language = en-US (English)
           fr-FR (French)
           it-IT (Italian)
           es-ES (Spanish)
           fi-FI (Finnish)
           de-DE (German)
           pt-PT (Portuguese)
           ja-JP (Japanese)
           zh-CN (Chinese)
           ko-KR (Korean)
           sv-SE (Swedish)
           pt-BR (Brazilian Portuguese)
           nl-NL (Dutch)
```
Determines the language used for server-side reminder messages. Consult Chapter 6 of your calendar server *Administrator's Guide* for details on server-side reminders and user languages.

DefaultReminder = 0/1
If set to 1, the Default Reminder for Agenda Entries and Day Events is set to Pop-up Reminder. For Tasks, only the Default Task Due Reminder is set to Pop-up Reminder, the Default Task Start Reminder is NOT set. Furthermore, The Daily Notes Default Reminder is also not set. The default is 0, or no reminders.

TimeBeforeReminder = <time_in_minutes>
This parameter is used to set the default reminder time. In other words, a value of 24 would mean that default reminders would appear 24 hours before the start of the event. Only the following values can be specified: 0, 2, 5, 10, 60, 120, 240 (minutes); 12, 24, 48, 96 (hours); 7, 14, 31 (days).

SMSServiceEnable = TRUE/FALSE
This parameter determines whether or not wireless notification is enabled for this user. The "MOB-ENABLE" user field is set according to the value of this parameter. The default value is "TRUE".

Default security to other users

```
ViewNormalEvent = YES/NO/TIME
ViewPersonalEvent = YES/NO/TIME
ViewConfidentialEvent = YES/NO/TIME
ViewNormalTask = YES/NO (user profiles only)
ViewPersonalTask = YES/NO (user profiles only)
ViewConfidentialTask = YES/NO (user profiles only)
```
The above parameters determine the default security rights granted to other users when creating events or tasks of these designations. For example, if `ViewNormalEvent` were set to `TIME`, only the time slot of the event would be visible to other users, not its title, location or description. Conversely, if `ViewNormalEvent` were set to `YES`, all details of the event would be visible to other users. If `ViewNormalEvent` were set to `NO`, the event would not be visible at all to other users.

The default value for all of the above parameters is `NO`.

All details of a *public* event are always visible to other users. There is no way to modify this behaviour using these parameters.

The `ViewNormalEvent` and `ViewNormalTask` settings map to the "Normal" Access Level on the client.

The `ViewPersonalEvent` and `ViewPersonalTask` settings map to the "Personal" Access Level on the client.

The `ViewConfidentialEvent` and `ViewConfidentialTask` settings map to the "Confidential" Access Level on the client.

```
CanBookMe = TRUE/FALSE
```
Setting this parameter to `TRUE` allows any undefined user to schedule with the user. Of course, this can be overridden by the user within the client. The default setting is `FALSE`.

Group and administrative rghts (user profiles only)

```
CreatePublicGroups = FALSE
```
This parameter determines whether or not users have the ability to create Public groups (i.e. groups available to all users in the database). The default is `FALSE`.

```
ManageAdmGroups = FALSE
```
This parameter determines whether or not users have the ability to create Admin groups. Like Public groups, Admin groups are available to all users in the database, except that Admin groups are not owned by the user who created them, but rather by the SYSOP. Why would someone want to create a SYSOP-owned group? If you

delete a user who has created a Public group, the Public group is deleted along with his/her data. Conversely, Admin groups created by that user will not be deleted. The default is FALSE.

```
ManageHolidays = FALSE
```
This parameter determines whether or not users have the ability to manage (i.e. create, modify or delete) holidays on the system.

X.400 address information (user profiles only)

The following parameters, when defined, can be useful for populating the database with a large number of users who share X.400 address information.

```
OU1 = <Org_unit_1>
OU2 = <Org_unit_2>
OU3 = <Org_unit_3>
OU4 = <Org_unit_4>
O = <Organization>
C = <Country>
A = <Administrative_Domain>
P = <Private_Member_Domain>
```

Miscellaneous

```
TimeZone = <time zone>
```
This parameter is used to define a different time zone for the user.

Admin and public groups

This section allows you to define groups that users and resources will be placed in as they are added to the system. Note that the groups must be created beforehand, and that there is a maximum of 10 groups per section.

```
Group0 = <group_name>
Group1 = <group_name>
Group2 = <group_name>
...
Group9 = <group_name>
```

Designates

A designate is a user assigned the right to modify the agenda of another user or resource.

This section allows you to define designates for users and resources. Note that designates must exist in the database beforehand, and that there is a maximum of 10 designates per section.

```
Designate0 = <designate_name>
Designate1 = <designate_name>
Designate2 = <designate_name>
...
Designate9 = <designate_name>
```

The <designate_name> argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. The "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they may need to be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

> **Note:** If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored.

Some example specifications are: `"S=Kilpi/G=Eeva"`, `"S=B*/G=Nicole/O=Acme"`, `"O=Acme/ID=1111/OU1=authors"`, `"S=Austen/G=Jane/EMAIL=mr_darcy@freemail.org"`

Table A–2   Accepted keys

Key	X.400 Field
S	Surname
G	Given name
I	Initials
ID	Identifier
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2

Table A–2   Accepted keys

Key	X.400 Field
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain
P	Private domain

B

Server Parameters

This appendix lists and describes all tunable parameters available to configure your calendar server. All parameters listed are located in the initialization file `/users/unison/misc/unison.ini`.

Each parameter's stated default value is used if that parameter is omitted from its configuration file. These defaults are not necessarily the optimal settings for your installation. The initialization files supplied with the software contain settings that provide a good starting point for further configuration. It is strongly recommended that for reference purposes you keep a copy, in either printed or electronic format, of these files before modification.

Configuration parameters

The types of behaviour that can be modified fall under the following sections:

- Controlling server behaviour
- Controlling server interactions with directory server
- Controlling client behaviour
- Controlling client connections to server

The following table lists all parameters alphabetically by section.

Table B–1 `unison.ini` ***configuration parameters***

Section	Parameter	Description
[ACE]	frameworkenable	Enable the ACE framework
	slibcachecount	Maximum number of shared libraries per type
	minbufsizetocompress	Minimum buffer size for compression
	workbufsize	Buffer size for compression and encryption
[ACE_PLUGINS_SERVER]	web_CAL_sharedkey	Web authentication — shared key
[AUTHENTICATION]	admindefault	Default authentication method for administrators
	default	Default authentication method for clients
	keepresourcepswincaldb	Location of resource passwords for authentication
	servicedefault	Default authentication method for other servers
	supported	Supported authentication methods for clients
[CLIENT]	minrefreshrate	Minimum interval for checks for new agenda entries (server-side enforcement)
[CLUSTER]	excludednodes	Excluded nodes for on-line registration
	masternode	Master node
	remotemasternode	Remote master node
[COMPRESSION]	admindefault	Default compression method for administrators
	default	Default compression method for clients
	servicedefault	Default compression method for other servers
	supported	Supported compression methods
[CWS]	emailcharsetmap	Character set conversion for e-mail
	enable	Automatic start of CWS daemon/service

Table B–1 `unison.ini` *configuration parameters*

Section	Parameter	Description
	`mailfiledelete`	Automatic deletion of temporary file for last mail message
	`mailhdroriginatorfromuser`	Content of the "From:" field of the mail header
	`mailhdrtoname`	UTF-8 for names in "To:" field of mail header
	`mailhost`	Host name of the SMTP mail server (obsolete)
	`maxtimepernode`	Maximum time spent processing requests per node
	`mimecontentcharset`	Character set for content portion of mail message
	`noreqsleep`	Sleep time between checks on request queue
	`sendmailpath`	Pathname of the mail program (obsolete)
	`smsnotifyprogram`	Short Message Service (SMS) notification program
	`smsnotifyprogramparam`	Short Message Service (SMS) notification program arguments
	`smtpmail`	Set mail transport
	`smtpmailhost`	Host name of the SMTP mail server
	`smtpmailmaxcommandlinesize`	Maximum size for sendmail command lines
	`smtpmailmaxrecipients`	Maximum number of recipients
	`smtpmailpath`	Pathname of the mail program
	`smtpmailprogram`	Mail program
	`startupsleep`	Time to sleep on start-up
	`trace`	Logging: verbose switch
	`unixmail`	Set mail transport (obsolete)
	`unixmailprogram`	Mail program (obsolete)

Table B–1 `unison.ini` ***configuration parameters***

Section	Parameter	Description
[DAS]	dir_itembindonread	Force DAS sign-on and sign-off for read-only operations
	dir_service	Set directory service
	dir_updcalonly	Allow users to update only calendar attributes
	enable	Automatic start of DAS daemon/service
	stats	Statistics logging: unidasd
[DB]	db_files	Maximum number of database files open per user
	db_pages	Number of pages in the database cache
[DBI]	dbi_name	Node database template
	dbversion	Node database version
[DOMAIN]	type	Node connection types
[ENCRYPTION]	admindefault	Default encryption method for administrators
	default	Default encryption method for clients
	needsauthenticate	Encryption methods requiring prior authentication
	servicedefault	Default encryption method for other servers
	supported	Supported encryption methods
[ENG]	activity	Statistics logging: user connections
	allowresourceconflict	Double-booking resources (server-side)
	allowsysoplogon_capi	Allow SYSOP logons for CAPI applications
	allowsysoplogon_unicp	Allow SYSOP logons from unicp utilities
	allowsysoplogon_uniical	Allow SYSOP logons from uniical
	calendarhostname	Cluster host name
	capi_storage	Supported CAPI version
	dac_configerrlog	Logging of configuration errors

Table B–1 `unison.ini` *configuration parameters*

Section	Parameter	Description
	`dac_failederrlog`	Logging of failure errors
	`dac_ignorederrlog`	Logging of non-critical errors
	`dac_itemget`	Enable item retrieval from directory server
	`dac_maxretry`	SNC to DAS connection retries
	`dac_miscerrlog`	Logging of miscellaneous errors
	`dir_internal_nodes`	Coexistence of LDAP and non-LDAP nodes
	`eventrefreshintervals`	Refresh intervals and agenda ranges
	`eventsearch_ clientwindowsize`	Size of the client event search result window
	`eventsearch_ commentsearchlength`	Search event comments
	`eventsearch_ maxlookthroughlimit`	Timeout for event search
	`evsearch_maxcount`	Maximum number of events to return
	`localcharset`	Character set for log files
	`itemuidmap`	X.400 field for UID
	`max_addrlogons`	Number of concurrent sessions from a specific Internet address
	`maxinstances`	Maximum number of instances of a recurring meeting, daily note, or day event (server-side)
	`max_userlogons`	Maximum number of concurrent sessions by a given user
	`notify`	Mail notification
	`passwords`	Case-sensitivity of passwords
	`readlocktimeout`	Maximum read lock time before termination
	`readmaxlocktime`	Maximum read lock time before release
	`resourcemailmap`	Resource mail mapping

Table B–1 `unison.ini` **configuration parameters**

Section	Parameter	Description
	standards	Calendar standards
	stats	Statistics logging: user sessions
	userlookthroughlimit	Maximum number of items to search
	usermailmap	User mail mapping
	usermobilemap	User mobile phone number mapping
	usermobiletypemap	User mobile phone type mapping
	usersearchmaxreturn	Maximum number of items to return
	usersmscprefmap	User alert preference mapping
	utf8_autoconvert	Enable conversion of data to UTF-8 format
	utf8_onfailprintmesg	Logging of failure to instantiate UTF-8 conversion functionality
	writelocktimeout	Maximum write lock time before termination
	writemaxlocktime	Maximum write lock time before release
[LCK]	lck_users	Maximum number of Engines
[LDAP]	admin	Location of the calendar server administrators
	admingroup	Group entry for calendar server administrators
	attr_country	Name of the "country" attribute
	attr_organization	Name of the "organization" attribute
	attr_generation	Name of the "generation qualifier" attribute
	attr_givenname	Name of the "given name" attribute
	attr_mail	Name of the "mail" attribute
	attr_uid	Calendar unique user identifier mapping
	attr_version	LDAP protocol version
	attrpreservelist	Attribute preserve list

Table B–1 `unison.ini` *configuration parameters*

Section	Parameter	Description
	`basedn`	Distinguished Name of the subtree containing calendar server entries
	`binddn`	Distinguished Name used for anonymous connections
	`bindpwd`	Password used for anonymous connections
	`certdbpath`	Location of file containing trusted certificates
	`charset`	Character set used by the directory server
	`dsa`	Name of directory server
	`dsausesearchtocompare`	Directory server support for ldap_compare() requests
	`groupfilter`	Search filter for groups
	`groupmemberattribute`	Name of "member" attribute in group object class
	`groupobjectclass`	Search filter for groups
	`host`	Name of directory server host
	`mgrdn`	Distinguished Name of the directory server administrator
	`port`	Port number of the LDAP directory server
	`resourcerelativedn`	Relative Distinguished Name for resources
	`secure-port`	Port to use for SSL connections
	`security`	Enable SSL connections
	`timelimit`	Maximum time to wait on an LDAP call
	`writedn`	Distinguished Name used for write operations
	`writednpassword`	Password used for LDAP write connections
`[LIMITS]`	`agendaview`	Default agenda view
	`allowattachments`	Allow agenda attachments

Table B–1 `unison.ini` ***configuration parameters***

Section	Parameter	Description
	autocontrol	Minimum interval for checks for new agenda entries (client-side enforcement)
	browserhelp	On-line help display
	groupviewmax	Maximum number of users in a group view
	mail	Enable mail notification
	maxattachmentsize	Maximum size of attachments
	maxmaildistr	Maximum number of people in a mail notification distribution list
	maxpasswordage	Password aging
	maxpersabentries	Maximum number of personal address book entries
	maxrecur	Maximum number of instances for a repeating meeting, daily note, or day event (client-side)
	maxremleadtime	Maximum lead time on a reminder
	maxsearchresult	Maximum number of search results displayed
	maxwinopen	Maximum number of open windows
	mincharsearch	Minimum number of characters in the Surname edit box
	publishab	Enable publishing of address books
	page-backward	"Previous" button in search dialogue box
	page-forward	"Next" button in search dialogue box
	pubgroups	Right to create public groups
	publishab	Enable publishing of address books
	remotemaxretry	Retry limit for remote data requests to server
	remotewait	Retry interval for remote data requests to server
	resourceconflicts	Double-booking resources (client-side)

Table B–1 `unison.ini` *configuration parameters*

Section	Parameter	Description
	secure-login	Secure sign-in
	settimezone	Permission to change default timezone
	signinmaxattempts	Maximum number of sign-in attempts
	singlelst	Single local storage
	ssignin	Allow automatic sign-in
	userlist_login	Show multiple user matches on sign-in
[LOCALE]	charset	Character set for content portion of mail message
[NOTIFY]	checkreminderinterval	Interval between checks for reminders
	ignoreoldreminders	Reminders to ignore
	limitremindercheck	Maximum time to check a node for reminders
	sms	Short Message Service (SMS) alerts
[OUTLOOK_CONNECTOR]	eventselectbegin	Number of days preceding current date to consult or return for queries
	eventselectend	Number of days following current date to consult or return for queries
[PRODUCT]	name	Product name
	version	Product version number
[QUOTA]	maxfolderentryperuser	Maximum number of entries in a folder
[SNC]	enable	Automatic start of the SNC daemon/service
	max_socket	Maximum number of connections
	request_chunk_size	Number of requests that are reset at a time
	snc_so_keepalive	Idle connections
	snc_so_rcvbuf	Size of the socket layer receive buffer
	snc_so_sndbuf	Size of the socket layer send buffer
	snc_tr_block	Block size for communications

Table B–1 `unison.ini` ***configuration parameters***

Section	Parameter	Description
	snc_tr_recv_timeout	Time-out for received transmissions
	snc_tr_send_timeout	Time-out for sent transmissions
	wait_sbh	Number of minutes to wait for remote node connection
[SYS]	sys_owner	User under whom processes run (UNIX only)
[TIMEZONE]	checksum	Checksum of the time zone rules file
	default	Default time zone
	rules	Time zone rules
[UTL]	adm_moduserpassword	Modification of user password
	backup_timeout	Backup operation timeout
	external_backup	Specify alternate backup utility
	external_restore	Specify alternate restore utility
	host	Host name of system
	restore_timeout	Restore operation timeout
[WEBLINK]	browser-path-win	Browser to launch for Windows clients
	command-description-offline	Off-line command description
	command-description-online	On-line command description
	command-name	Text to appear in Help menu and ToolTip
	download-fail	Error message to display if download fails
	download-mode	File transfer protocol
	mode	Enabling web access
	offline-source-mac	Source of web pages for Mac clients in off-line mode
	offline-source-version	Version of off-line web page
	offline-source-win	Source of web pages for Windows clients in off-line mode

Table B–1 `unison.ini` **configuration parameters**

Section	Parameter	Description
	`online-url`	Web page to load for clients working on-line
`[WEBLINK-FRENCH]`	`browser-path-win`	French-language browser to launch for Windows clients
	`command-description-offline`	Off-line French-language command description
	`command-description-online`	On-line French-language command description
	`command-name`	French-language text to appear in Help menu and ToolTip
	`download-fail`	Error message to display if download fails
	`download-mode`	File transfer protocol
	`mode`	Enabling French-language web access
	`offline-source-mac`	Source of French-language web pages for Mac clients in off-line mode
	`offline-source-version`	Version of French-language off-line web pages
	`offline-source-win`	Source of French-language web pages for Windows clients in off-line mode
	`online-url`	French-language web page to load for clients working on-line
`[YOURHOSTNAME, unidas]`	`connect_timeout`	Timeout for connecting to directory server
	`dir_adminmodcalonly`	Permissions for modifying directory server attributes
	`dir_adminupdcalonly`	Permissions for updating directory server attributes
	`numconnect`	Number of connections to directory server
`[YOURNODEID]`	`aliases`	Node alias(es)
	`localnodes`	Allow resources in remote nodes to appear as local
	`name`	Node name

Table B–1 `unison.ini` *configuration parameters*

Section	Parameter	Description
	relativedn	Relative DN for individual domains
	resourcerelativedn	Resource Relative DN for individual domains
	timezone	Node time zone
	version	Database version number

Product name

Section

[PRODUCT]

Parameter

name

Description

Specifies the name of the product. Set during installation, this value should not be edited or removed.

Accepted values

n/a

Default value

n/a

Product version number

Section

[PRODUCT]

Parameter

version

Description

Specifies the version number of your calendar server. Do not edit or remove this value.

Accepted values

n/a

Default value

n/a

Controlling server behaviour

Host name of system

Section

[UTL]

Parameter

host

Description

Specifies the host name of the system on which your calendar server is running. The server installation script sets this parameter. This parameter must always have a value set for it in the unison.ini file.

Accepted values

A valid (fully-specified) host name

Default value

None

Cluster host name

Section

[ENG]

Parameter

`calendarhostname`

Description

Specifies an alternate host name for the calendar server in cases when the system-defined host name should not be used.

The principal use for this parameter is to identify the calendar server host in UNIX environments using operating system clusters, where multiple hosts are running the calendar server in the same cluster for failover protection. In this case, you should set the value of this parameter to the name of the operating system cluster itself, rather than the name of any physical calendar server host.

Accepted values

A valid (fully-specified) host name

Default value

None

Calendar standards

Section

`[ENG]`

Parameter

`standards`

Description

A lists of supported Internet standards and related technologies, enclosed in curly braces {} and separated by commas. Do not change the value of this parameter without explicit instructions from application documentation or Oracle support personnel.

Accepted values

{} (no CAPI support)

{CAPI} (CAPI support with support for some ICAL2.0 attributes)

{CAPI, ICAL2.0} (CAPI support and support for all IETF ICAL 2.0 attributes)

Default value

{}

Supported CAPI version

Section

[ENG]

Parameter

capi_storage

Description

Specifies the version of Oracle's Calendar API (CAPI) that the server supports. The server reads this parameter only if the value of [ENG] standards includes ICAL2.0.

Accepted values

BASIC (pre-4.0 support for CAPI)

FH (support for CAPI 1.0)

OPTFH (support for CAPI 1.1 and higher)

Default value

OPTFH

User under whom processes run (UNIX only)

Section

[SYS]

Parameter

sys_owner

Description

For UNIX only, this parameter specifies the user under whom the calendar server processes run. In all cases, the calendar server executes services with the effective user controlling security set to unison.

Under NT all services run as System Account.

Accepted values
(UNIX)

unison

root

Default value
(UNIX)

unison

Automatic start of CWS daemon/service

Section
[CWS]

Parameter
enable

Description
Determines whether unicwsd, the Corporate-Wide Services daemon/service, automatically starts when the calendar server is brought up. You must set this to TRUE if your server configuration has multiple nodes or mail notification.

Accepted values
TRUE (start unicwsd automatically)

FALSE (do not start unicwsd automatically)

Default value
TRUE

Automatic start of the SNC daemon/service

Section
[SNC]

Parameter

enable

Description

Determines whether unisncd, the Synchronous Network Connections daemon/service, automatically starts when the calendar server is brought up. You must set this to TRUE if your server configuration contains multiple nodes or uses a directory server.

Accepted values

TRUE (start unisncd automatically)

FALSE (do not start unisncd automatically)

Default value

TRUE

Automatic start of DAS daemon/service

Section

[DAS]

Parameter

enable

Description

Determines whether unidasd, the Directory Access daemon/service, automatically starts when the calendar server is brought up. The unidasd daemon/service is required only for installations that connect to a directory server.

Accepted values

TRUE (start unidasd automatically)

FALSE (do not start unidasd automatically)

Default value

FALSE

Time to sleep on start-up

Section

[CWS]

Parameter

startupsleep

Description

Specifies the number of seconds the Corporate-Wide Services daemon/service waits (sleeps) at start-up before attempting to process any requests. This delay is intended to provide enough time for the SNC daemon/service to start up and establish the necessary connections to nodes. Increasing the value of this parameter may be necessary for servers with many nodes or connections, or where the bandwidth is low.

Accepted values

A positive integer

Default value

60

Sleep time between checks on request queue

Section

[CWS]

Parameter

noreqsleep

Description

Specifies the number of seconds the Corporate-Wide Services daemon/service waits (sleeps) when there are no requests in the local queue. After that time, the unicwsd again checks its queue for pending requests. This setting affects how long it takes to propagate data, such as remote user invitations, to other nodes. A low value may slow down the uniengd.

Accepted values

A positive integer up to a maximum value of $(2^{32} - 1)$

Default value

`60`

Maximum time spent processing requests per node

Section

`[CWS]`

Parameter

`maxtimepernode`

Description

Determines the maximum time, in seconds, that the CWS daemon/service spends processing requests for the same node. After it processes each request, the CWS daemon/service checks the total time it has spent processing requests for the node. If the total time exceeds `maxtimepernode`, the CWS daemon/service moves on to processing requests from another node, even if the current request queue is not empty. This ensures that the CWS daemon/service treats all nodes fairly, and ensures a more uniform replication delay for calendar data.

Accepted values

A positive integer

Default value

`30`

Interval between checks for reminders

Section

`[NOTIFY]`

Parameter

`checkreminderinterval`

Description

Determines the interval, in minutes, that the CWS daemon/service waits between checks for reminders.

Accepted values

A positive integer

Default value

2

Reminders to ignore

Section

[NOTIFY]

Parameter

ignoreoldreminders

Description

Determines which reminders the CWS daemon/service ignores when it checks for reminders. It ignores all reminders older than the number of minutes (from the current time) specified by this parameter.

Accepted values

A positive integer

Default value

30

Maximum time to check a node for reminders

Section

[NOTIFY]

Parameter

limitremindercheck

Description

Specifies the maximum amount of time, in seconds, that the CWS daemon/service spends at one time checking a node for reminders.

Accepted values

A positive integer

Default value

30

Number of connections to directory server

Section

[<YOURHOSTNAME>,unidas]

Parameter

numconnect

Description

Specifies the number of connections to establish to the directory server. A number of variables must be considered when setting this parameter. Guidelines are provided for installations contained within the following configuration:

- hardware configuration adequately supports the demands of the software (see Appendices A, B and C)

- clients used are not web-based (i.e. Windows, Mac or Motif clients)

- configured users per host does not exceed 5,000

- logged-on users per host does not exceed 2,500

- logged-on users per node does not exceed 1,000

- logged-on users per node is greater than 250

- connected nodes per host does not exceed 4

- number of nodes in a network does not exceed 10

- [ENG] dac_itemget = FALSE

Guidelines:

- using the `uninode` utility, establish 4 connections each way between a local node and a remote node

- using the `uninode` utility, establish 3 connections each way between local nodes

- set `numconnect` to 10% of logged-on users (value represented by the `[LCK]` `lck_users` parameter)

Administrators of installations that do not match the specified configuration should contact Oracle support for further assistance in setting this parameter.

Accepted values
Any positive integer up to a maximum value of 255

Default value
5

Timeout for connecting to directory server

Section
`[<YOURHOSTNAME>,unidas]`

Parameter
`connect_timeout`

Description
Determines the number of seconds the `unisncd` will wait before returning a timeout error when attempting to start the `unidasd` to connect to the directory server.

Accepted values
`0` (no timeout)

A positive integer

Default value
10

SNC to DAS connection retries

Section
[ENG]

Parameter
dac_maxretry

Description
Specifies the maximum number of retries the SNC daemon/service makes when attempting to establish a connection to the DAS daemon/service.

Accepted values
0 (no retries)

Any positive integer up to a maximum value of (232-1)

Default value
3

Maximum number of connections

Section
[SNC]

Parameter
max_socket

Description
Specifies the maximum number of connections the SNC daemon/service brokers among nodes in the node network.

Consult Oracle Support before setting this parameter. In most cases you instantiate all of the connections configured in the nodes.ini file. In certain configurations where you have a large number of nodes on the same machine, this parameter reduces the number of connections used, and thereby the amount of memory required, to instantiate the node network. Each connection has a socket and a uniengd process associated with it so the fewer the connections, the fewer the

number of processes and sockets required. See "Connecting nodes" in Chapter 6 of your calendar server's *Administrator's Guide* for guidelines on the number of connections to configure in the `nodes.ini` file.

Set this parameter high enough to ensure there is at least one connection from each node in the network to every other node in the network. Tune based on usage statistics.

The `[<YOURHOSTNAME>, unidas]` `numconnect` parameter configures the total number of connections to the DAS daemon/service that the SNC daemon/service brokers.

Accepted values
(UNIX)

A positive integer up to the maximum imposed by the following equation:

<flimit> - <#nodes> - 5 - `numconnect`

where:

- <flimit> is the maximum number of open files allowed per process, a limit imposed by the operating system

- <#nodes> is the number of included nodes in the node network

- `numconnect` is the value of the `[<YOURHOSTNAME>,unidas]` `numconnect` parameter

This equation ensures the SNC daemon/service has sufficient resources to establish connections to both nodes and to the DAS daemon/service. A value well under this maximum is recommended to avoid possible problems related to values close to operating system limits.

(NT)

A positive integer up to a maximum value of 250.

Default value
(UNIX)

<flimit> - <#nodes> - 5 - `numconnect`

(NT)

`250`

Number of minutes to wait for remote node connection

Section

[SNC]

Parameter

wait_sbh

Description

Specifies the number of minutes to wait if the SNC daemon/service is not able to connect to a remote node.

Accepted values

A positive integer

Default value

5

Number of requests that are reset at a time

Section

[SNC]

Parameter

request_chunk_size

Description

Specifies the number of requests that are reset at a time by the SNC daemon/service. When the SNC daemon/service establishes a connection, it examines the request queue of each local node and resets all requests labelled CANTSERVICE to NOTSERVICED. To minimize the time that another process may be made to wait for access to the node database while the SNC daemon/service resets the request queue (which is in the node database), this parameter allows the resetting to be performed in "chunks" of requests.

Accepted values

A positive integer

Default value

`25`

Block size for communications

Section

`[SNC]`

Parameter

`snc_tr_block`

Description

Specifies the block size, in bytes, to use for communications between a `uniengd` server and a `unidasd` server. Do not change this value without first consulting Oracle support.

Accepted values

`0` (use internal default value)

A positive integer

Default value

`0`

Time-out for received transmissions

Section

`[SNC]`

Parameter

`snc_tr_recv_timeout`

Description

Specifies the time-out value, in seconds, for received transmissions.

Accepted values

`0` (require an immediate response)

A positive integer

Default value

5

Time-out for sent transmissions

Section

[SNC]

Parameter

snc_tr_send_timeout

Description

Specifies the time-out value, in seconds, for sent transmissions.

Accepted values

0 (require an immediate response)

A positive integer

Default value

0

Size of the socket layer receive buffer

Section

[SNC]

Parameter

snc_so_rcvbuf

Description

Specifies the size, in bytes, of the socket layer receive buffer. Do not change this value without first consulting Oracle support.

Accepted values

0 (use internal default value)

A positive integer

Default value

0

Size of the socket layer send buffer

Section

[SNC]

Parameter

snc_so_sndbuf

Description

Specifies the size, in bytes, of the socket layer send buffer. Do not change this value without first consulting Oracle support.

Accepted values

0 (use internal default value)

A positive integer

Default value

0

Idle connections

Section

[SNC]

Parameter

snc_so_keepalive

Description

Determines whether or not the system keeps idle connections active.

If this parameter is set to TRUE, a network packet is sent periodically to determine whether or not the process on the other end of an idle connection is still running. If no acknowledgement is received from that process within a specified period of time, it is assumed to have terminated and the connection is no longer maintained.

If this parameter is set to FALSE, periodic checking on idle connections is not done, and the connections are maintained indefinitely.

Accepted values

TRUE (check idle connections)

FALSE (do not check idle connections)

Default value

TRUE

Statistics logging: user connections

Section

[ENG]

Parameter

activity

Description

Specifies whether or not to log signons and signoffs. The resulting log is useful for tracking server usage and for monitoring possible security violations. If you enable logging, you should closely monitor the size of the log file (/users/unison/log/act.log), as it can grow quickly.

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

FALSE

Statistics logging: user sessions

Section

[ENG]

Parameter

stats

Description

Specifies whether or not to log user session statistics (CPU consumption, user wait times, and network traffic). If you enable logging, you should closely monitor the size of the log file (/users/unison/log/stats.log), as it can grow quickly.

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

FALSE

Character set for log files

Section

[ENG]

Parameter

localcharset

Description

Defines the character set to use for data in log files. For example, if you set this parameter to MSCP932, the server will print all of the logs in the /users/unison/log directory in MSCP932.

This parameter is only checked if [ENG] utf8_autoconvert is set to TRUE.

It is recommended that you set this parameter to the same character set you use for the client. Using two different character sets increases the amount of memory required (both character sets must be loaded) and can affect performance.

Accepted values

HPROMAN

HPROMAN8

ISO-8859-1

MAC-JAPANESE

MAC-ROMAN

MSCP1252

MSCP932 (MSCP932 is a slight variation on Shift_JIS)

Shift_JIS

T61

UTF8

Default value

UTF8

Logging: verbose switch

Section

[CWS]

Parameter

trace

Description

Determines how much information is written to the unicwsd daemon/service log file (/users/unison/log/cws.log). Set this parameter to TRUE to enable a more verbose style of logging whereby each transaction performed by the CWS daemon/service is logged. This causes the log file to grow rapidly and should only be used for a short time for testing or debugging purposes.

Accepted values

TRUE (enable verbose logging)

FALSE (disable verbose logging)

Default value

FALSE

Statistics logging: unidasd

Section

[DAS]

Parameter

stats

Description

Determines whether unidasd statistics are generated and logged to the dasstats.log and the dsstats.log files. If you enable logging, you should closely monitor the size of the log files, as they can grow quickly.

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

FALSE

Logging of failure errors

Section

[ENG]

Parameter

dac_failederrlog

Description

Determines whether errors related to directory server access that appear in the client interface as "unexpected error" are logged to the `/users/unison/log/eng.log` file.

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

TRUE

Logging of configuration errors

Section

[ENG]

Parameter

`dac_configerrlog`

Description

Determines whether three directory server access errors are logged to the `/users/unison/log/eng.log` file. The three errors are: "unable to connect to the SNC daemon/service," "no connections to the directory access (DAS) daemon/service are currently available," and "the number of retries to obtain a connection has been attained; no connections to the directory access (DAS) daemon/service are configured."

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

TRUE

Logging of miscellaneous errors

Section

[ENG]

Parameter

dac_miscerrlog

Description

Determines whether three types of directory server access errors related to the client are logged to the /users/unison/log/eng.log file. The three errors are: password discrepancy due to changes made in the directory server through another application; an LDAP client-side error; an LDAP server-side error.

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

TRUE

Logging of non-critical errors

Section

[ENG]

Parameter

dac_ignorederrlog

Description

Determines whether non-critical directory server access errors are logged to the /users/unison/log/eng.log file.

Accepted values

TRUE (enable logging)

FALSE (disable logging)

Default value

TRUE

Logging of failure to instantiate UTF-8 conversion functionality

Section

[ENG]

Parameter

utf8_onfailprintmesg

Description

Determines whether an error message is logged to
/users/unison/log/eng.log if the server is unable to instantiate UTF-8
conversion functionality for a given user session. Enough information is logged in
the error message to determine why the functionality could not be created.

 Accepted values

TRUE (log an error message)

FALSE (do not log an error message)

Default value

TRUE

Enable item retrieval from directory server

Section

[ENG]

Parameter

dac_itemget

Description

Determines whether item retrieval (from a search or a signon) is from the directory
server or the internal cache. The internal cache gives better performance, but the
item may be slightly out of date.

Accepted values

TRUE (retrieve items from the directory server)

FALSE (retrieve items from the internal cache)

Default value

TRUE

Enable conversion of data to UTF-8 format

Section

[ENG]

Parameter

utf8_autoconvert

Description

Determines whether input data from the client is converted and stored in UTF-8 format by the server.

WARNING: Setting this parameter to FALSE can have adverse effects in installations that support clients on more than one platform or of more than one language.

Accepted values

TRUE (convert input data to UTF-8)

FALSE (do not convert input data)

Default value

TRUE

Number of pages in the database cache

Section

[DB]

Parameter

db_pages

Description

Specifies the number of pages for the database cache. The greater the value, the greater the amount of memory used and the better the performance. As the number increases beyond a certain point, the performance enhancement returns diminish.

Accepted values

A positive integer

Default value

8

Maximum number of database files open per user

Section

[DB]

Parameter

db_files

Description

Specifies the number of database files that may be open at any time for one user session. Increasing this number can improve performance in cases when this limit is repeatedly encountered.

Accepted values

A positive integer up to the maximum set by the operating system for number of open files per process.

Default value

(UNIX)

30

(NT)

170

Node database template

Section

[DBI]

Parameter

dbi_name

Description

Specifies the name of an empty node database to use as a template for node creation. Set during installation, this value should not be edited or removed.

Accepted values

n/a

Default value

n/a

Node database version

Section

[DBI]

Parameter

dbversion

Description

Specifies the node database version number. Set during installation, this value should not be edited or removed.

Accepted values

n/a

Default value

n/a

Database version number

Section

[<YOURNODEID>]

Parameter

version

Description

Specifies the version of the node database. This is a reference value set automatically during node creation. It must NEVER be manually edited.

Accepted values

n/a

Default value

n/a

Specify alternate backup utility

Section

[UTL]

Parameter

external_backup

Description

Specifies an alternate backup utility for unidbbackup to invoke. The server uses the value of this parameter to construct the following command line:

<external_backup value> [-f] -s <src> -d <dst>

where

- <external_backup value> is the value of this parameter

- **-f** indicates that the source is a file (absence of this flag indicates the source is a directory)

- **-s** specifies the source to back up (<src> may be any valid file or directory name)

- **-d** specifies the destination for the backup (<dst> may be any valid file or directory name)

The generated command line must be valid. It may be that you require an intermediate script to take this command line, create one which is valid, and then invoke the valid one. In this case, you set the value of external_backup to the appropriate value for invoking the intermediate script.

Accepted values
A valid path and file name

Default value
None

Specify alternate restore utility

Section
[UTL]

Parameter
external_restore

Description
Specifies an alternate restore utility for unidbrestore to invoke. The server uses the value of this parameter to construct the following command line:

<external_restore value> [-f] -s <src> -d <dst>

where

- <external_restore value> is the value of this parameter

- **-f** indicates that the source is a file (absence of this flag indicates the source is a directory)

- **-s** specifies the source to restore (<src> may be any valid file or directory name)

- **-d** specifies the destination for the restore process (<dst> may be any valid file or directory name)

The generated command line must be valid. It may be that you require an intermediate script to take this command line, create one which is valid, and then invoke the valid one. In this case, you set the value of external_restore to the appropriate value for invoking the intermediate script.

Accepted values
A valid path and file name

Default value
None

Backup operation timeout

Section
[UTL]

Parameter
backup_timeout

Description
Sets the maximum time, in seconds, that unidbbackup will keep any node database locked when using an external backup utility. If a node database is locked for longer than this value, unidbbackup will abort the entire backup operation. This parameter is only used when an alternate backup utility is specified using the [UTL] external_backup parameter. When the unidbbackup utility backs up the calendar database itself, the node backup time is not limited.

Note that the total backup time can easily exceed this value when multiple nodes are involved, since each individual node can take up to this amount of time.

Accepted values
A positive integer

Default value
3600

Restore operation timeout

Section

[UTL]

Parameter

restore_timeout

Description

Sets the timeout, in seconds, for the restore operation on the database.

Accepted values

A postive integer

Default value

3600

Node alias(es)

Section

[<YOURNODEID>]

Parameter

aliases

Description

Specifies the name or names of the nodes configured on a server. When multiple nodes are configured on a server, users must indicate to which node they want to connect. Since, in general, a name is easier to remember than a numeric node-ID, aliases can be configured.

WARNING: Administrators of Lexacom Enterprise Calendar Server installations should avoid changing this parameter once set.

Accepted values

A list of one or more aliases to a maximum of 255 characters, where each alias is an alphanumeric string containing at least one letter and no spaces, and each alias in the list is separated from the next by a comma.

Default value

None

Node name

Section

[<YOURNODEID>]

Parameter

name

Description

Specifies the name of the root directory for the node database found under /users/unison/db/nodes/<name>. The value of this parameter is automatically generated during node creation. The first node created is labelled 'N0', the second 'N1', and following up to 'N9'. Subsequent nodes continue the cycle through the alphabet from O to Z and then from A to L.

Accepted values

A code composed of a letter (A-Z) and a number (0-9)

Default value

n/a

Node time zone

Section

[<YOURNODEID>]

Parameter

timezone

Description

Indicates the time zone of the node. The server sets this parameter when it creates the node. Its value should never be changed.

The server sets this parameter to either the time zone specified by the administrator upon creation of the node, or, if the administrator does not specify one, the value of the `[TIMEZONE] default` parameter.

The `timezone` parameter allows nodes in a node network to have different time zones.

Accepted values

n/a

Default value

None

Relative DN for individual domains

Section

[<YOURNODEID>]

Parameter

relativedn

Description

Lexacom Enterprise Calendar Server only. Determines a location for the domain associated with this node in the LDAP hierarchy relative to the calendar server base DN.

Accepted values

Any valid DN, to a maximum of 255 characters.

Default value

None

Coexistence of LDAP and non-LDAP nodes

Section

[ENG]

Parameter

dir_internal_nodes

Description

Identifies all nodes with an internal directory in an installation where the network requires the coexistence of nodes using an LDAP directory and those with their own internal directory.

Accepted values

Valid node-IDs, separated by a comma and enclosed within {}. For example: dir_
internal_nodes = {10000,10001}

Default value

n/a

User mail mapping

Section

[ENG]

Parameter

usermailmap

Description

Specifies the attribute in the user record that contains users' e-mail addresses. For installations using the calendar server's internal directory only.

Accepted values

A valid X.400 key

Default value

None

Resource mail mapping

Section

`[ENG]`

Parameter

`resourcemailmap`

Description

Specifies the attribute in the resource record that contains resources' e-mail addresses. For installations using the calendar server's internal directory only.

Accepted values

A valid X.400 key

Default value

None

User mobile phone number mapping

Section

`[ENG]`

Parameter

`usermobilemap`

Description

Specifies the attribute in the user record that contains users' mobile phone numbers. For installations using the calendar server's internal directory only.

Accepted values

A valid X.400 key

Default value

None

User mobile phone type mapping

Section

[ENG]

Parameter

usermobiletypemap

Description

Specifies the attribute in the user record that contains users' mobile phone types. For installations using the calendar server's internal directory only.

Accepted values

A valid X.400 key

Default value

None

User alert preference mapping

Section

[ENG]

Parameter

usersmscprefmap

Description

Specifies the attribute in the user record that contains users' preferred notification formats. For installations using the calendar server's internal directory only.

Accepted values

A valid X.400 key

Default value

None

X.400 field for UID

Section

[ENG]

Parameter

itemuidmap

Description

Determines which X.400 field(s) holds the calendar server unique UID. Installations requiring 64 bytes for this information can use the two X.400 fields OU1 and OU2. Do not change the value of this parameter once it has been set. Doing so may result in database corruption. For installations using the calendar server's internal directory only.

Accepted values

A valid X.400 field name

OU1_OU2

Default value

None

Maximum number of items to search

Section

[ENG]

Parameter

userlookthroughlimit

Description

Specifies the maximum number of items (users or resources) the calendar server searches through before ending a search and returning the results to the client.

Accepted values

A positive integer up to a maximum value of `4294967295`

Default value

`4294967295`

Maximum number of items to return

Section

`[ENG]`

Parameter

`usersearchmaxreturn`

Description

Specifies the maximum number of items (users or resources) in a search result. Once the search result contains this number of items, the server ends the search and returns the results to the client.

Accepted values

A positive integer up to a maximum value of `4294967295`

Default value

`4294967295`

Timeout for event search

Section

`[ENG]`

Parameter

`eventsearch_maxlookthroughlimit`

Description

Specifies the maximum time in milliseconds to spend searching events. For native clients v. 5.0 or greater.

Accepted values

A positive integer

Default value

`5000`

Maximum number of events to return

Section

`[ENG]`

Parameter

`evsearch_maxcount`

Description

Specifies the maximum number of events to return from a search. For native clients v. 5.0 or greater.

Accepted values

A positive integer greater than 10.

Default value

`25`

Search event comments

Section

`[ENG]`

Parameter

`eventsearch_commentsearchlength`

Description

Specifies the maximum number of bytes to search through in an event's comments, starting at the beginning. For native clients v. 5.0 or greater.

Accepted values

0 (Disables searching in comments)

A positive integer

Default value

4096

Mail notification

Section

[ENG]

Parameter

notify

Description

Determines whether or not notification by SMTP-based UNIX mail is performed on event creation. This parameter applies only to pre-3.0 versions of the client.

Accepted values

TRUE (enable notification)

FALSE (disable notification)

Default value

TRUE

Set mail transport

Section

[CWS]

Parameter

smtpmail

Description

Determines whether mail notifications are sent using the SMTP mail server. This parameter should always be set to TRUE, regardless of whether mail notification is enabled. Mail notification is enabled or disabled using the [LIMITS] mail parameter.

This parameter supercedes unixmail. For backwards compatibility, if smtpmail is not set, and a value for the unixmail parameter can be found, that value is used.

Accepted values

TRUE (use the SMTP mail server)

FALSE (do not use the SMTP mail server)

Default value

FALSE

Set mail transport (obsolete)

Section

[CWS]

Parameter

unixmail

Description

This parameter is superceded by [CWS] smtpmail.

Mail program

Section

[CWS]

Parameter

smtpmailprogram

Description

Specifies the mail utility for transferring messages to the SMTP mail server. This parameter supercedes `unixmailprogram`. For backwards compatibility, if `smtpmailprogram` is not set, and a value for the `unixmailprogram` parameter can be found, that value is used.

Accepted values

(UNIX)

sendmail

postmail

(NT)

sendmail.exe

Default value

(UNIX)

sendmail

(NT)

sendmail.exe

Mail program (obsolete)

Section

[CWS]

Parameter

unixmailprogram

Description

This parameter is superceded by [CWS] smtpmailprogram.

Pathname of the mail program

Section

[CWS]

Parameter

smtpmailpath

Description

Specifies the directory pathname of the local mail utility. This parameter supercedes unixmailpath. For backwards compatibility, if smtpmailpath is not set, and a value for the unixmailpath parameter can be found, that value is used.

Accepted values

A valid pathname

Default value

(UNIX)

/usr/lib

(NT)

/users/unison/bin

Pathname of the mail program (obsolete)

Section

[CWS]

Parameter

sendmailpath

Description

This parameter is superceded by [CWS] smtpmailpath.

Host name of the SMTP mail server

Section

[CWS]

Parameter

smtpmailhost

Description

Specifies the name of the host on which the SMTP mail server is running. This parameter is meaningful only under NT. It supercedes [CWS] mailhost. For backwards compatibility, if smtpmailhost is not set, and a value for the mailhost parameter can be found, that value is used.

Accepted values

A valid host name

Default value

The host name of the machine on which the calendar server is running.

Host name of the SMTP mail server (obsolete)

Section

[CWS]

Parameter

mailhost

Description

This parameter is superceded by [CWS] smtpmailhost.

Maximum number of recipients

Section

[CWS]

Parameter

smtpmailmaxrecipients

Description

Specifies the maximum number of recipients for a mail message. If a mail is to be sent with more recipients than the value of this parameter, the CWS will split the list of recipients and call the sendmail program multiple times.

See also the [CWS] smtpmailmaxcommandlinesize parameter.

Accepted values

A positive integer

Default value

`100`

Maximum size for sendmail command lines

Section

`[CWS]`

Parameter

`smtpmailmaxcommandlinesize`

Description

Specifies the maximum size of the buffer passed to the sendmail program as a command-line argument. If the buffer size required is larger than this value, the CWS will split the list of recipients and call the sendmail program multiple times.

See also the `[CWS]` `smtpmailmaxrecipients` parameter.

Accepted values

A positive integer

Default value

`1024`

Automatic deletion of temporary file for last mail message

Section

`[CWS]`

Parameter

`mailfiledelete`

Description

Determines whether the temporary file containing the last sent mail message is deleted after the mail is sent. This parameter may be useful to check the calendar server behaviour if you are experiencing a problem with mail delivery.

The temporary file in which the server writes the last mail message can be found at `/users/unison/tmp/MAILMSG`.

Accepted values

TRUE (delete mail messages automatically)

FALSE (do not delete mail messages automatically)

Default value

TRUE

Content of the "From:" field of the mail header

Section

[CWS]

Parameter

mailhdroriginatorfromuser

Description

Determines whether the "From:" field of the mail header is the e-mail address of the sender.

Accepted values

TRUE ("from" field is same as "reply-to" field)

FALSE ("from" field is set to "unison,unison")

Default value

TRUE

UTF-8 for names in "To:" field of mail header

Section
[CWS]

Parameter
mailhdrtoname

Description
Determines whether or not to include names along with addresses in the "To:" field of the mail header. While addresses are constructed using ASCII characters (and hence present no display problem for mail readers), names may contain non-ASCII characters. In cases where the mail reader is unable to display the non-ASCII characters properly, it may be preferable to simply remove the names from the "To:" field altogether.

Accepted values
TRUE (include names)

FALSE (do not include names)

Default value
TRUE

Character set for content portion of mail message

Section
[CWS]

Parameter
mimecontentcharset

Description
Determines the character set to use to encode the content and subject portion of all MIME mail messages sent by the CWS daemon/service. For example, if you set the value of this parameter to MAC-ROMAN, the content and subject portions of all messages sent by the CWS daemon/service will be in MAC-ROMAN.

Accepted values

`"CN-GB"`

`"HPROMAN"`

`"HPROMAN8"`

`"ISO-8859-1"`

`"MAC-JAPANESE"`

`"MAC-ROMAN"`

`"MSCP1252"`

`"MSCP932"`

`"Shift_JIS"`

`"T61"`

`"UTF8"`

Note that the enclosing quotation marks must be present.

Default value

`"ISO-8859-1"`

Character set conversion for e-mail

Section

`[CWS]`

Parameter

`emailcharsetmap`

Description

A mapping table determining what client character sets to convert before sending e-mail. If the character set used by a client is not in this list, then the value of `[CWS]` `mimecontentcharset` is used for e-mail sent by that client.

If this parameter is left blank, the following mapping table will be used:

```
MSCP932     -->  ISO-2022-JP
EUC-JP      -->  ISO-2022-JP
```

```
SHIFT-JIS    -->   ISO-2022-JP
ISO-2022-JP  -->   ISO-2022-JP
MSCP936      -->   ISO-2022-CN
EUC-CN       -->   ISO-2022-CN
HP15CN       -->   ISO-2022-CN
CN-GB        -->   ISO-2022-CN
```

In other words, clients with the MSCP932, EUC-JP, SHIFT-JIS and ISO-20220-JP character sets will send e-mail in ISO-2022-JP; clients with the MSCP936, EUC-CN, HP15CN and CN-GB character sets will send e-mail in ISO-2022-CN.

If the [CWS] emailcharsetmap parameter is used to specify a mapping, that value will be added to this default table, overriding the default values only where necessary. For example, if the value of this parameter were set to {"MSCP932": "SHIFT-JIS"}, all clients using the MSCP932 character set would send mail in SHIFT-JIS, overriding the default mapping listed above. All other mappings listed in the table would still be in effect.

Accepted values

A list of charset mappings separated by commas and enclosed in {}, with the following syntax: "<charset used by client>": "<charset used for e-mail>"

Default value

See above

Character set for content portion of mail message

Section

[LOCALE]

Parameter

charset

Description

Determines the character set to use to encode the content portion of the mail message.

Accepted values

HPROMAN

```
HPROMAN8

ISO-8859-1

MAC-JAPANESE

MAC-ROMAN

MSCP1252

MSCP932

Shift_JIS
```
T61
```
UTF8
```

Default value
ISO-8859-1

Short Message Service (SMS) notification program

Section
[CWS]

Parameter
smsnotifyprogram

Description
Specifies the file name and location of the sendalert utility the calendar server uses to send alerts to Oracle's wireless delivery services.

Accepted values
Any valid path and file name

Default value

Short Message Service (SMS) notification program arguments

Section

[CWS]

Parameter

smsnotifyprogramparam

Description

Specifies the command-line argument that will be passed to the sendalert utility configured by the [CWS] smsnotifyprogram parameter.

Use this parameter to indicate to the sendalert utility the host name and port of your Oracle 9*i*AS PIM Notification Dispatcher. For details on how to find out the host name and port number of your Oracle 9*i*AS PIM Notification Dispatcher, see the "Alerts" chapter of your calendar server *Administrator's Guide*.

Accepted values

"-host <hostname> -port <portnumber>"

Default value

Default time zone

Section

[TIMEZONE]

Parameter

default

Description

Specifies the local time zone.

Accepted values

Any time zone that appears in the /users/unison/misc/timezone.ini file (e.g. EST-5EDT)

Default value
None

Time zone rules

Section
[TIMEZONE]

Parameter
rules

Description
Specifies the name of the file containing time zone rules.

Accepted values
A valid fully-specified file name

Default value
/users/unison/misc/timezone.ini

Checksum of the time zone rules file

Section
[TIMEZONE]

Parameter
checksum

Description
Contains the checksum of the time zone rules file. This value is preset and must not be altered under any circumstance.

Accepted values
n/a

Default value

n/a

Node connection types

Section

[DOMAIN]

Parameter

type

Description

Controls the possibility of connecting nodes. When set to corporate or portal, nodes may be connected only in a single network or cluster. When set to asp, all nodes are always disconnected and may not be linked in a node network.

Accepted values

asp

corporate

portal

Default value

corporate

Master node

Section

[CLUSTER]

Parameter

masternode

Description

Specifies a master node for the cluster.

Accepted values

A valid node-ID belonging to any node in the cluster

Default value

None

Remote master node

Section

[CLUSTER]

Parameter

remotemasternode

Description

Specifies the master node in the cluster. This parameter speeds up the replication of information to the master node when users are created using uniuser.

Accepted values

A valid node-ID belonging to any node in the cluster

Default value

None

Excluded nodes for on-line registration

Section

[CLUSTER]

Parameter

excludednodes

Description

Determines what nodes are excluded from on-line user registration. The server will not create users on listed nodes. Use this parameter to avoid registering users on your cluster's master node, or on nodes that are reaching maximum capacity.

Accepted values

A list of valid node-IDs or aliases belonging to any nodes in the cluster, separated by commas and enclosed in {}. For example:

```
{ 14, 446, 447 }
```

Default value

```
{ }
```

Allow SYSOP logons from uniical

Section

```
[ENG]
```

Parameter

```
allowsysoplogon_uniical
```

Description

Specifies whether `uniical` users may log in to the server as SYSOP.

Accepted values

`TRUE` (SYSOP logons allowed)

`FALSE` (SYSOP logons not allowed)

Default value

`TRUE`

Allow SYSOP logons from unicp utilities

Section

```
[ENG]
```

Parameter

```
allowsysoplogon_unicp
```

Description

Specifies whether users of the `unicp*` family of utilities may log in to the server as SYSOP.

Accepted values

`TRUE` (SYSOP logons allowed)

`FALSE` (SYSOP logons not allowed)

Default value

`TRUE`

Allow SYSOP logons for CAPI applications

Section

`[ENG]`

Parameter

`allowsysoplogon_capi`

Description

Specifies whether applications using Oracle's Calendar API (CAPI) can log in to the server as SYSOP.

Accepted values

`TRUE` (SYSOP logons allowed)

`FALSE` (SYSOP logons not allowed)

Default value

`FALSE`

Controlling server interactions with directory server

Set directory service

Section
[DAS]

Parameter
dir_service

Description
Specifies the directory service that the unidasd daemon/service accesses. The value of this parameter is also found as the name of the section in the /user/unison/misc/unison.ini file where the configuration parameters for the directory service are found. For example, if dir_service = LDAP, the [LDAP] section of the unison.ini file contains the LDAP directory service configuration parameters.

Accepted values
LDAP

Default value
LDAP

Name of directory server

Section
[LDAP]

Parameter
dsa

Description
Specifies the name of the LDAP directory server. This parameter is set during installation. Changing the value of this parameter may result in directory server corruption.

Accepted values

`OID` (Oracle Internet Directory)

Default value

None

Name of directory server host

Section

`[LDAP]`

Parameter

`host`

Description

Specifies the name of the machine hosting the LDAP directory server. If failovers for the directory server have been configured, they may be listed here. By default, the calendar server will attempt to establish a connection to the first server listed; if unable to do so, it will try the next.

Accepted values

A valid host name, fully-qualified domain name, or IP address

A list of directory servers, in the form `"<hostname>[:<port>] <hostname>[:<port>]"` For example:

`"host1:389 host2:389"`

Default value

None

Port number of the LDAP directory server

Section

`[LDAP]`

Parameter

`port`

Description

Specifies the port number of the LDAP directory server. If the [LDAP] host parameter contains a port number, the value of the [LDAP] port parameter will be ignored.

Accepted values

A valid port number

Default value

389

Character set used by the directory server

Section

[LDAP]

Parameter

charset

Description

Indicates the character set the LDAP directory server uses. This is the character set that the calendar server must use for data destined for the LDAP directory server.

Accepted values

HPROMAN

HPROMAN8

ISO-8859-1

MAC-JAPANESE

MAC-ROMAN

MSCP1252

MSCP932

Shift_JIS

T61

UTF8

Default value

UTF-8

LDAP protocol version

Section

[LDAP]

Parameter

attr_version

Description

Specifies the LDAP protocol version used by the directory server. Set the value of this parameter to 2 only if your LDAP directory server does not support the LDAPv3protocol.

Accepted values

2 (for LDAPv2)

3 (for LDAPv3)

Default value

3

Permissions for modifying directory server attributes

Section

[<YOURHOSTNAME>,unidas]

Parameter

dir_adminmodcalonly

Description

Specifies whether the calendar server's reserved users can modify all directory server attributes or only the calendar attributes in the directory server.

Accepted values

TRUE (permit modifications to calendar attributes only)

FALSE (permit modifications to all attributes)

Default value

FALSE

Permissions for updating directory server attributes

Section

[<YOURHOSTNAME>,unidas]

Parameter

dir_adminupdcalonly

Description

Specifies whether the calendar server's reserved users can update all directory server attributes or only the calendar attributes in the directory server.

Accepted values

TRUE (permit updates to calendar attributes only)

FALSE (permit updates to all attributes)

Default value

FALSE

Attribute preserve list

Section

[LDAP]

Parameter

attrpreservelist

Description

Specifies a list of attributes (a "preserve list") which are not to be deleted when a calendar user is deleted (i.e. when the user's calendar attributes as well as their `ctCalUser` object class are deleted). If the calendar user entries also use the `inetOrgPerson` object class, you should configure this parameter as follows:

`{employeeNumber, givenName, initials, mail, ou}`

Any fields mapped to attributes outside of the `ctCalUser` object class (e.g. `attr_organization = uid`) should also be added to the "preserve list".

Accepted values

A list of strings, separated by commas and enclosed in {}, where each string in the list is the name of a user attribute

Default value

`{}`

Calendar unique user identifier mapping

Section

`[LDAP]`

Parameter

`attr_uid`

Description

Determines the directory server attribute name that the calendar server uses as a unique user identifier. Do not change the value of this parameter unless you also change the attribute your Oracle Internet Directory uses to authenticate SSO logins. If you change that attribute on your directory server, you must change the value of this parameter.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

`uid`

Name of the "country" attribute

Section

[LDAP]

Parameter

attr_country

Description

Determines the attribute name that the LDAP directory server uses for the "country" attribute.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

None

Name of the "generation qualifier" attribute

Section

[LDAP]

Parameter

attr_generation

Description

Determines the attribute name that the LDAP directory server uses for the "generation qualifier" attribute.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

generationQualifier

Name of the "organization" attribute

Section

[LDAP]

Parameter

attr_organization

Description

Determines the attribute name that the LDAP directory server uses for the "organization" attribute.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

None

Name of the "given name" attribute

Section

[LDAP]

Parameter

attr_givenname

Description

Determines the attribute name that the LDAP directory server uses for the "given name" attribute.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

givenName

Name of the "mail" attribute

Section

[LDAP]

Parameter

attr_mail

Description

Determines the attribute name that the LDAP directory server uses for the "mail" attribute.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

mail

Name of the "uid" attribute

Section

[LDAP]

Parameter

attr_uid

Description

Determines the attribute name that the LDAP directory server uses for the "uid" attribute.

Accepted values

Any attribute name defined in the LDAP directory server schema

Default value

uid

Name of group object class

Section
[LDAP]

Parameter
groupobjectclass

Description
Determines the name of the object class for the group of calendar server administrators.

Accepted values
groupOfUniqueNames

Default value
groupOfUniqueNames

Name of "member" attribute in group object class

Section
[LDAP]

Parameter
groupmemberattribute

Description
Determines the name of the "member" attribute that is in the object class specified by the groupobjectclass parameter.

Accepted values
uniqueMember

Default value
uniqueMember

Location of the calendar server administrators

Section

[LDAP]

Parameter

admin

Description

Specifies part of the Distinguished Name (DN) of the location under which calendar server administrators will be created. The DN of this location is constructed by appending the value of the basedn parameter to the value of the admin parameter. For example, where admin = "ou=calendar servers" and basedn = "o=acme", the DN for the location under which calendar server administrators will be created is "ou=calendar servers, o=acme".

Accepted values

A valid Distinguished Name or Relative Distinguished Name (see your LDAP directory server documentation for further information on the correct form)

Default value

None

Group entry for calendar server administrators

Section

[LDAP]

Parameter

admingroup

Description

Specifies part of the Distinguished Name (DN) of the group entry for calendar server administrators (the administrators are added to this group). The DN of the group entry is constructed by appending the value of the basedn parameter to the value of the admingroup parameter. For example, where admingroup = "cn=calendar server admins" and basedn = "o=acme", the DN for the

group entry of calendar server administrators is `"cn=calendar server admins, o=acme"`.

Accepted values
A valid Distinguished Name or Relative Distinguished Name (see your LDAP directory server documentation for further information on the correct form)

Default value
`cn=Calendar Server Admins`

Distinguished Name of the subtree containing calendar server entries

Section
`[LDAP]`

Parameter
`basedn`

Description
Specifies the Distinguished Name of the LDAP directory server subtree containing calendar entries.

Accepted values
A valid Distinguished Name of a maximum of 255 characters (see your LDAP directory server documentation for further information on the correct form)

Default value
None

Relative Distinguished Name for resources

Section
`[LDAP]`

Parameter

`resourcerelativedn`

Description

Specifies a location for resources in the LDAP directory relative to the calendar server base DN (specified by the value of the `[LDAP] basedn` parameter).

If a full Distinguished Name is specified when creating a new resource, that value will be used and the value of this parameter will be disregarded.

Accepted values

A valid Distinguished Name (see your LDAP directory server documentation for further information on the correct form)

Default value

None

Resource Relative DN for individual domains

Section

`[<YOURNODEID>]`

Parameter

`resourcerelativedn`

Description

For Lexacom Enterprise Calendar Server only. Specifies the Resource Relative DN in which to locate resources for the domain associated with this node. This DN is relative to the domain's Relative DN (see the `[YOURNODEID] relativedn` parameter).

For all servers, if a full Distinguished Name is specified when creating a new resource, that value will be used and the value of this parameter will be disregarded.

Accepted values

A valid Distinguished Name of a maximum of 255 characters (see your LDAP directory server documentation for further information on the correct form)

Default value

None

Distinguished Name used for anonymous connections

Section

[LDAP]

Parameter

binddn

Description

Specifies the Distinguished Name the LDAP directory server uses for anonymous connections.

Accepted values

A valid Distinguished Name (see your LDAP directory server documentation for further information on the correct form)

Default value

None

Distinguished Name of the directory server administrator

Section

[LDAP]

Parameter

mgrdn

Description

Specifies the Distinguished Name of the LDAP directory server administrator.

Accepted values

A valid Distinguished Name (see your LDAP directory server documentation for further information on the correct form)

Default value

None

Password used for anonymous connections

Section

[LDAP]

Parameter

bindpwd

Description

Specifies the password for the LDAP user specified by the value of the [LDAP]
binddn parameter.

You must encrypt the password using the uniencrypt utility before entering it in
the unison.ini file. See the uniencrypt documentation in Appendix C, "Utilities".
The encrypted password must be preceded by the encrpytion method used to
generate it (acipher in most cases) and enclosed in double-quotes.

Accepted values

"{acipher}<encrypted_value>"

Default value

None

Distinguished Name used for write operations

Section

[LDAP]

Parameter

writedn

Description

Specifies the Distinguished Name the calendar server uses for all write operations
on the directory server.

Accepted values

A valid Distinguished Name (see your LDAP directory server documentation for further information on the correct form)

Default value

None

Password used for LDAP write connections

Section

[LDAP]

Parameter

writednpassword

Description

Specifies the password for the LDAP user specified by the value of the [LDAP] writedn parameter.

You must encrypt the password using the uniencrypt utility before entering it in the unison.ini file. See the uniencrypt documentation in Appendix C, "Utilities". The encrypted password must be preceded by the encrpytion method used to generate it (acipher in most cases) and enclosed in double-quotes.

Accepted values

"{acipher}<encrypted_value>"

Default value

None

Modification of user password

Section

[UTL]

Parameter

adm_moduserpassword

Description

Determines whether the calendar server administrator can modify the password of a calendar user.

Accepted values

TRUE (can modify the password)

FALSE (cannot modify the password)

Default value

TRUE

Directory server support for ldap_compare() requests

Section

[LDAP]

Parameter

dsausesearchtocompare

Description

Indicates whether the LDAP directory server handles ldap_compare() calls properly or must use ldap_search() calls to simulate ldap compare() calls.

Accepted values

TRUE (uses ldap_search() calls)

FALSE (handles ldap_compare() calls properly)

Default value

FALSE

Search filter for groups

Section

[LDAP]

Parameter

groupfilter

Description

Specifies the LDAP filter the calendar server uses when searching groups in the directory server.

The default value of this parameter exposes all groups to the calendar client; users will be able to see all groups in the directory server, and any members of those groups who are also calendar users. However, if there are groups in the directory server that consist entirely of non-calendar users, the calendar client will display these groups with no members.

To avoid this, you may wish to create a custom object class such as "calendarGroup", and apply this object class only to the LDAP groups that you wish to be visible through the calendar client. Then, extend the value of this parameter to include that object class. For example, the new value might be:

(&(member=*)(objectclass=groupOfNames)(objectclass=calendarGroup))

For more details, see Chapter 3 of your calendar server *Administrator's Guide*.

Accepted values

Any valid filter, up to a maximum length of 150 characters

Default value

(&(uniqueMember=*)(objectclass=groupOfUniqueNames))

Maximum time to wait on an LDAP call

Section

[LDAP]

Parameter

timelimit

Description

Specifies the maximum time, in seconds, that the server waits on an LDAP call before returning a timeout error to the client. Note that the timeout settings in the directory server take precedence over this parameter.

Accepted values

0 or a positive integer. A value of 0 means no timeout ever occurs and causes the server to wait until the directory server returns either a result or an error.

Default value

120

Enable SSL connections

Section

[LDAP]

Parameter

security

Description

Enables SSL connections to the LDAP directory server. If this parameter is set to TRUE, the calendar server uses the LDAPSSL functions contained in the libraries specified by the [LDAP] ldapssllibname and [LDAP] lberssllibname parameters to connect to the directory server. Otherwise it uses the LDAP functions contained in the libraries specified by the [LDAP] ldaplibname and [LDAP] lberlibname parameters to connect to the directory server.

Accepted values

TRUE (enable SSL connections)

FALSE (disable SSL connections)

Default value

FALSE

Port to use for SSL connections

Section

[LDAP]

Parameter

secure-port

Description

Determines the port to use for SSL connections. This parameter is only checked if [LDAP] security is set to TRUE.

Accepted values

Any value in the range 1 to 65535

Default value

636

Location of file containing trusted certificates

Section

[LDAP]

Parameter

certdbpath

Description

Specifies the path and/or file name of the file containing the certificates used by the SSL. This parameter is only used when the value of the [LDAP] security parameter is set to TRUE.

Accepted values

A valid path and file name

A valid file name

cert7.db (the database file used by Netscape Communicator 4.x and up)

A valid path name. In this case the calendar server adds the file name cert7.db to the path.

Note that on Windows platforms, any path information specified in the value of this parameter must use UNIX syntax, with forward-slashes '/' instead of backslashes

'\', and omitting the drive letter from the beginning of the string. The file must also exist on the same drive as the calendar server installation directory.

Default value

cert7.db

Controlling client behaviour

Allow agenda attachments

Section

[LIMITS]

Parameter

allowattachments

Description

Determines whether or not the client allows attachments for meetings or tasks.

Accepted values

TRUE (allow attachments)

FALSE (do not allow attachments)

Default value

FALSE

Maximum size of attachments

Section

[LIMITS]

Parameter

maxattachmentsize

Description

Determines the maximum size, in bytes, for attachments to meetings, tasks and other agenda entries. This parameter is only checked if the [LIMITS] allowattachments parameter is set to TRUE. Suggested value is 102400 (100K).

Accepted values

Any positive integer up to a maximum value of (2^{32}-1)

Default value

2^{32}-1

Minimum interval for checks for new agenda entries (client-side enforcement)

Section

[LIMITS]

Parameter

autocontrol

Description

Determines the minimum number of minutes that a user can set as the interval between agenda refresh calls to the server (i.e. between each check for new agenda entries).

If this value is less than lck_users/60, the value of lck_users/60 takes precedence, to a maximum value of 45. For example, if autocontrol = 15 and lck_users = 1200, no refresh occurs before 20 (i.e. 1200/60) minutes has elapsed.

Note that this parameter has been superseded by the [CLIENT] minrefreshrate parameter, which enforces the behaviour on the server side instead of on the client side. It is included here for backward compatibility with older clients.

Accepted values

Any positive integer up to a maximum value of (2^{16}-1)

Default value

15

Minimum interval for checks for new agenda entries (server-side enforcement)

Section

[CLIENT]

Parameter

minrefreshrate

Description

Determines the minimum number of minutes that a user can set as the interval between agenda refresh calls to the server (i.e. between each check for new agenda entries).

Note that this value overrides the [LIMITS] autocontrol parameter, and does not take into account the value of the [LCK] lck_users parameter as autocontrol does.

Note also that setting the value of this parameter too low can have serious consequences upon the performance of the calendar system. The more system resources and database access time are devoted to automatic idle refreshes, the slower the perceived performance of on-demand requests can become. Tune this parameter according to the number of logged-on users you experience at peak hours, and according to the number of database requests per second your hardware can comfortably accommodate.

For example, if testing has established acceptable performance benchmarks at one automatic refresh request per second, then for an environment of 1000 users, this parameter should not be set to an interval lower than 1000 seconds, or approximately seventeen minutes. The value provided at installation time should serve as an acceptable limit for all but the most exceptional installations.

Accepted values

Any positive integer up to a maximum value of (2^{16}-1)

Default value

15

Maximum number of instances for a repeating meeting, daily note, or day event (client-side)

Section

`[LIMITS]`

Parameter

`maxrecur`

Description

Specifies the maximum number of instances the client allows a user to create for a single repeating meeting, daily note, or day event.

This parameter is now outdated. For Windows and Macintosh clients v. 5.0 and above, use the `[ENG] maxinstances` parameter instead to control this behaviour. However, it is recommended that you ensure the `[LIMITS] maxrecur` and `[ENG] maxinstances` parameters be set to the same value, to ensure full compatibility between all clients.

Accepted values

A positive integer up to a maximum value of $(2^{32}-1)$

Default value

`60` (60 instances per meeting, note, or day event)

Maximum number of instances of a recurring meeting, daily note, or day event (server-side)

Section

`[ENG]`

Parameter

`maxinstances`

Description

Determines the maximum number of instances of a recurring meeting, daily note, or day event the calendar server can create. It is recommended that you ensure the

[LIMITS] maxrecur parameter be set to the same value as [ENG] maxinstances to ensure full compatibility between all clients.

Accepted values
An integer in the range 1-65535

Default value
100

Maximum lead time on a reminder

Section
[LIMITS]

Parameter
maxremleadtime

Description
Specifies the maximum number of days in advance of an event that a user can set a reminder to ring.

Accepted values
Any positive integer up to a maximum value of $(2^{32}-1)$

Default value
21

Double-booking resources (client-side)

Section
[LIMITS]

Parameter
resourceconflicts

Description

Determines whether the client allows users to double-book resources. This parameter should always be set with the same value as the `[ENG]` `allowresourceconflict` parameter.

Accepted values

TRUE (allow double-bookings)

FALSE (do not allow double-bookings)

Default value

TRUE

Double-booking resources (server-side)

Section

`[ENG]`

Parameter

`allowresourceconflict`

Description

Determines whether the server allows double-booking of resources. This parameter should always be set with the same value as the `[LIMITS] resourceconflicts` parameter.

Accepted values

TRUE (allow double-bookings)

FALSE (do not allow double-bookings)

Default value

FALSE

Default agenda view

Section

`[LIMITS]`

Parameter

`agendaview`

Description

Determines the default view in which the client opens agenda windows. For clients 4.1 and above.

Accepted values

`0` (day view)

`1` (week view)

`2` (month view)

Default value

`0`

Maximum number of open windows

Section

`[LIMITS]`

Parameter

`maxwinopen`

Description

Determines the maximum number of windows (views) that can be opened at the same time in the user interface.

Accepted values

Any positive integer up to a maximum value of $(2^{32}-1)$

Default value

`7`

Maximum number of users in a group view

Section

[LIMITS]

Parameter

groupviewmax

Description

Specifies the maximum number of users that the client can display in a group view.

Accepted values

A positive integer up to the value of $(2^{32}-1)$

Default value

100

Allow resources in remote nodes to appear as local

Section

[<YOURNODEID>]

Parameter

localnodes

Description

Specifies which remote resources to consider local for client scheduling purposes. If you want users on separate but connected nodes to view and treat all resources as local (a common situation when two or more nodes are in close geographic proximity), enter the relevant node-ID(s) after this parameter. Nodes must be connected to enable this feature.

Accepted values

Valid node-IDs, separated by a comma

Default value

n/a

Case-sensitivity of passwords

Section

[ENG]

Parameter

passwords

Description

Determines whether client password verification is case-sensitive. Only used for installations using the calendar server's internal directory.

Accepted values

case (case sensitive)

ignorecase (case insensitive)

Default value

case

Allow automatic sign-in

Section

[LIMITS]

Parameter

ssignin

Description

Determines whether a user can use the desktop clients' automatic sign-in feature to sign in to the calendar server without providing a password. For Windows clients 4.2 and above and Macintosh clients 4.5 and above.

See also the [LIMITS] ssigninrestrictions parameter.

Accepted values

TRUE (allow automatic sign-in)

FALSE (force user to always supply a password)

Default value

FALSE

Restrictions on automatic sign-in

Section

[LIMITS]

Parameter

ssigninrestrictions

Description

Restricts the automatic sign-in feature of desktop clients to secure operating systems. The automatic sign-in feature will be unavailable for clients running on Windows 95/98 and Mac OS 7/8.

See also the [LIMITS] ssignin parameter.

Accepted values

TRUE (restrict automatic sign-in to secure operating systems)

FALSE (allow automatic sign-in from any operating system)

Default value

TRUE

Password aging

Section

[LIMITS]

Parameter

maxpasswordage

Description

Controls password aging. The value represents the number of days that a password can exist before users are required to change it.

Accepted values

Any positive integer up to a maximum value of (2^{32}-1)

Default value

`76543` (for all practical purposes, password aging is OFF)

Maximum number of sign-in attempts

Section

`[LIMITS]`

Parameter

`signinmaxattempts`

Description

Determines how many unsuccessful sign-in attempts are allowed before the client closes. Native clients v. 5.0 and greater only.

Accepted values

An integer between 1 and 2^{32}

Default value

`5`

Show multiple user matches on sign-in

Section

`[LIMITS]`

Parameter

`userlist_login`

Description

Determines whether or not to show a list of matching users when more than one fits the specified sign-in credentials. Native clients v. 6.0 and greater only.

Accepted values

TRUE (Display the list of matching users)

FALSE (Don't display the list)

Default value

TRUE

Secure sign-in

Section

[LIMITS]

Parameter

secure-login

Description

Determines whether or not to restrict information given about incorrect sign-in credentials. Native clients v. 6.0 and greater and Web clients v. 3.0 and greater only.

Accepted values

TRUE (Display only that the credentials supplied are incorrect)

FALSE (Display more user-friendly error messages on incorrect sign-in)

Default value

FALSE

Single local storage

Section

[LIMITS]

Parameter

`singlelst`

Description

Disables the client's Different Local Storage dialog. If this parameter is set to `TRUE`, only one user may access the calendar server from a given client machine. If another user tries to sign in, he or she will be forced to quit the application.

Accepted values

`TRUE` (disable different local storage)

`FALSE` (allow different local storage)

Default value

`FALSE`

Allow users to update only calendar attributes

Section

`[DAS]`

Parameter

`dir_updcalonly`

Description

Determines whether users can update only calendar attributes, or calendar and non-calendar attributes within their entries.

Accepted values

`TRUE` (permit updates only to calendar attributes)

`FALSE` (permit updates to any attributes)

Default value

`FALSE`

Force DAS sign-on and sign-off for read-only operations

Section
[DAS]

Parameter
dir_itembindonread

Description
Forces the Directory Access Server to sign in to the directory server for read-only operations (e.g. searching).

Read operations are carried out using the anonymous profile. If that profile has limited rights, directory server searches will be constrained by those rights.

Note that write operations always require a bind.

Accepted values
TRUE (demand credentials for searches)

FALSE (permit searches without calling credentials)

Default value
FALSE

Right to create public groups

Section
[LIMITS]

Parameter
pubgroups

Description
Determines whether users holding the necessary access rights can create public groups.

Accepted values

TRUE (permit users to create public groups)

FALSE (do not permit users to create public groups)

Default value

TRUE

Permission to change default timezone

Section

[LIMITS]

Parameter

settimezone

Description

Determines whether the user is permitted to save time zone changes for future client sessions.

Accepted values

TRUE (permit users to set a different time zone)

FALSE (do not permit users to set a different time zone)

Default value

FALSE

Enable mail notification

Section

[LIMITS]

Parameter

mail

Description

Determines whether mail notification is enabled, regardless of how the client is set up.

Accepted values

TRUE (enable mail notification)

FALSE (disable mail notification)

Default value

TRUE

Short Message Service (SMS) alerts

Section

[NOTIFY]

Parameter

sms

Description

Determines whether wireless alerts are enabled. See your calendar server Administrator's Guide for details on the available wireless alert services.

Accepted values

TRUE (enable SMS notification)

FALSE (disable SMS notification)

Default value

FALSE

Maximum number of people in a mail notification distribution list

Section

[LIMITS]

Parameter

maxmaildistr

Description

Specifies the maximum number of users in a mail notification distribution list.

Accepted values

Any positive integer up to a maximum value of $(2^{32}-1)$

Default value

30

Minimum number of characters in the Surname edit box

Section

[LIMITS]

Parameter

mincharsearch

Description

Determines the minimum number of search characters that the user must supply in the Surname field when performing a directory search from the client. The default value of 0 allows a user to execute a search without limits and retrieve the complete database of users.

Accepted values

0

A positive integer up to a maximum value of $(2^{32}-1)$

Default value

0

Maximum number of search results displayed

Section
[LIMITS]

Parameter
maxsearchresult

Description
Determines the maximum number of items (users or resources) that the search dialogue box lists in both the calendar clients and the Admin GUI. Tune this parameter relative to the size of your installation. If you use an external directory server, tune this parameter to match any search limits configured in the directory server. Consult the relevant CorporateConnect appendix for your directory server, or your directory server documentation to determine what these limits are and how to configure them.

Accepted values
Any positive integer up to the value of $(2^{32}-1)$

Default value
100 (list only 100 entries at a time)

"Next" button in search dialogue box

Section
[LIMITS]

Parameter
page-forward

Description
Determines whether the "next" button is enabled in the item search dialogue box for users and resources of the calendar clients and the Admin GUI.

When a search is performed, the [LIMITS] maxsearchresult parameter determines the maximum number of search results to return to the client. Assume page-forward is TRUE, maxsearchresult is set to 100, and you search for all

users whose surname begins with "S". If there are 220 such users in the database, the search dialogue will present you with the first 100 users. You may then click on the "next" button to see the next 100 users, and click again to see the last 20.

This functionality is disabled when the server is connected to a directory server.

Accepted values

TRUE (enable the "next" button)

FALSE (disable the "next" button)

Default value

TRUE

"Previous" button in search dialogue box

Section

[LIMITS]

Parameter

page-backward

Description

Determines whether the "previous" button is enabled in the search dialogue box for users and resources. This button performs the reverse operation of the [LIMITS] page-forward parameter, allowing the user to return to previously-listed entries of the search result.

This functionality is disabled when the calendar server is connected to a directory server.

Accepted values

TRUE (enable the "previous" button)

FALSE (disable the "previous" button)

Default value

TRUE

Size of the client event search result window

Section
[ENG]

Parameter
eventsearch_clientwindowsize

Description
Specifies the number of rows in the client's event search result list box. For native clients v. 5.0 or greater.

Accepted values
A positive integer

Default value
20

Number of days preceding current date to consult or return for queries

Section
[OUTLOOK_CONNECTOR]

Parameter
eventselectbegin

Description
For Oracle Outlook Connector only. Sets the number of days preceding the current date that will be searched or returned for all database queries.

Accepted values
Any positive integer up to the value of the number of days between the current date and January 1, 1991.

Default value

180

Number of days following current date to consult or return for queries

Section

[OUTLOOK_CONNECTOR]

Parameter

eventselectend

Description

For Oracle Outlook Connector only. Sets the number of days following the current date that will be searched or returned for all database queries.

Accepted values

Any positive integer up to the value of the number of days between the current date and December 31, 2037.

Default value

730

Enable address books

Section

[GENPREFS]

Parameter

offlineab

Description

For native clients 5.1 and above. Enables and disables the use of address books.

Accepted values

TRUE (enable address books)

`FALSE` (disable address books)

Default value

`TRUE`

Enable publishing of address books

Section

`[LIMITS]`

Parameter

`publishab`

Description

For clients 4.1 and above. Enables the publishing of address books.

Accepted values

`TRUE` (enable the publishing of address books)

`FALSE` (disable the publishing of address books)

Default value

`TRUE`

Maximum number of personal address book entries

Section

`[LIMITS]`

Parameter

`maxpersabentries`

Description

For clients 4.0 and above. Determines the maximum number of personal address book entries.

Accepted values

Any positive integer up to a maximum value of $(2^{32}-1)$

Default value

2000

Maximum number of entries in a folder

Section

[QUOTA]

Parameter

maxfolderentryperuser

Description

Determines the maximum number of entries in an address book folder.

Accepted values

0 (no entries)

A positive integer up to a maximum value of $(2^{32}-1)$

Default value

2000

Refresh intervals and agenda ranges

Section

[ENG]

Parameter

eventrefreshintervals

Description

Configures the refresh intervals and agenda ranges, in seconds, that Oracle Outlook Connector uses when it queries the server for new events.

This parameter is a list of intervals, separated by commas and enclosed in {}. Each interval in the list has the following format:

: -<lower bound> +<upper bound>

where <interval>, <lower bound> and <upper bound> are all expressed in seconds. The <interval> determines the refresh interval. The <lower bound> and <upper bound> determine a range of time.

Every <interval> specifies a time when the calendar server should refresh Outlook with the associated range of agenda data. For example, the entry {900: -0 +172800} specifies that every 15 minutes (<interval> of 900 seconds) the server should refresh Outlook with an agenda range beginning at the current time (<lower bound> of 0 seconds) and continuing through 2 days following (<upper bound> of 172800 seconds).

Accepted values

<interval>, <lower bound> and <upper bound> are integers in the range 0-65535

Default value

{0: -86400 +518400, 2700: -604800 +10886400, 79200: -0 +0}

The first interval specifies that all client-initiated queries for events have a minimum agenda range of one day previous (-86400) through to six days (+518400) following the time of the query. The second interval tells the client to query the server every 45 minutes (2700) for events in the range of one week previous (-604800) to six weeks (+10886400) from the time of the query. The third interval tells the client to query the server every 22 hours (79200) for all events.

On-line help display

Section

[LIMITS]

Parameter

browserhelp

Description

For Windows clients version 4.5 or greater. Displays the on-line help in an ordinary web browser rather than the Microsoft HTML Help Viewer. If a browser is specified by the [WEBLINK] browser-path-win parameter, that browser is used.

For best results, please ensure that Java and Javascript are enabled in the browser.

Accepted values

`TRUE` (Forces on-line help to be displayed in a browser)

`FALSE` (Displays on-line help in Microsoft's HTML Help Viewer, if installed; if not, displays on-line help in a browser)

Default value

`FALSE`

Enabling web access

Section

`[WEBLINK]`

Parameter

`mode`

Description

Determines whether web access is enabled in the client, and how it is configured. This functionality gives Windows and Macintosh clients the ability to access and display a web page.

If this parameter is set to `custom`, ALL of the other `[WEBLINK]` parameters must have values specified in the `unison.ini` file. If this is not the case, the value of this parameter reverts to the default (`cst`).

Accepted values

`off` or `cst` (disable web access)

`custom` (enable web access and take values for all of the other `[WEBLINK]` parameters from `unison.ini`.)

Default value

`cst`

Browser to launch for Windows clients

Section
[WEBLINK]

Parameter
browser-path-win

Description
For Windows clients, this parameter determines which browser to launch for web access. For Windows clients 4.5 and greater, it also determines a browser for the on-line help when [LIMITS] browserhelp is set to TRUE, or when Microsoft HTML Help Viewer is not installed. This parameter determines the location of the web browser on the local machine of each signed-on user.

Accepted values
A valid path and filename

Default value
None

Text to appear in Help menu and ToolTip

Section
[WEBLINK]

Parameter
command-name

Description
Determines the text string that appears in the Help menu for the web access item, as well as in the pop-up ToolTip accompanying the Toolbar web access icon.

Accepted values
A string with a maximum of 150 characters in length

Default value

(when [WEBLINK] mode = cst) Product Information

On-line command description

Section

[WEBLINK]

Parameter

command-description-online

Description

For Windows clients, determines the description of the web access command that appears on the status bar when the client is on-line. For Macintosh clients 4.2 or earlier, determines the description of the web access command that appears in balloon help when the client is on-line.

Accepted values

A string with a maximum of 150 characters in length

Default value

(when [WEBLINK] mode = cst) Click here to access product information on the web.

Off-line command description

Section

[WEBLINK]

Parameter

command-description-offline

Description

For Windows clients, determines the description of the web access command that appears on the status bar when the client is off-line. For Macintosh clients 4.2 and earlier, determines the description of the web access command that appears in balloon help when the client is off-line.

Accepted values

A string with a maximum of 150 characters in length.

Default value

(when `[WEBLINK]` mode = cst) `Click here to access product information on the web.`

File transfer protocol

Section

`[WEBLINK]`

Parameter

`download-mode`

Description

Determines the file transfer protocol to use when downloading the web pages for viewing in off-line mode.

Accepted values

`file` (use the file sharing protocol)

Default value

(when `[WEBLINK]` mode = cst) `file`

Web page to load for clients working on-line

Section

`[WEBLINK]`

Parameter

`online-url`

Description

Determines the web page to load for clients working on-line.

Accepted values

A valid URL with a maximum of 150 characters in length

Default value

None

Source of web pages for Windows clients in off-line mode

Section

[WEBLINK]

Parameter

offline-source-win

Description

For Windows clients. Determines the source of the web page to display when working in off-line mode. This parameter is only checked when [WEBLINK] mode = custom.

Accepted values

A valid path and file name with a maximum of 150 characters in length

Default value

None

Source of web pages for Mac clients in off-line mode

Section

[WEBLINK]

Parameter

offline-source-mac

Description

For Macintosh clients. Determines the source of the web page to display when working in off-line mode.

Accepted values

A string with a maximum of 150 characters in length which obeys the following format and restrictions:

`<user>:<pw>@[<zone>]:<AFP Server Name>[,<ip>[,<port>]];<volume path>:`

where:

- `<user>` is a maximum of 31 characters in length
- `<pw>` is a maximum of 8 characters in length
- `<zone>` is a maximum of 31 characters in length; default value is the asterisk symbol "*" (without the quotes)
- `<AFP Server Name>` is a maximum of 31 characters in length
- `<ip>` is a maximum of 31 characters in length
- `<port>` is a numerical value 0 to 32767 inclusive; default value is 548
- `<volume path>` is a maximum of 64 characters in length

This parameter is only checked when `[WEBLINK] mode = custom`.

Default value

None

Version of off-line web page

Section

`[WEBLINK]`

Parameter

`offline-source-version`

Description

Determines the version of the off-line web page. This provides a mechanism for the administrator to ensure that the most recent version of the off-line web page is the one being accessed. This parameter is only checked when `[WEBLINK] mode = custom`.

Accepted values

A character string with a maximum length of 150 characters.

Default value

None

Error message to display if download fails

Section

[WEBLINK]

Parameter

download-fail

Description

Determines the error message to display if a download fails.

Accepted values

A character string with a maximum length of 150 characters.

Default value

(when [WEBLINK] mode = cst) Your on-line account did not supply all the information needed to support this operation. Please contact your system administrator.

Enabling French-language web access

Section

[WEBLINK-FRENCH]

Parameter

mode

Description

Determines whether French-language web access is enabled in the client, and how it is configured. This functionality gives Windows and Macintosh clients the ability to access and display a web page.

If this parameter is set to `custom`, ALL of the other `[WEBLINK-FRENCH]` parameters must have values specified in the `unison.ini` file. If this is not the case, the value of this parameter reverts to the default (`cst`).

Accepted values

`off` or `cst` (disable web access)

`cst` (enable web access and take the values for all of the other `[WEBLINK-FRENCH]` parameters directly from the code; ignore any values set in the `unison.ini` file for these other parameters)

`custom` (enable web access and take values for all of the other `[WEBLINK-FRENCH]` parameters from `unison.ini`.)

Default value

`cst`

French-language browser to launch for Windows clients

Section

`[WEBLINK-FRENCH]`

Parameter

`browser-path-win`

Description

For Windows clients, this parameter determines which browser to launch for French-language web access. This parameter determines the location of the web browser on the local machine of each signed-on user.

Accepted values

A valid path and filename

Default value

None

French-language text to appear in Help menu and ToolTip

Section

[WEBLINK-FRENCH]

Parameter

command-name

Description

Determines the French-language text string that appears in the Help menu for the web access item, as well as in the pop-up ToolTip accompanying the Toolbar web access icon.

Accepted values

A string with a maximum of 150 characters in length

Default value

(when [WEBLINK-FRENCH] mode = cst) Information sur le produit

On-line French-language command description

Section

[WEBLINK-FRENCH]

Parameter

command-description-online

Description

For Windows clients, determines the description of the web access command that appears on the status bar when the client is on-line. For Macintosh clients 4.2 or earlier, determines the description of the web access command that appears in balloon help when the client is on-line.

Accepted values

A string with a maximum of 150 characters in length

Default value

(when [WEBLINK-FRENCH] mode = cst) Lancer le navigateur Web
pour visualiser l'information sur le produit

Off-line French-language command description

Section

[WEBLINK-FRENCH]

Parameter

command-description-offline

Description

For Windows clients, determines the description of the web access command that
appears on the status bar when the client is on-line. For Macintosh clients 4.2 or
earlier, determines the description of the web access command that appears in
balloon help when the client is on-line.

Accepted values

A string with a maximum of 150 characters in length

Default value

(when [WEBLINK-FRENCH] mode = cst) Lancer le navigateur Web
pour visualiser l'information sur le produit

French-language web page to load for clients working on-line

Section

[WEBLINK-FRENCH]

Parameter

online-url

Description
Determines the French-language web page to load for clients working on-line.

Accepted values
A valid URL with a maximum of 150 characters in length

Default value
None

File transfer protocol

Section
`[WEBLINK-FRENCH]`

Parameter
`download-mode`

Description
Determines the file transfer protocol to use when downloading web pages for viewing in off-line mode.

Accepted values
`file` (use the file sharing protocol)

Default value
(when `[WEBLINK-FRENCH]` mode = cst) `file`

Source of French-language web pages for Windows clients in off-line mode

Section
`[WEBLINK-FRENCH]`

Parameter
`offline-source-win`

Description

For Windows clients. Determines the source of the web page to display when working in off-line mode. This parameter is only checked when `[WEBLINK-FRENCH] mode = custom`.

Accepted values

A valid path and file name with a maximum of 150 characters in length

Default value

None

Source of French-language web pages for Mac clients in off-line mode

Section

`[WEBLINK-FRENCH]`

Parameter

`offline-source-mac`

Description

For Macintosh clients. Determines the source of the web page to display when working in off-line mode.

Accepted values

A string with a maximum of 150 characters in length which obeys the following format and restrictions:

<user>:<pw>@[<zone>]:<AFP Server Name>[,<ip>[,<port>]];<volume path>:

where:

- <user> is a maximum of 31 characters in length

- <pw> is a maximum of 8 characters in length

- <zone> is a maximum of 31 characters in length; default value is the asterisk symbol "*" (without the quotes)

- <AFP Server Name> is a maximum of 31 characters in length

- `<ip>` is a maximum of 31 characters in length

- `<port>` is a numerical value 0 to 32767 inclusive; default value is 548

- `<volume path>` is a maximum of 64 characters in length

This parameter is only checked when `[WEBLINK-FRENCH]` mode = custom.

Default value
None

Version of French-language off-line web pages

Section
`[WEBLINK-FRENCH]`

Parameter
`offline-source-version`

Description
Determines the version of the off-line web page. This provides a mechanism for the administrator to ensure that the most recent version of the off-line web page is the one being accessed. This parameter is only checked when `[WEBLINK-FRENCH]` mode = custom.

Accepted values
A character string with a maximum length of 150 characters.

Default value
None

Error message to display if download fails

Section
`[WEBLINK-FRENCH]`

Parameter
`download-fail`

Description

Determines the error message to display if a download fails.

Accepted values

A character string with a maximum length of 150 characters.

Default value

(when `[WEBLINK-FRENCH] mode = cst`) `Le telechargement s'est`
`acheve correctement. Les fichiers de mode déconnecte ont ete`
`restaures.`

Controlling client connections to server

Enable the ACE framework

Section

`[ACE]`

Parameter

`frameworkenable`

Description

Enables authentication, compression, and encryption. Note that if you disable the
ACE module, the calendar server uses the built-in `cs-basic` authentication
method. In other words, the calendar server always uses an authentication method.

Accepted values

`TRUE` (enable authentication, compression, encryption)

`FALSE` (disable authentication, compression, encryption)

Default value

`TRUE`

Maximum number of shared libraries per type

Section

[ACE]

Parameter

slibcachecount

Description

Determines the number of shared libraries that can be loaded at the same time for each type of method (authentication, compression, encryption).

Due to a limitation of IBM AIX in which shared libraries cannot be reloaded once removed from memory, the default value of this parameter is higher than for other platforms.

Accepted values

A positive integer up to a maximum value of 32768

Default value

20 (IBM AIX only)

3 (all other platforms)

Minimum buffer size for compression

Section

[ACE]

Parameter

minbufsizetocompress

Description

Specifies the minimum size in bytes required in order for a buffer to be compressed.

Accepted values

A positive integer up to a maximum value of 32768

Default value

700

Buffer size for compression and encryption

Section

[ACE]

Parameter

workbufsize

Description

Specifies the size, in bytes, of the buffer to allocate for compression and encryption.

Accepted values

A positive integer up to a maximum value of 32768

Default value

4096

Supported authentication methods for clients

Section

[AUTHENTICATION]

Parameter

supported

Description

Specifies a list of the authentication methods the calendar server supports for clients.

Both the cs-basic and the cs-standard methods use the calendar server name and password of a user to authenticate that user. Both encrypt the user password; cs-standard also encrypts the user name. This encryption is independent of the negotiated encryption method. The server applies the negotiated encryption on top of this encryption.

The cs-basic authentication method works with all calendar clients, regardless of client version. It pre-dates the calendar server ACE module.

cs-standard is the recommended authentication method to use where the client supports it. It offers a higher level of security (better authentication and encryption) than cs-basic.

Only the cs-basic and cs-standard methods currently work with LDAP directory servers.

Accepted values

A list of one or more of the following, separated by commasand enclosed in {}:

cs-basic

cs-standard

Default value

{cs-basic, cs-standard}

Default authentication method for clients

Section

[AUTHENTICATION]

Parameter

default

Description

Specifies the default authentication method the calendar server uses for clients. See the description of the [AUTHENTICATION] supported parameter for more information on supported methods.

Accepted values

Any method in the list specified by the [AUTHENTICATION] supported parameter.

Default value

cs-standard

Default authentication method for administrators

Section
[AUTHENTICATION]

Parameter
admindefault

Description
Specifies the default authentication method the calendar server uses for administrative sessions using the Calendar Administrator. See the description of the [AUTHENTICATION] supported parameter for more information on supported methods.

Accepted values
Any method in the list specified by the [AUTHENTICATION] supported parameter.

Default value
The value of the [AUTHENTICATION] default parameter

Default authentication method for other servers

Section
[AUTHENTICATION]

Parameter
servicedefault

Description
Specifies a default encryption method for the calendar server to use for communications with other calendar servers that request connections.

The server uses this default, along with the list of supported encryption methods, when it negotiates ACE methods with another calendar server initiating a request.

Accepted values

Any method in the list of supported encryption methods specified by the
`[AUTHENTICATION]` `supported` parameter.

Default value

The value of the `[AUTHENTICATION]` `default` parameter.

Location of resource passwords for authentication

Section

`[AUTHENTICATION]`

Parameter

`keepresourcepswincaldb`

Description

Determines whether resource passwords are stored in the calendar server's internal
database or in the database of the configured authentication mechanism.

Accepted values

`TRUE` (resource passwords stored in the calendar server database)

`FALSE` (resource passwords stored in the authentication mechanism database)

Default value

`TRUE`

Supported compression methods

Section

`[COMPRESSION]`

Parameter

`supported`

Description

Specifies a list of the compression methods the calendar server supports. Currently, only the Oracle `cs-simple` compression method is supported. This method uses simple run-length encoding compression, a very fast and efficient compression method for calendar data.

Accepted values

A list of one or more of the following, separated by commas and enclosed in { }:

`cs-simple`

`none`

Default value

`{cs-simple, none}`

Default compression method for clients

Section

`[COMPRESSION]`

Parameter

`default`

Description

Specifies the default compression method the calendar server uses for clients.

Accepted values

Any method in the list specified by the `[COMPRESSION] supported` parameter.

Default value

`cs-simple`

Default compression method for administrators

Section

`[COMPRESSION]`

Parameter

admindefault

Description

Specifies the default compression method the calendar server uses for administrative sessions using the Calendar Administrator. See the description of the [COMPRESSION] supported parameter for more information on supported methods.

Accepted values

Any method in the list specified by the [COMPRESSION] supported parameter.

Default value

The value of the [COMPRESSION] default parameter

Default compression method for other servers

Section

[COMPRESSION]

Parameter

servicedefault

Description

Specifies a default compression method for communications with other calendar servers that attempt to connect to this server.

The server uses this default, along with the list of supported compression methods, when it negotiates ACE methods with another calendar server initiating a request.

Accepted values

Any method in the list specified by the [COMPRESSION] supported parameter.

Default value

The value of the [COMPRESSION] default parameter.

Supported encryption methods

Section

[ENCRYPTION]

Parameter

supported

Description

Specifies a list of the encryption methods the calendar server supports.

The cs-light method scrambles data with a randomly generated key. It is very fast and offers minimal impact on performance, but is recommended for minimal-security installations.

The cs-acipher1 method is slower than the cs-light method, but offers much more secure encryption.

Accepted values

A list of one or more of the following, separated by commas and enclosed in {}:

cs-light

cs-acipher1

Default value

{cs-light, cs-acipher1, none}

Encryption methods requiring prior authentication

Section

[ENCRYPTION]

Parameter

needsauthenticate

Description

Specifies a list of encryption methods that require authentication prior to use. These methods are only available after the calendar client or another server authenticates itself to this calendar server. The initial ACE negotiation cannot include any of the methods listed by this parameter.

Accepted values

A list of any methods in the list specified by the `[ENCRYPTION] supported` parameter, separated by commas and enclosed in {}.

Default value

`{}`

Default encryption method for clients

Section

`[ENCRYPTION]`

Parameter

`default`

Description

Specifies the default encryption method the calendar server uses for clients.

Accepted values

Any method in the list specified by the `[ENCRYPTION] supported` parameter.

Default value

`none`

Default encryption method for administrators

Section

`[ENCRYPTION]`

Parameter

admindefault

Description

Specifies the default encryption method the calendar server uses for administrative sessions using the Calendar Administrator. See the description of the [ENCRYPTION] supported parameter for more information on supported methods.

Accepted values

Any method in the list specified by the [ENCRYPTION] supported parameter.

Default value

The value of the [ENCRYPTION] default parameter

Default encryption method for other servers

Section

[ENCRYPTION]

Parameter

servicedefault

Description

Specifies a default encryption method for the calendar server to use for communications with other calendar servers that request connections.

The server uses this default, along with the list of supported encryption methods, when it negotiates ACE methods with another calendar server initiating a request.

Accepted values

Any method in the list of supported encryption methods specified by the [ENCRYPTION] supported parameter.

Default value

The value of the [ENCRYPTION] default parameter.

Web authentication — shared key

Section
[ACE_PLUGINS_SERVER]

Parameter
web_CAL_sharedkey

Description
Specifies the shared key to compare with the value of the client `webcal.ini`
[ACE_PLUGINS_CLIENT] web_CAL_sharedkey parameter.

Accepted values
A valid path and file name

Default value
None

Maximum number of Engines

Section
[LCK]

Parameter
lck_users

Description
Specifies the maximum number of Engine processes permitted. The value of this
parameter should be carefully considered. It must allow for enough Engines to
service both client access and SNC connections; however setting the value higher
than required wastes system resources.

Accepted values
A positive integer up to 2000 (NT) or 5000 (UNIX)

Default value
100

Maximum number of concurrent sessions by a given user

Section

[ENG]

Parameter

max_userlogons

Description

Specifies the maximum number of concurrent "named" sessions that each user may invoke. A session is "named" if it is associated with a specific user and "unnamed" if it is not associated with a specific user.

Accepted values

0 (no limit)

A positive integer

Default value

0

Number of concurrent sessions from a specific Internet address

Section

[ENG]

Parameter

max_addrlogons

Description

Specifies the maximum number of concurrent "unnamed" sessions that can be invoked by a single client, that is, from a single Internet address. A session is "named" if it is associated with a specific user and "unnamed" if it is not associated with a specific user.

Accepted values

0 (no limit)

A positive integer

Default value

0

Maximum read lock time before termination

Section

[ENG]

Parameter

readlocktimeout

Description

Determines the number of consecutive seconds that the server can lock the database for a client read request. If this maximum is exceeded, the uniengd server and the associated user session terminate, and the timeout is logged to eng.log.

Accepted values

0 (no limit)

A positive integer up to the value of $(2^{32}-1)$

Default value

60

Maximum write lock time before termination

Section

[ENG]

Parameter

writelocktimeout

Description

Determines the number of consecutive seconds that the server can lock the database for a client write request. If this maximum is exceeded, the uniengd server and the associated user session terminate, and the timeout is logged to eng.log.

Accepted values

0 (no limit)

A positive integer up to the value of $(2^{32}\text{-}1)$

Default value

60

Maximum read lock time before release

Section

[ENG]

Parameter

readmaxlocktime

Description

For newer operations, determines the number of consecutive milliseconds that an operation can hold a read lock on the calendar database. If this maximum is exceeded, the lock will be released. If the process has not been completed, it will then re-lock the calendar database.

Accepted values

0 (no limit)

A positive integer up to the value of $(2^{32}\text{-}1)$

Default value

250

Maximum write lock time before release

Section
[ENG]

Parameter
writemaxlocktime

Description
For newer operations, determines the number of consecutive milliseconds that an operation can hold a write lock on the calendar database. If this maximum is exceeded, the lock will be released. If the process has not been completed, it will then re-lock the calendar database.

Accepted values
0 (no limit)

A positive integer up to the value of $(2^{32}-1)$

Default value
150

Retry interval for remote data requests to server

Section
[LIMITS]

Parameter
remotewait

Description
Specifies the number of seconds the client waits before retrying a call to the server for data from a remote server.

Accepted values
A positive integer up to the value of $(2^{32}-1)$

Default value

2

Retry limit for remote data requests to server

Section

[LIMITS]

Parameter

remotemaxretry

Description

Specifies the number of retries of a request for remote node data that the client makes before returning an error.

Accepted values

A positive integer up to the value of $(2^{32}-1)$

Default value

5

C

Utilities

This appendix contains full instructions on the usage and syntax of all utilities shipped with your calendar server. Note that the installation script does not install UNIX-only utilities on Windows NT platforms. All utilities are installed in the /users/unison/bin directory.

The following table lists all utilities in alphabetical order.

Table C–1   Calendar server utilities

Script	Function
UNIADDNODE	Create a new calendar server node or re-initialize an existing one
UNIADMRIGHTS	Manage the administration rights of users.
UNIARCH (UNIX ONLY)	Create a tar archive of the calendar server.
UNICHECK (UNIX ONLY)	Verify the calendar server file system.
UNICKSUM	Generate a checksum for a file.
UNICLEAN (UNIX ONLY)	Clean up the calendar server file system (remove transient files and set permissions).
UNICLR_IPC (UNIX ONLY)	Clear IPC resources consumed by the calendar server.
UNICPINR	Copy resource data from a file created by unicpoutr to a calendar server node.
UNICPINU	Copy the contents of a file of user data created by unicpoutu to a calendar server node.
UNICPOUTR	Copy resource data from a calendar server node into a file.

Table C–1   Calendar server utilities

Script	Function
UNICPOUTU	Copy user data from a calendar server node to a file.
UNICPR	Format of the file created by `unicpoutr` and read by `unicpinr`.
UNICPU	Format of the file created by `unicpoutu` and read by `unicpinu`.
UNIDBBACKUP	Create an archive of the calendar server.
UNIDBCONV	Convert a version 2.50 or 2.60 node database to a 2.61 node database.
UNIDBFIX	Check, repair, defragment and maintain a calendar server node database.
UNIDBRESTORE	Restore the contents of a calendar server from a backup created by `unidbbackup`.
UNIDOMAIN	Create a calendar domain.
UNIDSACISETUP	Set the access control information in the directory server for the calendar server ADMIN group. (external directory only)
UNIDSDIFF	Find and delete differences between a calendar server node and a directory server. (external directory only)
UNIDSSEARCH	List all users in a directory server who are not calendar server users. (external directory only)
UNIDSSYNC	Synchronize the information in a calendar server node with that in a directory server. (external directory only)
UNIDSUP	Report the status of the directory server. (external directory only)
UNIENCRYPT	Encrypt a password for inclusion in a calendar server configuration file
UNIGRPLS	Display both the public and administrative groups in a calendar server database.
UNILOGONS	Display calendar server SIGNON/SIGNOFF statistics.
UNIMVUSER	Move a user from one calendar server node to another.

Table C–1   Calendar server utilities

Script	Function
UNINODE	Administer a calendar server node network.
UNIPASSWD	Change a user password on a calendar server database.
UNIPING	Ping a calendar server node or nodes.
UNIREQDUMP	View, and optionally delete, requests in the queue of the Corporate-Wide Services (CWS) daemon.
UNIRES	List, add, or delete calendar server resources, or modify the information associated with them.
UNIRMOLD	Remove old events and tasks from agendas in a calendar server database.
UNIRNDEL	Delete a remote node from a local calendar server node database.
UNISTATS	Propagate deletions in the local information of one node to another node in the network.
UNISIZEOF	Compute the size of the calendar server installation.
UNISLICE (UNIX ONLY)	Extract information from calendar server log files.
UNISNAPSHOT	Compile calendar server information for diagnostic purposes.
UNISNCDUMP	Retrieve statistics from the calendar server's Synchronous Network Connection daemon/service.
UNISTART	Start up the calendar server.
UNISTAT	Produce a report on a calendar server node.
UNISTATS	Display summary statistics of the data in a calendar server stats file.
UNISTATUS	Determine the status of the calendar server.
UNISTOP	Shut down the calendar server.
UNISYNCREFRESH	Refresh synchronization records.
UNITZINFO	Print information about a calendar server time zone.
UNIUSER	List, add, or delete calendar users; modify the information associated with them.
UNIVERSION	Verify the version of the calendar server.

Table C–1   Calendar server utilities

Script	Function
UNIWHATOS (UNIX ONLY)	Determine whether the the calendar server package runs under the current operating system.
UNIWHO	Display information on signed-on calendar users.

UNIADDNODE

uniaddnode - Create a new calendar server node or re-initialize an existing one.

SYNTAX

Internal Directory

uniaddnode -n <node-ID> [-t <timezone>][-a <nodealias>] [-r] [-y]

Directory Server

uniaddnode -n <node-ID> -w <DmPsw> [-p <SysOpPsw>] [-t <timezone>] [-a <nodealias>] [-r] [-y]

uniaddnode -v
uniaddnode -h

DESCRIPTION

This utility creates and initializes a new calendar server node for use with either an internal directory or a directory server. It can also re-initialize an existing node.

uniaddnode runs only if the calendar server is down.

OPTIONS

-a

<nodealias>

Specify an alias for the node. <nodealias> is a descriptive word (it cannot contain spaces).

-n

<node-ID>

Specify the node-ID. The node-ID must be unique across all nodes in the network.

-p

<SysOpPsw>

Provide the SYSOP password for the node. This option is only required for directory servers. If the password is not provided on the command line, prompting for it occurs. For internal directories, the SYSOP password can be set after creation of the node using the `unipasswd` utility.

-r

Re-initialize the node.

> **Warning:** **All existing node data is lost.**

Note that in the case of a directory server, all users and resources must first be removed from the node before it can be re-initialized.

-t

<timezone>

Specify a time zone for the node. The default is the time zone set during installation of the calendar server. Time zones can be obtained from the `unitzinfo` utility, the `/users/unison/misc/timezone.ini` file, or the calendar server Reference Manual, Appendix D, "Time Zone Table".

-w

<DmPsw>

Provide the directory server password for unrestricted access (i.e. the password associated with the value of the `[LDAP]` mgrdn parameter in the `unison.ini` file). This option is only required for installations using a directory server. If the password is not specified on the command line, prompting for it occurs.

-y

Used with the **-r** option to auto-confirm the re-initialization.

-v

Print the current version number of `uniaddnode`.

-h

Print a usage message explaining how to run `uniaddnode`.

EXAMPLES

■ Create a node with node-ID "44", an alias of "admin", and the time zone of New York City for a calendar server using a directory server:

```
% uniaddnode –n 44 –a admin –t EST5EDT –w DmPsw –p sysOpPsw
unidsndini: working, please wait ...
Creation of reserved users successful.
Creation of Administrators group successful.
uniaddnode: unidsndini done
uniaddnode: unidbi done
```

The following entry now appears in the [<YOURNODEID>] section of the `/users/unison/misc/unison.ini` file.

```
[44]
name = <internally-assigned value>
version = A.02.62
aliases = admin
timezone = EST5EDT
```

FILES

`/users/unison/misc/unison.ini`
This is the calendar server configuration file. For each new node, a node entry is created in this file by the `uniaddnode` utility.

EXIT STATUS

Exit values are:

0 Success

Any non-zero value signals an error.

UNIADMRIGHTS

`uniadmrights` - Manage the administration rights of users.

SYNTAX

```
uniadmrights [–ls] [[–hday] [–pgrp] [–opgrp] | –all] [–n <node-ID>] [–host
<hostname>] [–p <SysOpPsw>]
```

```
uniadmrights -e <user> [-add | -del] [[-hday] [-pgrp] [-opgrp] | -all] [-n
<node-ID>] [-host <hostname>] [-p <SysOpPsw>]

uniadmrights -default [-add | -del] [[-hday] [-pgrp] [-opgrp] | -all] [-n
<node-ID>] [-host <hostname>] [-p <SysOpPsw>]

uniadmrights -v
uniadmrights -h
```

DESCRIPTION

This utility allows the SYSOP to grant certain administration rights to users as well as to revoke these rights. It can also be used to determine the rights held by each user.

The existing rights are granted on a per-node basis and apply to:

- holiday administration

- administrative groups

- public groups

By default, uniadmrights lists ALL rights that have been granted by the SYSOP. Note that the -**ls** option is mutually exclusive with the -**add** option, and with the -**del** option.

The calendar server must be up to run uniadmrights.

Note: Use the ManageHolidays, ManageAdmGroups, and CreatePublicGroups keywords in the user.ini file to automatically grant one or more of these administration rights on user creation.

OPTIONS

-add

Grant a right. Used with the -**e** option.

-all

Add or delete ALL rights held by the user when used with the -**e** option (and either the -**add** or -**del** option). List all users holding rights when used with the -**ls** option.

-default

Set rights for all users with the default administrative rights profile (applied when the `ManageAdmGroups`, `ManageResources` and `ManageHolidays` parameters are not present in or are commented out of `user.ini`).

-del

Remove a right. Used with the **-e** option.

-e

<user>

Specify the user. If more than one match for the user is found in the database, `uniadmrights` fails. If no action (-**add**/-**del**/-**all)** is specified along with this option, the default behaviour is to grant the specified right(s) to the user; if no rights are specified, ALL rights are granted to the user. See FORMAT OF THE <user> ARGUMENT for details on the <user> argument.

-hday

The holiday administration right. This right allows the user to set which holidays appear in the agendas of all users in the node. Note that no designates are associated with holiday administration; only those users granted the holiday right by the SYSOP may administer holidays.

-host

<hostname>

Specify the host. Required if the host is remote.

-ls

List all granted rights. This is the default behaviour when no option has been specified.

-n

<node-ID>

Specify the node. Required if more than one node exists on the host.

-opgrp

The public groups right. Allows the user to create public groups. The user, as owner of the public group, can make modifications to the group as well as delete the group

itself. Since there are no designates associated with a public group, only its creator
(owner) will be able to make modifications to it, or delete it.

-p
<SysOpPsw>

Provide the SYSOP password; required if one is set. If this option is not used and a
password is required, uniadmrights prompts the user for it.

-pgrp
The administrative groups right. Allows the user to create, delete, and/or modify
administrative groups. Any user holding this right can delete and/or modify an
existing administrative group, regardless of whether or not they are its creator.
Since there are no designates associated with an administrative group, only those
users holding this right will be able to modify or delete an administrative group.

-v
Print the current version number of uniadmrights.

-h
Print a usage message explaining how to run uniadmrights.

FORMATS

FORMAT OF THE <user> ARGUMENT
The <user> argument is a string of the form "key=value/key=value/...", where
"key" is one of those listed below, and "value" is any string. Both "key" and
"value" are case insensitive. The "value" string may be terminated by a wild card
symbol (*). If a forward slash "/" is to be included in a string, it should be escaped
with the character "\" to prevent it from being interpreted as a key-value pair
delimiter - i.e. "S=Hoopla/OU1=R\/D".

If, in a UNIX environment, a shell will be processing the string (e.g. the string is
provided on the command line or is passed as an argument to the utility), the string
should be enclosed in quotation marks. Furthermore, if characters meaningful to the
shell are included in the string, they may need to be escaped (i.e. preceded by the
escape character "\") to prevent the shell from interpreting them.

> **Note:** If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored.

Some example specifications are: `"S=Kilpi/G=Eeva"`, `"S=B*/G=Nicole/O=Acme"`, `"O=Acme/ID=1111/OU1=authors"`

Table C–2   Accepted keys

Key	X.400 Field
S	Surname
G	Given name
I	Initials
ID	Identifier
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain
P	Private domain

EXAMPLES

- List all users with administration rights (where only one node exists so the node-ID need not be specified):

  ```
  % uniadmrights
  ```

- List all users with holiday administration rights on node 80:

  ```
  % uniadmrights –ls –hday –n 80
  ```

- List all users with public group administration rights on remote host gravel (only one node exists on gravel):

  ```
  % uniadmrights -pgrp -host gravel
  ```

- Grant holiday administration rights to Don Martin in R&D, at node 80:

  ```
  % uniadmrights -e "S=Martin/G=Don/OU1=r&d" -add -hday -n 80
  ```

- Grant public group administration rights to all users in node 80 who currently hold no rights:

  ```
  % uniadmrights -default -add -pgrp -n 80
  ```

- Remove all rights from Joan Bean on remote host montreal (only one node exists on montreal, so node-ID is not specified):

  ```
  % uniadmrights -e "S=Bean/G=Joan" -del -all -host montreal
  ```

WARNINGS

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities.

In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

UNIARCH (UNIX ONLY)

uniarch - Create a tar archive of the calendar server.

SYNTAX

```
uniarch [-d] [-y] [-t | -f <filename>]

uniarch -v
uniarch -h
```

DESCRIPTION

`uniarch` creates a backup of the calendar server. By default, the entire `/users/unison` directory is archived.

You must invoke `uniarch` from outside of the directory or directories it is backing up. For example, to back up the entire calendar server, you invoke `uniarch` from outside of the `/users/unison` directory.

`uniarch` can only be run if the calendar server is down.

> **Warning: uniarch backs up the calendar server internal database. If a directory server is being used, its database should also be backed up.**

OPTIONS

-d

Back up only the contents of `/users/unison/db/nodes`, the calendar server database.

-f

<filename>

Specify the name of the archive file. If this option is not used, prompting for the filename occurs.

-t

Force the tar default device to be used for the archive destination file.

-y

By default, `uniarch` asks for confirmation before proceeding with the creation of the archive. This option tells `uniarch` to automatically proceed, without prompting for confirmation. Default if there is no tty associated with the calling process.

-v

Print the current version number of uniarch.

-h

Print a usage message explaining how to run uniarch.

EXAMPLES

- Archive the entire /users/unison directory:

```
% uniarch
uniarch: working, please wait ...
uniarch: input tar archive destination file name: jan07-99.bkup
uniarch: archive "/users/unison" and redirect to "jan07-99.bkup"? (y/n)
uniarch: archive completed
```

- Archive only the calendar server database, supplying the name of the destination archive file on the command line:

```
% uniarch -d -f jan07-99-db.bkup
uniarch: working, please wait ...
uniarch: archive "/users/unison/db/nodes" and redirect to
"jan07-99-db.bkup"? (y/n)

uniarch: archive completed
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIB2LENDIAN

unib2lendian - Convert a calendar server node database from a format for big-endian UNIX processors to a format for little-endian Windows NT processors. For more details on this utility, contact Oracle support.

UNICHECK (UNIX ONLY)

unicheck - Verify the calendar server file system.

SYNTAX

unicheck [-nowarn] [-nodb | -maxdb <n>] [-c]

unicheck -v
unicheck -h

DESCRIPTION

unicheck verifies the calendar server file system. The utility first checks that the version of the calendar server is intended to run on the local operating system. If this is not the case, unicheck prompts the user to determine whether or not they wish to continue. If the version runs on the local operating system, unicheck then verifies:

1. that all necessary files and directories are present

2. that the permissions, and owner and group information are correctly set on the files and directories.

Any discrepancies are reported. Unless an entire file or directory is missing, any problems found are fixed running uniclean.

unicheck should be run periodically to ensure that the file system is in good order.

unicheck can be run whether the calendar server is up or down.

OPTIONS

-maxdb

<n>

Specifies the maximum number of node databases unicheck should consider. For example, if <n>=30, unicheck checks the files of only the first 30 databases.

-nowarn

Do not print warning messages (error messages are still printed).

-nodb

Do not check database files.

-c

Computes a system-independent checksum for each static file. If this option is used, output should be redirected to a file for future use.

-v

Print the current version number of `unicheck`.

-h

Print a usage message explaining how to run `unicheck`.

EXAMPLES

■ Run `unicheck` (for brevity, sections of the output have been replaced by [...]):

```
% unicheck
unicheck: checking all directories
unicheck: checking directory "/users/unison"
unicheck: checking directory "/users/unison/tmp"
[...]
unicheck: checking files in directory "/users/unison/bin"
unicheck: checking files in directory "/users/unison/misc"
[...]
unicheck: checking versions of files in directory "/users/unison/bin"
unicheck: check completed
```

■ Run `unicheck`, suppressing any warning messages and computing a checksum for each file (for brevity, sections of the output have been replaced by [...]):

```
% unicheck -nowarn -c
unicheck: checking all directories
unicheck: checking directory "/users/unison"
unicheck: checking directory "/users/unison/tmp"
[...]
unicheck: checking files in directory "/users/unison/bin"
unicheck: checking files in directory "/users/unison/misc"
unicheck: checking files in directory "/users/unison/man"
[...]
unicheck: checking versions of files in directory "/users/unison/bin"
unicheck: computing checksums
unicksum: checksum of the file "/users/unison/misc/timezone.ini" is 17289
unicksum: checksum of the file "/users/unison/bin/addme" is 33775
[...]
unicheck: check completed
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNICKSUM

unicksum - Generate a checksum for a file.

SYNTAX

```
unicksum <filename>

unicksum -v
unicksum -h
```

DESCRIPTION

unicksum generates a checksum for a file that is used to determine whether or not differences exist between two instances of the same file.

unicksum runs whether the calendar server is up or down.

OPTIONS

-v

Print the version number of unicksum.

-h

Print a usage message explaining how to run unicksum.

EXAMPLES

■   Generate a checksum for the unitzinfo executable:

```
% unicksum unitzinfo
unicksum: checksum of the file "unitzinfo" is 18187
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNICLEAN (UNIX ONLY)

`uniclean` - Clean up the calendar server file system.

SYNTAX

```
uniclean

uniclean -v
uniclean -h
```

DESCRIPTION

`uniclean` cleans up the calendar server file system by removing some transient files and ensuring file/directory and owner/group permissions are properly set.

`uniclean` can be run when the calendar server is up or down.

OPTIONS

-v

Print the current version number of `uniclean`.

-h

Print a usage message explaining how to run `uniclean`.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

SEE ALSO
unicheck

UNICLR_IPC (UNIX ONLY)

uniclr_ipc - Clear IPC resources consumed by the calendar server.

SYNTAX
uniclr_ipc [-s] [-q]

uniclr_ipc -v
uniclr_ipc -h

DESCRIPTION

uniclr_ipc clears IPC (Inter-Process Communication) resources consumed by the calendar server. By default, all IPC resources are freed. The -**s** and -**q** options are available to selectively clear only semaphore or message-queue resources respectively.

uniclr_ipc can only be run if the calendar server is down.

OPTIONS

-s

Clear semaphore-related resources only.

-q

Clear message-queue related resources only.

-v

Print the version number of uniclr_ipc.

-h

Print a usage message explaining how to run uniclr_ipc.

EXAMPLES

In all of the following examples, the calendar server is down.

- Clear all IPC resources:

```
% uniclr_ipc
uniclr_ipc: working, please wait...
uniclr_ipc: ipc resources cleared
```

- Clear only the semaphore resources:

```
% uniclr_ipc -s
uniclr_ipc: working, please wait...
uniclr_ipc: ipc semaphore-related resources cleared
```

- Clear only the message-queue resources:

```
% uniclr_ipc -q
uniclr_ipc: working, please wait...
uniclr_ipc: ipc message-queue related resources cleared
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNICPINR

unicpinr - Copy resource data from a file created by unicpoutr to a calendar server node.

SYNTAX

```
unicpinr [-add] [-f <filename>] [-start <day> <month> <year>] [-end <day>
<month> <year>] [-host <hostname>] <Node-ID> [-p <SysOpPsw>]

unicpinr -ls [<filename(s)>]

unicpinr -v
unicpinr -h
```

DESCRIPTION

Copies a file containing resource data (created with the `unicpoutr` utility) into a calendar server node. The utility can be used in conjunction with `unicpoutr` to move a resource from one node to another, or to add the agenda of one resource to that of another (see EXAMPLES).

By default, the resource specified in the file must already exist in the destination calendar server node. If this is not the case, the -**add** option is used to add it.

`unicpinr` can only be run if the calendar server is up.

It is important to understand how `unicpinr` handles the information in the file during the copy into the destination node.

■   Resource identifier

These are the values for the keys R, N, CA, S, G, ID, LOC, PHONE, EXT, FAX (see RESOURCE IDENTIFIER KEYS below for details on these keys). Only non-null values are output to the file by `unicpoutr` so not all keys may have a value in the file.

`unicpinr` uses these values to uniquely identify an existing resource in the destination node.

■   Password and agenda-specific preferences

Where the resource already exists in the destination node, these values are already set and `unicpinr` does NOT overwrite them with those in the input file.

■   Agenda information

Where a resource already exists in the destination node, `unicpinr` simply adds the agenda information in the input file to the existing agenda.

All events listed in the file are copied into the destination node with the resource as the owner. Where appropriate, the description of each event contains extra data indicating the invitees to the event, their status, and the original creator and owner. Recurring or repeating instances of an event are disconnected from each other and copied in as individual events.

The -**start** and -**end** options can be used to import only those events that fall within the specified time.

> **Note:** The `unicpinr` utility does not consult the `resource.ini` files when importing resources.

OPTIONS

-add

Add the resource to the database before copying in the file. It is an error to specify this option if the resource already exists in the node. In the case of a directory server, the resource is created under the baseDN.

-end

<day> <month> <year>

Set the end dates of the events to be processed. By default, all events in the file are created; this option and the -start option allow you to exclude certain events. Dates must be expressed in the form "day month year". Years must be specified using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-f

<filename>

Specify the input file name. The file must have been created with the `unicpoutr` utility. By default, standard input is used.

-host

<hostname>

Specify the host on which the specified node can be found. The default is the local host.

-ls

List the file name followed by the name of the resource it contains for each specified file name. Files not created with the `unicpoutr` command are not listed. If no file names are specified, the files of the current directory (.) are examined.

-p

<SysOpPsw>

Provide the SYSOP password. If this option is not used, prompting for the password occurs.

-start

<day> <month> <year>

Set the start date of the events to be processed. By default, all events in the file are created; this option and the -**end** option allow you to exclude certain events. Dates must be expressed in the form "day month year". Years must be specified using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-v

Print the current version number of `unicpinr`.

-h

Print a usage message explaining how to run `unicpinr`.

RESOURCE IDENTIFIER KEYS

Table C–3   Accepted keys

Key	Description
R	Resource name
N	Resource number
CA	Capacity
S	Contact's surname
G	Contact's given name
LOC	Location
PHONE	Phone number
EXT	Phone extension
FAX	Fax phone number

EXAMPLES

MOVE A RESOURCE FROM ONE NODE TO ANOTHER

`unicpinr` is used in conjunction with `unicpoutr` and `unires` to move a resource from one node to another. In the following example, the resource "betacam" will be moved from node 30 to 35.

1. Verify that the resource to be moved exists in node 30:

```
% unires -ls "R=Betacam" -n 30
R=Betacam/CA=1/ID=1234
```

2. Copy out the resource data to a file:

```
% unicpoutr "R=Betacam" -f betacam.dat 30
```

3. Delete the resource from the node. This is normal practice as you do not usually want the same resource to exist in two different nodes.

```
% unires -del "R=Betacam" -n 30
```

4. Add the resource to the destination node:

```
% unicpinr -add -f betacam.dat 35
```

ADD THE AGENDA OF ONE RESOURCE TO THAT OF ANOTHER RESOURCE

`unicpinr` can be used in conjunction with `unicpoutr` to add the agenda of one resource to that of another resource. This example adds the agenda for "PineNook" to the agenda for "OakCranny" and at the same time changes the capacity of "OakCranny" to 5.

1. Copy out the resource data for PineNook (from node 30) to a file:

```
% unicpoutr "R=PineNook" -f pinenook.dat 30
```

2. Edit the file and modify the resource identifier to match that for OakCranny

```
% vi pinenook.dat
```

3. Copy in the file to OakCranny in node 30. Since this resource exists, the password, and agenda-specific preferences are not overwritten.

```
% unicpinr -f pinenook.dat 30
```

The agenda information for PineNook has been added to the existing agenda information for OakCranny.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNINGS

Agenda size and processing time

Depending on the size of the agenda in the file, `unicpinr` may take some time to complete.

Limitations of this utility

The `unicp` family of utilities have the following limitations that must be considered.

■ *Events*

From the perspective of a moved user (or resource), each of the moved events in the new agenda is a personal event with enough data in the description to determine who created the event and who the attendees are. All links are broken but there is sufficient information in the description to allow the links to be rebuilt.

Note also that where the agenda of one user (or resource) is being added to that of another, double-booking may occur.

■ *Deleting a user (or resource)*

When a user (or resource) is moved to a new node, that user (or resource) should be deleted from the old node (using `uniuser -del` (or `unires -del`)).

When a resource is deleted, all traces of that resource are removed. Thus, that resource is no longer an invitee to events.

When a user is deleted, all traces of that user are removed. Thus, that user is no longer an invitee to events created by other users. Furthermore, and most importantly, all events created by the user are deleted. As a consequence, any user in the old node who was invited to an event by the moved user, will no longer be able to view the event.

- *Moving several users (and/or resources) at a time*

  If several users (and/or resources) are to be moved, it is best to perform the move in three phases:

  1. Copy the information on each user (and/or resource) from the source node to a file (using `unicpoutu` and/or `unicpoutr`).

  2. Delete each user (and/or resource) from the source node.

  3. Copy the information on each user (and/or resource) into the destination node using (`unicpinu` and/or `unicpinr`).

  This ensures that information on any links among the users (and/or resources) being moved is not lost (see "Deleting a resource" above).

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

SEE ALSO

`unicpoutr, unicpr`

UNICPINU

`unicpinu` - Copy the contents of a file of user data created by `unicpoutu` to a calendar server node.

SYNTAX

`unicpinu [-add ] [-f <filename>] [-start <day> <month> <year>] [-end <day> <month> <year>] [-host <hostname>] <node-ID> [-p <SysOpPsw>]`

`unicpinu -ls [<filename(s)>]`

```
unicpinu -v
unicpinu -h
```

DESCRIPTION

unicpinu copies a file containing user data (created by unicpoutu) into a calendar server node. The utility can be used in conjunction with unicpoutu to add the agenda of one user to that of another user (see EXAMPLES). Although it can also be used, in conjunction with unicpoutu, to move a user from one node to another, unimvuser is the proper utility for moving users. unimvuser prevents the data loss that occurs when moving users with unicpoutu and unicpinu.

By default, the user specified in the file must already exist in the destination calendar server node. If this is not the case, they can be added using the -**add** option.

unicpinu can only be run if the calendar server is up.

It is important to understand how unicpinu handles the information in the input file during the copy into the destination node:

- X.400 name and address

  These are the values for the keys S, G, I, and X, and the keys OU1, OU2, OU3, OU4, O, C, A and P respectively (see NAME AND ADDRESS KEYS below for details on these keys). Only non-null values are output to the file by unicpoutu so not all keys may have a value in the file.

  unicpinu uses these values to uniquely identify an existing user in the destination node.

- Personal information, password, and agenda-specific preferences

  Personal information includes employee number, phone number, extension, fax number, job title and office mailing address.

  Where the user already exists in the destination node, these values are already set and unicpinu does NOT overwrite them with those in the input file.

- Agenda information

  Where a user already exists in the destination node, unicpinu simply adds the agenda information in the input file to the existing agenda.

  All events listed in the file are copied into the destination node with the user as the owner. Where appropriate, the description of each event contains extra data indicating the invitees to the event, their status, and the original creator and

owner. Recurring or repeating instances of an event are disconnected from each other and copied in as individual events.

The -**start** and -**end** options can be used to import events and completed tasks that fall within a specified range. Incomplete tasks are always imported.

Warning: Holidays are output by unicpoutu as meetings, and therefore input by unicpinu as meetings. Only the existing holidays in the destination node appear as holidays in the user's agenda.

Note: The unicpinu utility does not consult the user.ini files when importing users.

OPTIONS

-add

Add the user to the database and then copy in the user's agenda. It is an error to specify this option if the user already exists. Note that for directory servers, the user must already exist in the directory server (all of the X.400 key-value pairs specified in the input file must match), and must not already be a calendar user.

-end

<day> <month> <year>

Set the end date for the events and tasks to be processed. By default, all events and tasks in the file are created; this option and the -**start** option allow you to exclude certain events and tasks. Dates must be expressed in the form "day month year". Years must be expressed using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-f

<filename>

Specify the input file name. The file must be created with the unicpoutu utility. If this option is not specified, standard input is used.

-host

<hostname>

Specify the host on which the specified node is found. The default is the local host.

-ls

<filename(s)>

Print the filename followed by the X.400 name and address of the user contained in the file, for each specified file name. Files not created by the `unicpoutu` command are not listed. If no file names are specified, the files in the current directory (.) are examined.

-p

<SysOpPsw>

Provide the SYSOP password. If this option is not used, prompting for the password occurs.

-start

<day> <month> <year>

Set the start date for the events and tasks to be processed. By default, all events and tasks in the file are created; this option and the -end option allow you to exclude certain events and tasks. Dates must be expressed in the form "day month year". Years must be expressed using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-v

Print the current version number of `unicpinu`.

-h

Print a usage message explaining how to run `unicpinu`.

X.400 NAME, AND ADDRESS KEYS

Table C–4   Accepted keys

Key	Description
S	Surname
G	Given name
I	Initials
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain
P	Private domain

EXAMPLES

MOVE A USER FROM ONE NODE TO ANOTHER

`unicpinu` is used in conjunction with `unicpoutu` and `uniuser` to move a user from one node to another. In this example the user "Sarah Herman" will be moved from node 20 to 44, and one of her organizational units changed from "Sales" to "R&D".

> **Warning:**   Use this procedure ONLY if unimvuser cannot handle the move you need to make. See WARNINGS below for information on the data that is lost during this procedure.

1.   Verify that the user to be moved exists in node 20:

```
% uniuser -ls "S=Herman/G=S*" -n 20
S=Herman/G=Sarah/OU1=Dallas/OU2=Sales/ID=1234
```

2. Copy the user's agenda and user information to a file:

```
% unicpoutu "G=Sara*/S=Herman -f sherman.dat 20
```

3. Delete the user from node 20. This is normal practice as the same user should not exist in two different nodes. In the case of a directory server, this step is required if the subsequent `unicpinu -add` command is to succeed.

```
% uniuser -del "G=Sara*/S=Herman" -n 20
```

4. Add the user to the destination node:

```
% unicpinu -add -f sherman.dat 44
S=Herman/G=Sarah/OU1=Dallas/OU2=Sales/ID=1234
```

ADD THE AGENDA OF ONE USER TO THAT OF ANOTHER USER

`unicpinu` can be used in conjunction with `unicpoutu` to add one user's agenda to that of another user. This example adds Sarah Herman's agenda to Yannick Olafsen's agenda.

1. Copy Sarah Herman's user data (from node 20) to a file:

```
% unicpoutu "G=Sara*/S=Herman" -f sherman.dat 20
```

2. Edit the `sherman.dat` file to modify the X.400 name and address to match that contained in the database for Yannick Olafsen.

```
% vi sherman.dat
```

3. Copy the file to node 24. Since Yannick Olafsen already exists as a user in node 24, his personal information, password, and agenda preferences are not overwritten.

```
% unicpinu -f sherman.dat 24
```

The agenda information for Sarah Herman is added to the existing agenda information for Yannick Olafsen.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNINGS

Agenda size and processing time

Depending on the size of the agenda in the file, `unicpinu` may take some time to complete.

Limitations of this utility

The `unicp` family of utilities have the following limitations that must be considered.

- *Events*

  From the perspective of a moved user (or resource), each of the moved events in the new agenda is a personal event with enough data in the description to determine who created the event and who the attendees are. All links are broken but there is sufficient information in the description to allow the links to be rebuilt.

  Note also that where the agenda of one user (or resource) is being added to that of another, double-booking may occur.

- *Deleting a user (or resource)*

  When a user (or resource) is moved to a new node, that user (or resource) should be deleted from the old node (using `uniuser -del` (or `unires -del`)).

  When a resource is deleted, all traces of that resource are removed. Thus, that resource is no longer an invitee to events.

  When a user is deleted, all traces of that user are removed. Thus, that user is no longer an invitee to events created by other users. Furthermore, and most importantly, all events created by the user are deleted. As a consequence, any user in the old node who was invited to an event by the moved user, will no longer be able to view the event.

- *Moving several users (and/or resources) at a time*

  If several users (and/or resources) are to be moved, it is best to perform the move in three phases:

1. Copy the information on each user (and/or resource) from the source node to a file (using `unicpoutu` and/or `unicpoutr`).

2. Delete each user (and/or resource) from the source node.

3. Copy the information on each user (and/or resource) into the destination node using (`unicpinu` and/or `unicpinr`).

This ensures that information on any links among the users (and/or resources) being moved is not lost (see "Deleting a user (or resource)" above).

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

SEE ALSO

`unicpoutu, unicpu`

UNICPOUTR

`unicpoutr` - Copy resource data from a calendar server node into a file.

SYNTAX

```
unicpoutr res [-f <filename>] [-host <hostname>] [-start <day> <month> <year>]
[-end <day> <month> <year>] <node-ID> [-p <SysOpPsw>]

unicpoutr -v
unicpoutr -h
```

DESCRIPTION

`unicpoutr` copies a resource's data from a calendar server node to a file. It can be used in conjunction with the `unicpinr` utility to move a resource from one node to another as well as to copy the resource agenda from one resource to another.

`unicpoutr` can only be run if the calendar server is up.

The `res` argument must match a single resource or an error is reported. See FORMAT OF THE res ARGUMENT below for details on how to specify this argument.

unicpoutr copies the following information to the file (see unicpr for more information concerning the format and content of the output file):

- resource name

- resource password

- resource information (capacity, phone, etc.)

- agenda-specific preferences

- agenda information

  Agenda information includes the past and future events either owned by the resource or to which the resource is invited. Holiday events are not included unless the -**holiday** option is used. The -**start** and -**end** options may be used to export those events with an attendance record which falls within a specified time period.

The following information is NOT copied to the file:

- access control lists associated with the resource (this includes a description of designate rights granted to and by the resource)

OPTIONS

-end
<day> <month> <year>

Set the end date of the events to be processed. By default, all events are output; this option and the -start option allow you to exclude certain events. Dates must be expressed in the form "day month year". Years must be specified using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-f
<filename>

Specify the output file name. The file must not exist. By default, the standard output is used.

-host

Specify the host on which the database for the specified node is found. The default is the local host.

-p
<sysOpPsw>

Provide the SYSOP password. If this option is not used, prompting for the password occurs.

-start
<day> <month> <year>

Set the start date of the events to be processed. By default, all events are output; this option and the -end option allow you to exclude certain events. Dates must be expressed in the form "day month year". Years must be specified using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-v
Print the current version number of `unicpoutr`.

-h
Print a message explaining how to run `unicpoutr`.

FORMATS

FORMAT OF THE res ARGUMENT
The res argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. The "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. "R=betacam\/loaner/S=Khupfer".

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

Note that if the ID key-value pair is specified in the res argument, all other key-value pairs specified along with it are ignored.

Table C–5   Accepted keys

Key	Field
R	Resource name
N	Resource number
CA	Capacity
S	Contact's surname
G	Contact's given name
ID	Identifier
LOC	Location
PHONE	Phone number
EXT	Phone extension
FAX	Fax phone number

EXAMPLES

- To copy the resource data for the resource "Kitchen" from node 20 to the file `kitchen.dat`:

  ```
  % unicpoutr "R=Kitchen" -f kitchen.dat 20
  ```

- To perform the same task, ignoring events before January 10, 1998:

  ```
  % unicpoutr "R=Kitchen" -f kitchen.dat -start 10 1 1998 20
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNINGS

Agenda size and processing time

Depending on the size of the agenda, `unicpoutr` may take some time to complete.

Limitations of this utility

The `unicp` family of utilities have the following limitations that must be considered.

- *Events*

  From the perspective of a moved user (or resource), each of the moved events in the new agenda is a personal event with enough data in the description to determine who created the event and who the attendees are. All links are broken but there is sufficient information in the description to allow the links to be rebuilt.

  Note also that where the agenda of one user (or resource) is being added to that of another, double-booking may occur.

- *Deleting a user (or resource)*

  When a user (or resource) is moved to a new node, that user (or resource) should be deleted from the old node (using `uniuser -del` (or `unires -del`)).

  When a resource is deleted, all traces of that resource are removed. Thus, that resource is no longer an invitee to events.

  When a user is deleted, all traces of that user are removed. Thus, that user is no longer an invitee to events created by other users. Furthermore, and most importantly, all events created by the user are deleted. As a consequence, any user in the old node who was invited to an event by the moved user will no longer be able to view the event.

- *Moving several users (and/or resources) at a time*

  If several users (and/or resources) are to be moved, it is best to perform the move in three phases:

  1. Copy the information on each user (and/or resource) from the source node to a file (using `unicpoutu` and/or `unicpoutr`).

  2. Delete each user (and/or resource) from the source node.

3. Copy the information on each user (and/or resource) into the destination node using (`unicpinu` and/or `unicpinr`).

This ensures that information on any links among the users (and/or resources) being moved is not lost (see "Deleting a user (or resource)" above).

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

SEE ALSO

`unicpinr, unicpr`

UNICPOUTU

`unicpoutu` - Copy user data from a calendar server node to a file.

SYNTAX

`unicpoutu user [-f <filename>] [-host <hostname>] [-start <day> <month> <year>] [-end <day> <month> <year>] [-holiday] <node-ID> [-p <SysOpPsw>]`

`unicpoutu -v`
`unicpoutu -h`

DESCRIPTION

`unicpoutu` copies a user's data from a calendar server node to a file. It can be used in conjunction with the `unicpinu` utility to copy an agenda from one user to another. Although it can also be used, in conjunction with `unicpinu`, to move a user from one node to another, `unimvuser` is the proper utility for moving users. `unimvuser` prevents the data loss that occurs when moving users with `unicpoutu` and `unicpinu`.

`unicpoutu` can only be run if the calendar server is up.

The user argument must match a single user or an error is reported. See FORMAT OF THE user ARGUMENT below for details on how to specify this argument.

`unicpoutu` copies the following information to the file (see `unicpu` for more information concerning the format and content of the output file):

■ the user's X.400 name and address

■ the user's password

■ the user's personal information. This includes the employee number, phone number, extension, fax number, job title and office mailing address

■ the user's agenda-specific preferences

■ the user's agenda information:

This includes the past and future events either owned by the user or to which the user is invited. Holiday events are not included unless the -**holiday** option is used. The -**start** and -**end** options may be used to export events falling within a specified time period.

Also included are all incomplete tasks and, by default, all completed tasks. The -**start** and -**end** options may be used to export completed tasks falling within a specified time period.

The following information is NOT copied to the file:

■ the access control lists associated with the user -- this includes a description of those rights granted to and by the user, such as designate or viewing rights

■ the user's groups

OPTIONS

-end

<day> <month> <year>

Set the end date of the events and tasks to be processed. By default, all events and tasks are output; this option and the -start option allow you to exclude certain events and tasks. Dates must be expressed in "*day month year*" form. Years must be specified using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-f

Specify the output file name. The file must not exist. By default, standard output is used.

-holiday

Include the holidays from the user's agenda in the output file. Holidays are output as meetings, with all users in the node included as attendees to the meeting. If the user's agenda is subsequently input into a new node using `unicpinu`, only the existing holidays in the new node appear as holidays in the user's agenda; the holidays from the old node appear as meetings.

-host

<hostname>

Specify the host on which the specified node can be found. The default is the local host.

-p

<sysOpPsw>

Provide the SYSOP password. If this option is not used, prompting for the password occurs.

-start

<day> <month> <year>

Set the start date of the events and tasks to be processed. By default, all events and tasks are output; this option and the -**end** option allow you to exclude certain events and tasks. Dates must be expressed in "day month year" form. Years must be specified using four digits. Some legal dates are "12 mar 1995", "15 october 1994", "25 12 1995" (for December 25, 1995). Variations such as "mar 12 1995" or "12 dec" are illegal and produce an error message.

-v

Print the current version number of `unicpoutu`.

-h

Print a usage message explaining how to run `unicpoutu`.

FORMATS

FORMAT OF THE user ARGUMENT

The user argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. The "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

Note: If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored.

Table C–6   Accepted keys

Key	X.400 Field
S	Surname
G	Given name
I	Initials
ID	Identifier
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain

Table C–6 Accepted keys

Key	X.400 Field
P	Private domain

EXAMPLES

■ To copy the user data for "Herman, Sarah" from node 20 to the file "sherman.dat":

```
% unicpoutu "S=Herman/G=Sa*" -f sherman.dat 20
```

■ To perform the same task, ignoring tasks and events before January 10, 1998:

```
unicpoutu "S=Herman/G=Sa*" -f sherman.dat -start 10 1 1998
20
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNINGS

Agenda size and processing time

Depending on the size of the agenda, `unicpoutu` may take some time to complete.

Limitations of this utility

The `unicp` family of utilities have the following limitations that must be considered.

■ *Events*

From the perspective of a moved user (or resource), each of the moved events in the new agenda is a personal event with enough data in the description to determine who created the event and who the attendees are. All links are broken but there is sufficient information in the description to allow the links to be rebuilt.

Note also that where the agenda of one user (or resource) is being added to that of another, double-booking may occur.

■ *Deleting a user (or resource)*

When a user (or resource) is moved to a new node, that user (or resource) should be deleted from the old node (using `uniuser -del` (or `unires -del`)).

When a resource is deleted, all traces of that resource are removed. Thus, that resource is no longer an invitee to events.

When a user is deleted, all traces of that user are removed. Thus, that user is no longer an invitee to events created by other users. Furthermore, and most importantly, all events created by the user are deleted. As a consequence, any user in the old node who was invited to an event by the moved user will no longer be able to view the event.

■ *Moving several users (and/or resources) at a time*

If several users (and/or resources) are to be moved, it is best to perform the move in three phases:

1. Copy the information on each user (and/or resource) from the source node to a file (using `unicpoutu` and/or `unicpoutr`).

2. Delete each user (and/or resource) from the source node.

3. Copy the information on each user (and/or resource) into the destination node using (`unicpinu` and/or `unicpinr`).

This ensures that information on any links among the users (and/or resources) being moved is not lost (see "Deleting a user (or resource)" above).

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

SEE ALSO

`unicpinu`, `unicpu`

UNICPR

unicpr - Format of the file the `unicpoutr` utility creates, and the `unicpinr` utility reads.

DESCRIPTION

The `unicpoutr` utility creates, and the `unicpinr` utility reads, an ASCII file with the following format.

```
{
<resource identification>
}
K Events:
<event descriptions>
```

Except for the open brace bracket and close brace bracket which respectively open and close the *<resource identification>* section of the file, each line of the file begins with a single character code which defines the data stored on that line. A space follows the single character code. The `unicpinr` utility ignores blank lines and lines beginning with unknown codes.

Codes in the *<resource identification>* section are not legal in the *<event descriptions>* section, and vice versa.

The following describes the lines that the *<resource identification>* section may contain.

Table C–7   Accepted lines

Code and Arguments	Data Type	Description
E *encrypt_flag*	boolean	File encryption flag; currently only False is available
F *file_type*	string	File type; currently this is the string "Unison Export File"
I *password*	string	Resource's unencrypted password
N *number*	integer	Number of events in the file

Table C–7   Accepted lines

Code and Arguments	Data Type	Description
P *preferences*	integers	Resource's display and notification preferences; the 12 integers, from first to last, are:
		ShowEventTitles
		StartDay (in minutes)
		EndDay (in minutes)
		StartWeek display
		Display in time increments
		Display days
		Display time format
		Periodic refresh
		Refresh frequency
		Mail notification
		Reminders
		Lead time before reminders
R a *name*	string	Resource's name
R b *number*	string	Resource's number
R c *capacity*	string	Resource's capacity
R d *phone_number*	string	Resource's phone number
R e *extension*	string	Resource's extension number
R f *fax_number*	string	Resource's fax number
V *version#*	string	Version number; currently this is A.02.53
X *contact_data*	string	Contact's X.400 data

The following describes the lines that the *<event descriptions>* section may contain.

Table C–8   Accepted lines

Code and Arguments	Data Type	Description
S *date*	string	Event start time in (UNIAPI_TIME) or as a date specification
D *duration*	integer	Event duration in minutes
T *title*	string	Event title
G *location*	string	Event location
I *class*	integer	Event class (normal, holiday, ...)
R *type&priority*	string	Event type (normal, public,...) followed by priority (1, 2,...)
M *creator*	string	Event creator
W *owner*	string	Event owner
A *attending reminder leadtime*	string	Indicates whether resource is attending (TRUE/FALSE); Visual Reminder (0, 1); Leadtime (in minutes)
C *description*	string	Event description; this may span several lines and include a list of attendees. When it does span more than one line, each subsequent line must begin with "C" as well.
O		End of an event instance

EXAMPLES

■   The following is an example of a file containing one event from the kitchen resource agenda.

```
{
F Unison Export File
V A.04.00
E FALSE
X S=Baker/G=James
R a Kitchen
R b 12
R c 10
R d (123)-456-7890
R e 217
R f (123)-456-7891
I Password
```

```
P 4 480 1140 1 15 127 2 0 15 0 0 0
#These define display and notification preferences as follows:
#       4     ShowEventTitles
#       480   StartDay at 8:00am
#       1140  EndDay at 7:00pm
#       1     StartWeek display on Sunday
#       15    Display in time increments of 15 min.
#       127   Display all days Sunday to Saturday
#       2     Display time in am/pm format
#       0     Periodic refresh disabled
#       15    Refresh frequency 15 minutes
#       0     Mail notification disabled
#       0     No reminders
#       0     No lead time before reminders
N 1
}

K Events:
S 2262975
D 75
T Lunch
G Kitchen
I 0
R N0
M Kitchen
W Kitchen
A TRUE 1 5
C Lunch time
C David Robinson
C Kathy Bates
O
```

SEE ALSO

unicpinr(8), unicpoutr(8).

UNICPU

unicpu - File format of the file the unicpoutu utility creates, and the unicpinu
utility reads.

DESCRIPTION

The `unicpoutu` utility creates, and the `unicpinu` utility reads, an ASCII file with the following format.

```
{
<user identification>
}
K Events:
<event descriptions>
K Tasks:
<task descriptions>
```

Except for the open brace bracket and close brace bracket which respectively open and close the *<user identification>* section of the file, each line of the file begins with a single character code which defines the data stored on that line. A space follows the single character code. The `unicpinu` utility ignores blank lines and lines beginning with unknown codes.

Codes in the *<user identification>* are not legal in either of the other two sections, and vice versa.

The following describes the lines that the *<user identification>* section may contain.

Table C–9   Accepted lines

Code and Arguments	Data Type	Description
E *encrypt_flag*	boolean	File encryption flag; currently only False is available
F *file_type*	string	File type; currently this is the string "Unison Export File"
I *password*	string	User's unencrypted password
N *events tasks*	integers	Number of events, tasks in the file; *events* is the number of events, *tasks* is the number of tasks

Table C–9   Accepted lines

Code and Arguments	Data Type	Description
P *preferences*	integers	User's display and notification preferences; the 12 integers, from first to last, are:
		ShowEventTitles
		StartDay (in minutes)
		EndDay (in minutes)
		StartWeek display
		Display in time increments
		Display days
		Display time format
		Periodic refresh
		Refresh frequency
		Mail notification
		Reminders
		Lead time before reminders
U a *address*	string	User's address. This can span multiple lines, and when it does, each additional line must also begin with "U a ".
U b *empl_number*	string	User's employee number
U c *phone_number*	string	User's phone number
U d *fax_number*	string	User's fax number
U e *extension*	string	User's extension number
U f *job_title*	string	User's job title
V *version#*	string	Version number; currently this is A.02.51
X *user_data*	string	User's X.400 data

The following describes the lines that the *<event descriptions>* section may contain.

Table C–10   Accepted lines

Code and Arguments	Data Type	Description
S *date*	string	Event start time in (UNIAPI_TIME) or as a date specification
D *duration*	integer	Event duration in minutes
T *title*	string	Event title
G *location*	string	Event location
I *class*	integer	Event class (normal, holiday, ...)
R *type&priority*	string	Event type (normal, public,...) followed by priority (1, 2,...)
M *creator*	string	Event creator
W *owner*	string	Event owner
A *attending reminder leadtime*	string	Indicates whether user is attending (TRUE/FALSE); Visual Reminder (0, 1); Leadtime (in minutes)
C *description*	string	Event description; this may span several lines and include a list of attendees. When it does span more than one line, each subsequent line must begin with "C" as well.
O		End of an event instance

The following describes the lines that the *<task descriptions>* section may contain.

Table C–11   Accepted lines

Code and Arguments	Data Type	Description
S *starttime*	string	Task start time in (UNIAPI_TIME) or as a date specification
D *endtime*	string	Task end time as a date specification
T *title*	string	Task title
R *priority*	integer	Task priority
L *compl_level*	integer	Completion level
M *creator*	string	Task creator

Table C–11   Accepted lines

Code and Arguments	Data Type	Description
W *owner*	string	Task owner
C *description*	string	Task description; this may span several lines. When it does span more than one line, each subsequent line must begin with "C" as well.
O		End of a task instance

EXAMPLES

■   The following is an example of a file containing two events and one task from the agenda of James Baker.

```
{
F Unison Export File
V A.02.51
E FALSE
X S=Baker/G=James/I=T/OU1=Labs/OU2=SysAdmin
U a Calendar Server Corporation,
U a 1234 Software Blvd.,
U a Suite 999,
U a Software Valley, CA 99999.
U b 12
U c (123)-456-7890
U d (123)-456-7891
U e 215
U f System Administrator
I Password
P 4 480 1140 1 15 127 2 0 15 0 0 0
#These define display and notification preferences as follows:
#       4     ShowEventTitles
#       480   StartDay at 8:00am
#       1140  EndDay at 7:00pm
#       1     StartWeek display on Sunday
#       15    Display in time increments of 15 min.
#       127   Display all days Sunday to Saturday
#       2     Display time in am/pm format
#       0     Periodic refresh disabled
#       15    Refresh frequency 15 minutes
#       0     Mail notification disabled
#       0     No reminders
#       0     No lead time before reminders
```

```
N 2 1
}
K Events:
S 2262975
D 75
T Friday R&D meeting
G Conference Room
I 0
R N0
M Baker James
W Baker James
A TRUE 1 5
C Discuss next week's activities.
C James Baker
C David Robinson
C Kathy Bates
O
S D=25/Y=2000/M=April/T=00:00/z=EST5EDT
D 1440
T Company Holiday
I 1
R A2
M Robinson David
W Robinson David
A TRUE 0 0
O
K Tasks:
S D=1/Y=2000/M=April/T=8:00/z=EST5EDT
D D=30/Y=2000/M=April/T=17:00/z=EST5EDT
T System Overhaul.
R 3
L 70
M Baker James
W Baker James
C Upgrade OS version from A.02.50 to A.04.51
O
```

SEE ALSO

unicpinu(8), unicpoutu(8)

UNIDBBACKUP

`unidbbackup` - Create an archive of a calendar server node and related configuration information.

SYNTAX

```
unidbbackup -d <dst>

unidbbackup -v
unidbbackup -h
```

DESCRIPTION

`unidbbackup` creates a backup of a calendar server node and its related configuration information. More specifically, it creates a backup of the `/users/unison/misc` directory and the `/users/unison/db` directory. As the information in these two directories is interrelated, it is important to ensure they are backed up at the same time.

`unidbrestore` is the complementary utility to `unidbbackup`. By default, these utilities perform a copy of the source to the destination. If behaviour other than a straight copy is needed, an alternate backup/restore command can be specified using the `[UTL] external_backup` and `external_restore` parameters in the `unison.ini` file. See FILES below for details on how to specify an alternate backup command.

> **Warning:**   **The backup and restore commands are inverse operations so if alternate commands are used, it is of critical importance to ensure they do in fact perform the inverse operation of each other. The integrity of the database is at stake.**

`unidbbackup` can be run when the calendar server is either up or down.

> **Warning:** unidbbackup backs up the calendar server internal database. If a directory server is being used, its database should also be backed up.

OPTIONS

-d

<dst>

Specify the destination for the archive, where <dst> is a directory name.

-v

Print the current version number of unidbbackup.

-h

Print a usage message explaining how to run unidbbackup.

EXAMPLES

- Back up the calendar server to the directory `/backups/cserver/jan.7.99`:

  ```
  % unidbbackup –d /backups/cserver/jan.7.99
  ```

EXIT STATUS

Exit values are:

0 Success

Any non-zero value signals an error

FILES

`/users/unison/misc/unison.ini`
The following keys in the [UTL] section of this file are of relevance to this utility:

- `lock_timeout`

  This key sets the timeout, in seconds, for the lock operation on the database.

- `backup_timeout`

  This key sets the timeout, in seconds, for the backup operation on the database.

- `external_backup`

  This key provides a way for an alternate backup utility to be invoked by `unidbbackup`. `unidbbackup` uses the value of this key, along with the arguments supplied to `unidbbackup` on the command line, to construct (and subsequently invoke) the following command line:

  ```
  value_of_external_backup [-f] -s <src> -d <dst>
  ```

  where:

  – <dst> specifies the destination for the backup (`unidbbackup` constructs this from the <dst> argument specified by the user on the `unidbbackup` command line)

  – <src> specifies the source to be backed up (`unidbbackup` constructs this argument based on the information it finds in the `/users/unison/misc/unison.ini` file)

  – -**f** indicates that the source is a file (absence of this flag indicates the source is a directory)

  `unidbbackup` iteratively invokes the generated command line until all of the required database files are backed up, locking and unlocking the database for each iteration.

  The administrator must ensure that the generated command line is in fact a valid one for the alternate utility. It may be that an intermediate utility is required to take this command line, create one which is valid, and then invoke it. In this case, `external_backup` would be set to invoke the intermediate utility.

  The accepted value for `external_backup` is any command line. There is no assigned default value for this key.

SEE ALSO
`unidbrestore`

UNIDBCONV

`unidbconv` - Convert a version 2.50, 2.60 or 2.61 node database to version 2.62.

SYNTAX
```
unidbconv -n <node-ID> | all [-p <pagevalue>]
```

```
unidbconv -v
unidbconv -h
```

DESCRIPTION

unidbconv converts a version 2.50, 2.60 or 2.61 node database to version 2.62. In general you do not invoke this utility directly (a conversion is done automatically during the upgrade to a newer version of the calendar server). The last two digits of the "version" parameter in the [<YOURNODEID>] section of the unison.ini file indicate the version of the node.

> **Warning: You should back up the calendar server before invoking unidbconv as this utility overwrites the existing database.**

The calendar server must be down to run unidbconv.

OPTIONS

-n

<node-ID> | all

Perform the conversion on the specified node only (if <node-ID> is used) or on all nodes (if all is used).

-p

<pagevalue>

Specify the number of cache pages to use for scanning and rebuilding key files. If this option is not used, the default value of 256 is used instead.

-v

Print the version number of unidbconv.

-h

Print a usage message explaining how to run unidbconv.

EXAMPLES

■ Convert all calendar server node databases to version 2.61 node databases:

```
% unidbconv -n all
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIDBFIX

`unidbfix` - Check, repair, defragment and maintain a calendar server node database.

SYNTAX

```
unidbfix -c [-pix] -n <node-ID> | all [-r] [-sfgn] [-kp <pagevalue>]

unidbfix -f [-pix] -n <node-ID> | all [-r] [-y] [-sfgn] [-kp <pagevalue>]

unidbfix -d [-pix] -n <node-ID> | all [-r] [-y] [-sfgn] [-kp <pagevalue>]

unidbfix -export [-pix] -n <node-ID> | all [-r] [-kp <pagevalue>]

unidbfix -import [-pix] -n <node-ID> | all [-r] [-y] [-kp <pagevalue>]

unidbfix -ck -n <node-ID> | all [-r] [-y] [-kp <pagevalue>]

unidbfix -k -n <node-ID> | all [-r] [-y] [-kp <pagevalue>]

unidbfix -i [-pix] -n <node-ID> | all [-r] [-kp <pagevalue>]

unidbfix -v
unidbfix -h
```

DESCRIPTION

unidbfix checks for and repairs database corruptions and/or inconsistencies, and/or defragments and compresses a node database. You should run unidbfix as part of a regular database maintenance program.

Warning: **Database corruption may occur if you do not use the version of unidbfix that ships with, or is compatible with, the version of the calendar server you are running. Consult Oracle Support if you have any questions on compatibility.**

Warning: **Before invoking this utility with one of the -f, -d, or -import options it is highly recommended that you make a backup of the database. You only need to back up the data (*.dat) files as unidbfix can reconstruct the key (*.key) files from the data files.**

---

unidbfix carries out checks/repairs on the following parts of the database of the specified node:

- Remote Nodes
- Host Node
- Nextslot and File Size
- Records
- Calendar-dependent Data Fields
- Pointers
- Delete Chain
- Key Files

unidbfix runs in one of eight different modes as listed below. If, in any mode, unidbfix makes a fix, it reports that fix. The scan phases for each mode appear in the order in which they occur. See the **NOTES** section for additional information on the Remote Nodes, Bins, and File Fragmentation scan phases.

Table C–12 `unidbfix` *modes*

Mode	Option	Scan Phases	Changes Database
check	**-c**	File Sizes	No
		Nodes	
		Remote Nodes	
		Records	
		Sets	
		Bins	
		Dchain	
		Key Check	
		Database Info	
fix	**-f**	File Sizes	Yes
		Nodes	
		Remote Nodes	
		Records	
		Sets	
		Bins	
		Dchain	
		File Fragmentation	
		Key Build	
		Database Info	
defragment	**-d**	Files Sizes	Yes
		Nodes	
		Remote Nodes	
		Records	
		Sets	
		Bins	
		Dchain	
		File Fragmentation	
		Key Build	

Table C–12 `unidbfix` **modes**

Mode	Option	Scan Phases	Changes Database
import	**-import**	RemoteNodes	Yes
		Key Build	
export	**-export**	Remote Nodes	No
check key	**-ck**	Key Build (in check mode)	No
fix key	**-k**	Key Build (in fix mode)	Yes
info	**-i**	Database Info	No

`unidbfix` only runs if the calendar server is down.

OPTIONS

-c

Run in check mode. `unidbfix` reports all database corruptions and inconsistencies but takes no action to correct them (use fix mode to do this). If `unidbfix` detects an error, it stops the check after the scan phase in which it finds the errors. For instance, if it discovers an error during the File Sizes scan phase, it terminates on completion of this scan phase. It does not proceed to the Nodes scan phase.

-ck

Run in check key mode. Checks only the key files of the database.

-d

Run in defragment mode. In this mode `unidbfix` frees space occupied by deleted records and then compresses the database. To ensure database consistency, `unidbfix` checks the database for errors and fixes any it finds before it proceeds with defragmentation.

Warning: While it is possible to interrupt unidbfix during the defragmentation phase using a kill -9, this causes irreversible damage to the database.

-export

Run in export mode. In export mode unidbfix writes remote node information from the database to the remotenode.ini file. Note that it writes only the non-null fields for each remote node to the file. See the REMOTE NODES SCAN PHASE note for an example of how to use the -**export** mode.

-f

Run in fix mode. Fix and clean up the database. This fixes all errors detected in check mode. In some circumstances unidbfix may be forced to delete data (e.g. where corruption to the data is such that unidbfix is unable to repair it, or where orphan data cannot be safely re-integrated).

-i

Run in info mode. In this mode unidbfix outputs various database statistics to the dbfix.log file.

-import

Run in import mode. In import mode unidbfix writes remote node information from the remotenode.ini file to the database. See the REMOTE NODES SCAN PHASE note for an example of how to use the -**import** mode as well as warnings on its use.

-k

Run in fix key mode. Rebuilds only the key files of the database.

-kp

<pagevalue>

Specifies the number of cache pages to use for scanning and rebuilding key files. If this option is not used, the default value of 256 is used instead.

-n

<node-ID> | all

Specify the node to check/fix/defragment. Use `-n all` to scan all the nodes on a computer.

-pix

Turn off the progress indicator. By default `unidbfix` outputs a progress indicator, for each utility it calls, to standard error.

-r

Overwrite the `/users/unison/log/dbfix.log` log file, rather than append output to it.

-sfgn

Turn on foreign node checking and fixing. Use only if you have foreign nodes and items.

> **Warning:** Using this option when you do not have foreign nodes and items may result in changes to the database. These changes may or may not be problematic.

-y

Turn fix and defragmentation confirmation message off.

-v

Print the current version number of unidbfix.

-h

Print a usage message, and a short description of each option.

EXAMPLES

- Check the consistency of node 35:

  ```
  % unidbfix -c -n 35
  ```

- Fix node 12:

  ```
  % unidbfix -f -n 12
  ```

- Defragment and compress node 10 and overwrite the log file:

```
% unidbfix –d –n 10 –r
```

FILES

`/users/unison/log/dbfix.log`
`unidbfix` writes any errors it finds and/or any fixes it makes, to this file. It lists each error as a DATABASE ERROR, and each repair as a Fix. `unidbfix` can repair any database error it finds. Totals of all errors found, fixes made, and records deleted during fixing, appear at the end of the file. Note that the total number of database errors need not equal the total number of fixes. You do not normally need to consult this file.

`/users/unison/log/unison.ini`
Consult this file for a listing of all local nodes, with their corresponding directory names and node-IDs.

`remotenode.ini`
`unidbfix` uses this file in import and export modes. It creates this file in a node's `perm` directory the first time it runs on the node. The file contains a listing of all the remote node records and their data fields. The information for each remote node is as follows:

```
[Node–ID]
RN_NUMCONNECT:     any number zero and above
RN_ACCESSMETHOD:   must be 2
RN_SERVICENAME:    must be "unieng"
RN_HOSTNAME:       name of the remote host
```

Node-ID is the remote node identification number. It must be enclosed in square brackets and it must start a line. A field can have a null value. If any field has an invalid value, `unidbfix` returns an error message, and does not make the change for the remote node with the error.

The following sample `remotenode.ini` file contains two remote nodes: the first has the node-ID 730 and the name "NewYork"; the second has the node-ID 631 and the name "LosAngeles".

```
[730]
RN_NUMCONNECT = 2
RN_ACCESSMETHOD = 2
RN_SERVICENAME = "unieng"
RN_HOSTNAME = "NewYork"

[631]
RN_NUMCONNECT = 2
RN_ACCESSMETHOD = 2
```

```
RN_SERVICENAME = "unieng"
RN_HOSTNAME = "LosAngeles"
```

`unidbfix.lck`
This is a lock file which prevents multiple instances of `unidbfix` from running on the same node simultaneously. `unidbfix` creates this in the `perm` directory of the node on which it is running. In the event that a kill -9 or a system crash prevents `unidbfix` from running to completion, this file remains in place. It may be manually deleted.

EXIT STATUS

Exit values are:

0 Success

No errors found (check mode)

Errors found but fixed (fix mode)

Successfully defragmented (defragment mode)

Successful import (import mode)

Successful export (export mode)

1 Errors Found

Errors were found (check mode)

2 Usage error

3 User interrupt

4 Aborted

Another instance of `unidbfix` was running on the node.

5 Stopped

`unidbfix` either found errors in the remote node records while in fix or check mode, or it could not find the `remotenode.ini` file. It needed more information to be able to continue checking or fixing.

NOTES

KEY FILES

Note that `unidbfix` rebuilds the key files of the database in fix, defragment, import, and fix key modes. If `unidbfix` is interrupted during any of these modes, the key files may have been deleted and not yet rebuilt. For this reason, it is highly recommended that you run `unidbfix` again after an interruption.

BINS AND FILE FRAGMENTATION SCAN PHASES

In the Bins and the File Fragmentation scan phases, `unidbfix` rebuilds files without checking for, or reporting, previously existing errors. In all other scan phases all errors reported in check mode are reported in fix mode before being fixed.

REMOTE NODES SCAN PHASE

For this scan phase to run, the node's `remotenode.ini` must exist, and its contents must agree with the list of remote nodes in the database. When one of these conditions is not met, you can use the -**export** and -**import** modes to rectify the situation. The explanations that follow use the node-ID "43".

1. CONDITION: A `remotenode.ini` file does not exist for node 43. In this case, generate one from the remote node list in the database:

```
% unidbfix -export -n 43
```

2. CONDITION: The remote node list in the database does not agree with the information in the `remotenode.ini` file for node 43. In this case, rectify the discrepancy as follows.

First write the remote node information from the database to the `remotenode.ini` file for node 43:

```
% unidbfix -export -n 43
```

Make any required edits to the resulting `remotenode.ini` file.

Warning: Edit with care! Errors in this file may lead to unwanted deletion of records when the file is imported. For this reason it is highly recommended that you back up the database before running unidbfix in -import mode.

Update the database with the modified file:

```
% unidbfix -import -n 43
```

Warning: Use uninode, not unidbfix -import, to administer the node network. Use unidbfix with the -import option only to fix corruptions in the remote node connection information in the database.

SEE ALSO

unistart, unistop, uninode

UNIDBRESTORE

unidbrestore - Restore a calendar server node and configuration information from a backup created by unidbbackup.

SYNTAX

unidbrestore -s <src> [-d <dst>] [-n <node-ID>]

unidbrestore -v
unidbrestore -h

DESCRIPTION

unidbrestore - restores the node and configuration information of a calendar server from a backup created by unidbbackup.

> **Warning:** By default, the destination directory for the restore is /users/unison. This means that the restore overwrites the existing files of the calendar server database. Thus, this utility should be used with extreme care to ensure the calendar server database is not inadvertently corrupted. A more careful approach would be to use the -d option to specify a different directory for the restore and then copy the individual files from the restored directory into the /users/unison directory.

unidbbackup is the complementary utility to unidbrestore. By default, these utilities perform a copy of the source to the destination. If behaviour other than a straight copy is needed, an alternate backup/restore command can be specified using the [UTL] external_backup and external_restore parameters in the unison.ini file. See FILES below for details on how to specify an alternate restore command.

> **Warning:** The backup and restore commands are inverse operations so if alternate commands are used, it is of critical importance to ensure they do in fact perform the inverse operation of each other. The integrity of the database is at stake.

unidbrestore can only be run when the calendar server is down.

> **Warning:** unidbrestore restores the calendar server's internal database. If a directory server is being used, its database is untouched by unidbrestore. Therefore, if you restore a calendar server node after deleting users, you will have to add them back into the directory server. Similarly, if you restore a single node after changing node network information, you will encounter errors due to the conflict between the current network configuration and the restored node's old network information. Contact Oracle support for more details if this occurs to you.

OPTIONS

-d

Specify the destination for the restore. By default this is the /users/unison directory.

-n

<node-ID>

Specify a node to restore.

-s

<src>

Specify the backup source, where <src> is a directory name.

-v

Print the current version number of unidbrestore.

-h

Print a usage message explaining how to run unidbrestore.

EXAMPLES

- Restore node 45 of the calendar server backup /backups/cserver/jan.7.99 to the directory /users/unison:

  ```
  % unidbrestore -s /backups/cserver/jan.7.99 -n 45
  ```

EXIT STATUS

Exit values are:

0 Success

Any non-zero value signals an error

FILES

/users/unison/misc/unison.ini
The following parameters in the [UTL] section are of relevance to this utility:

- lock_timeout

  This parameter sets the timeout, in seconds, for the lock operation on the database.

- restore_timeout

This parameter sets the timeout, in seconds, for the restore operation on the database.

■   `external_restore`

This parameter provides a way for an alternate restore utility to be invoked by `unidbrestore`. `unidbrestore` uses the value of this parameter, along with the arguments supplied to `unidbrestore` on the command line, to construct (and subsequently invoke) the following command line:

`value_of_external_restore [-f] -s <src> -d <dst>`

where

–   `-d <dst>` specifies the destination for the restore (`unidbrestore` constructs this from the dst argument supplied on the `unidbrestore` command or if no argument was supplied, uses the default)

–   `-s <src>` specifies the source to be restored (unidbrestore constructs this from the `src` argument supplied on the unidbrestore command line)

–   `-f` indicates that the source is a file (absence of this flag indicates the source is a directory)

`unidbrestore` iteratively invokes the generated command line until all of the required database files are restored, locking and unlocking the database for each iteration.

It is up to the user to ensure that the generated command line is in fact a valid one for the alternate utility. It may be that an intermediate utility is required to take this command line, create one which is valid, and then invoke it. In this case, "external_restore" would be set to invoke the intermediate utility.

The accepted value for "external_restore" is any command line. There is no assigned default value for this key.

SEE ALSO
`unidbbackup`

UNIDOMAIN

`unidomain` - Create a new calendar domain. For Business-to-Business (B2B) deployments only.

SYNTAX

```
unidomain -add -hostalias <alias> -domainid <id> -domainname <name> -t
<timezone> -adminpassword <password> [-cds <hostname>] [-n <node-ID>] [-p
<sysOpPassword>] [-disablepasswordmod] [-capacity <num>] [-reldn <dn>]
[-resreldn <dn>] [-logo <imagename>] [-sitelogo <imagename>] [-sitename <name>]
[-templatedir <dir>] [-supportemail <addr>] [-manageuseroff]
[-manageresourceoff] [-manageholidayoff] [-manageeventcaloff] [-disable] [-y]

unidomain -listaliases [-cds <hostname>] [-p <sysOpPassword>]

unidomain -listdomainids [-cds <hostname>] [-p <sysOpPassword>]

unidomain -v
unidomain -h
```

DESCRIPTION

unidomain is a command-line alternative to the Calendar Administrator for creating calendar domains. A domain is the representation on the calendar server of a single customer account. Each domain is associated with a single calendar server node. For more information on domains and the Calendar Domain Service, consult your calendar server *Administrator's Guide*.

You must use the -**domainid** option to specify a Domain ID for the new domain. This Domain ID must be unique in the Calendar Domain Service specified by the -**cds** option. To obtain a current list of all domain IDs in use on a given Calendar Domain Service, run unidomain with the -**listdomainids** option.

You must also use the -**hostalias** option to specify a calendar server alias for the new domain. The required alias is NOT simply the hostname of the calendar server, but one of the parameters listed in the [HOSTS] section of the /users/unison/misc/domain/domain.ini file. To see a list of all available calendar server aliases and the names of their corresponding hosts, run unidomain with the -**listaliases** option.

By default, unidomain will display the information you have specified for the new domain, and prompt you to continue if that information is correct.

unidomain will write many of the domain properties you specify in the /users/unison/misc/domain/domain.ini file. These parameters may be changed after domain creation. See Chapter 5 of your calendar server's *Administrator's Guide* for details on the parameters involved.

unidomain will also write a value for the domain.ini [YOURDOMAINID] adm.adminurl parameter. This URL provides an address for the Domain Administrator to manage the new domain through the Calendar Administrator.

unidomain can only be run when the calendar server is up.

OPTIONS

-add
Creates a new calendar domain.

-adminpassword
<password>

Specify an initial password for the Domain Administrator of the domain being created.

-capacity
Specify the maximum number of users and event calendars combined that may be created in the new domain. This value may be changed later using the Calendar Administrator. If this option is not used, the default value of 10 is assumed.

-cds
<hostname>

Specify a Calendar Domain Service host. If this option is not used, the local host is assumed.

-disable
Disable the new calendar domain. The Calendar Administrator will still be able to manage the domain normally, but clients featuring Domain-ID sign-in will be blocked (Outlook Connector 2.0 and greater, native clients 5.1 and greater, Web clients 2.2 and greater).

-disablepasswordmod
Prevent the Domain Administrator of the new domain from changing his or her password through the Calendar Administrator.

-domainid

<id>

Specify a Domain ID for the new domain. This ID is usually a short string representative of the customer account. The value of the <id> argument may be any non-null alphanumeric string beginning with a letter that is unique in the Calendar Domain Service indicated by the -**cds** option. Domain IDs are case-insensitive; the Domain ID "acme1" cannot be used if the ID "ACME1" already exists. See also the -**listdomainids** option.

-domainname

<name>

Specify the name of the company represented by the new domain.

-hostalias

<alias>

Specify the Calendar Domain Service alias of the calendar server host on which the new domain is to be created. The value of the <alias> argument must be one of the aliases configured in the [HOSTS] section of the /users/unison/misc/domain/domain.ini file. See also the -**listaliases** option.

-listaliases

Lists all available Calendar Domain Service aliases and calendar server hostnames currently in use on the Calendar Domain Service host specified by the -**cds** option.

-listdomainids

Lists all Domain IDs currently in use on the Calendar Domain Service specified by the -**cds** option.

-logo

<imagename>

Specify an image file to display for this domain in the Calendar Administrator. The value of the <imagename> argument may be a filename within the Calendar Administrator's configured images directory or a fully qualified URL.

-manageeventcaloff

Remove from the Domain Administrator of this domain the right to manage event calendars and their events through the Calendar Administrator.

-manageholidayoff

Remove from the Domain Administrator of this domain the right to manage holidays through the Calendar Administrator.

-manageresourceoff

Remove from the Domain Administrator of this domain the right to manage resources through the Calendar Administrator.

-manageuseroff

Remove from the Domain Administrator of this domain the right to manage users through the Calendar Administrator.

-n

<node-ID>

Specify a node to associate with this domain. If this option is not used, unidomain will use the first empty node it finds. If you use this option, the node you specify must already exist on the host indicated by the -**hostalias** option, and must not be associated with any other domain. Use of this option is strongly discouraged in most circumstances.

-p

<sysOpPassword>

Specify the SYSOP password; this password should be the same for all calendar server nodes. if you do not supply the password, unidomain will prompt for it.

-reldn

<dn>

Specify a Relative DN for this domain, relative to the calendar server base DN. Required only for installations using an external LDAP directory server.

-resreldn

Specify a Resource Relative DN for this domain, relative to the domain's Relative DN (specified by the -**reldn** option). Only for installations using an external LDAP directory server. If this option is not used, the default value of `"ou=Resources"` is assumed.

-sitelogo

<imagename>

Specify an image file to display in the Calendar Administrator to represent the provider for this domain. The value of the <imagename> argument may be a filename within the Calendar Administrator's configured images directory or a fully qualified URL.

-sitename

<name>

Specify the name of this domain's provider as it should appear in the Calendar Administrator. The value of the <name> argument will override the Calendar Administrator's default Provider Name for this domain only. Other existing domains and future domains will not be affected.

-supportemail

<addr>

An e-mail address to display in the Calendar Administrator for the Domain Administrator to contact in case of errors or problems.

-t

<timezone>

The time zone of the new domain. Use any of the values listed in the calendar server Reference Manual, Appendix D, "Time Zone Table".

-templatedir

<dir>

Specify a template directory for the Calendar Administrator to use for this domain. The value of the <dir> argument may be either a path relative to the Calendar Administrator CGI or FastCGI file, or it may be a fully qualified URL. The value of the <dir> argument will override the Calendar Administrator's default template

directory for this domain only. Other existing domains and future domains will not be affected.

-y
Proceed without asking for confirmation.

-v
Print the current version number of `unidomain`.

-h
Print a usage message explaining how to run `unidomain`.

EXAMPLES

■   To view a list of all Domain IDs already in use:

```
% unidomain -listdomainids -cds cds.oracle.com

The following Domain IDs already exist:

ACME1
ACME2
TESTCO
```

■   To view a list of all Calendar Domain Service aliases currently available, and the names of the calendar server hosts to which the aliases point:

```
% unidomain -listaliases -cds cds.oracle.com

Alias                  Hostname

calserver3             host3.oracle.com
calserver2             host2.oracle.com
calserver1             host1.oracle.com
```

■   To create a domain for Acme Corporation on the server represented by the alias "`calserver1`", granting the Domain Administrator the right to manage users, resources and holidays but not event calendars:

```
% unidomain -add -hostalias calserver1 -domainid acme1 -domainname "Acme
Corporation" -adminpassword sesame -t EST5EDT -cds cds.oracle.com
-manageeventcaloff -p sysopPassword

unidomain: Calendar domain provisioned successfully.
```

FILES

`/users/unison/misc/domain/domain.ini`
This configuration files is located on the Calendar Domain Service host, and contains most of the information that can be specified in the command-line argument given to this utility. See Chapter 5 of your calendar server's *Administrator's Guide* for details on the parameters involved.

EXIT STATUS

Exit values are:

0 Success

Any non-zero value signals a failure

UNIDSACISETUP

`unidsacisetup` - Set the access control information in the directory server for the calendar server ADMIN group.

SYNTAX

```
unidsacisetup [-w <mgrDnPwd>]

unidsacisetup -info

unidsacisetup -v
unidsacisetup -h
```

DESCRIPTION

`unidsacisetup` sets the directory server access control information (ACI) for the calendar server ADMIN group. Although you can use directory server utilities to set ACIs, it is advisable to use `unidsacisetup` to ensure the ACI for the ADMIN group is properly set. Most calendar server utilities do not run unless the ACI for the ADMIN group is set.

This utility should be run every time a new calendar server ADMIN group is created, i.e. every time the `[LDAP] admingroup` parameter in the `unison.ini` file is changed.

`unidsacisetup` runs whether the calendar server is up or down. The directory server, however, must be running.

OPTIONS

-info

Display the list of directory servers for which this utility can create access control information.

-w

<mgrDnPwd>

Provide the directory server manager password (this is the password associated with the `[LDAP] mgrdn` parameter in `unison.ini`). If this option is not used, `unidsacisetup` prompts the user for the password.

-v

Print the version number of `unidsacisetup`.

-h

Print a usage message explaining how to run `unidsacisetup`.

EXAMPLES

- Display the list of directory servers for which `unidsacisetup` can set ACI:

  ```
  % unidsacisetup -info
  ```

- Set the ACI for the calendar server ADMIN group:

  ```
  % unidsacisetup
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIDSDIFF

`unidsdiff` - Find and delete differences between a calendar server node and a directory server.

SYNTAX

```
unidsdiff [-d] [-noprompt] [-n <node-ID>] [-host <hostname>] [-p <SysOpPsw>]
[-verbose]

unidsdiff -v
unidsdiff -h
```

DESCRIPTION

This utility finds all users and resources in a calendar server node without a match in the directory server and vice versa. By default, it only reports discrepancies. Use the -**d** option to delete discrepancies.

The calendar server assigns each user and resource a unique identifier called an `xItemId`. `unidsdiff` first checks that each `xItemId` (for the specified node) in the directory server:

1. is unique

2. has a single user or resource associated with it

3. is expressed in a valid format

If `unidsdiff` detects an `xItemId` which does not pass one of these checks, it aborts; directory server utilities must be used to correct the problem. Otherwise `unidsdiff` proceeds to verify that:

4. all users and resources in the calendar server node appear in the directory server (if the -**d** option was used, any users or resources appearing only in the calendar server node are removed)

5. all calendar users and resources in the directory server appear in the calendar server node (if the -**d** option was used, any calendar users or resources appearing only in the directory server are removed from the directory server, i.e. they no longer appear as calendar users in the directory server).

The calendar server must be up to run `unidsdiff`.

OPTIONS

-d

Delete the differences found. The user is prompted to confirm each deletion. Without the -**d** option, unidsdiff simply lists the differences.

-host

<hostname>

Specify the host to connect to. Required if host is remote.

-n

<node-ID>

Specify a node. Required if more than one exists.

-noprompt

Disable prompting when used with the -**d** option.

-p

<SysOpPsw>

Provide the SYSOP password.

-verbose

Display all Distinguished Names in the directory associated with the node.

-v

Print the current version number of unidsdiff.

-h

Print a usage message explaining how to run unidsdiff.

EXAMPLES

- Run unidsdiff on node 10:

```
% unidsdiff -n 10 -host inkpen
Enter SYSOP password:
unidsdiff: detected 0 duplicate "ctCalXItemId" attributes in directory
unidsdiff: detected 0 multi-valued "ctCalXItemId" attributes in directory
```

```
unidsdiff: detected 0 badly-formed "ctCalXItemId" attributes in directory
unidsdiff: detected 0 calendar-stores without a matching directory entry
unidsdiff: detected 0 calendar directory entries without a matching
calendar-store
```

In this case, no discrepancies were found between the directory server and the calendar server. A verbose version of the same command would result in the following output:

```
% unidsdiff -n 10 -host inkpen -verbose
Enter SYSOP password:
DN="cn=Lorde Audre,o=Acme,c=us"<ctCalXItemID010:00346>
DN="cn=Kilpi Eeva,o=Acme,c=us"<ctCalXItemID010:00347>
:
:
DN="cn=Cohen Leonard,o=Acme,c=us"<ctCalXItemID010:00484>
DN="cn=Atwood Margaret,o=Acme,c=us"<ctCalXItemID010:00485>
DN="cn=Brossard Nicole,o=Acme,c=us"<ctCalXItemID010:00486>
unidsdiff: detected 0 duplicate "ctCalXItemId" attributes in directory
unidsdiff: detected 0 multi-valued "ctCalXItemId" attributes in directory
unidsdiff: detected 0 badly-formed "ctCalXItemId" attributes in directory
unidsdiff: detected 0 calendar-stores without a matching directory entry
unidsdiff: detected 0 calendar directory entries without a matching
calendar-store
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNINGS

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

UNIDSSEARCH

unidssearch - List all users in a directory server who are not calendar users.

SYNTAX

```
unidssearch [-f <LDAPfilter>] [-c <#ofDNs>] [-n <node-ID>]

unidssearch -v
unidssearch -h
```

DESCRIPTION

unidssearch lists all users in the directory server who are not calendar users. The output of this command may be redirected to a file, modified as needed, and subsequently used as input to uniuser (using the -**ex** option). See OUTPUT FORMAT for information on the format of the file output by unidssearch.

The calendar server must be up to run unidssearch.

In Business-to-Business (B2B) deployments, unidssearch will produce no output if run on a node that is not part of a calendar domain.

OPTIONS

-f

<LDAPfilter>

Specify a raw LDAP filter to combine ("AND") with the default filter to retrieve users from an LDAP directory. Refer to your directory server documentation for exact attributes that can be specified in the LDAP filter. The values specified in the filter must be in the configured character set of the directory server (e.g. UTF-8, T.61). The default filter is:

```
[&(objectClass=organizationalPerson)(|(!(ctCalXItemId=*))
(!(ctCalXItemId=*:*)))]
```

-c

<#ofDNs>

Limit the number of results returned to this number.

-n

<node-ID>

Required for Business-to-Business (B2B) deployments. Returns only users within the relative DN of the domain identified with the specified node.

-v

Print the current version number of unidssearch.

-h

Print a usage message explaining how to run unidssearch.

FORMATS

OUTPUT FORMAT

The content of the file output by unidssearch has the following format:

```
A did=cn=jdoe, o=Acme, c=US
A did=cn=confroom4, o=Acme, c=US
```

Each entry has an initial "A" character, followed by a "did". The "A" flags the user as one to add to the directory server as a calendar user. The "did" is the Directory ID or Distinguished Name of the user, uniquely identifying that user in the Directory Server.

The format of this file is the same as that required for the input file to the uniuser -ex command. If this is the intended use of the file, additional user data may be appended to the "did", in X.400 format. For example:

```
A did=cn=jdoe, o=Acme, c=US/G=John/OU=Sales
```

EXAMPLES

- Obtain a listing of all directory server users who are not calendar users and redirect the output to a file:

  ```
  % unidssearch > dsonly.txt
  ```

- Obtain a listing of 50 directory server users who are not calendar users:

  ```
  % unidssearch -c 50
  ```

- Obtain a listing of only those directory server users whose surnames begin with "Smith" (the specified filter conforms to the requirements of the directory server being used):

  ```
  % unidssearch -f "(sn=Smith*)"
  ```

WARNINGS

Directory server warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

SEE ALSO

uniuser

UNIDSSYNC

unidssync - Synchronize the information in a calendar server node with that in a directory server.

SYNTAX

unidssync [-n <node-ID>] [-host <hostname>] [-p <SysOpPsw>]

unidssync -v
unidssync -h

DESCRIPTION

unidssync is only used when connected to an external directory server. This utility synchronizes the information in a calendar server node with that in the directory server.

unidssync should be run when other applications using the directory server have changed directory server entries without the knowledge of the calendar server,

AND when the [ENG] dac_itemget parameter in unison.ini is set to "FALSE" to enhance performance (in this case, the calendar server retrieves its information from the internal store rather than from the directory server).

These conditions might allow discrepancies to arise between the information in the internal store of the calendar server node and that in the directory server. unidssync eliminates discrepancies, using the directory server as the authority. It should be run as part of a regular maintenance program.

The calendar server must be up to run unidssync.

OPTIONS

-host

<host>

Specify the host. Required if connecting to a remote host.

-n

<node-ID>

Specify the node. Required if more than one node exists.

-p

<sysOpPsw>

Provide the SYSOP password. If it is not provided on the command line, prompting for it occurs.

-v

Print the current version number of unidssync.

-h

Print a usage message explaining how to run unidssync.

EXAMPLES

- Synchronize the contents of node 10 on host "fergus" with the directory server information for that node:

```
% unidssync -n 10 -host fergus
unidssync: 152 internal calendar directory entries to synchronize
```

```
SYNCHRONIZING <10,234> <S="Okeefe",G=Georgia)<U>
DONE
SYNCHRONIZING <10,235> <S="Whittome",G=Irene)<U>
DONE
SYNCHRONIZING <10,236> <S="Cornell",G=Joseph)<U>
DONE
:
:
SYNCHRONIZING <10,383> <S="Goodwin",G=Betty)<U>
DONE
SYNCHRONIZING <10,384> <S="Dickson",G=Jennifer)
<U> DONE
SYNCHRONIZING <10,385> <S="Wagschal",G=Marian)
<U> DONE
SYNCHRONIZING <10,386> <S="Giacometti",G=Alberto)
<U> DONE
unidssync: 152 internal calendar directory entries synchronized
```

WARNINGS

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIDSUP

unidsup - Report the status of the directory server.

SYNTAX

```
unidsup [-q] [-host <hostname>]

unidsup -v
unidsup -h
```

DESCRIPTION

unidsup reports whether or not the directory server is running.

The calendar server must be up to run unidsup.

OPTIONS

-host

<hostname>

Provide the name of the calendar server host. If this option is not present, the local host is assumed.

-q

Operate in quiet mode (produces no output when the directory server is up).

-v

Print the version number of unidsup.

-h

Print a usage message explaining how to run unidsup.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIENCRYPT

`uniencrypt` - Encrypt a password for inclusion in a calendar server configuration file.

SYNTAX

```
uniencrypt -m <encryption_method> -s <string>

uniencrypt -v
uniencrypt -h
```

DESCRIPTION

`uniencrypt` uses the encryption method specified by the -**m** option to encrypt the password specified by the -**p** option. Any password supplied in a calendar server configuration file (such as those specified by the `[LDAP]` `bindpwd` and `writednpassword` parameters) must first be encrypted using this utility.

`uniencrypt` returns the encrypted password preceded by the encryption method used to generate it. For example, `{acipher}ruyr84jf`. Generally, this entire value, including the encryption method and curly braces, should be enclosed in double quotes and included as the value of the password specified in the calendar server configuration file. For example:

```
[LDAP]
bindpwd = "{acipher}ruyr84jf"
```

`uniencrypt` can be run when the calendar server is up or down.

OPTIONS

-m

<encryption_method>

Specifies the encryption method to use. Accepted values currently include only `acipher`, a proprietary affine cipher encryption method, and `encd`. If this argument is not used, `acipher` will be used by default.

-s

<string>

Specifies the string to encrypt. If this is option is not used, `uniencrypt` will prompt for the string to encrypt.

-v

Print the current version number of `uniencrypt`.

-h

Print a usage message explaining how to run `uniencrypt`.

EXAMPLES

- Encrypt the password "secure" using the default encryption method:

```
% uniencrypt -s secure
{acypher}u0bI1inm
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIGRPLS

`unigrpls` - Display both the public and administrative groups in a calendar server database.

SYNTAX

```
unigrpls [-grp <groupname>] [-members] [-host <hostname>] [-n <node-ID>] [-p
<SysOpPsw>]

unigrpls -v
unigrpls -h
```

DESCRIPTION

`unigrpls` prints both the public and administrative groups in the specified calendar server node. By default, all groups are displayed along with the total number of members in each. The -**members** option is used to display each member in the group.

Note that if a directory server is used, any groups created in the directory server are also included in the output of unigrpls. If members are listed, only the members of the directory server group who are also calendar users are output.

unigrpls can only be run if the calendar server is up.

OPTIONS

-grp
<groupname>

Specify a group.

-members
Print the individual members for each group output.

-host
<hostname>

Specify the host on which the operation is to be performed. The default is the local host.

-n
<node-ID>

Specify the node on which the group is located. Required if more than one node is configured.

-p
<sysOpPsw>

Specify the SYSOP password. Without this option, prompting for the password occurs.

-v
Print the current version number of unigrpls.

-h
Print a usage message explaining how to run unigrpls.

EXAMPLES

- To display all groups in node 20 on the remote host "jupiter":

  ```
  % unigrpls –host jupiter –n 20
  ```

- To display all members of the group "Managers" in node 10 on the local host:

  ```
  % unigrpls –grp "Managers" –members –n 10
  ```

- To show all groups and all members where only one node is configured:

  ```
  % unigrpls –members
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIL2BENDIAN

`unil2bendian` - Convert a calendar server node database from a format for little-endian Windows NT processors to a format for big-endian processors. For more details on this utility, contact Oracle support.

UNIL2LENDIAN

`unil2lendian` - Convert a calendar server node database from a format for little-endian Windows NT processors to a format for little-endian UNIX processors (Red Hat Linux or Tru64 UNIX) and vice-versa. For more details on this utility, contact Oracle support.

UNILOGONS

`unilogons` - Display calendar server SIGNON/SIGNOFF statistics.

SYNTAX

```
unilogons [–s <starttime>] [–e <endtime>] [–i <interval>] [–f <filename>]
```

```
unilogons -t -s <starttime> -e <endtime> -i <interval> [-f <filename>]

unilogons -t [<time>] [-f <filename>]

unilogons -v
unilogons -h
```

DESCRIPTION

`unilogons` displays the signon and signoff activity of users on a calendar server at a specific time or during a specific time period. By default it uses the information in the `/users/unison/log/act.log` file. The -**f** option may be used to specify another input file.

The -**t** option displays activity at a precise moment, while the -**s** and -**e** options display activity during a defined period. The -**i** option specifies a regular time interval (e.g. every 15 minutes) within a specified period.

By default, all activity between the default start-time (the first minute of the current day) and the default end-time (the current system time) is displayed.

The calendar server must be up to run `unilogons`.

OPTIONS

-e

<endtime>

Specify an end time for the statistics. Without this option, the default end time is the current time of the current day. See TIME ARGUMENT FORMAT below for details on how to specify <endtime>.

-f

<filename>

Specify the name of the input file. By default the input file is `/users/unison/log/act.log`. The input file specified with the -**f** option must be in the same format as the `act.log` file.

-i

Specify a time interval. The default interval is <endtime> minus <starttime>. See INTERVAL ARGUMENT FORMAT below for details on how to specify interval.

-s

<starttime>

Specify a start time for the statistics. Without this option, the default start time is the first minute of the current day. See TIME ARGUMENT FORMAT below for details on how to specify <starttime>.

-t

[<time>]

If used without the -**s**, **e**, and -**i** options, this displays statistics for the current time (-**t**) or for a given time (-**t** <time>). When used together with all of the -**s**, -**e**, and -**i** options, the -**t** (without a time argument) restricts output to activity at only the precise times determined by the interval (-**i**) argument. See the last two EXAMPLES for sample output of the -**s**, -**e**, -**i** options both with and without the -**t** option. See TIME ARGUMENT FORMAT below for details on how to specify time.

-v

Print the current version number of `unilogons`.

-h

Print a usage message explaining how to run `unilogons`.

FORMATS

FORMAT OF THE time ARGUMENTS

The <starttime>, <endtime>, and <time> arguments may each be expressed as either:

- `day month [year] [time]` or
- `[month day] time [year]`

where

- `day`

  is a number between 1 and 31;

- `month`

is either the full name of the month or the first three letters of the full name (e.g. jan, feb, mar, etc.) (month is case-insensitive);

- `year`

  must be 1991 or higher and must be specified using four digits; and

- `time`

  is in the form HH:MM or HH:MM:SS (HH is an integer between 0 and 23, MM is an integer between 0 and 59, and SS is an integer between 0 and 59).

The order of the individual elements in the argument is unimportant. What is important is that either day and month be specified, or time be specified. For example, the following are all valid:

```
Feb 22 1996 10:00:00
22 february 10:00:00
10:00:00  february 22 1996
1996 feb 22
feb 22
10:00:00
```

Default values for day, month, year and time are current day, current month, current year and current system time respectively.

Any missing field in time (HH, MM, or SS) is replaced with the current HH, MM, or SS value. Thus, if the current date and time is March 12 1998 10:12:34, and only HH:MM are specified in the argument, the SS becomes "34":

```
-e 12:41 -> March 12 1998 12:41:34
-s 12:41 -> March 12 1998 12:41:34
```

If none of the time fields are specified, starttime defaults to the first minute of the day, and endtime defaults to the last minute of the day:

```
-s feb 22 -> feb 22 1998 00:00:00
-e feb 22 -> feb 22 1998 23:59:59
```

FORMAT OF THE interval ARGUMENT

The interval argument must be an integer greater than zero and be input as minute, hour or day as follows:

```
minutes: 1m, 2m, etc. up to 999999999m  (9 digits)
hours:   1h, 2h, etc. up to 9999999h    (7 digits)
days:    1d, 2d, etc. up to 99999d      (5 digits)
```

EXAMPLES

- Display the current number of logged-on users:

```
unilogons -t
```

- Display the number of users logged-on at 3:00 p.m. on October 6, 1998:

```
unilogons -t oct 6 1998 15:00
```

This would produce the following output:

```
Time 1: Oct 6 1998 15:00:00
-----------------------------------
Client                    Logged-On
Name & Version

unisncd                   2
Windows/32/CorporateTime  1
-----------------------------------
Totals:                   3
```

- Display the number of users logged-on at 3:00 p.m. on October 6, 1998, and at each 15-minute interval, up to 5:00 p.m. on October 6, 1998.

```
unilogons -t -s oct 6 1998 15:00:00 -e oct 6 1998 17:00:00 -i 15m
```

A sample section of the output from this command shows the form of what is output for each of the times 15:00:00, 15:15:00, 15:30:00, etc., up to 17:00:00. (Compare this with the output of the next example, where the -**t** is removed from the command line.)

```
Time 1: Oct 6 1998 15:00:00
-----------------------------------
Client                    Logged-On
Name & Version

unisncd                   2
Windows/32/CorporateTime  1
-----------------------------------
Totals                    3
```

- Output the signon/signoff statistics for a defined period of time (from 3:00 p.m. to 5:00 p.m. on October 6, 1998), providing cumulative statistics for each of the 15-minute intervals in the period. Note how the output from this command line differs from that of the previous example where the -**t** was included.

```
unilogons -s oct 6 1998 15:00:00 -e oct 6 1998 17:00:00 -i 15m
```

For each of the 15-minute time intervals within the entire time period, output similar to the following is displayed:

```
Time Period 1: From Oct 6 1998 15:00:00  Till Oct 6 1998 15:15:00
------------------------------------------------------------------------------
Client                      Logons    Logoffs    Average Time   Median Time
Name & Version                                   Logged-On(hrs) Logged-On(hrs)

Not Available                  0         2          20.71          23.98
unisncd                        2         0           9.83           9.83
Windows/32/CorporateTime       4         4           0.02           0.02
------------------------------------------------------------------------------
Totals                         6         6
```

FILES
`/users/unison/log/act.log`
By default `unilogons` obtains its information from this file. Note that this file is only created if the `[ENG] activity` parameter in `unison.ini` is set to "`TRUE`".

WARNINGS

Input file and processing time
`unilogons` may take some time to finish depending on the size of the input file.

Input file and disk space
The disk space requirement to run `unilogons` is one and a half times the input file. Thus, if the size of the input file is 8 Mb, approximately 12 Mb of free disk space is required to run `unilogons`. `unilogons` creates its temporary files in the `/users/unison/tmp` directory so sufficient free space must exist in that directory.

EXIT STATUS
Exit values are:

0 Success

1 Failure

UNIMVUSER

`unimvuser` - Move a user from one calendar server node to another.

SYNTAX

```
unimvuser -u <user> -host1 <hostname1> -host2 <hostname2> -n1 <node-ID1> -n2
<node-ID2> [-p1 <sysOpPsw1>] [-p2 <sysOpPsw2>] [-up <userPsw>] [-UID <preserve>]
[-verbose]

unimvuser -v
unimvuser -h
```

DESCRIPTION

`unimvuser` moves a user from one calendar server node to another.

Note: `unimvuser` only works if all hosts in the network use a server greater than 4.0. Use `unicpoutu` and `unicpinu` to move users in a node network that includes nodes created by earlier versions of the calendar server.

Note: See the WARNINGS section below before attempting to move a user from a 5.0 or greater node to a 4.0 or earlier node.

The move operation makes the following changes to the user information:

- Any designate rights granted by the moved user are removed.

- Any admin groups created by the user are not moved to the new node.

- Any public groups created by the moved user are made into private groups.

- In installations using an internal directory, the user's password is not retained. See the -**up** option below.

`unimvuser` logs these changes, along with the rest of its activity, in the `/users/unison/log/unimvuser.log` file.

It is important to understand that the move operation may still be in progress even after `unimvuser` has successfully completed. In particular, work is being done by the destination node (the node to which the user has moved) and by remote nodes (where other users reside who may have invited the user). Until the work is complete, the moved user sees an incomplete agenda.

The time required to complete the move operation depends on the number of requests waiting in the request queue of the Corporate-Wide Services

daemon/service. For this reason, it is advisable to run `unimvuser` during off-peak hours for the calendar server.

In addition, the user being moved should not attempt to sign in to the calendar server before `unimvuser` has completed, nor should any other user attempt to work as a designate for the user being moved. Any changes made under these circumstances will be lost.

Never run more than one `unimvuser` operation at the same time. Even if the users involved are on different nodes and you run `unimvuser` on different calendar server hosts, the users may share some meetings or events; this scenario can cause database corruptions.

`unimvuser` can move a user from a node using an external directory server to a node using the calendar server's internal directory, but cannot move a user from a node using the calendar server's internal directory to a node using an external directory server.

Always use the most recent version of `unimvuser`, even when moving a user between nodes on calendar server hosts of earlier versions. For example, if your node network has two calendar server hosts of version 5.1 and one host of version 5.2, you should use the `unimvuser` utility in the `bin` directory of the 5.2 server.

In addition, it is recommended that you always invoke `unimvuser` from the command line rather than through the Windows NT administration tool.

Be aware also that differences in the configurations between the source host and the destination host may cause problems or block the move entirely. For example, if the maximum number of instances of a recurring meeting (`unison.ini` `[ENG]` `maxinstances` parameter) on the source server is set higher than on the destination server, and the user to be moved owns a recurring meeting with more instances than the destination host allows, the move will fail.

Note also that users cannot be specified using only the `UID` key/value pair.

The calendar server must be up to run `unimvuser`.

OPTIONS

-host1
<hostname1>

Specify the host name of the source node.

-host2

<hostname2>

Specify the host name of the destination node.

-n1

<node-ID1>

Specify the source node.

-n2

<node-ID2>

Specify the destination node.

-p1

<sysOpPsw1>

Provide the SYSOP password for the source node. If this option is not used, prompting for the password occurs.

-p2

<sysOpPsw2>

Provide the SYSOP password for the destination node. If this option is not used, prompting for the password occurs.

-u

<user>

Specify the user to be moved. See FORMAT OF THE user ARGUMENT below for details on the proper specification of the user argument. For directory servers, the user must already exist in the directory server used by the destination node.

-UIDpreserve

Preserve original CAPI event UIDs. This option is required if CAPI is used on both the source and the destination node.

-up

<userPsw>

Internal directory only. Specifies a new password for the user. If this option is not used, the user will be able to log into the calendar server without a password. In the case of a directory server, this option has no effect since the password is stored in the directory server and thus remains unchanged.

-verbose

Use verbose mode.

-v

Print the current version number of `unimvuser`.

-h

Print a usage message explaining how to run `unimvuser`.

FORMATS

FORMAT OF THE user ARGUMENT

The user argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. For all keys except the ID key, the "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

> **Note:** If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored. Further note that the ID key-value pair may be specified without using the ID key, i.e. "-u 256" is a valid specification and is equivalent to "-u ID=256".

Table C–13 Accepted keys

Key	X.400 Field
S	Surname
G	Given name
I	Initials
ID	Identifier
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain
P	Private domain
PHONE	Phone number
EXT	Phone extension
FAX	Fax phone number
EMPL-ID	Employee number
JOB-TITLE	Job title

EXAMPLES

■ Move the user with ID 354 from node 12 on host "horus" to node 25 on host "nut":

```
% unimvuser –u "ID=354" –host1 horus –host2 nut –n1 12 –n2 25
```

FILES

`/users/unison/log/unimvuser.log`
`unimvuser` logs its activity in this file.

WARNINGS

Moving a User from a 5.0 or Greater Node to a 4.0 Node

If you move a user from a 5.0 or greater node to a 4.0 node, the user loses the following data:

■ additional meeting information introduced in 5.0

■ additional task information introduced in 5.0

■ folders

■ journal entries

■ sticky notes (in the case of Outlook users)

For this reason, performing such a move is not recommended. `unimvuser` warns you of this data loss and asks for confirmation before proceeding with the move.

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

SEE ALSO

uniuser

UNINODE

uninode - Administer a calendar server node network.

SYNTAX

uninode -add [-nologinfo] <hostname>

uninode -apply [-y | -n] [-nologinfo] [<SysOpPsw>]

uninode -cws [-nologinfo] [<node-ID> | <hostname> | <group>]

uninode -edit [-e <editor>] [<SysOpPsw>]

uninode -import [-nologinfo] [<SysOpPsw>]

uninode -init [-nologinfo] [<SysOpPsw>]

uninode -reset [-nologinfo] [<node-ID> | <hostname> | <group>] [<SysOpPsw>]

uninode -retry [-nologinfo] [<node-ID> | <hostname> | <group>] [<SysOpPsw>]

uninode -snc [-nologinfo] [<node-ID> | <hostname> | <group>]

uninode -test <node-ID> | <hostname> | <group>

uninode -v
uninode -h

DESCRIPTION

uninode is a centralized tool for setting up and administering a calendar server
node network. See EXAMPLES for an example of setting up a node network. Use
uninode to add and remove nodes from the node network, as well as to add and
remove connections between nodes. Also use uninode to make queries about the
node network configuration and about the status of remote connections.

uninode uses the node network configuration information in the nodes.ini file to configure the node network. Only one nodes.ini file should exist for a node network, regardless of how many calendar servers are linked. Furthermore, you manage the calendar server node network, that is you run uninode, from the machine on which this file exists. When your node network includes coexistence of multiple calendar server versions, always make sure that the host of the nodes.ini file is of the most recent version.

The <sysOpPsw> is the SYSOP password for the node in the calendar server network with the lowest node-ID on the machine hosting the nodes.ini file.

<node-ID>, <hostname> and <group> each restrict uninode's actions to certain nodes in the nodes.ini file. <node-ID> restricts uninode to the specified node, <hostname> to the nodes on the specified host, and <group> to the nodes in the specified group. <group> may be one of the following:

- all

  all included (+) and all excluded (-) nodes

- included

  all included (+) nodes

- excluded

  all excluded (-) nodes

<group> may also be a customized group name. Consult your calendar server *Administrator's Guide* for further details on the meaning of each of these values. If none of these values are specified, uninode will assume the value all.

If you are using a directory server, you may want to run unidssync on each node before running uninode to ensure that the local information in each node is synchronized with what is in the directory server. Note that all nodes in a calendar server node network must use the same directory server.

uninode only runs if the calendar server is up.

uninode is not used with Business-to-Business (B2B) deployments.

OPTIONS

-add

Add all nodes found on the specified host to the `nodes.ini` file. This option first determines which nodes exist on the specified host. It then removes all lines for that host in the `nodes.ini` file, and finally adds a line for each node found on the host. Nodes are added as excluded nodes. You must edit the `nodes.ini` file to include them in the network.

-apply

Apply the configuration in the `nodes.ini` file. The <node-ID>, <group>, and <hostname> arguments restrict the application to specified nodes in the `nodes.ini` file.

`uninode` first verifies that:

- the syntax of the `nodes.ini` file is correct
- the specified host name or specified node-ID is valid
- the `uniengd` and `unisncd` servers are up
- the version of `uniengd` is greater than A.01.15
- the SNC daemon is running
- the `nodes.ini` file exists only on the host currently running `uninode`
- all nodes in the node network are available

If any of these verifications fails, `uninode` terminates.

Otherwise, it proceeds to check the remote node information in each of the nodes involved, and if it finds there are entries missing, it prompts the user to confirm the addition of the missing entries. Use the **-y** or **-n** option to automatically provide a response. Note that `uninode` does not delete any surplus entries from any of the nodes.

-cws

Print the following information for each connection between two nodes. This includes information from the CWS daemon/service.

- EX

  The number of TCP/IP connections, between the two nodes, configured in the `nodes.ini` file.

- CO

  The actual number of TCP/IP connections between the two nodes.

- Q-SIZE

  The number of CWS requests currently in the CWS queue.

- IN-PROCESS

  The number of CWS requests processed.

- IMPORT-DIR

  The number of items (users and resources) in the local copy of the remote directory.

-edit

<editor>

Safely edit a COPY of the `nodes.ini` file using the specified text editor. `uninode` first performs the verifications described in the -**apply** option and terminates if any of the verifications fails. If all verifications succeed, it invokes the editor. On exit from the editor `uninode` parses the edited file, and, if it does not find any errors, updates the original `nodes.ini` file. If `uninode` finds errors in the edited file, it prompts the user to either re-edit the file or abort the operation.

-import

Same as -**apply** with the -**y** option.

-init

Construct a `nodes.ini` file from the currently running node network configuration. The node with the lowest node-ID on the machine hosting the `nodes.ini` file is the one from which `uninode` begins construction of the file. If a `nodes.ini` file already exists, `uninode` prompts for confirmation to overwrite it.

-n

Prevent correction of any directory inconsistency when you use the -**apply** option.

-nologinfo

Do not write to the log file. By default, `uninode` logs any errors, as well as any output it sends to the screen, to the `uninode.log` file.

-reset

Reset the statistics of a Synchronous Network Connection (SNC) daemon. It is recommended that you reset all nodes at the same time by running `uninode`

`-reset all`. Resetting the statistics allows the administrator to compare the statistics for different nodes at a later time.

-retry

Restart the retry mechanism of an SNC daemon. When there are fewer connections available than are configured, the SNC daemon attempts to acquire new connections at specific time intervals. It retries at intervals of 1, 2, 4, 8, 16, 32, and finally every 64 minutes. This option resets the interval to 1 minute. One use of this option might be to run `uninode -retry all` after a network-related problem is solved.

-snc

Print the following information on the TCP/IP connections for the specified node, or for each node in the specified group or on the specified host.

- EX

  The number of TCP/IP connections to the node configured, as per the information in the `nodes.ini` file.

- CO

  The actual number of TCP/IP connections to the node.

- AV

  The number of connections to the node currently available.

- US

  The number of connections to the node currently in use.

- LOST

  The number of times the SNC daemon lost a connection to the node.

- RETRY

  The time (expressed in the format <mm>:<ss> format) before the next attempt to reconnect a lost connection.

- QUEUE

  The number of requests currently in the queue.

- CANCEL

  The number of cancelled requests.

- CHECK

  The number of checks for queued requests. Checks are performed when a connection is waiting in the queue.

- GRANTED

  The number of requests for connections the SNC daemon/service granted since it started.

-test

Verify that it is possible to connect to a node or group of nodes. See the -**apply** option for a list of the items `uninode -test` verifies.

-y

Auto-confirm the correction of any directory inconsistency when you use the -**apply** option.

-v

Print the current version number of `uninode`.

-h

Print a usage message explaining how to run `uninode`.

EXAMPLES

CREATE A NODE NETWORK

You have a company with offices in three different countries. Each office runs its own calendar server. You want to set up a node network and manage it from the calendar server running on "gravlax" in Sweden.

1. Log on to "gravlax" and create a `nodes.ini` file.

   ```
   % uninode -init
   ```

   Since no node network currently exists, `uninode` creates an empty `nodes.ini` file with sample lines included as comments.

2. Add the nodes from each of the three calendar servers.

   ```
   % uninode -add gravlax
   % uninode -add gnocchi
   % uninode -add biryani
   ```

3. Examine the contents of the `nodes.ini` file.

```
% cat nodes.ini
- H=biryani/N=32
- H=biryani/N=31
- H=gnocchi/N=25
- H=gnocchi/N=24
- H=gnocchi/N=23
- H=gnocchi/N=22
- H=gnocchi/N=21
- H=gravlax/N=13
- H=gravlax/N=12
- H=gravlax/N=11
```

4. Edit the file to configure the node network.

```
% vi /users/unison/misc/nodes.ini
```

The `nodes.ini` file now contains the following lines.

```
% cat /users/unison/misc/nodes.ini
+ H=biryani/N=32/ALIAS=salesIndia/GR=india
+ H=biryani/N=31/ALIAS=adminIndia/GR=india
- H=gnocchi/N=26/ALIAS=tempItaly/GR=italy
+ H=gnocchi/N=25/ALIAS=supportItaly/GR=italy
+ H=gnocchi/N=24/ALIAS=financeItaly/GR=italy
+ H=gnocchi/N=23/ALIAS=r&dItaly/GR=italy
+ H=gnocchi/N=22/ALIAS=salesItaly/GR=italy
+ H=gnocchi/N=21/ALIAS=adminItaly/GR=italy
- H=gravlax/N=16/ALIAS=tempSweden/GR=sweden
+ H=gravlax/N=13/ALIAS=r&dSweden/GR=sweden
+ H=gravlax/N=12/ALIAS=salesSweden/GR=sweden
+ H=gravlax/N=11/ALIAS=adminSweden/GR=sweden
included:2
india:+2
italy:+3
sweden:+2
```

The node network has the following characteristics:

- It has ten nodes.

- There are two excluded nodes (nodes 16 and 26).

- There are two connections going from each node in the node network to every other node in the node network. For example, two connections go from node 32

to node 13, and two go from node 13 to node 32. A single connection is unidirectional.

- The "india" group of nodes has two additional connections going from each node in the "india" group to each of the other nodes in the "india" group. Similarly, the "italy" group of nodes has three additional connections between each of the nodes in the "italy" group, and the "sweden" group has two additional connections between each of the nodes in the "sweden" group.

In this configuration, the total number of connections from node 13 is 22 (two to each of the other nine nodes in the network gives 18, plus two to each of the other two included nodes in the "sweden" group gives 4).

Consult your calendar server's *Administrator's Guide* for rules on configuring connections between nodes.

Next, apply the configuration. Since this is the first time that nodes "see" other nodes, you expect inconsistencies in their remote node directories. For this reason you use the -y option.

```
% uninode -apply -y
```

During execution of this command, `uninode` prints out information on the work it is performing. For example:

```
Processing node 11
connected to gravlax, node 11
connected to gravlax, node 12
added 11->12, TCP/IP connection
placed a request in the CWS queue to get node 12 user directory
```

FILES

`/users/unison/misc/nodes.ini`
Contains the list of nodes and the rules that describe the calendar server's node network configuration.

`/users/unison/log/uninode.log`
By default, `uninode` logs any errors, as well as any output it sends to the screen, to this file.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIPASSWD

`unipasswd` - Change a calendar server SYSOP or CWSOP password.

SYNTAX

`unipasswd [-sysop | -cwsop] [-n <node-ID>] [-host <hostname>] [-p <sysOpPsw>]`

`unipasswd -v`
`unipasswd -h`

DESCRIPTION

`unipasswd` changes the password of the SYSOP or CWSOP of a given node. Note that the -**sysop** and -**cwsop** options are all mutually exclusive.

The maximum length of a password is 15 characters.

`unipasswd` only runs if the calendar server is up.

If you use a Business-to-Business (B2B) deployment, always omit the -**n** option when changing SYSOP and CWSOP passwords. By default, `unipasswd` will change these passwords for all nodes. See WARNINGS.

OPTIONS

-cwsop
Change the password of the CWSOP.

-host
<hostname>

Specify the host on which the operation is to be performed. The default is the local host.

-n
<node-ID>

Specify the node on which the password is to be changed. Required if more than one node exists.

-p

<sysOpPsw>

Provide the current SYSOP password.

-sysop

Change the password of the SYSOP.

-v

Print the current version number of unipasswd.

-h

Print a usage message explaining how to run unipasswd.

EXAMPLES

■   Change the SYSOP password on node 20 on the remote host "jupiter":

```
% unipasswd -host jupiter -n 20
```

■   Change the local CWSOP password where only one node is configured:

```
% unipasswd -cwsop
```

WARNINGS

Business-to-Business (B2B) deployments and the Calendar Administrator

Do not change the SYSOP and CWSOP passwords for individual nodes. Doing so may cause the Calendar Administrator web interface to fail. In addition, after changing the SYSOP password, make sure that you update the [ADMINSETTINGS] serverpassword and serveroldpassword parameters in uniwebadm.ini. Failure to do so may cause an interruption in Calendar Administrator services.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIPING

`uniping` - Ping another calendar server.

SYNTAX

```
uniping [-host <hostname>] [-n <node-ID>] [-u <user>] [-p <password>] [-domain]
[-i <sec>] [-allnodes] [-s <size>] [-stats] [-log] [-time]

uniping -v
uniping -h
```

DESCRIPTION

`uniping` sends echo messages to a node or a node network. Receiving nodes reply to the message, and `uniping` prints the elapsed time between sending the original message and receiving the replies. Use this utility to verify that a node is up, or to measure server response time under various load conditions.

Before sending any messages, `uniping` first authenticates the specified user on the specified node. `uniping` only sends messages if this authentication is successful.

`uniping` runs whether the calendar server is up or down.

OPTIONS

-allnodes
Send the echo message to all nodes connected to the node network containing the specified node.

-domain
Business-to-Business (B2B) deployments only. Used with the -**host** option. Checks whether the specified host is running Calendar Domain Services.

-host

<hostname>

Specify the name of a calendar server host. If this option is not used, the local host is assumed.

-i

<sec>

Specify an interval in seconds after which `uniping` repeats its echo message. If this option is not used, `uniping` sends one echo message to each specified node.

-log

Print errors to a log file (`/users/unison/log/uniping.log`).

-n

<node-ID>

Specify a node to connect to. Required if more than one node exists on the calendar server specified by the -**host** option.

-p

<password>

Provide the password for the user specified by the -**u** option. If you do not use the -**u** option, use -**p** to provide the SYSOP password. If you do not use the -**p** option, `uniping` will prompt you for the password.

-s

<size>

Specify the size of the echo message in bytes. The default is 64 bytes.

-stats

Display statistics on startup.

-time

Display the time at which each message is sent.

-u

<user>

Specify a user name to use for authentication. If this option is not used, SYSOP is used by default. Please note: this user name must be valid on the node specified by the -**n** option. Use the -**p** option to specify this user's password. See "FORMAT OF THE user ARGUMENT" below.

-v

Print the version number of `uniping`.

-h

Print a usage message explaining how to run `uniping`.

FORMATS

FORMAT OF THE user ARGUMENT

The user argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. For all keys except the ID key, the "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

> **Note:** If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored. Further note that the ID key-value pair may be specified without using the ID key, i.e. "-u 256" is a valid specification and is equivalent to "-u ID=256".

Table C–14   Accepted keys

Key	X.400 Field
S	Surname
G	Given name
I	Initials
ID	Identifier
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain
P	Private domain
PHONE	Phone number
EXT	Phone extension
FAX	Fax phone number
EMPL-ID	Employee number
JOB-TITLE	Job title

EXAMPLES

- Ping node 14 of a calendar server on the host "Scribe", using the SYSOP user account:

```
% uniping -host scribe -n 14
Enter password:

scribe,14: 40 ms.
```

- Ping all nodes in the node network containing node 60, using the user "Dashiell Hammett", and displaying the time of each sent message:

```
% uniping -n 60 -u "S=Hammett/G=Dashiell" -allnodes -time
Enter password:

Fri Jul 07 10:23:41 2000: scribe,14: 40 ms.
Fri Jul 07 10:23:41 2000: scribe,60: 0 ms.
Fri Jul 07 10:23:41 2000: scribe,66: 114 ms.
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

UNIREQDUMP

`unireqdump` - View, and optionally delete, requests in the queue of the Corporate-Wide Services (CWS) daemon.

SYNTAX

```
unireqdump [-delete] [-excl <filter>] [-incl <filter>] [-item <itemnum>]
[-remotenode <node-ID>] [-prompt] [-reqid <ID>] [-trcode <code>] [-n <node-ID>]
[-p <sysOpPsw>]

unireqdump -v
unireqdump -h
```

DESCRIPTION

`unireqdump` outputs the set of requests currently in the queue of the Corporate-Wide Services daemon/service, `unicwsd`. The utility is also used to delete requests from the queue (using the -**delete** option).

By default, all requests in the queue are output. The -**excl**, -**incl**, -**item**, -**remotenode**, -**reqid**, and -**trcode** options allow you to select requests satisfying specific criteria. These options are applied successively so each of the requests in the output must meet the combined criteria for all of the options specified.

Numeric arguments can be either decimal or hexadecimal values (where hexadecimal values are prefixed by "`0x`"). The single exception is the ID argument to the -**reqid** option, where a hexadecimal value is always assumed, even if the "`0x`" prefix is not present.

`unireqdump` can only be run if the calendar server is up.

OPTIONS

-delete

Enable the deletion option. After each request is output, the user is prompted to confirm whether or not they wish to delete it. The -**prompt** option may be used along with this option to tell `unireqdump` to automatically delete ALL of the requests in the output, without prompting for confirmation.

-excl

<filter>

Set an exclusion filter. Requests matching this filter are excluded from the output. The possible filters are:

- *serviced:* Requests already serviced

- *notserviced:* Requests not yet serviced

- *unserviced:* Synonym for "notserviced"

- *cantservice:* Requests that cannot be serviced

- *suspended:* Requests that have been queued pending reactivation of an item's SMS notifications

-incl

<filter>

Set an inclusion filter. Requests matching this filter are included in the output. The possible filters are listed under the -**excl** option.

-item

<itemnum>

Select only requests matching the specified item number. itemnum is the numeric ID of the item.

-n

<node-ID>

Specify the node from which the request originated. Required if more than one node exists on the server running unireqdump. The SYSOP password unireqdump requires is the one for this node.

-p

<sysOpPsw>

Provide the SYSOP password. If the password is not supplied on the command line, prompting for it occurs.

-prompt

Used with the -**delete** option to tell unireqdump to automatically delete all of the requests in the output, without prompting for confirmation. Use this option with care!

-remotenode

<node-ID>

Select only requests destined for the specified remote node.

-reqid

<ID>

Select the request with the specified ID. ID is a hexadecimal value (it is not necessary to prefix the value with "0x", though doing so causes no harm).

-trcode

<code>

Select requests with the specified transaction code. The transaction code may be expressed numerically (the numeric values are available in the documentation for the calendar programming interface), or as one of the following strings:

```
agendaget
attendadd
echo
eventattend
eventcreated
eventdeleted
eventmodified
foreignerdeleted
instanceadded
instancemodified
itemdeleted
itemmodified
mailmessagepost
nodeitemsget
notifynewevent
notifynewinstance
securityadd
securitydeleted
securitymodified
```

-v

Print the current version number of `unireqdump`.

-h

Print a usage message explaining how to run `unireqdump`.

EXAMPLES

- Select all requests in the queue which originate in node 10:

  ```
  unireqdump -n 10 -p sysOpPsw
  ```

- Output all requests in the queue which originate in node 10, and interactively prompt for confirmation to delete each one:

  ```
  unireqdump -delete -n 10 -p sysOpPsw
  ```

- Output all requests in the queue except those already serviced (a single node exists on this server so the -**n** option is unnecessary):

```
unireqdump –excl serviced –p sysOpPsw
```

- Output all unserviced requests with the "eventattend" transaction code, orginating in node 10 and destined for the remote node 20, and interactively prompt for confirmation to delete each one:

```
unireqdump –delete –excl serviced –remotenode 20 –trcode eventattend –n 10
–p sysOpPsw
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIRES

`unires` - List, add, or delete calendar resources, or modify the information associated with them.

SYNTAX

Listings

```
unires –ls [<res>] [–format <format>]  [–host <hostname>] –n <node-ID> [–p
<sysOpPsw>]
```

```
unires –format <format>  [–host <hostname>] –n <node-ID> [–p <sysOpPsw>]
```

Addition

```
unires –add <res> [–s <sections>]  [–host <hostname>] –n <node-ID> [–p
<sysOpPsw>]
```

Deletion

```
unires –del <res> [–y]  [–host <hostname>] –n <node-ID> [–p <sysOpPsw>]
```

```
unires –desdel <res>  [–host <hostname>] –n <node-ID> [–p <sysOpPsw>]

unires –grpdel <res>  [–host <hostname>] –n <node-ID> [–p <sysOpPsw>]
```

Modification

```
unires –mod <res> –s <sections> | –m <modifier>  [–host <hostname>] –n <node-ID>
[–p <sysOpPsw>]
```

Multiple additions, deletions, modifications

```
unires –ex <filename> [–s <sections>] [–k]  [–host <hostname>] –n <node-ID> [–p
<sysOpPsw>]
```

Other

```
unires –s [<sections>] –n <node-ID> [–p <sysOpPsw>]

unires –info

unires –v
unires –h
```

DESCRIPTION

unires is used to list, add, or delete calendar resources, and to modify the
information associated with them. Resources are identified by their names so each
must be unique. Details on how to specify the *res* argument are given in the
FORMAT OF THE res ARGUMENT section below. Note that the -**ls**, -**add**, -**del**,
-**grpdel**, -**desdel**, and -**mod** options are all mutually exclusive.

The information associated with a resource is a combination of the key-value pairs
described in the FORMAT OF THE res ARGUMENT below and that contained in
the resource.ini file (which includes the resource preferences, security, personal
group, admin group membership, and the list of users permitted to work as a
designate for the resource).

uniuser is the complementary utility for users.

unires can only be run if the calendar server is up.

OPTIONS

-add

Create a new resource. If the resource already exists, an error is reported. The information associated with the new resource is a combination of what is specified in the *res* argument, and the default values in the resource.ini file. By default, when unires reads the resource.ini file, it considers only the default values in the [GEN] section. Use the -**s** option to apply default values from other sections of the resource.ini file. Use the -**ex** option to create multiple resources.

When using the calendar server's internal directory, the "R" key is mandatory for the -**add** option.

For external directory server contexts, the DID (Directory ID) should be specified in the *res* argument. If no DID is specified, unires will generate one from the "R=" value. See EXAMPLES.

-del

<res>

Delete the specified resource. unires prompts for confirmation before performing the deletion unless the -**y** option is used. To delete more than one resource, use the -**ex** option.

-desdel

<res>

Delete the list of designates for the specified resource. If more than one resource matches <res>, the command fails.

-edit

This option exists only under UNIX. It allows you to first output the list of existing calendar resources to a file, then edit the file to make desired modifications, and finally input the changes back into the node. The following sequence of commands is performed:

```
% unires –ls –n node-ID > file
% vi file
% unires –ex file –n node-ID
% rm file
```

-ex

Perform the additions, deletions and/or modifications specified in *filename*. Each line of the file must begin with one of the characters '.', '#', 'A', 'a', 'D', 'd', 'M', 'm', 'S', 's', or '+'. This initial character specifies the action to take, as follows:

Character	Action
'.' or '#'	ignore the line
'A' or 'a'	add the resource
'D' or 'd'	delete the resource
'M' or 'm'	modify the resource
'S' or 's'	update the resource with the settings from the resource.ini file
'+'	treat the line as the continuation of the previous line; note that key-value pairs cannot break over lines

The initial character must be followed by a space and a resource specification. In the case of a modification, the ID for the resource must be supplied; it is used to identify the resource, while the other key-value pairs specified along with it are applied as the modifications. See EXAMPLES.

For each deletion specified in the file, unires prompts for confirmation before performing the deletion. The -**y** option may be used to automatically provide confirmation.

One way to create this file is to save the output of unires -ls to a file. This can then be edited and input to unires -ex.

The -**s** sections option may be used with -**ex** to set the corresponding resource information to the default values contained in the specified sections of the resource.ini file.

For directory servers, the DID can be followed by data in X.400 format. Note that the ability to modify the name of a resource (the "R" key) depends on the directory server.

-format
<format>

This option is used to select resource information fields and customize the format of the output. The -**info** option lists the parameters that can be used to specify the customized format. These are also listed in the FORMAT OF THE res ARGUMENT

section below. If this option is not used, all resource information fields are output, and a default presentation is used. See EXAMPLES.

-grpdel

<res>

Delete the specified resource from all admin groups. A single resource must match <res> or the command fails.

-host

<hostname>

Specifies a host on which to look for the node specified by the -**n** option. Required for remote hosts. If -host is not present, unires will assume the local host. For modification or deletion, if -**host** is specified and -**n** is not, unires will search for a master node on the specified host. If a master node is found, unires will use it to locate the selected users.

-info

List the format parameters used with the -**format** option.

-k

Used with the -**ex** option to force unires to process all lines in the file even if it encounters an error. Errors are sent to standard error; use file redirection to capture these to a file.

-ls

[<res>]

If a resource is specified, -**ls** lists that resource (in the specified node). Each output line contains a period ('.'), a space, the resource description, and the ID of the resource. If no resource is specified, all resources in the node are listed. The -**format** option can be used with -**ls** to configure the presentation of the output (see EXAMPLES).

-m

<modifier>

Specify a modification to make to the resource information. The modifier is a string of the same form as the <res> argument with the following exceptions: the ID key may NOT be specified, and the PSW key MAY be specified.

-mod

<res>

Modify the information associated with the specified resource. This option is used with either the -**s** sections or the -**m** modifier option. When used with the -**s** option, it sets the corresponding information for the resource to the values contained in the specified sections of the resource.ini file. When used with the -**m** modifier option, modifications are made to the key-value pairs listed in the FORMAT OF THE res ARGUMENT. In the case of a directory server, the ability to modify the name of a resource (the R key) may depend on the directory server.

-n

<node-ID>

Specify a node.

-p

<sysOpPsw>

Provide the SYSOP password for the specified node. If this option is not used, prompting for the password occurs.

-s

[<sections>]

Evaluate all or some of the contents of the resource.ini file and output information from it to standard output.

Use this option without the <sections> argument to determine what sections exist in the resource.ini file. The output in this case is a list of all section names along with their respective "Info" key-value pairs.

Use the <sections> argument to evaluate all of the key-value pairs in the specified sections. In this case all key-value pairs in the specified sections are output, along with any errors detected in these pairs. This is one way to validate the contents of the resource.ini file. See the -**add** and -**mod** options for information on using -**s** <sections> to apply values from resource.ini.

The <sections> argument is a list of one or more section names, each separated by a forward slash (e.g. "GEN/GR1/GR2" specifies the sections GEN, GR1 and GR2). Evaluation is done from left to right. Thus, in the above example, GEN is evaluated first, GR1 second, and GR2 last. Where the same key appears in more than one section, the value of the last instance evaluated takes precedence.

-y

Used with the -**del** and -**ex** options to auto-confirm the deletion(s).

-v

Print the current version number of `unires`.

-h

Print a usage message explaining how to run `unires`.

FORMATS

FORMAT OF THE res ARGUMENT

The res is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. For all keys except the ID key, the "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/G=James\/Jim"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

Note that if the ID key-value pair is specified in the *res* argument, all other key-value pairs specified along with it are ignored. Also note that the PSW key may only be specified in the *res* argument for the -**add** option.

The format parameters listed in the third column below are used with the -**format** option to configure the presentation of a listing (see EXAMPLES).

Table C–15   Accepted keys

Key	Field	Format parameter
R	Resource name	%R
N	Resource number	%N
CA	Capacity	%CA

Table C–15   Accepted keys

Key	Field	Format parameter
S	Contact's surname	%S
G	Contact's given name	%G
ID	Identifier	%ID
LOC	Location	%LOC
PHONE	Phone number	%PHONE
EXT	Phone extension	%EXT
FAX	Fax phone number	%FAX
PSW	Password	None
	(maximum of 15 characters; -**add** only)	

EXAMPLES

LISTINGS

- List all resources in node 12 with "HPLaser" in their name:

  ```
  % unires -ls "R=HPLaser*" -n 12
  Enter SysOp password:

  . R=HPLASER dorian/S=Wilde/G=Oscar/ID=438
  . R=HPLASER sula/S=Morrison/G=Toni/ID=512
  ```

- List all resources in node 12 with "HPLaser" in their name; output the resource name, the name and surname of the contact person, and separate each field by a colon:

  ```
  % unires -ls "R=HPLaser*" -format "%r Contact: %g %s" -n 12
  Enter SysOp password:

  HPLASER dorian    Contact: Oscar Wilde
  HPLASER sula    Contact: Toni Morrison
  ```

- List all resources in node 12; output the resource name, the name and surname of the contact person, and separate each field by a colon:

  ```
  % unires -format "%r: %g %s" -n 12
  ```

```
Enter SysOp password:

HPLASER dorian: Oscar Wilde
HPLASER sula: Toni Morrison
RedRoom: Jeannette Winterson
BlueRoom: James Thurber
Apple Laptop: Audre Lorde
SGI Indy: Milan Kundera
Olivetti Typewriter: Dashiell Hammett
```

ADDITION

■ Add the oak-panelled conference room to node 12:

```
% unires –add "R=oakroom" –n 12
```

■ Perform the same addition for a directory server:

```
% unires –add "DID=cn=oakroom,o=Acme,c=US" –n 12
```

DELETION

■ Delete the resource "Olivetti Typewriter" from node 12:

```
% unires –del "R=Olivetti Typewriter" –n 12
```

MODIFICATION

■ Modify the capacity of the oak-panelled conference room in node 12 to "15":

```
% unires –mod "R=oakroom" –m "CA=15" –n 12
```

MULTIPLE ADDITIONS, DELETIONS, MODIFICATIONS

Multiple additions, deletions, modifications are done using the -**ex** option. In this example two new resources are added (a BetaCam and a VCR; the contact for the BetaCam is specified at the same time as the add is being done), the capacity of an existing resource is modified, and an existing resource is deleted. A directory server is being used.

1. Create the file:

```
% vi multiple.dat
% cat multiple.dat
A DID=cn=betacam,o=Acme,c=US/S=Fellini/G=Fred
A DID=cn=vcr1,o=Acme,c=US
M R=RedRoom/CA=25/ID=441
D R=BlueRoom
```

Note that in the case of the modification, the resource is the one which has ID=441; the capacity of the resource is modified to "25".

2. Input the file to `unires`:

```
% unires -ex multiple.dat -n 12
```

3. Enter SYSOP password:

```
unires: added "cn=betacam,o=Acme,c=US"
unires: added "cn=vcr1,o=Acme,c=US"
unires: modified "RedRoom"
unires: deleted "BlueRoom"
```

FILES

`unires.ini`
This file specifies possible calendar resource configurations. See also the calendar server Reference Manual, Appendix A, "User and Resource Parameters".

WARNINGS

Directory Server Warning
It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

Modification of Key-Value Pairs
This utility uses the [UTL] `adm_modres<field>` parameters in `unison.ini` to determine whether or not it can modify the corresponding resource key-value pairs. For example, if `adm_modrescapacity` is set to "FALSE", then the value associated with the CA key cannot be modified by this utility.

If no corresponding `adm_modres<field>` for a particular key appears in the file, the utility assumes it CAN modify it.

Deleting resources with large agendas
Deleting resources with a large numbers of meetings and events can take a long time and cause a decrease in performance for other calendar users. It is

recommended that you delete such resources outside of normal hours, or at least not at times of peak calendar usage.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

SEE ALSO

`uniuser` is the complementary utility for users

UNIRMOLD

`unirmold` - Remove old events and tasks from agendas in a calendar server database.

SYNTAX

```
unirmold [<user>] [-event | -task] [-n <numOfDays>] [-y] <node-ID> [-p
<sysOpPsw>] [-include <types>] [-sync]

unirmold -res [<resource>] [-n <numOfDays>] [-y] <node-ID> [-p <sysOpPsw>]

unirmold -v
unirmold -h
```

DESCRIPTION

`unirmold` removes events and/or tasks older than a specified number of days from agendas in a calendar server database.

To remove an event from an agenda, `unirmold` "uninvites" the owner of the agenda from the event. This has two consequences: the event no longer appears in that agenda AND the owner of that agenda no longer appears on the list of invitees to the event. The update to the list of invitees propagates as necessary to the other nodes in the node network.

By default, `unirmold` removes all events and tasks older than 90 days from all user agendas in the node and all events older than 90 days from all resource agendas in the specified node. The -**res** option restricts `unirmold` to events in resource agendas. The <user> argument restricts `unirmold` to the agendas of the specified users. See FORMAT OF THE <user> ARGUMENT below for details on how to specify <user>.

When using `unirmold` in -**res** mode, you may specify a resource filter using the <resource> argument. See FORMAT OF THE <resource> ARGUMENT below for details on how to specify <resource>.

> **Note:** `unirmold` only removes tasks if the start date, the due date, and the completion date are all older than the specified number of days.

`unirmold` only runs if the calendar server is up.

OPTIONS

-event

Delete only old events. By default `unirmold` deletes both events and tasks from the user agenda.

-include

<types>

Delete events which are special types of events. Currently this option applies only to events which are either Outlook journal entries or sticky notes. The <types> argument is one or more of the following: `journal`, `sticky`. For example, to delete both events which are journal entries and events which are sticky notes, use `-include journal sticky`. To delete only events which are sticky notes, use `-include sticky`.

-n

<numOfDays>

Delete events and tasks that are more than <numOfDays> days old from the agenda. If you do not use this option, the default value is 90 days. The minimum value is 30 days.

-p

<sysOpPsw>

Provide the SYSOP password for the node. If you do not use this option, `unirmold` prompts for the password.

-res

<resource>

Remove all events in resource agendas only. You may specify a filter to select specific resources by providing the <resource> argument. See FORMAT OF THE <resource> ARGUMENT below for details.

-sync

Removes all sync records from the user agenda.

-task

Delete only old tasks from the user agenda. If this option is not used, both events and tasks are deleted.

-y

Eliminate prompting for confirmation of any deletions.

-v

Print the current version number of `unirmold`.

-h

Print a usage message explaining how to run `unirmold`.

FORMATS

FORMAT OF THE <user> ARGUMENT

The user argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. The "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

> **Note:** If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored.

Table C–16   Accepted keys

Key	X.400 Field
S	Surname
G	Given name
I	Initials
ID	Identifier
X	Generation
OU1	Organizational Unit 1
OU2	Organizational Unit 2
OU3	Organizational Unit 3
OU4	Organizational Unit 4
O	Organization
C	Country
A	Administration domain
P	Private domain

FORMAT OF THE <resource> ARGUMENT

The <resource> is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. For all keys except the ID key, the "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/G=James\/Jim"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

Note that if the ID key-value pair is specified in the <resource> argument, all other key-value pairs specified along with it are ignored.

Table C–17   Accepted keys

Key	Field	Format parameter
R	Resource name	%R
N	Resource number	%N
ID	Identifier	%ID

EXAMPLES

- Remove all events and tasks from the node network that are owned by users in node 10, and all events that are owned by resources in node 10:

  ```
  % unirmold 10
  ```

- Remove all events in the node network that are more than 30 days old and are owned by users in node 10 with the surname "Wembley":

  ```
  % unirmold "s=wembley" –event –n 30 10
  ```

- Remove all events in the node noetwork that are more than 30 days old and owned by any resource in node 10.

  ```
  % unirmold –res –n 30 10
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIRNDEL

`unirndel` - Delete a remote node from a local calendar server node database.

SYNTAX

`unirndel remoteNoteId [<sysOpPsw>] [-n <node-ID>]`

`unirndel -v`
`unirndel -h`

DESCRIPTION

`unirndel` deletes all references to a remote node from the database of a local node. By default the local node is the one with the name "N1". `unirndel` should only be used to delete a remote node created for test purposes. You should consult Oracle Support before using `unirndel`.

It is recommended that you back up the local `/users/unison/db` directory before running `unirndel`.

`unirndel` runs only if the calendar server is up.

OPTIONS

-n

<node-ID>

Specifies the node-ID of the local node database from which the remote node should be deleted.

-v

Print the current version number of `unirndel`.

-h

Print a usage message explaining how to run `unirndel`.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIRNSYNCH

unirnsynch - Propagate deletions in the local information of one node to another node in the network.

SYNTAX

```
unirnsynch -rn <node-ID> [-rhost <hostname>] [-rp <remote-sysop-password>] -n
<node-ID> [-host <hostname>] [-p <sysop-password>]

unirnsynch -v
unirnsynch -h
```

DESCRIPTION

unirnsynch is used to propagate deletions in the local information of one node to another node in the network. Each node in a node network contains both local information and remote node information, where:

- **local information** is a list of the users and resources belonging to that node

- **remote node information** is a list of the users and resources belonging to each of the other nodes in the node network.

The remote node information of a given node is constructed from the local information of each of the other nodes in the node network.

Changes to the local information of a node are normally automatically propagated to the remote node information of all other nodes in the network. However, if for any reason discrepancies do occur, the remote node information can be updated using unirnsynch and/or uninode. uninode (using the -**apply** option) may be used to add missing entries while unirnsynch is used to delete entries which no longer exist in the local information.

uninode -cws all may be used to determine whether or not discrepancies exist (see the IMPORT-DIR field of the output).

The calendar server must be up to run unirnsynch.

OPTIONS

-host

<hostname>

Specify the host where the node that has had deletions to local information resides.

-n

<node-ID>

Specify the node-ID of the node that has had deletions to its local information.

-p

<sysOpPsw>

Provide the SYSOP password of the node that has had deletions to its local information.

-rhost

<hostname>

Specify the host where the node that is to have its remote node information updated resides. Default is the local host.

-rn

<node-ID>

Specify the node-ID of the node that is to have its remote node information updated.

-rp

<remote-sysOpPsw>

Provide the SYSOP password of the node that is to have its remote node information updated.

-v

Print the current version number of `unirnsynch`.

-h

Print a usage message explaining how to run `unirnsynch`.

EXAMPLES

■   Propagate deletions to entries in node 30 on host "pepper" to the remote node information of node 20 on host "salt":

```
% unirnsynch –rn 20 -rhost salt -rp remote-sysOpPsw –n 30 –host pepper –p
sysOpPsw
```

EXIT STATUS

Exit values are:

0 Success

1 Usage error

2 System error

SEE ALSO

uninode

UNISIZEOF

unisizeof - Compute the size of the calendar server installation.

SYNTAX

```
unisizeof [–db | –n <node-ID>]

unisizeof –v
unisizeof –h
```

DESCRIPTION

unisizeof computes the size of a calendar server installation. By default, it determines the size of the /users/unison directory, including all database nodes and the calendar server (executables, * . ini files, etc.). Use the **-db** option to determine the size of the entire database and the **-n** option to determine the size of a single database node.

unisizeof runs whether the calendar server is up or down.

OPTIONS

-db

Compute the size of the entire database. The entire database is made up of all nodes on the server.

-n

<node-ID>

Compute the database size of the specified node.

-v

Print the version number of unisizeof.

-h

Print a usage message explaining how to run unisizeof.

EXAMPLES

■   Determine the size of the calendar server installation:

```
% unisizeof
unisizeof: total size of the calendar server 44216K
```

■   Determine the size of the entire database:

```
% unisizeof –db
unisizeof: total size of the calendar server database is 10010K
```

■   Determine the size of the database for node 10:

```
% unisizeof –n 10
unisizeof: database size for nodeid [10] is 760K
```

FILES

/users/unison/misc/unison.ini
Used to determine the default node (i.e. the node for which "name = N1" in this file) when unisizeof is used with the -**db** option.

EXIT STATUS

Exit values are:

    0 Success

    1 Failure

    2 Usage error

    3 User interrupt

UNISLICE (UNIX ONLY)

`unislice` - Extract information from the calendar server's log files.

SYNTAX

```
unislice <logFile(s)> [-s <starttime>] [-e <endtime>]

unislice -v
unislice -h
```

DESCRIPTION

`unislice` extracts information from the specified log file(s) and sends it to standard output. The `unisnapshot` utility uses `unislice` to gather information contained in log files. The <logFile(s)> argument is a list of one or more log files; each must be a fully-specified pathname separated from the others by a space. `unislice` can run on most of the log files in the `/users/unison/log` directory.

`unislice` runs whether the calendar server is up or down.

OPTIONS

-e

<endtime>

Set an end time. Only log file information with time stamps prior to this time are included in the output. Thus, if an end time of January 1 is set, no information from the 1st of January is included. See FORMAT OF starttime, endtime ARGUMENTS below for details on how to specify these arguments.

-s

Set a start time. Only log file information with time stamps on or after this time are included in the output. See FORMAT OF starttime, endtime ARGUMENTS below for details on how to specify these arguments.

-v

Print the current version number of `unislice`.

-h

Print a usage message explaining how to run `unislice`.

FORMATS

FORMAT OF THE time ARGUMENTS

Each of these arguments can take one of the forms:

- "day month [year] [time]"

- "day month [time] [year]"

- "month day [year] [time]"

- "month day [time] [year]"

where

- day

  is a number between 1 and 31;

- month

  is either the full name of the month or one of the following abbreviations: jan, feb, mar, apr, aug, sep, sept, oct, nov, dec (month is case-insensitive);

- year

  is specified using four digits; and

- time

  is in the form HH:MM or HH:MM:SS (HH is an integer between 0 and 23).

EXAMPLES

- Output the full contents of the `uniengd` log file:

  ```
  % unislice /users/unison/log/eng.log
  ```

- Output all `uniengd` messages logged on February 7th 1995:

  ```
  % unislice /users/unison/log/eng.log -s 7 feb 1995 -e feb 8 1995
  ```

- Output all `eng.log` messages after 1 PM, July 7:

  ```
  % unislice /users/unison/log/eng.log -s july 7 13:00
  ```

- Output all `eng.log` messages before 9 AM, October 15, 1995:

  ```
  % unislice /users/unison/log/eng.log -e oct 15 9:00 1995
  ```

- Output all `eng.log` messages logged in a 45-second period starting at 10 AM, January 30:

  ```
  % unislice /users/unison/log/eng.log -s jan 30 10:00:00 -e jan 30 10:00:46
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

SEE ALSO

unisnapshot

UNISNAPSHOT

`unisnapshot` - Compile calendar server information for diagnostic purposes.

SYNTAX

```
unisnapshot [<date>] [-nolog] [-p <sysOpPsw>]

unisnapshot [-s <starttime>] [-e <endtime>] [-nolog] [-p <sysOpPsw>]

unisnapshot -v
unisnapshot -h
```

DESCRIPTION

`unisnapshot` assembles information used by support staff to diagnose most calendar server problems. Should a problem ever arise, only this file need be supplied to support staff.

Output is written to the `unisnapshot.log` file in the `/users/unison/log` directory. `unisnapshot` can be restricted to include log file information logged during a single day, or during a specified time period. This reduces the amount of irrelevant information in the output.

Under Windows operating syatems, `unisnapshot` requires the SYSOP password for each node. See the -**p** option below for more information.

See FORMAT OF THE date ARGUMENT below for details on how to specify <date>.

`unisnapshot` can be run whether the calendar server is up or down.

OPTIONS

-e
<endtime>

Set an end time. Only log file information with time stamps prior to this time are included in the output of `unisnapshot`. Thus, if an end time of January 1 is set, no information from the 1st of January is included. <endtime> is a string of the same format as <date>.

-nolog
Prevent `unisnapshot` from including log file information in its output.

-p
<sysOpPsw>

This option exists only under Windows operating systems. Specify the SYSOP password to use to connect to each node. If you use this option, the SYSOP password must be the same for all nodes. If you do not use this option under Windows, `unisnapshot` prompts for the SYSOP password for the first node at the time it connects to that node. For each subsequent node, it prompts for the SYSOP password only if the SYSOP password for that node is different from the last SYSOP password entered.

-s

<starttime>

Sets a start time. Only log file information with time stamps on or after this time are included in the output of unisnapshot. <starttime> is a string of the same format as date.

-v

Print the current version number of unisnapshot.

-h

Print a usage message explaining how to run unisnapshot.

FORMATS

FORMAT OF THE date ARGUMENT

The date argument takes one of the forms:

- "day month [year] [time]"

- "day month [time] [year]"

- "month day [year] [time]"

- "month day [time] [year]"

where

- day

    is a number between 1 and 31;

- month

    is either the full name of the month or one of the following abbreviations: jan, feb, mar, apr, aug, sep, sept, oct, nov, dec (month is case-insensitive);

- year

    is specified using four digits; and

- time

    is in the form HH:MM or HH:MM:SS (HH is an integer between 0 and 23).

If no year is specified, the default is the current year.

EXAMPLES

■ Assemble all information:

```
% unisnapshot
```

■ Assemble all information except that contained in the log files:

```
% unisnapshot -nolog
```

■ Assemble all information about February 7th 1998:

```
% unisnapshot 7 feb 1998
```

■ Assemble all information about the period after 1 PM, July 7:

```
% unisnapshot -s july 7 13:00
```

■ Assemble all information about the period before 9 AM, October 15, 1998:

```
% unisnapshot -e oct 15 9:00 1998
```

■ Assemble all information about the 45-second period starting at 10 AM, January 30:

```
% unisnapshot -s jan 30 10:00:00 -e jan 30 10:00:46
```

FILES

/users/unison/log/unisnapshot.log
This is the file where unisnapshot writes its output. If a previous file exists at the time unisnapshot is invoked, it is overwritten.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNING

unisnapshot may take some time to complete.

UNISNCDUMP

unisncdump - Retrieve statistics from the calendar server's Synchronous Network Connection daemon/service.

SYNTAX

unisncdump [-host <hostname>] [-n <node-ID>] [-p <sysOpPsw>] [-screen]

unisncdump -v
unisncdump -h

DESCRIPTION

unisncdump retrieves statistics from the unisncd daemon and writes them to the /users/unison/log/unisncdump.log file. Included are the number of configured and available connections for each service.

OPTIONS

-host

<hostname>

Specify the host on which the unisncd is located.

-n

<node-ID>

Specify the calendar server node. Required if more than one node exists.

-p

<sysOpPsw>

Provide the SYSOP password. If this option is not used, prompting for the password occurs.

-screen

Display the output on the screen instead of writing it to the log file.

-v

Print the version number of unisncdump.

-h

Print a usage message explaining how to run `unisncdump`.

EXAMPLES

■ Dump the `unisncd` statistics for node 11 on host "oregano" to the screen (the node network contains two nodes: 11 and 12).

```
% unisncdump -screen -n 11 -host oregano
Enter SysOp password:
---------------------------------------------
DATE = Mon Sep 28 14:50:08 1998
PID = 1314
Host = oregano
Service = unieng,12
Transactions: Request = 0
              Check Request = 0
              Cancel Request = 0
              Free = 0

Connections:  Configured = 2
              Available = 2
              Granted = 0
              Request queue = 0
              Failed = 0
              Last failure = 0
              Next attempt = 0
              Attempt timeout = 0
              Max wait before retry = 3840
```

FILES

`/users/unison/log/unisncdump.log`
`unisncdump` writes to this file by default.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNISTART

`unistart` - Start up the calendar server.

SYNTAX

`unistart [-bypass] [-nocws | -cws] [-nosnc | -snc] [-das] [-r] [-w <secs>] [-c]`
`[-n] [-s]`

`unistart -v`
`unistart -h`

DESCRIPTION

`unistart` starts the calendar server. The default action is to start all 5 the calendar server daemons/services, in the following order: `unilckd`, `uniengd`, `unidasd` (if a Directory Server is being used), `unisncd`, and `unicwsd`. If you start any of the daemons/services manually, you must respect this order to avoid problems. For example, if you start `unisncd` manually, you must be certain that `unilckd`, `uniengd`, and `unidasd` (if you are running a Directory Server) are all running, and that `unicwsd` is not running.

By default, `unistart` calls `unicheck` to check the file system. It invokes `unicheck` with the option "`-maxdb 30`", restricting the check to including only the first 30 node databases.

> **Note:** -bypass, -c, -cws, -das, -n, -nosnc, -s, -snc and -w are all UNIX-only options.

OPTIONS

-bypass

By default, `unicheck` is run before the daemons/services are started. This option causes `unistart` to execute without running `unicheck`.

-c

Same as -**cws**. Included for backwards compatibility.

-cws

Starts only `unicwsd`, the Corporate-Wide services daemon/service, provided that the `unilckd` and `uniengd` daemons/services are up. To avoid problems, you

should also be certain that `unidasd` (if you are running a Directory Server) and `unisncd` are running.

-das

Starts only `unidasd`, provided that the `unilckd` and `uniengd` daemons/services are up. Note that this daemon is used only with directory servers. To avoid problems, you should also be certain that `unisncd` and `unicwsd` are not running.

-n

Same as -**nocws**. Included for backwards compatibility.

-nocws

By default, `unicwsd` is started if the `[CWS] enable` parameter in `unison.ini` is set to "`TRUE`". This option overrides this setting and prevents `unicwsd` from being started. If this option is used under NT, `unicwsd` can be brought up later by simply running `unistart` again without this option.

-nosnc

By default, `unisncd` is started if the `[SNC] enable` parameter in `unison.ini` is set to "`TRUE`". This option overrides this setting and prevents `unisncd` from being started.

-r

Removes the log files `act.log`, `cws.log`, `das.log`, `dbv.log`, `eng.log`, `lck.log`, and `snc.log` from the `/users/unison/log` directory. Note that new log files are immediately created for `cws.log`, `das.log`, `eng.log`, `lck.log`, and `snc.log`.

-s

Same as -**w**. Included for backwards compatibility.

-snc

Starts only `unisncd`, provided that the `unilckd` and `uniengd` daemons/services are up. To avoid problems, you should also be certain that `unidasd` (if you are running a Directory Server) is running and `unicwsd` is not running.

-w

When a process is started there is a short delay before a ps(1) command acknowledges the existence of the process. It is therefore appropriate to sleep until a ps(1) verifies that the process is running. The default period is 2 seconds. This option can be used to override this default; the <secs> argument specifies the number of seconds to sleep.

-v

Print the current version number of `unistart`.

-h

Print a usage message explaining how to run `unistart`.

EXAMPLES

- Start the calendar server without running `unicheck`; remove the old log files at the same time:

  ```
  % unistart -bypass -r
  ```

- Start the calendar server; do not run the Corporate-Wide Services daemon/service:

  ```
  % unistart -nocws
  ```

- Start the Corporate-Wide Services daemon/service at some later point:

  ```
  % unistart -cws
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNISTAT

`unistat` - Produce a report on a calendar server node.

SYNTAX

```
unistat [-l | -p] [-s | -g] [-m] [-sn] -n <node-ID>

unistat -v
unistat -h
```

DESCRIPTION

`unistat` produces a report for the specified node and sends it to standard output. `unistat` prompts the user for the SYSOP password for the node. The following information is included in the report:

■ For each user: the X.400 name, the X.400 Organizational Units, the number of events, instances, and attendees owned by the user, the size (in bytes) of any attached files, the size (in bytes) of any event descriptions, the size (in bytes) of any extra information attached to events, and the size (in bytes) of the user's agenda.

■ For each resource: the resource identifier, the number of events, instances, and attendees owned by the resource, the size (in bytes) of any attached files, the size (in bytes) of any event descriptions, the size (in bytes) of any extra information attached to events, and the size (in bytes) of the resource's agenda.

■ A list of public groups and their owners.

The calendar server must be up for `unistat` to run.

Note: The -l and -p options are mutually exclusive, as are the -s and -g options.

OPTIONS

-g
Print the list of public groups.

-l
Print the report in 128 characters per line mode. If this option is not used, the default is 80 characters per line.

-m
Print the members of the public groups.

-n

<node-ID>

Generate statistics for the specified node.

> **Note:**   On UNIX, the -n is optional; a node-ID may be specified on the command line without being preceded by -n.

-p

Create a PostScript report. File redirection should be used to capture this to a file.

-s

Print database statistics.

-sn

Print serial numbers. Obsolete with Oracle Collaboration Suite.

-v

Print the current version number of `unistat`.

-h

Print a usage message explaining how to run `unistat`.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNISTATS

`unistats` - Display summary statistics of the data in a calendar server stats file.

SYNTAX

```
unistats [-s <starttime>] [-e <endtime>] [-f <filename>] [-server <version>]
[-client <entry>] [-n <node-ID>] [-user <name> | -res <resource> | -ruser]
[-all]

unistats -v
unistats -h
```

DESCRIPTION

Displays summary statistics of the data in a calendar server stats file. By default, the input file is /users/unison/log/stats.log. Filters (i.e. the -**server**, -**client**, -**n**, -**user**, -**res**, -**ruser** options) may be used to compile statistics from a subset of the information in the stats file.

The default output is a summary for each unique client. Different versions of the same client are treated as separate clients, and a summary is output for each. The -**all** option displays a summary incorporating all clients. All output is displayed in 122-character-wide format. A complete list of all output fields is given in the OUTPUT section below.

OPTIONS

-all

Display summary incorporating all interface clients.

-client

<entry>

Display summary statistics on a specific calendar client. <entry> is the name and version of that client. See FORMAT OF THE entry, name, AND resource ARGUMENTS below for details on how to specify <entry>.

-e

<endtime>

Specify end time for statistics. If this option is not used, the default is the current time of the current day of the current month of the current year. See FORMAT OF THE time ARGUMENT below for details on how to specify <endtime>.

-f

<filename>

Specify the file to be used. This file must be in the same format as the default input file `/users/unison/log/stats.log`. This option is commonly used where a file has been created from an existing `stats.log` file and is supplied as input to `unistats`.

-n

<node-ID>

Display summary statistics on a specific node. <node-ID> is a calendar server node-ID.

-res

<resource>

Display summary statistics on a specific resource. <resource> is the name and/or identification number of the resource. See FORMAT OF THE entry, name, AND resource ARGUMENTS for details on how to specify <resource>.

-ruser

Display summary statistics on all reserved users (e.g. SYSOP).

-s

<starttime>

Specify a start time for the statistics. If this option is not used, the default start time is "Jan 1 1991 00:00:00". See FORMAT OF THE time ARGUMENT below for details on how to specify <starttime>.

-server

<version>

Display summary statistics on a specific calendar server. <version> is the version number of that server (e.g. A.02.90).

-user

Display summary statistics on a specific user. <name> is some combination of the surname, given name, and organizational units of the user. See FORMAT OF THE entry, name, AND resource ARGUMENTS below for details on how to specify <name>.

-v

Print the current version number of `unistats`.

-h

Print a usage message explaining how to run `unistats`.

FORMATS

FORMAT OF THE entry, name, AND resource ARGUMENTS

Each of the arguments <entry>, <name>, and <resource> is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. The "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

Note that if the ID key-value pair is specified in the -**res** argument, all other key-value pairs specified along with it are ignored.

Table C–18   Accepted keys

OPTION	KEY	MEANING OF THE KEY
-client	N	Client Name
-client	V	Client Version
-user	S	Surname
-user	G	Given name
-user	OU1	Organizational Unit 1

Table C–18 Accepted keys

OPTION	KEY	MEANING OF THE KEY
-user	OU2	Organizational Unit 2
-user	OU3	Organizational Unit 3
-user	OU4	Organizational Unit 4
-res	R	Resource Name
-res	ID	Resource ID

Some example specifications are:

```
-client "N=Windows CorporateTime - 32 Bit/V=version 4.1"
-user "S=Carter/G=Angela"
-res "R=laptop"
-res "ID=328"
```

FORMAT OF THE time ARGUMENT

The <starttime> and <endtime> arguments may be expressed as either:

- "<day> <month> [<year>] [<time>]" or

- "[<month> <day>] <time> [<year>]"

where

- <day> is a number between 1 and 31;

- <month> is either the full name of the month or the first three letters (e.g. jan, feb, mar, etc.) (month is case-insensitive);

- <year> must be 1991 or higher and must be specified using four digits;

- <time> is in the form HH:MM or HH:MM:SS (HH is an integer between 0 and 23, MM is an integer between 0 and 59, and SS is a number between 0 and 59).

The order of the individual elements in the argument is unimportant. What is important is that either day and month be specified, or time be specified. The following are all valid examples:

```
Feb 22 1996 10:00:00
22 february 10:00:00
10:00:00 february 22 1996
1996 feb 22
feb 22
```

```
10:00:00
```

Default values for <day>, <month>, <year>, and <time> are the current day, current month, current year and current system time respectively.

Any missing field in <time> (HH, MM, or SS) is replaced with the current HH, MM, or SS value. E.g. if the current date and time is March 12 1998 10:41:34, and only HH:MM are specified in the argument, the SS becomes "34":

```
-e 12:41 -> March 12 1998 12:41:34
-s 12:41 -> March 12 1998 12:41:34
```

If none of the time fields are specified, <starttime> defaults to the first minute of the day, and <endtime> defaults to the last minute of the day:

```
-s feb 22 -> feb 22 1998 00:00:00
-e feb 22 -> feb 22 1998 23:59:59
```

OUTPUT

All output fields displayed by unistats are explained here, in the order in which they will be seen:

Table C–19   unistats CLIENT output fields

CLIENT fields	Description
CLIENT	Name and Version of the calendar client
SYSTEM	Name of the host operating system of the calendar server
SERVER	The calendar server version
SIGNONS	Number of records used for the summary statistics of this client
SESSION AVERAGE	Average session time
CPU TOTAL	Total cpu time for all SIGNONS
CPU MEDIAN	Median cpu time
CPU AVERAGE	Average cpu time; "usr" stands for user and "sys" stands for system
NETWORK TOTAL	Total number of bytes exchanged between the client and calendar server host

Table C–19   unistats CLIENT output fields

CLIENT fields	Description
NETWORK MEDIAN	Median of NETWORK TOTAL
NETWORK AVERAGE	Average of NETWORK TOTAL; "snd" stands for send and "rcv" stands for receive
CALLS	Total number of function calls

Table C–20   unistats FUNCTION NAME output fields

FUNCTION NAME fields:	Description
CALL (%)	Percentage of all calls for this function
TIME (W)	Greatest user response time for function to process one call
TIME (A)	Average user response time for processing this function
CPU (%)	Percentage of all cpu time taken by this function
CPU (%U)	Percentage (of CPU(%) above) taken by the user
CPU (%S)	Percentage (of CPU(%) above), taken by the system
CPU (W)	Greatest cpu time taken by this function to process one call
CPU (A)	Average cpu time taken by this function to process one call
NET (%)	Percentage of all network i/o used by this function
NET (%S)	Percentage (of NET(%) above) of data sent
NET (%R)	Percentage (of NET(%) above) of data received

EXAMPLES

- Get the summary statistics of the data from the default file (/users/unison/log/stats.log):

  ```
  % unistats
  ```

- Get the summary statistics of all "windows" clients from the file myfile.log:

  ```
  % unistats -client "N=window*" -f myfile.log
  ```

■ Get the summary statistics of user "Don Martin" from server "A.02.90" only:

```
% unistats –user "s=martin/g=don" –server "A.02.90"
```

■ Get summary statistics of the resource "projector" from "motif" clients only:

```
% unistats –res "R=projector/ID=901" –client "N=Motif"
```

■ Get summary statistics for July 19:

```
% unistats –s jul 19 –e jul 19
```

■ Get summary statistics for all users of all clients:

```
% unistats –all –user "S=*"
```

■ Get summary statistics of all reserved users in node 70:

```
% unistats –ruser –n 70
```

FILES

`/users/unison/log/stats.log`
By default, `unistats` obtains its information from this file. The `[ENG] stats` parameter in `unison.ini` must be set to "`TRUE`" to enable `uniengd` to log information to this file.

`/users/unison/log/unistats.log`
`unistats` logs any errors in this file.

EXIT STATUS

Exit values are:

0 Success

1 usage error

2 system error

UNISTATUS

`unistatus` - Determine the status of the calendar server.

SYNTAX

`unistatus [–f] [–d] [–s] [–q] [–n] [–e]`

```
unistatus -v
unistatus -h
```

DESCRIPTION

By default `unistatus` determines which of the calendar server daemons/services are running and prints their current status to standard output. The -**d**, -**f**, -**n**, -**q**, and -**s** options are currently supported only on UNIX platforms.

`unistatus` runs whether the calendar server is up or down.

OPTIONS

-d

Produce a report for daemon processes only.

-e

Alter the default exit status values to provide information about the calendar server daemon/services. See EXIT STATUS below for the values and their meanings.

-f

Produce an extensive ps-like report, taking into account the distinction between daemons and servers. The calendar server may have the following daemons and servers running:

- `uniengd` daemon: always runnning

- `unilckd` daemon: always running

- `unieng` server: one for each signed-on user

- `unicwsd` daemon: runs if corporate-wide services are enabled

- `unisncd` daemon: runs if remote-node services are enabled and/or a directory server is being used

- `unidasd` daemons and servers: runs if a directory server is being used

-n

Suppress the printing of the messages "CORPORATE-WIDE SERVICES are down" or "REMOTE-NODE SERVICES are down" when the `unilckd` and `uniengd` daemons/services are running but the `unicwsd` or `unisncd` daemons/services are not.

-q

Force the quiet version of the command which does not produce the usual status message.

-s

Produce a report for server processes.

-v

Print the current version number of `unistatus`.

-h

Print a usage message explaining how to run `unistatus`.

EXAMPLES

- Run `unistatus` under NT.

```
% unistatus
unistatus: unicwsd is running
unistatus: unisncd is running
unistatus: unidasd is running
unistatus: uniengd is running
unistatus: unilckd is running
```

- Produce a full report on all calendar server daemons and servers on a UNIX system.

```
% unistatus -f
   UID    PID   PPID   STIME  TIME  COMMAND  CLASS
unison  23846      1  Sep 25  2:02  unilckd daemon
unison  23851      1  Sep 25  0:00  uniengd daemon
unison  23927      1  Sep 25  0:00  unidasd daemon
unison  24007      1  Sep 25  0:00  unisncd daemon
unison  24014      1  Sep 25  0:28  unicwsd daemon
unison  24117  23851  Sep 25  0:18  uniengd server
unison  24120  23851  Sep 25  0:16  uniengd server
unison  24123  23851  Sep 25  0:16  uniengd server
unison  24132  23851  Sep 25  0:00  uniengd server
unison  24153  23851  Sep 25  0:00  uniengd server
unison  24111  23927  Sep 25  0:00  unidasd server
unison  24129  23927  Sep 25  0:00  unidasd server
unison  24151  23927  Sep 25  0:00  unidasd server
unison  24172  23927  Sep 25  0:00  unidasd server
```

```
unison  24178  23927  Sep 25  0:00  unidasd server
unistatus: the calendar server is up
```

■ Produce a report on the calendar server processes under UNIX.

```
% unistatus -s
   UID    PID   PPID   STIME  TIME  COMMAND  CLASS
unison  24117  23851  Sep 25  0:18  uniengd server
unison  24120  23851  Sep 25  0:16  uniengd server
unison  24123  23851  Sep 25  0:16  uniengd server
unison  24132  23851  Sep 25  0:00  uniengd server
unison  24153  23851  Sep 25  0:00  uniengd server
unison  24111  23927  Sep 25  0:00  unidasd server
unison  24129  23927  Sep 25  0:00  unidasd server
unison  24151  23927  Sep 25  0:00  unidasd server
unison  24172  23927  Sep 25  0:00  unidasd server
unison  24178  23927  Sep 25  0:00  unidasd server
unistatus: the calendar server is up
```

■ Produce a report on calendar server daemon processes under UNIX.

```
% unistatus -d
   UID    PID   PPID   STIME  TIME  COMMAND  CLASS
unison  23846      1  Sep 25  2:02  unilckd daemon
unison  23851      1  Sep 25  0:00  uniengd daemon
unison  23927      1  Sep 25  0:00  unidasd daemon
unison  24007      1  Sep 25  0:00  unisncd daemon
unison  24014      1  Sep 25  0:28  unicwsd daemon
unistatus: the calendar server is up
```

EXIT STATUS

The default exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

Use of the -**e** option alters the default exit values to encode the status of the various calendar server daemons/services. These values are as follows:

■ 0...127: Success. This value is the sum of one or more of the values 1, 2, 4, 8, 16, 32, and 64, where:

- 1 means `uniengd` servers are running

- 2 means `unicws` daemon is running

- 4 means `uniengd` daemon is running

- 8 means `unilckd` daemon is running

- 16 means `unisncd` daemon is running

- 32 means `unidasd` servers are running

- 64 means `unidasd` daemon is running

- 253 Interrupted

- 254 Usage error

- 255 Failure

UNISTOP

`unistop` - Shut down the calendar server.

SYNTAX

```
unistop [-y] [-bypass] [-ser] [-cws] [-snc] [-das] [-dass] [-w <secs>] [-l
<level>] [-cl <level>]

unistop -v
unistop -h
```

DESCRIPTION

`unistop` shuts down all or part of a running calendar server. By default, all daemons/services are shut down, in the following order: `unicwsd`, `unisncd`, `unidasd` (if a directory server is being used), `uniengd`, and `unilckd`. If any users are currently signed-on, `unistop` prompts for confirmation before proceeding with the shutdown. If you stop any of the daemons/services manually, you must respect this order to avoid problems. For example, if you stop `unisncd` manually, you must be certain that `unicwsd` is not running and that `unilckd`, `uniengd`, and `unidasd` (if you are running a Directory Server) are all running.

The -**bypass**, -**cl**, -**das**, -**dass**, -**l**, -**ser**, -**snc**, -**w**, and -**y** options are all UNIX-only. The -**l** and -**cl** options are mutually exclusive.

unistop can only be run if the calendar server is at least partially up (i.e. one or more daemons/services are running).

OPTIONS

-bypass

Allow unistop to execute even if another unistart or unistop process is running.

-cl

Allow cycling of shutdown levels from 0 up to the specified level, when the current level fails. Thus, levels 0, 1, 2, & 3 are tried until either the specified level is reached or the shutdown is successful. The user is NOT prompted for any confirmation.

> **Warning:**   **A shutdown at level 3 may corrupt the calendar server database. See the -l option for more information.**

-cws

Shut down only the Corporate-Wide Services daemon/service (unicwsd). The unilckd and uniengd daemons/services must be running for this option to succeed. To avoid problems, you should also be certain that unisncd and unidasd (if you are running a Directory Server) are both running.

-das

Stop only the unidasd daemons and servers. These are used only with a directory server. The unilckd and uniengd daemons/services must be running for this option to succeed. To avoid problems, you should also be certain that unicwsd and unisncd are both running.

-dass

Stop only the unidasd servers. These are used only with a directory server.

-l

The shutdown levels are 0, 1, 2 and 3, with the normal level (and the default) being 0. The higher the number, the more severe the shutdown. You should always begin with 0 and only in the rare event that this fails should you proceed to 1, then 2 and finally 3.

> **Warning: A level 3 shutdown is guaranteed to succeed but it may corrupt the calendar server database. Contact qualified support staff before performing a shutdown at this level. unistop prompts for confirmation before proceeding with a level 3 shutdown.**

-ser

Shut down only the `uniengd` servers. The `unilckd` and `uniengd` daemons must be running for this option to succeed. Each `uniengd` server is associated with a signed-on calendar user; the `uniengd` daemon is always present when the calendar server is running. If this option is not used, both `uniengd` daemons and servers are shut down.

-snc

Shut down only the `unisncd` daemon. The `unilckd` and `uniengd` daemons must be running for this option to succeed. To avoid problems, you should also be certain that `unidasd` (if you are running a Directory Server) is running and `unicwsd` is not running.

-w

<secs>

When a process is stopped there is a short delay during which the `ps`(1) command continues to acknowledge the existence of the process. It is therefore necessary to sleep for a short period of time until a `ps` verifies that the process is no longer running. The default delay is 4 seconds. This option overrides the default; the <secs> parameter specifies the number of seconds to sleep.

-y

By default, if there are any users signed on to the calendar server, a prompt is issued to confirm that a shutdown is desired. This option causes `unistop` to automatically proceed with the shutdown even if there are users signed on. The shutdown of each of the active `uniengd` servers proceeds in such a way as to ensure the integrity of the database.

-v

Print the current version number of `unistop`.

-h

Print a usage message explaining how to run `unistop`.

EXAMPLES

- Shut down the calendar server.

  ```
  % unistop
  ```

- Shut down the Corporate-Wide Services daemon.

  ```
  % unistop -cws
  ```

- Shut down only the directory server daemons and servers under UNIX.

  ```
  % unistop -das
  ```

- Execute a level 1 shutdown under UNIX after a level 0 shutdown has failed.

  ```
  % unistop -l 1
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNISYNCREFRESH

`unisyncrefresh` - Refresh calendar server sync records.

SYNTAX

```
unisyncrefresh [-node <node-ID>] [-host <hostname>] [-fr <date>] [-p
<SYSOPpassword>]

unisyncrefresh -v
```

```
unisyncrefresh -h
```

DESCRIPTION

unisyncrefresh refreshes calendar server synchronization records.

If you are experiencing long refresh times when piloting newer calendar clients including 3.0 versions of the Sync clients and Outlook Connector 3.3 with a small number of users, you can increase performance by creating a script to run unisyncrefresh automatically every 5 to 15 minutes.

unisyncrefresh can only be run if the calendar server is up.

OPTIONS

-fr

<date>

Forces a specified start date for the refresh. This option is not recommended for general use; consult Oracle support for instructions on using this option in specific circumstances.

-host

<hostname>

Specifies a host on which to look for the node specified by the -**n** option. Required for remote hosts. If -**host** is not present, unisyncrefresh will assume the local host. If -**host** is specified and -**n** is not, unisyncrefresh will search for a master node on the specified host.

-n

<node-ID>

Specify a node. If -**n** is not used, unisyncrefresh will search for a master node located on the host specified by the -**host** option. If no master node exists, -**n** is required.

-p

<SYSOPpassword>

Provide the SYSOP password for the specified node. If this option is not used, unisyncrefresh will prompt for the password.

-v

Print the current version number of `unisyncrefresh`.

-h

Print a usage message explaining how to run `unisyncrefresh`.

EXAMPLES

- Refresh the sync records on node 45 of the local host.

  ```
  % unisyncrefresh -n 45
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNITZINFO

`unitzinfo` - Print information about a calendar server time zone.

SYNTAX

```
unitzinfo [-c] [-l] [-t <timezone>] [-node <node-ID>] [-y <year>]

unitzinfo -v
unitzinfo -h
```

DESCRIPTION

Extracts information from the calendar server time zone table found in the `/users/unison/misc/timezone.ini` file. By default, only the information for the configured time zone and the current year used by the calendar server is printed in an 80-character-wide format.

The calendar server table contains time zone information from the year 1991 to 2074 inclusive.

`unitzinfo` can be run whether the calendar server is up or down.

OPTIONS

-c

List the time zone information by country. Time zones within a country are listed in sequence. The printed fields are:

Table C–21   Time zone fields

Field	Description
COUNTRY	Country name
TIMEZONE	Time zone name
ST	The difference (in hours) from GMT
DST	The difference (in hours) from GMT during Daylight Savings Time (DST)
EFFECTIVE PERIOD	The period when DST is in effect

-l

Print the information in 132-character-wide ("large") output format.

-node

<node-ID>

Specify the node. This option causes the information for the time zone configured for the node to be output.

-t

<timezone>

Specify the name of the time zone to print. If *timezone* has the value "all", the complete list of time zones is printed.

-y

<year>

Specify the year for which the time zone information will be output (e.g. to view the DST period for that year). <year> must be specified using four digits. The default is the current year.

-v

Print the current version number of unitzinfo.

-h

Print a usage message explaining how to run unitzinfo.

EXAMPLES

■ Display the time zone information associated with node 20:

```
% unitzinfo –node 20
EST5EDT  Eastern Standard Time, Eastern Daylight Time
         U.S.A. (Eastern), Canada (Eastern), Bahamas,
         Haiti, Turks & Caicos
         Hours from GMT: -5h
          Daylight Saving Time : -4h (Apr 4,1999 - Oct 30,1999)
```

FILES

/users/unison/misc/timezone.ini
This file contains the time zone descriptions used by the calendar server.

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

UNIUSER

uniuser - List, add, or delete calendar users, or to modify the information associated with them.

SYNTAX

Listings

uniuser –ls [<user>] [–format <format>] [–host <hostname>] –n <node-ID> [–p <sysOpPsw>]

```
uniuser –format <format> –n <node-ID> [–p <sysOpPsw>]
```

Addition

```
uniuser -add <user> [-s <sections>] [-host <hostname>] -n <node-ID> [-p
<sysOpPsw>]
```

Deletion

```
uniuser -del <user> [-y] [-host <hostname>] [-n <node-ID>] [-p <sysOpPsw>]
```

```
uniuser -desdel <user> [-host <hostname>] [-n <node-ID>] [-p <sysOpPsw>]
```

```
uniuser -grpdel <user> [-host <hostname>] [-n <node-ID>] [-p <sysOpPsw>]
```

Modification

```
uniuser -mod <user> -s <sections> | -m <modifier> [-host <hostname>] [-n
<node-ID>] [-p <sysOpPsw>]
```

Multiple additions, deletions, modifications

```
uniuser -ex <filename> [-s <sections>] [-k] [-host <hostname>] [-n <node-ID>]
[-p <sysOpPsw>]
```

```
uniuser -edit [-host <hostname>] [-n <node-ID>] [-p <sysOpPsw>]
```

Other

```
uniuser -s [<sections>] [-host <hostname>] -n <node-ID> [-p <sysOpPsw>]
```

```
uniuser -info
```

```
uniuser -v
uniuser -h
```

DESCRIPTION

uniuser can list, add, or delete calendar users, or modify the information associated with them. See the FORMAT OF THE user ARGUMENT section below for details on how to specify the *user* argument. Note that the -**ls**, -**add**, -**del**, -**grpdel**, -**desdel**, and -**mod** options are all mutually exclusive.

The information associated with a calendar user is a combination of the key-value pairs described in the FORMAT OF THE user ARGUMENT below, and that contained in the user.ini file. This includes user preferences, security,

administrative rights, X.400 information, personal group, admin group membership and the list of persons permitted to work as a designate for the user.

It is recommended that you use `uniuser` to modify only user attributes that are specific to the calendar server. Any attributes that can be modified using the Oracle Internet Directory administration tools directly should not be modified through `uniuser`.

The calendar server must be up to run `uniuser`.

`unires` is the complementary utility for resources.

OPTIONS

-add

<user>

Create a new calendar user. The information associated with the new user is a combination of what is specified in the <user> argument and the default values in the `user.ini` file. By default, when `uniuser` reads the `user.ini` file, it considers only the values in the `[GEN]` section. Use the -**s** option to apply values from other sections of the `user.ini` file. Use the -**ex** option to add multiple users.

When using the calendar server's internal directory, the "S" key is mandatory for the -**add** option.

For external directories, the user must already exist in the directory server. The DID (Directory ID) for the user must be specified, and it must be in DN (Distinguished Name) format. This can be followed by data in X.400 format. See EXAMPLES.

-del

<user>

Delete the specified calendar user. `uniuser` prompts for confirmation before performing the deletion unless the -**y** option is used. If more than one user is to be deleted, the -**ex** option must used.

This operation can take a long time for users with very large agendas, and may have an impact on the performance of the calendar server for other users. It is recommended that you only delete users in off-peak hours.

-desdel

Delete the list of designates for the specified user. If more than one user matches <user>, the command fails.

-edit

This option only exists under UNIX. It allows you to first output the list of existing calendar users to a file, then edit the file to make desired modifications, and finally to input the changes back into the node. The following sequence of commands is performed:

```
% uniuser -ls -n node-ID > file
% vi file
% uniuser -ex file -n node-ID
% rm file
```

-ex

<filename>

Perform the additions, deletions, and/or modifications specified in the file <filename>. Each line of the file must begin with one of the characters '.', '#', 'A', 'a', 'D', 'd', 'M', 'm', 'S', 's', or '+'. This initial character specifies the action to take, as follows:

Character	Action
'.' or '#'	ignore the line
'A' or 'a'	add the user
'D' or 'd'	delete the user
'M' or 'm'	modify the user
'S' or 's'	update the user with the settings from the user.ini file
'+'	treat the line as the continuation of the previous line; note that key-value pairs cannot break over lines

The initial character must be followed by a space and a user specification. In the case of a modification, the ID must be specified; it alone is used to identify the user, while the other key-value pairs specified along with it are applied as the modifications. See EXAMPLES.

For each deletion specified in the file, uniuser prompts for confirmation before performing the deletion. The **-y** option is used to automatically provide confirmation.

One way to create this file is to save the output of uniuser -ls to a file. This can then be edited and input to uniuser -ex.

The -**s** sections option may be used with -**ex** to set the corresponding user information to the default values contained in the specified sections of the user.ini file.

For directory servers, the most common way of adding many calendar users is to first use unidssearch to output the list of all non-calendar users to a file. This file can then be modified (if necessary), and input to uniuser using the -**ex** option. unidssearch outputs in the same "key=value/key=value/...." format that uniuser requires for input. See EXAMPLES.

-format

<format>

This option is used to select user information fields and to customize the format of the output. The -**info** option lists the parameters that can be used to specify the customized format. These are also listed in the FORMAT OF THE user ARGUMENT section below. If this option is not used, all user information fields are output, and a default presentation format is used. See EXAMPLES.

-grpdel

<user>

Delete the specified user from all admin groups. A single user must match <user> or the command fails.

-host

<hostname>

Specifies a host on which to look for the node specified by the -**n** option. Required for remote hosts. If -host is not present, uniuser will assume the local host. For modification or deletion, if -**host** is specified and -**n** is not, uniuser will search for a master node on the specified host. If a master node is found, uniuser will use it to locate the selected users.

-info

List the format parameters used with the -**format** option.

-k

Used with the -**ex** option to force uniuser to continue processing all lines in the file even if it encounters an error. Errors are sent to standard error; file redirection may be used to capture these to a file.

-ls

[<user>]

If a user is specified, -**ls** lists that user (in the specified node). If no user is specified, all users in the node are listed. The -**format** option can be used with -**ls** to configure the presentation of the output (see EXAMPLES).

-m

<modifier>

Specify a modification to be made to the user information of a particular user. The modifier is a string of the same form as the user argument with the following exceptions: the ID key may NOT be specified, and the PSW, PUBLISHEDTYPE and GLOBALREADONLY keys MAY be specified.

-mod

<user>

Modify the information associated with the specified user. This option is used with either the -**s** or the -**m** options. When used with the -**s** option, it sets the corresponding information for the user to the values contained in the specified sections of the user.ini file. When used with the -**m** option, modifications are made to the user's X.400 information.

> **Note:** It is recommended that you use uniuser to modify only user attributes that are specific to the calendar server. Any attributes that can be modified using the Oracle Internet Directory administration tools directly should not be modified through uniuser.

-n

<node-ID>

Specify a node. If -**n** is not used, `uniuser` will search for a master node located on the host specified by the -**host** option. If a master node is found, `uniuser` will use it to locate or distribute the specified users. If no master node exists, -**n** is required.

-p

<sysOpPsw>

Provide the SYSOP password for the specified node. If this option is not used, prompting for the password occurs.

-s

[<sections>]

Evaluate all or some of the contents of the `user.ini` file and output information from it to standard output.

Use this option without the <sections> argument to determine what sections exist in the `user.ini` file. The output in this case is a list of all section names along with their respective "Info" key-value pairs.

Use the <sections> argument to evaluate all of the key-value pairs in the specified sections. In this case all key-value pairs in the specified sections are output, along with any errors detected in these pairs. This is one way to validate the contents of the `user.ini` file. See the -**add** and -**mod** options for information on using -**s** <sections> to apply values from the `user.ini` file.

The <sections> argument is a list of one or more section names, each separated by a forward slash (e.g. "GEN/GR1/GR2" specifies the sections GEN, GR1 and GR2). Evaluation is done from left to right. Thus, in the above example, GEN is evaluated first, GR1 second, and GR2 last. Where the same key appears in more than one section, the value of the last instance evaluated takes precedence.

-y

Used with the -**del** and -**ex** options to auto-confirm the deletion(s).

-v

Print the current version number of `uniuser`.

-h

Print a usage message explaining how to run `uniuser`.

FORMATS

FORMAT OF THE user ARGUMENT

The user argument is a string of the form "key=value/key=value/...", where "key" is one of those listed below, and "value" is any string. Both "key" and "value" are case insensitive. For all keys except the ID key, the "value" string may be terminated by a wild card symbol (*). If a forward slash "/" is to be included in a string, it should be escaped with the character "\" to prevent it from being interpreted as a key-value pair delimiter - i.e. `"S=Hoopla/OU1=R\/D"`.

If, in a UNIX environment, a shell will be processing the string (e.g. the string is provided on the command line or is passed as an argument to the utility), the string should be enclosed in quotation marks. Furthermore, if characters meaningful to the shell are included in the string, they should be escaped (i.e. preceded by the escape character "\") to prevent the shell from interpreting them.

> **Note:** If the ID key-value pair is specified in the user argument, all other key-value pairs specified along with it are ignored.

The format parameters listed in the third column below are used with the -**forma**t option to configure the presentation of a listing (see EXAMPLES).

Table C–22 Accepted keys

Key	X.400 Field	Format Parameter
S	Surname	%S
G	Given name	%G
I	Initials	%I
ID	Identifier	%ID
X	Generation	%X
OU1	Organizational Unit 1	%OU1
OU2	Organizational Unit 2	%OU2
OU3	Organizational Unit 3	%OU3
OU4	Organizational Unit 4	%OU4
O	Organization	%O

Table C–22   Accepted keys

Key	X.400 Field	Format Parameter
C	Country	%C
A	Administration domain	%A
P	Private domain	%P
PHONE	Phone number	%PHONE
EXT	Phone extension	%EXT
FAX	Fax phone number	%FAX
EMPL-ID	Employee number	%en
JOB-TITLE	Job title	%jt
EMAIL	<value of [ENG] usermailmap parameter>	None
PSW	Password (maximum of 15 characters)	None
MOB-ENABLE	Mobile notification (ON/OFF)	None
MOB-FILTERRANGE	Time range of SMS filter (hh:mm-hh:mm)	None
MOB-FILTERRANGEACTION	Filter action (NOTHING/SUSPEND/DISCARD)	None
MOB-NOTIFY	Notify of new, modified and deleted events (ON/OFF)	None
MOB-PHONE	User's mobile phone number	None
MOB-REMINDER	Set mobile device alarm for vCalendar reminders (ON/OFF)	None
MOB-SUSPEND	Disable SMS (ON/OFF)	None
MOB-TYPE	Type of mobile device (NOK.A61XX (vCalendar note), NOK.A61XX- (text message), NOK.A61XX+ (vCalendar and text), NOK.A91XX, or TEXT for generic device)	None

Table C–22   Accepted keys

Key	X.400 Field	Format Parameter
SMSC-PHONE	User's preferred SMSC number	None
REMINDER-AUDIBLE	Audible reminder (ON/OFF)	None
REMINDER-VISUAL	Visual reminder (ON/OFF)	None
REMINDER-SERVEREMAIL	E-mail reminder (ON/OFF)	None
REMINDER-SERVERSMS	SMS reminder (ON/OFF)	None
REMINDER-UPCOMING	Unpcoming reminder (ON/OFF)	None
PUBLISHEDTYPE	Published type (NOTPUBLISHED, PUBLISHED, EVENTCALENDAR; -**add** only)	None
GLOBALREADONLY	Global read access (ON/OFF; -**add** only)	None
LANG	Language (en-US, fr-FR, it-IT, es-ES, fi-FI, de-DE, pt-PT, ja-JP, zh-CN, ko-KR, sv-SE, pt-BR, nl-NL)	None

EXAMPLES

LISTINGS

- List all users in node 23 whose last names begin with "W":

```
% uniuser -ls "s=W*" -n 23
Enter SysOp password:
S=Whitman/G=Walt/ID=154/C=US
S=Winterson/G=Jeannette/ID=114/C=England
```

- List all users in node 23 whose last names begin with "W"; output only their surname and given name, separated by a colon:

```
% uniuser -ls "s=W*" -format "%s:%g" -n 23
Enter SysOp password:
Whitman:Walt
```

```
Winterson:Jeannette
```

■ List all users in node 23; output their surname and given name, separated by a colon:

```
% uniuser -format "%s:%g" -n 23
Enter SysOp password:
Brossard:Nicole
Dillard:Annie
Jansson:Tove
Kilpi:Eeva
Kundera:Milan
Lorde:Audre
Morrison:Toni
Sanchez:Sonia
Whitman:Walt
Winterson:Jeannette
```

ADDITION

■ Add the user "Maya Angelou" to node 24.

```
% uniuser -add "S=Angelou/G=Maya" -n 24
```

■ Perform the same addition on a directory server.

```
% uniuser -add "DID=cn=Maya Angelou, o=Acme, c=US" -n 24
```

DELETION

■ Delete the user "Eeva Kilpi" from node 24:

```
% uniuser -del "S=Kilpi/G=Eeva" -n 24
```

MODIFICATION

1. Modify Milan Kundera's entry to reflect recent changes to the [GEN] section of the user.ini file (Milan Kundera exists in node 23). Look at the values in the GEN section of the user.ini file and ensure they are all valid:

```
% uniuser -s "GEN" -n 23
StartDay = 08h00
EndDay = 18h00
TimeInc = 30
ShowSunday = FALSE
```

```
ShowSaturday = FALSE
TimeFormat = 2
RefreshFrequency = 60
DefaultReminder = 0
TimeBeforeReminder = 10
MailNotification = TRUE
OU1 =
OU2 =
OU3 =
OU4 =
O =
C =
A =
P =
TimeZone =
ViewNormalEvent = TIME
ViewPersonalEvent = TIME
ViewConfidentialEvent = TIME
ViewNormalTask = NO
ViewPersonalTask = NO
ViewConfidentialTask = NO
CanBookMe = TRUE
```

2. Proceed with the modification:

```
% uniuser –mod "S=Kundera/G=Milan" –s "GEN" –n 23
```

3. Modify Milan Kundera's OU1 value to "authors":

```
% uniuser –mod "S=Kundera/G=Milan" –m "ou1=authors" –n 23
```

MULTIPLE ADDITIONS, DELETIONS, MODIFICATIONS

Multiple additions, deletions, and modifications are done using the -**ex** option. In this example, three new calendar users are added, one modified, and one deleted. A directory server is being used.

1. Output all users in the directory server who are not currently calendar users:

```
% unidssearch > multiple.dat
% cat multiple.dat
A DID=cn=Italo Calvino,o=Acme, c=US
A DID=cn=Herman Hesse,o=Acme, c=US
A DID=cn=Doris Lessing,o=Acme, c=US
A DID=cn=Anja Kauranen,o=Acme, c=US
```

2. Modify the data in the file: change the country for Italo Calvino to Italy; add the modification to Walt Whitman's first name; add the deletion of Nicole Brossard.

```
% vi multiple.dat
% cat multiple.dat
A DID=cn=Italo Calvino,o=Acme, c=US/C=Italy
A DID=cn=Herman Hesse,o=Acme, c=US
A DID=cn=Doris Lessing,o=Acme, c=US
A DID=Anja Kauranen,o=Acme, c=US
M G=Walter/ID=154
D G=Nicole/S=Brossard
```

In the case of the modification, the ID is used to find the user, and the given name is modified to "Walter".

3. Input the file to `uniuser`:

```
% uniuser -ex multiple.dat 23
Enter SYSOP password:
uniuser: added "cn=Italo Calvino,o=Acme, c=US"
uniuser: added "cn=Herman Hesse,o=Acme, c=US"
uniuser: added "cn=Doris Lessing,o=Acme, c=US"
uniuser: added "cn=Anja Kauranen,o=Acme, c=US"
uniuser: modified "Whitman,Walt"
uniuser: deleted "Brossard,Nicole"
```

Note that if this example did not use a directory server, the input file would contain the following:

```
% cat multiple.dat
A S=Calvino/G=Italo/C=Italy
A S=Hesse/G=Herman
A S=Lessing/G=Doris
A S=Kauranen/G=Anja
M G=Walter/ID=154
D G=Nicole/S=Brossard
```

FILES

`/users/unison/misc/user.ini`
This file specifies possible calendar user configurations. See also the calendar server Reference Manual, Appendix A, "User and Resource Parameters".

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

WARNINGS

Modification of Key-Value Pairs

This utility uses the `[UTL]` `adm_moduser<field>` parameters in `unison.ini` to determine whether or not it can modify the corresponding key-value pairs for the user. For example, if `adm_modusersurname` were set to `FALSE`, then the value associated with the S key could be modified by this utility.

If no corresponding `adm_moduser<field>` for a particular key appears in the file, the utility assumes it CAN modify it.

Directory Server Warning

It is important to understand the implications of the directory server configuration for calendar server utilities. In a supplier-consumer configuration, the scheduling of updates between the consumer and supplier may result in temporary differences between the two. This may mean that a calendar server utility is reading from a consumer directory server that has not yet been synchronized with its supplier.

Deleting users with large agendas

Deleting users with a large numbers of meetings and events can take a long time and cause a decrease in performance for other calendar users. It is recommended that you delete such users outside of normal hours, or at least not at times of peak calendar usage.

SEE ALSO

`unires` is the complementary utility for resources.

UNIVERSION

`universion` - Verify the version of the calendar server.

SYNTAX

```
universion [-all] [-nowarn]

universion -v
universion -h
```

DESCRIPTION

`universion` displays the version number of the calendar server and checks all scripts and binaries to see if their versions are up to date.

`universion` runs whether the calendar server is up or down.

OPTIONS

-all

Display version number for each component of the calendar server.

-nowarn

Suppress warning messages.

-v

Print the current version number of `universion`.

-h

Print a usage message explaining how to run `universion`.

EXAMPLES

- Display the version number of the calendar server and check that all of its components are up to date:

  ```
  % universion
  ```

- Display the version number of the calendar server and each of its components; check that all components are up to date:

  ```
  % universion -all
  ```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 Warning error

4 Severe error

5 Critical error

6 User interrupt

UNIWHATOS (UNIX ONLY)

`uniwhatos` - Determine whether the calendar server package runs under the current operating system.

SYNTAX

```
uniwhatos

uniwhatos -v
uniwhatos -h
```

DESCRIPTION

`uniwhatos` determines whether the current operating system is the one under which the calendar server package is intended to run. If a discrepancy exists, `uniwhatos` outputs the operating system(s) the package runs under.

The calendar server installation procedure calls this utility to determine whether or not to proceed with the installation.

`uniwhatos` runs whether the calendar server is up or down.

OPTIONS

-v

Print the current version number of `uniwhatos`.

-h

Print a usage message explaining how to run `uniwhatos`.

EXAMPLES

■   An example of `uniwhatos` finding no discrepancy between the actual and
    expected operating systems:

```
% uniwhatos
HP-UX 10.x
```

■   An example of `uniwhatos` finding a discrepancy between the actual and
    expected operating systems:

```
% uniwhatos
uniwhatos: package for wrong OS installed
expect: HP-UX B.10.x
actual: HP-UX B.9.01
```

EXIT STATUS

Exit values are:

0   Success

1   Failure

2   Usage error

3   Warning error

4   Severe error

5   Critical error

6   User interrupt

UNIWHO

`uniwho` - Display information on signed-on calendar users.

SYNTAX

```
uniwho [-a] [-t] [-j] [-x] [-f] [-p <pattern>]

uniwho -v
uniwho -h
```

DESCRIPTION

This utility allows the system manager to determine who is using the calendar server, where they are signed-on from, and the process-ID associated with their session. The options allow the display of various combinations of process-id, network address, node-ID, and user information.

This information is essential in certain situations. For example, when a user has done an abnormal shutdown of a client (say a powerdown while their client is active) the associated server process for that client continues to remain active for a fixed period of time. If the calendar server is configured (via the `[ENG] max_ userlogons` parameter in `unison.ini`) to limit the number of sessions per user to 1, this user will not be able to log on again until their server process had terminated. `uniwho` allows the system manager to find the process-ID of the session and terminate it.

`uniwho` can only be run if the calendar server is up.

OPTIONS

-a

Display the alias associated with the default network address.

-f

Display telephone, job-title and X.400-address when available.

-j

Display job-title when available.

-p

<pattern>

Display information for sessions which contain <pattern> in their information. For example, the pattern "128.192.64.96" would result in the display of session information for those logged on from this IP address. Matching is performed on all fields (network address/alias, telephone number, job-title, X.400 address), regardless of which of these may have been specified on the command line.

-t

Display telephone number when available.

-x

Display X.400 address when available.

-v

Print the current version number of uniwho.

-h

Print a usage message explaining how to run uniwho.

EXAMPLES

- Display the list of all signed-on calendar server users; display the machine alias rather than the network address in the output:

```
% uniwho -a
   PID          ALIAS NODEID XITEMID  USER
  2120  ark.boat.com 12     12,2      CWSOP,na
 24091 sail.boat.com 12     12,316    Barnes,Pat,B
 24298  row.boat.com 12     12,311    Beck,Tom,V
```

- Display the list of all signed-on calendar server users in "Quality Assurance":

```
% uniwho -p Quality
   PID         ADDRESS NODEID XITEMID  USER
 24298   199.88.48.81 12     12,311    Beck,Tom,V
```

- Display full information for all of the signed-on calendar server users:

```
% uniwho -f
   PID       ADDRESS NODEID XITEMID USER
  2120  199.88.48.6 12        12,2  CWSOP,na
 24091 199.88.48.81 12      12,316  Barnes,Pat,B
                                    Engineer/R&D
                                    /barnesp@acme.com
                                    /738-1000/123
 24298 199.88.48.85 12      12,311  Beck,Tom,V
                                    Technician
                                    QualityAssurance/QA
                                    /tomb@acme.com
                                    /738-2000/015
```

EXIT STATUS

Exit values are:

0 Success

1 Failure

2 Usage error

3 User interrupt

NOTES

Logging

`uniwho` starts a calendar server process. If activity logging has been enabled (via the `[ENG] activity` parameter in `unison.ini`), the start-up and shutdown of this process is logged.

D

Time Zone Table

Setting the time zone correctly is crucial to the success of your calendar server installation. Determine applicable time zones prior to creating nodes.

> **Note:** The time zone of a node cannot be changed once that node has been created.

Table D–1   Time Zones

Country	Current Time Zone Notation
Afghanistan	UCT-4:30
Albania	MET-1METDST
Algeria	UCT-1
American Samoa	UCT11
Andorra	MET-1METDST
Angola	UCT-1
Anguilla	UCT4
Antigua and Barbuda	UCT4
Argentina	SAT3
Armenia	UCT-4
Aruba	UCT4
Australia (Lord Howe Island)	LHT-10:30LHDT
Australia (New South Wales; Capitol Territory; Victoria)	EST-10EDT

Table D–1   Time Zones

Country	Current Time Zone Notation
Australia (Northern Territory)	UCT-9:30
Australia (Queensland)	UCT-10
Australia (South Australia and Broken Hill)	CST-9:30CDT
Australia (Tasmania)	TST-10TDT
Australia (Western)	UCT-8
Austria	MEZ-1MESZ
Azerbaijan	UCT-3
Bahamas	EST5EDT
Bahrain	UCT-3
Bangladesh	UCT-6
Barbados	UCT4
Belarus	EET-2EETDST
Belgium	MET-1METDST
Belize	UCT6
Benin	UCT-1
Bermuda	AST4ADT
Bhutan	UCT-6
Bolivia	UCT4
Bonaire	UCT4
Bosnia Herzegovina	MET-1METDST
Botswana	UCT-2
Brazil (East; Including All Coast and Brasilia)	EBST3EBDT
Brazil (Fernando de Noronha)	NORO2
Brazil (Trinity of Acre)	ACRE5
Brazil (West)	WBST4WBDT
British Virgin Islands	UCT4

Table D–1 Time Zones

Country	Current Time Zone Notation
Brunei	UCT-8
Bulgaria	EET-2EETDST
Burkina Faso	UCT
Burma	UCT-6:30
Burundi	UCT-2
Cambodia	UCT-7
Cameroon	UCT-1
Canada (Atlantic)	AST4ADT
Canada (Central)	CST6CDT
Canada (Eastern)	EST5EDT
Canada (Mountain)	MST7MDT
Canada (Newfoundland)	NST3:30NDT
Canada (Pacific and Yukon)	PST8PDT
Canada (Saskatchewan)	EST5
Cape Verde	UCT1
Cayman Islands	UCT5
Central African Republic	UCT-1
Chad	UCT-1
Chile	CST4CDT
Chile (Easter Island)	EIST6EIDT
China	CST-8
Christmas Islands	UCT-7
Cocos (Keeling) Islands	UCT-6:30
Colombia	UCT5
Congo	UCT-1
Cook Islands	UCT10
Costa Rica	UCT6

Table D–1   Time Zones

Country	Current Time Zone Notation
Cote d'Ivoire	UCT
Croatia	MET-1METDST
Cuba	UCT5
Curacao	UCT4
Cyprus	EET-2EETDST
Czech Republic	MET-1METDST
Denmark	MET-1METDST
Djibouti	UCT-3
Dominica	UCT4
The Dominican Republic	UCT4
Ecuador	UCT5
Ecuador (Galapagos Islands)	UCT6
Egypt	EST-2EDT
El Salvador	UCT6
Equatorial Guinea	UCT-1
Eritrea	UCT-3
Estonia	EET-2EETDST
Ethiopia	UCT-3
Faroe Islands	WET0WETDST
Fiji	UCT-12
Finland	EET-2EETDST
France	MET-1METDST
French Guiana	SAT3
French Polynesia	UCT10
Gabon	UCT-1
The Gambia	UCT
Georgia	EUT-4EUTDST

Table D–1   Time Zones

Country	Current Time Zone Notation
Germany	MEZ-1MESZ
Ghana	UCT
Gibraltar	MET-1METDST
Greece	EET-2EETDST
Greenland (Scorsbysund)	EUT1EUTDST
Greenland (Thule)	AST4ADT
Grenada	UCT4
Guadeloupe	UCT4
Guam	UCT-10
Guatemala	UCT6
Guinea Bissau	UCT
Guyana	UCT3
Haiti	EST5EDT
Hawaii	UCT10
Honduras	UCT6
Hong Kong	UCT-8
Hungary	MET-1METDST
Iceland	UCT
India	UCT-5:30
Indonesia (Central)	UCT-8
Indonesia (East)	UCT-9
Indonesia (West)	UCT-7
Iran	UCT-3:30
Iraq	IST-3IDT
Ireland	GMT0BST
Israel	IST-2IDT
Italy	MET-1METDST

Table D–1   Time Zones

Country	Current Time Zone Notation
Jamaica	UCT5
Japan	JST
Johnston Islands	UCT10
Jordan	JST-2JDT
Juan Fernandez Islands	UCT5
Kazakhstan	EUT-6EUTDST
Kenya	UCT-3
Kiribati	UCT-12
Kuwait	UCT-3
Kyrgyzstan	UCT-5
Laos	UCT-7
Latvia	EET-2EETDST
Lebanon	EUT-2EUTDST
Leeward Islands	UCT4
Lesotho	UCT-2
Liberia	UCT
Libya	UCT-2
Liechtenstein	MET-1METDST
Lithuania	EET-2EETDST
Luxembourg	MET-1METDST
Macao	UCT-8
Macedonia	MET-1METDST
Madagascar	UCT-3
Malawi	UCT-2
Malaysia	MST-8
Maldives	UCT-5
Mali	UCT

Table D–1   Time Zones

Country	Current Time Zone Notation
Malta	MET-1METDST
Mariana Islands	UCT-10
Martinique	UCT4
Mauritania	UCT
Mauritius	UCT-4
Mayotte	UCT-3
Mexico	CST6CDT
Mexico (Baja N.)	PST8PDT
Mexico (Baja S.)	MST7MDT
Midway Islands	UCT11
Moldova	EET-2EETDST
Monaco	MET-1METDST
Mongolia	EUT-8EUTDST
Montenegro	MET-1METDST
Montserrat	UCT4
Morocco	UCT
Mozambique	UCT-2
Namibia	UCT-2
Nauru	UCT-12
Nepal	UCT-5:45
The Netherlands Antilles	UCT4
The Netherlands	MET-1METDST
New Caledonia	UCT-11
New Hebrides	UCT-11
New Zealand	NZST-12NZDT
New Zealand (Chatham Island)	CIST-12:45CIDT
Nicaragua	UCT6

Table D–1   Time Zones

Country	Current Time Zone Notation
Niger	UCT-1
Nigeria	UCT-1
Niue Islands	UCT11
Norfolk Island	UCT-11:30
North Korea	KST
Norway	MET-1METDST
Oman	UCT-4
Pakistan	UCT-5
Palau	UCT-9
Panama	UCT5
Papua New Guinea	UCT-10
Paraguay	UCT4
Peru	UCT5
Philippines	UCT-8
Pitcairn Island	UCT-9
Poland	MET-1METDST
Portugal (Azores)	EUT1EUTDST
Portugal (Madeira)	PWT0PST
Puerto Rico	UCT4
Qatar	UCT-3
Reunion	UCT-4
Romania	EET-2EETDST
Russia (Moscow)	MST-3MDT
Russian Fed. Zone 1 (Kaliningrad)	RFT-2RFTDST
Russian Fed. Zone 10 (Magadan)	RFT-11RFTDST
Russian Fed. Zone 11 (Petropavlovsk-Kamchatsky)	RFT-12RFTDST
Russian Fed. Zone 2 (St. Petersburg)	RFT-3RFTDST

Table D–1   Time Zones

Country	Current Time Zone Notation
Russian Fed. Zone 3 (Izhevsk)	RFT-4RFTDST
Russian Fed. Zone 4 (Ekaterinburg)	RFT-5RFTDST
Russian Fed. Zone 5 (Novosibirsk)	RFT-6RFTDST
Russian Fed. Zone 6 (Krasnojarsk)	RFT-7RFTDST
Russian Fed. Zone 7 (Irkutsk)	RFT-8RFTDST
Russian Fed. Zone 8 (Yakatsk)	RFT-9RFTDST
Russian Fed. Zone 9 (Vladivostok)	RFT-10RFTDST
Rwanda	UCT-2
Saint Pierre & Miquelon	NAST3NADT
San Marino	MET-1METDST
Sao Tome and Principe	UCT
Saudi Arabia	UCT-3
Senegal	UCT
Serbia	MET-1METDST
The Seychelles	UCT-4
Sierra Leone	UCT
Singapore	UCT-8
Slovakia	MET-1METDST
Slovenia	MET-1METDST
Solomon Islands	UCT-11
Somalia	UCT-3
South Africa	SAST-2
South Georgia	UCT3
South Korea	KST
Spain	MET-1METDST
Spain (Canary Islands)	WET0WETDST
Sri Lanka	UCT-5:30

Table D–1   Time Zones

Country	Current Time Zone Notation
St. Helena	UCT
St. Kitts-Nevis	UCT4
St. Lucia	UCT4
St. Vincent and the Grenadines	UCT4
Sudan	UCT-2
Suriname	UCT3
Swaziland	UCT-2
Sweden	MET-1METDST
Switzerland	MEZ-1MESZ
Syria	SST-2SDT
Tahiti	UCT10
Taiwan	UCT-8
Tajikistan	UCT-5
Tanzania	UCT-3
Thailand	UCT-7
Togo	UCT
Tonga	UCT-13
Trinidad and Tobago	TTST4
Tunisia	UCT-1
Turkey	EET-2EETDST
Turkmenistan	UCT-5
Turks & Caicos Islands	EST5EDT
Tuvalu	UCT-12
Uganda	UCT-3
Ukraine	EET-2EETDST
Ukraine (Simferopol)	EUT-3EUTDST
United Arab Emirates	UAEST-4

Table D–1   Time Zones

Country	Current Time Zone Notation
United Kingdom	GMT0BST
Uruguay	SAT3
US Virgin Islands	UCT4
USA (Alaska)	NAST9NADT
USA (Aleutian Islands)	AST10ADT
USA (Arizona)	MST7
USA (Central)	CST6CDT
USA (Eastern)	EST5EDT
USA (Indiana)	EST5
USA (Mountain)	MST7MDT
USA (Pacific)	PST8PDT
Uzbekistan	UCT-5
Vanuatu	UCT-11
Vatican City	MET-1METDST
Venezuela	UCT4
Vietnam	UCT-7
Wake Islands	UCT-12
Wallis & Futana Islands	UCT-12
Western Samoa	UCT11
Windward Islands	UCT4
Yemen	UCT-3
Zaire (Kasai)	UCT-2
Zaire (Kinshasa)	UCT-1
Zambia	UCT-2
Zimbabwe	UCT-2

E

Extensions to Directory Server Schema

This appendix presents the calendar server's extensions to the LDAP directory server schema. Consult your directory server documentation for information on the rest of your directory server schema.

- Object class extensions
- Default mappings for attribute names

Object class extensions

There are three object class extensions to the directory server schema. Each directory server entry should contain an instance of only one of these object classes. Each class is mutually exclusive with each of the other classes.

Table E–1   Calendar server object classes

Object Class	Description
ctCalUser	The object class for calendar server users. Note that a calendar server user entry is usually added to an existing user entry in the directory server.
ctCalAdmin	The object class for calendar server reserved users.
ctCalResource	The object class for calendar server resources.

Calendar server object classes

The following tables present the ctCalUser, ctCalAdmin, and ctCalResource object classes respectively.

Each calendar server object class is composed of attributes specific to that class, and attributes inherited from superior classes. All attributes specific to a calendar server

object class have the prefix "ctCal" and are of type "case ignore string". See "Attribute definitions" on page E-6 for descriptions of each of the attributes.

ctCalUser object class

Requires:

- objectClass

Allows:

- c
- ctCalAccess
- ctCalAccessDomain
- ctCalAdmd
- ctCalCountry
- ctCalDefaultNoteReminder
- ctCalDefaultReminder
- ctCalDefaultTaskReminder
- ctCalDisplayPrefs
- ctCalFlags
- ctCalHost
- ctCalLanguageId
- ctCalMobileTelephoneType
- ctCalNodeAlias
- ctCalNotifMechanism
- ctCalOperatingPrefs
- ctCalOrganization
- ctCalOrgUnit1
- ctCalOrgUnit2
- ctCalOrgUnit3
- ctCalOrgUnit4

- ctCalPasswordRequired
- ctCalPreferredSMSCTelephoneNumber
- ctCalPrmd
- ctCalPublishedType
- ctCalRefreshPrefs
- ctCalServerVersion
- ctCalSMSTimeRange
- ctCalSysopCanWritePassword
- ctCalTimezone
- ctCalXItemId
- employeeNumber
- generationQualifier
- givenName
- initials
- mail
- title
- uid

ctCalAdmin object class

Requires:
- objectClass
- ctCalXItemId

Allows:
- c
- cn
- ctCalAccess
- ctCalAccessDomain

- ctCalAdmd
- ctCalCountry
- ctCalFlags
- ctCalHost
- ctCalLanguageId
- ctCalNodeAlias
- ctCalOrganization
- ctCalOrgUnit1
- ctCalOrgUnit2
- ctCalOrgUnit3
- ctCalOrgUnit4
- ctCalPasswordRequired
- ctCalPrmd
- ctCalServerVersion
- ctCalSysopCanWritePassword
- ctCalXItemId
- facsimileTelephoneNumber
- generationQualifier
- givenName
- initials
- mail
- o
- ou
- postalAddress
- sn
- telephoneNumber
- userPassword

ctCalResource object class

Requires:

- objectClass

- cn

Allows:

- ctCalAccess

- ctCalAccessDomain

- ctCalDefaultNoteReminder

- ctCalDefaultReminder

- ctCalDefaultTaskReminder

- ctCalDisplayPrefs

- ctCalFlags

- ctCalHost

- ctCalLanguageId

- ctCalNodeAlias

- ctCalNotifMechanism

- ctCalOperatingPrefs

- ctCalPasswordRequired

- ctCalRefreshPrefs

- ctCalResourceCapacity

- ctCalResourceNumber

- ctCalServerVersion

- ctCalSysopCanWritePassword

- ctCalTimezone

- ctCalXItemId

- facsimileTelephoneNumber

- givenName

- mail

- postalAddress

- sn

- telephoneNumber

- userPassword

Attribute definitions

The following two tables provide a description of all attributes associated with the calendar server object classes. The first describes attributes specific to the calendar server object classes, and the second describes attributes inherited from superior classes. Note that in the case of inherited attributes, the attribute name may vary with the directory server.

Table E–2   ctCal* attribute definitions

Attribute	Description
ctCalAccess	Allow/deny access of calendar server user.
ctCalAccessDomain	Internet domain or IP from which the calendar server user is allowed to access their calendar server data. For future use.
ctCalAdmd	X.400 Administration Management Domain Name (A).
ctCalCountry	Country. Not currently in use.
ctCalDefaultNoteReminder	*Type*: Visual, audible, mail, all or none. *Len*: Number of minutes before the event for reminder.
ctCalDefaultReminder	*Type*: Visual, audible, mail, all or none. *Len*: Number of minutes before the event for reminder.
ctCalDefaultTaskReminder	*Type*: Visual, audible, mail, all or none. *Len*: Number of minutes before the event for reminder.

Table E–2   ctCal* attribute definitions

Attribute	Description
ctCalDisplayPrefs	Display preferences.
	StartDay: Time in minutes to start day display. *EndDay*: Time in minutes to end day display. *WeekStart*: Specifies the first day of the week (i.e. usually Sunday or Monday). *TimeIncrement*: Time increment in minutes for display. *ActiveDays*: Specifies days to display (e.g. week days only). *TimeFormat*: Specifies time format (AM/PM or 24h) for display.
ctCalFlags	For future use.
ctCalHost	Host name, or IP address in dotted notation, of the computer hosting the calendar server user's data.
ctCalLanguageId	Preferred language for incoming mail notification. For future use.
ctCalNodeAlias	Mnemonic name of the node on which the calendar server user's data is stored.
ctCalNotifMechanism	Specifies mechanism used to notify attendees (usually mail).
ctCalOperatingPrefs	Operating preferences.
ctCalOrganization	User's organization. Not currently in use; "o" attribute is used for this information.
ctCalOrgUnit1	X.400 Organizational Unit 1(OU1).
	User's organization. Not currently in use; "ou" attribute is used for this information.
ctCalOrgUnit2	X.400 Organizational Unit 2 (OU2).
ctCalOrgUnit3	X.400 Organizational Unit 3 (OU3).
ctCalOrgUnit4	X.400 Organizational Unit 4 (OU4).
ctCalPasswordRequired	Specifies if the user must provide a password to have access to his calendar data.
ctCalPrmd	X.400 Private Management Domain Name (P).

Table E–2   ctCal* attribute definitions

Attribute	Description
ctCalPublishedType	Calendar type: Not Published, Published, Event Calendar
ctCalRefreshPrefs	Refresh preferences; *State*: on ∕ off *Frequency*: Interval in minutes between refreshes.
ctCalResourceCapacity	Capacity of resource.
ctCalResourceNumber	Identification number of resource.
ctCalServerVersion	Version number of the calendar server hosting the user's data.
ctCalSMSTimeRange	Time range during which the user does not want to receive SMS messages. Values are expressed in minutes of the day, separated by a colon, e.g. 9 PM to 9 AM is expressed as "1260:540".
ctCalSysopCanWritePassword	Specifies if the calendar server SYSOP can overwrite the user's password.
ctCalTimezone	Current time zone of the user.
ctCalXItemId	Identification number of the node on which the calendar server user's data is stored as well as the identification number of the calendar server user's item.
ctCalMobileTelephoneType	Mobile telephone type.
ctCalPreferredSMSCTelephoneNumber	Preferred SMSC telephone number.

Table E–3   Inherited attribute definitions

Attribute name	Class	Description
c	n∕a	Country.
cn	person	Common name of the administrator or resource.
employeeNumber	inetOrgPerson	Employee number of the user.
facsimileTelephoneNumber or fax	organizationalPerson	FAX phone number of the administrator or resource.

Table E–3   Inherited attribute definitions

Attribute name	Class	Description
generationQualifier or gq	n/a	Generation qualifier.
givenName or gn	inetOrgPerson	Given name of the user. In the case of a resource, this is the given name of the contact.
initials	inetOrgPerson	User's initials.
mail or rfc822MailBox	inetOrgPerson	Email address.
mobile	inetOrgPerson	Mobile telephone number.
o	n/a	Organization of the user.
ou	organizationalPerson	Organizational unit of the user.
postalAddress	organizationalPerson	Mailing address of the administrator or resource.
sn or surname	person	Surname of the administrator or resource.
telephoneNumber	person	Telephone number of the administrator or resource.
title	organizationalPerson	Job title of the user.
uid	inetOrgPerson	User identification number.
userPassword	person	Password with which the administrator or resource binds to the directory server.

Default mappings for attribute names

You may choose to change the default names your calendar server uses for certain attributes to ensure these map properly into your directory server schema.

You change these attribute names through configuration parameters contained in the /users/unison/misc/unison.ini file. These parameters are listed here, along with their default values.

> **Warning:** Do not change the value of the attr_uid parameter unless you have changed the attribute used by your Oracle Internet Directory server for SSO login. See the calendar server *Administrator's Guide*, Chapter 3, "Administration," for details.

Table E–4   Configuration parameters for LDAP attribute names

Configuration parameter	Default value
attr_accessdomain	"ctCalAccessDomain"
attr_access	"ctCalAccess"
attr_address	"postalAddress"
attr_admindomain	"ctCalAdmd"
attr_capacity	"ctCalResourceCapacity"
attr_commonname	"cn"
attr_country	" "
attr_defaultnotereminder	"ctCalDefaultNoteReminder"
attr_defaultreminder	"ctCalDefaultReminder"
attr_defaulttaskreminder	"ctCalDefaultTaskReminder"
attr_displayprefs	"ctCalDisplayPrefs"
attr_employeeid	"employeeNumber"
attr_fax	"facsimileTelephoneNumber"
attr_flags	"ctCalFlags"
attr_generation	"generationQualifier"
attr_givenname	"givenName"
attr_groupname	"cn"
attr_ host	"ctCalHost"
attr_initials	"initials"
attr_jobtitle	"title"
attr_langid	"ctCalLanguageId"
attr_mail	"mail"

Table E–4   Configuration parameters for LDAP attribute names

Configuration parameter	Default value
attr_member	"member"
attr_mobile	"mobile"
attr_mobiletype	"ctCalMobileTelephoneType"
attr_nodealias	"ctCalNodeAlias"
attr_notifmechanism	"ctCalNotifMechanism"
attr_objclass	"objectClass"
attr_operatingprefs	"ctCalOperatingPrefs"
attr_organization	""
attr_orgunit1	"ou"
attr_orgunit2	"ctCalOrgUnit2"
attr_orgunit3	"ctCalOrgUnit3"
attr_orgunit4	"ctCalOrgUnit4"
attr_passwordrequired	"ctCalPasswordRequired"
attr_password	"userPassword"
attr_phone	"telephoneNumber"
attr_privmdomain	"ctCalPrmd"
attr_publishedtype	"ctCalPublishedType"
attr_refreshprefs	"ctCalRefreshPrefs"
attr_resourcename	"cn"
attr_resourcenumber	"ctCalResourceNumber"
attr_serverversion	"ctCalServerVersion"
attr_smscpref	"ctCalPreferredSMSCTelephoneNumber"
attr_surname	"sn"
attr_sysopcanwritepassword	"ctCalSysopCanWritePassword"
attr_timezone	"ctCalTimezone"
attr_uid	"uid"
attr_uniquemember	"uniquemember"

Table E–4   Configuration parameters for LDAP attribute names

Configuration parameter	Default value
attr_version	" "
attr_xitemid	"ctCalXItemId"

Index