



The GMS Reference Guide

Third Edition

Gordano Ltd

The GMS Reference Guide

Copyright © Gordano Ltd, 1995-2015. All rights reserved. Printed in the United Kingdom.

Published by Gordano Ltd,
Unit 1, Yeo Bank Business Park, Kenn Road, Clevedon, North Somerset, BS21 6UW.

Printing History:

Oct 2002 First Edition

May 2003 Second Edition

July 2015 Third Edition

ISBN

GMS, Gordano and Gordano Ltd and their logos are trademarks of Gordano Ltd.

Many of the designations used by manufacturers and sellers to distribute their products are claimed as trademarks. Where those designations appear in this book, and Gordano Ltd was aware of a trademark claim, the designations have been printed in capitals or initial capitals

Written by Brian Dorricott, John Stanners, Dean Fenton, Jason Hall and Dean Packer.

Copyright © Gordano Ltd, 1995-2015

GMS

WARNING: YOU SHOULD CAREFULLY READ THE LICENCE AGREEMENT PROVIDED WITH THIS MANUAL BEFORE USING THIS SOFTWARE PACKAGE. INSTALLING THE SOFTWARE ONTO YOUR COMPUTER INDICATES YOUR ACCEPTANCE OF THESE TERMS AND CONDITIONS. IF YOU DO NOT WISH TO ACCEPT ALL OF THESE TERMS, YOU SHOULD STOP INSTALLING THIS SOFTWARE NOW AND DESTROY ALL COPIES OF THE SOFTWARE AND ALL MANUALS AND OTHER DOCUMENTS SUPPLIED WITH IT.

NTMail is a registered trademark of Gordano Ltd.

The Gordano Logo is a registered trademark of Gordano Ltd.

Juce is a registered trademark of Gordano Ltd.

NT is a registered trademark of Northern Telecom Ltd.

Windows NT is a trademark of Microsoft Corporation in the USA and other countries.

All other trademarks are acknowledged.

Patents

Gordano owns a number of patents on its software as listed below:

Autoport

Gordano's "Autoport" technology is patented in the United Kingdom under patent number GB2391649.

A patent application has been filed in the United States and is pending approval.

Maintaining software and data (Automatic Updates)

Gordano's "Maintaining software and data" technology is patented in the United Kingdom under patent number GB2374163.

A patent application has been filed in the United States and is pending approval.

Anti-spam filter (Sender Verification)

Gordano's "Anti-spam Filter" technology is patented in the United Kingdom under patent number GB2385965 and in the United States under patent number 7574476.

Transitory E-mail Addresses

Gordano's "Transitory E-mail Address" technology is patented in the United Kingdom under patent number GB2398399.

Table of Contents

1	Introduction	1
1.1	Who Should Read this Guide?	1
1.2	Other Guides from Gordano	2
1.3	Conventions	2
2	Compliance with RFCs	3
2.1	RFCs	3
2.2	RFC Override Options	3
2.3	RFC Reference List	4
3	Tools	7
3.1	Mail.exe	8
3.2	Mail.exe Parameters	9
	Append	10
	CNAME	11
	Domain	11
	Dump DNS Cache	11
	End of line	11
	Extract List	12
	Fix Addresses	12
	Help	13
	IM	13
	Index	13
	Insert List	13
	Kick	14
	List of Users	14
	Load	14
	Load Share	15
	Logon	16
	Max Wait	16
	MX Lookup	16
	Post	17
	Purge DNS Cache	17
	Query Server	18
	Run Script	18
	Sleep	19
	Sortlist	19
	Start Gordano services	19
	Stop Gordano services	19
	Transfer	19
	Update (Reread Registry)	20
	Verbose	20
	Version	20
	Wait	20
	Watch	21
	Dump Current Configuration	21
3.3	Mail.exe Reported Error Codes	23

3.4	Load Sharing	24
	Step By Step Guide to Setting Up Load Sharing	26
3.5	AddUser.exe	27
	Add Alias	28
	Add Domain	28
	Add User	28
	Delete Alias	29
	Delete Domain	29
	Delete User	29
	Set Variable	29
	Get Variable	30
	Interactive Mode	30
	Batch Mode	31
3.6	Admin Applet	31
	Adding an Administrator	31
	Admin Server port	32
3.7	Gordano Inspector	33
	Introduction	33
	Getting started	33
	The Action menu	34
	The View menu	35
	The Help menu	35
3.8	Mailbox Monitor	36
3.9	Performance Monitor (Perfmon)	37
	Operation	37
	SMTP parameters	40
	POST parameters	42
	POP parameters	43
3.10	Regedt32	45
3.11	Ping and Traceroute	46
	Ping	46
	Traceroute (Tracert)	47
3.12	Nslookup	48
3.13	Finger	48
4	Configuration — The Registry	49
4.1	Structure	49
	Registry structure (Windows)	49
	Configuration Database structure (Unix)	50
	Under both structures there are three sets of definitions -	
	The global parameters, domain specific parameters and	
	user parameters.	50
	Global, domain-specific and user parameters	50
	How values are stored in the Registry or Configuration	
	database	51
	Details used for NT SAM users	51
	Warnings	51
	Entering IP addresses	51
4.2	The parameters and their meanings	53
	Global parameters	53

Domain-specific parameters	97
User parameters	106
Private parameters	113
System Parameters	116
5 GMS Communication Server Configuration Database Parameters	119
5.1 Introduction	119
5.2 GMS Communication Server Parameters.....	121
Membership file format	121
List parameters	122
Values of Access parameters	127
Global parameters	146
User parameters	147
5.3 Parameter Changes Between Versions 3 and 6	151
6 Registry Parameters GMS WebMail Professional	153
Global Variables	153
Domain Variables	155
User Variables.....	157
7 Registry Parameters GMS Mobile.	167
User Variables.....	167
8 Collaboration Server Registry Parameters	169
8.1 Introduction	169
8.2 Parameters	170
Global Parameters.....	170
Domain Parameters.....	172
User Parameters	172
9 Registry Parameters GMS WAPMail	173
Global/Domain Variables.....	173
User Variables.....	174
10 GMS Anti-Spam Registry Parameters	175
10.1 Introduction	175
10.2 Parameters.....	176
Global parameters	176
Domain-specific parameters	182
11 Virus Scanner Registry Parameters	185
11.1 Introduction	185
11.2 Parameters	186
Global Parameters.....	186
Domain Parameters.....	187
12 Registry Parameters GMS Windows IM Client.	189
Variables.....	189

13 Directory Structure	.191
13.1 Global Directories	.191
Backup	.193
Bin	.193
ListIn	.193
ListMan	.193
Log	.193
MesLog	.193
MML	.193
MySQL	.193
Out	.193
Profiles	.194
SpamMes	.194
Temp	.194
13.2 Domain Directory Structure	.194
BadMes	.195
MesLog	.195
Users	.195
WWW	.195
13.3 Profiles Directory Structure	.195
13.4 Changing the Directory Structure	.196
14 Services	.197
14.1 Interactions between Services	.197
14.2 ESMTP (Enhanced SMTP)	.198
8BitMIME	.198
Auth	.198
Delivery Status Notification (DSN)	.198
Enhanced Status Codes	.199
ETRN	.199
Pipelining	.200
Restart	.200
Size	.201
VRFY	.201
XTND	.201
14.3 POST, POP, SMTP, IMAP, LIST and WWW Switches	.203
15 Customising the User Interface	.205
15.1 Background on Page Structure	.205
15.2 Replacing Pages	.205
16 Redirect, Template and Postservers Files	.207
16.1 The Redirect File	.208
Syntax	.208
Operation	.210
16.2 Postservers.txt	.211
16.3 Template File Format	.212
17 SharePoint	.215
17.1 Installation	.215

17.2	Configuration.....	215
17.3	Calendar Specific Properties	217
18	Gizmos	219
18.1	What are Gizmos	222
18.2	Gizmo Functionality	222
	Naming Conventions	222
	Display Name	223
	Description	223
	External Scripts	223
	Administration Gizmos	223
	Access Rights	223
	Javascript File	223
	HTTP Object	224
	Installation	224
18.3	Object Methods	225
18.4	Helper Methods	226
	General APIs	226
	Current Selection Values	227
	Menu APIs	228
	Panel APIs	229
	Tree APIs	229
	Content APIs	230
	Window APIs	230
	Ajax Requests	230
	Greasemonkey helpers	231
	jQuery	231
18.5	Sharing Gizmos	232
19	Themes	233
19.1	What is a Theme	234
19.2	Adding Custom Themes	235
19.3	Sharing Themes	236
20	Robots	237
20.1	Domain Robots	237
20.2	User Robots	237
20.3	Accepting the E-mail Message	237
20.4	Sending an E-mail Message	237
20.5	Example Code	238
21	DLLs	241
21.1	Why use a DLL?	241
21.2	Available DLLs	241
	Transmission DLLs	241
	SMTP DLLs	248
	User Authentication DLLs	256
	Extenal Folder Manager (EFM) DLL	260
	Message parsing and log DLLs	270

22 Using ColdFusion Tags	.273
22.1 Example ColdFusion Routines	.274
Logging on as an administrator	.274
Adding a user	.274
Changing the user's attributes	.275
Displaying the user's attributes	.276
Enumerating users in a domain	.277
Deleting a user	.278
Add a domain	.279
23 Gordano Accessory Pack	.281
23.1 Gizmos	.281
23.2 Mailbox Inspector	.281
23.3 NTMetrn Utility	.282
23.4 MML Examples and Documentation	.282
Delivery Scripts	.282
Timed Events	.282
GUI	.282
23.5 Robots	.283
Archive	.283
Detach	.284
Domain (free)	.284
eSarah	.285
Filter	.285
Rotate (free)	.285
23.6 Full to Virtual Domain Conversion	.285
Command Line Parameters	.285
23.7 Upgrade V3 Mailboxes	.285
23.8 Example SMTP DLL	.286
23.9 Example EFM DLL	.286
23.10 Virtual to Full Domain Conversion	.286
Command Line Parameters	.286
23.11 COM DLL	.286
General Functions	.287
Authentication Functions	.287
User Functions	.288
Alias Functions	.289
Domain Functions	.289
Global Variable Functions	.291
Message Queue Functions	.291
Message Functions	.291
Example for Message Functions	.293
23.12 Monitoring Software	.293
23.13 WatchMail	.294
Install	.294
Uninstall	.294
The Output	.294
SECURITY WARNING!	.295
23.14 GMS Watch	.295

24 Porting/Migration Tools	297
24.1 Exchange	297
24.2 Linux/Solaris	297
SendMail	297
Unix	298
25 Log Contents	301
25.1 Log Levels	301
25.2 Log file contents	301
25.3 Log File names	303
25.4 Logging Bit Map	304
25.5 Stats Log Contents	305
26 Error Messages	307
26.1 DNS Resolver Errors	308
26.2 Error Codes in Logs	309
PT Log codes	309
Three-digit SMTP error codes in logs	311
Enhanced SMTP error codes	312
26.3 Four-digit SMTP error codes	314
27 Security Check	317
27.1 Disallowing Relay	317
27.2 LocalIP	317
27.3 Local Clients	318
27.4 UnknownUserAction	318
27.5 Disable VRFY	318
27.6 Restrict Access to Config GUI	319
27.7 Password Policies	319
Index	321
Licence Agreements	333
Installation and Contact Information	343

1 Introduction

This guide is the Reference Guide for the Gordano Messaging Suite (GMS) of products on the Windows operating system. It has these sections:

- Compliance with RFCs — lists the protocols (Request for Comments documents — RFCs) that GMS Mail is based upon. The cases where GMS Mail can override an element of a protocol's definition are also given. The relevant RFCs are listed.
- Gordano Tools — describes utilities provided with GMS. The main tools covered are Mail.exe, Mailbox Inspector, Mailbox Monitor and the Performance Monitor. There is a brief section on using the Registry Editor. The diagnostic tools Ping, Traceroute, Nslookup and Finger are also described.
- Configuration — the Registry or Configuration file, depending upon the operating system you are using. This section gives detailed information on the general parameters Gordano products may use. This lets you set up those parameters which the user interface does not give access to, as well as detailing the other parameters. Parameters comprise three groups — global, domain-specific and user.
- Directory Structure — describes the directories the Gordano Installation creates and uses.
- Services — describes the features of Enhanced SMTP (ESMTP) that the products support. It also describes the switches used by POST, POP and SMTP, IMAP and LIST.
- Customising the User Interface — explains how to alter Gordano's Web Interface. You can edit the header and footer of each page, and also the first page and support pages.
- Robots — gives background on robots (executables) and an example program.
- DLLs (Windows)— gives listings of DLLs which are available, including user authentication DLLs and transmission DLLs.
- Shared Libraries (UNIX) — gives listings of shared libraries which are available, including user authentication shared libraries and transmission shared libraries.
- Log Contents — explains how to interpret the Gordano logs.
- Error Messages — describes SMTP and enhanced SMTP codes.

1.1 Who Should Read this Guide?

This guide is for administrators using any GMS products.

1.2 Other Guides from Gordano

The following guides provide additional information:

- *GMS Administrator's Guide* — describes how to install and use GMS.
- *GMS Communication Server* — describes GMS Communication Server, which manages lists of e-mail addresses.
- *MML Programmer's Guide* — describes the Mail Meta Language (MML) which you can use to modify GMS operation.
- *Gordano Users Guide* — Gives information regarding the use of GMS products.

1.3 Conventions

The following conventions are used in this guide:

Convention	Used for
Courier	Lines of code and e-mail messages.
<i>Italic</i>	Names of other guides, products and services.
<value>	Reference to information you must provide.
Option>page	The ">" abbreviates a sequence of actions. For example, "choose Content>Global Filters" means click the Content button on the toolbar, then choose the Global Filters page.

The following symbols are used in this guide:



Tip — gives optional extra information you may want to act on. You can ignore these if you wish.



Information — gives additional explanation of points. You should read these.



Warning — warns of areas where you could damage some element of your system. You must read these.

2 Compliance with RFCs

This section lists the Request for Comments (RFC) open protocols that Gordano Products are based upon. These are published by the Internet Engineering Task Force (IETF) and are freely available on the Internet. Some of these documents are available on the Gordano Ltd. Web site.

2.1 RFCs

Gordano's application of protocols complies with the following RFCs:

- SMTP — RFCs 821, 822, 1652, 1846, 1854, 1869, 1870, 1891, 1893, 2034 and 2554.
- ESMTP — RFC 1894.
- POP — RFCs 1734 and 1939.
- IMAP — RFCs 1730 and 2060.
- MIME — RFCs 1741, 2045 and 2046.
- Other RFCs — 1123, 1321 and 2142.

A detailed list of these RFCs is given at the end of this section.

2.2 RFC Override Options

There are some cases where Gordano Products can override an element of an RFCs definition, if you want it to do this. The following options can be enabled/disabled using the System>Compliance page in the configuration interface:

- Enforce RFC822 header.
- Enforce requirement for CRLF at end of line (the default is on).
- Enforce RFC1894 Delivery Status Notification (DSN).
- Allow RFC1123 returning of mail (the default setting is on).
- Correction of SMTP From: & To: clauses.
- MSMail Address fix.
- POP3 DELE works immediately (this contradicts RFC 1939).
- Show the real host name in the SMTP Received clause.
- SMTP resolve hostname.

2.3 RFC Reference List

The following table gives details of some of the main RFCs covering components of e-mail. You might want to refer to these:

RFC	Standard
-----	----------

821	<i>Simple Mail Transfer Protocol</i> , J. Postel, 1982
822	<i>Standard for the Format of ARPA Internet Text Messages</i> , D. Crocker, 1982
974	<i>Mail Routing and the Domain Name System</i> , C. Partridge, 1986
1035	<i>Domain Names Implementation and Specification</i> , P. Mockupetris, 1981
1123	<i>Requirements for Internet Hosts - Application and Support</i> , R. Barden, 1989
1321	<i>The MD5 Message-Digest Algorithm</i> , R. Rivest, 1992
1521	<i>Multipurpose Internet Mail Extensions</i> , N. Freed, 1993
1652	<i>SMTP Service Extension for 8bit-MIME transport</i> , J. Klensin, WG Chair, 1994
1730	<i>Internet Message Access Protocol - Version 4</i> , M. Crispin, 1994
1734	<i>POP3 AUTHentication command</i> , J. Myers, 1994
1741	<i>MIME Content Type for BinHex Encoded Files</i> , P. Faltstrom, 1994
1845	<i>SMTP Service Extension for Checkpoint/Restart</i>
1846	<i>SMTP 521 Reply Code</i> , A. Durand, F. Dupont, September 1995
1854	<i>SMTP Service Extension for Command Pipelining</i> , A Cargille and WG Chair, 1995
1869	<i>SMTP Service Extensions</i> J. Klensin, WG Chair, 1995
1870	<i>SMTP Service Extension for Message Size Declaration</i> , J. Klensin, WG Chair, 1995
1891	<i>SMTP Service Extension for Delivery Status Notification</i> . K.Moore, 1996
1893	<i>Enhanced Mail System Status Codes</i> , G. Vaudreuil, 1996
1894	<i>An Extensible Message Format for Delivery Status Notifications</i> , K. Moore & G. Vaudreuil, 1996
1939	<i>Post Office Protocol - Version 3</i> , May 1996
1985	<i>SMTP Service Extension for Remote Message Queue Starting</i>
2034	<i>SMTP Service Extension for Returning Enhanced Error Codes</i> , N.Freed, 1996
2045	<i>Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies</i> , N. Borenstein & N. Freed, 1996
2046	<i>Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types</i> , N. Freed, N. Borenstein, 1996

RFC Standard

- 2060 *Internet Message Access Protocol - Version 4 Revision 1*, M. Crispin, 1996
- 2142 *Mailbox Names for Common Services, Roles And Functions*, D.Crocker, 1997
- 2197 SMTP Service Extension for Command Pipelining
- 2505 Anti-Spam Recommendations for SMTP MTAs
- 2554 *SMTP Service Extension for Authentication*, J.Myers, 1999

3 Tools

This section describes the Gordano tools, as follows:

- Mail.exe — the general purpose configuration and diagnostic program for Gordano products.
- Mailbox Inspector — this maintenance tool shows the contents of domains and mailboxes. You can use it to rename mailboxes, delete messages from them, etc.
- Mailbox Monitor — monitors the allocation of threads and the sending and receipt of e-mail. It gives a dynamic view of the contents of the log files.
- Perfmon — the performance monitor. This shows the activity of each service.
- Regedt32 — the Registry editor, used to set Gordano product parameters.
- Ping and Traceroute — used to check connectivity and routing between hosts.
- Nslookup — lets you look up MX records when you have a problem.
- Finger — lets you obtain information on users whose e-mail address you know.



Several of these tools are run from the command prompt. To open this, choose Start, Program Files, Command Prompt. In the window which this opens, type the name of the relevant command, followed by any options.

3.1 Mail.exe

Mail.exe is a general purpose configuration and diagnostic program for Gordano products. It can also be used in conjunction with a script file to perform updates in batch fashion. It has many command line switches and several diagnostic options, a mail-to option, options to make services re-read the Registry or load the Registry from a script file, and lastly some routines for helping with dial-up connections.

Mail.exe will accept a sequence of commands and work through them in order. However, if an error occurs it exits at that point.

This table summarises the command line switches (insert a space between any switch and its parameter):

Switch	Description
-append <filename>	Appends POP entries from a file to the NT Registry or UNIX configuration files.
-cname <host>	Looks up C Name and returns the named host's IP address.
-domain <domain>	Specifies domain for use with the append, extractlist, listusers, insertlist, loadshare, load and sortlist options.
-dumpdns<file>	Dumps contents of DNS Cache to file.
-eol	Specifies line endings to be used with '-post' option.
-extractlist [<list>][,<file>][,<order>]	Extracts members of the named list
-fixaddress	Corrects addresses which are not RFC-compliant. Used with '-post'.
-help <switch>	Prints help on named switch or lists all switches.
-im	Sends an Instant Message to a specified account
-index <mailbox>	Rebuilds an index file for an existing mailbox.
-insertlist <list>,<file>	Inserts e-mail addresses from file into list.
-kick	Kicks the POST service, forcing any waiting mail to be sent.
-listusers <filename>	Writes a list of all the users in the Registry to the specified file.
-load <file>	Loads the contents of a POP script into the Registry.
-loadshare	Creates a load sharing array.

Switch	Description
-logon	Set the user name and password for commands that require them.
-maxwait <n seconds>	Sets maximum wait time for any operation.
-mx <host>	Attempts to find the MX records for the named host.
-post <file, mail address, host>	Mails the file to the e-mail address, using the specified host as mail server.
-purgedns	Purges the DNS Cache
-query <host>,<password>	Queries the named host for status information.
-runscript <script>	Runs an MML script.
-sleep <time>	Stops Mail.exe for <time> seconds. Usually used with another switch.
-sortlist	Specifies the order the list is sorted.
-start	Starts all Gordano services.
-stop	Stops all Gordano services.
-transfer	Specify a user to which vCards will be transfered when deleting accounts.
-update <service>	Update — forces one or all Gordano services to re-read the Registry.
-verbose	Verbose — outputs the starting time of each operation.
-version	Prints Mail.exe version number.
-wait <service>	Forces Mail.exe to wait for the named service to become free.
-watch{t s i r}	Displays statistics on mail services.
-y	Dumps the current Gordano configuration to a text file called Setup.txt in the current directory, or with the -yr switch restores earlier Setup.txt file. (See page 21)

3.2 Mail.exe Parameters

The following sections describe each of the switches listed above.



Some commands will request an administrator name and password to be entered before they can be processed.

Append

Tells Mail.exe to append the POP entries in the given file to those in the Registry. Mail.exe assumes that user names are unique — if it finds a second user in the file with the same name, the old one is overwritten. Completely new entries from the file are added, but if there is an identical entry in the file and the Registry, the Registry entry is not replaced.

Entries in the file to be appended should take this form:

```
mailbox username [-[password]]
```

where *mailbox* is the name of the file that the user's mail is stored in. Normally this would be called *inbox.mbx*, so the line would look like this:

```
inbox.mbx username password
```

There are three ways to define passwords:

- `-<password>` — the leading hyphen means any existing password will be replaced by the new password.
- `<password>` — If there is no leading hyphen and the user already has a password but has changed it (using the Password Server), their new password will not be overwritten.
- `"-"` — entering just a hyphen means use the same password as in the NT User Database.

Syntax

```
mail -append <filename>
```

Example

```
C:\mail>type user.txt
inbox.mbx username1 pwd
inbox.mbx username2 pwd
```

This should produce the following output, when *username1* already exists:

```
C:\Gordano\bin\mail -domain domain.dom -append user.txt
Reading details from file...
About to update Registry with new values...
username = postmaster, domainId = 1
username = username1, domainId = 1
/

Update completed...
New users:    1
Changed:     1
Unchanged:   0
Removed:     0
```

CNAME

Looks up the CName of the specified host and returns its IP address.

Syntax

mail -c<hostname>

Examples

C:\Gordano\bin\mail -cmail.test.dom

Official name: mail.test.dom (194.70.57.66)

Domain

Specifies the domain for use with the '-append', '-extractlist', '-listusers', '-insertlist', '-loadshare', '-load' and '-sortlist' options.

Syntax

mail -domain <domain>

Example

This loads the users from the file test.txt into the domain domain.dom:

```
C:\mail> mail -domain domain.dom -insertlist test.txt
```

Dump DNS Cache

Dumps the contents of the DNS Cache to the file specified. Use the -v option as well to produce a verbose dump.

Syntax

mail -dumpdns <file>

End of line

Forces the use of one of the following line endings with the '-post' option:

- C — use CR line endings.
- L — use LF line endings.

If no parameters are passed to -eol, the default setting of CRLF is used.

Syntax

mail -eol C -post...

Extract List

Extracts the e-mail addresses of people who have joined the named list and writes this information either to the specified file or, if "-" is used instead of a file name, to the console.

The information produced can be edited then reloaded using the -insertlist command.

Syntax

```
mail -domain domain.dom -extractlist <listname> [<filename> [_sortlist  
<order>]]
```

Example

To display in a file, called names.txt, the contents of list example_list, sorted by domain name, you would use this command:

```
C:\Gordano\bin\mail -domain domain.dom -extractlist example_list  
names.txt -sortlist D
```



For details of -sortlist see "Sortlist" on page 19.

Fix Addresses

Automatically fixes any badly formatted addresses, making them RFC-compliant. This option is used with the '-post' option.

Syntax

```
mail -fixaddress
```

Example

```
C:\Gordano\bin\mail -fixaddress -post filename,to...
```

Help

Displays help. Enter **mail -h** to list all the commands that mail.exe will accept. If you want more details about one switch, enter its letter after the h.

Syntax

mail -help [<switch>]

Example

C:\Gordano\Bin>mail -help post

Command: -post filename to [hostname [subject [from]]]

This command allows you to post a file to a mail server. You must define the filename and to fields. If hostname is missing, the local machine will be used. If subject is missing, "Message from MAIL" will be used. If no from is included, the FROM: clause will be NULL (so the mail will be deleted if undelivered). If you would like to have spaces in the subject, you must enclose the subject in quotes, for example:

MAIL -post file.txt david post.company.com "The time of day"

IM

Allows you to send an instant message to any user that is currently logged on to the messenger service.

The message is not displayed in the recipients messenger window until the <CRLF>.<CRLF> has been entered.

Syntax

mail -logon <postmaster account> <password> -im

Example

C:\Gordano\Bin>mail -logon postmaster password -im

Please enter a destination address and message, end with <CRLF>.<CRLF>

Address:

Index

Rebuilds an index file for an existing mailbox.

Syntax

mail -index <mailbox>

Example

C:\Gordano\bin\mail -domain domain -index user

Insert List

Inserts the e-mail addresses listed in a file into the specified list. This command can use the file generated by the extract list (-extractlist) option.

In the file, each e-mail address must appear on a single line. Lines which start with a hash (#) symbol are treated as comments.

Syntax

```
mail -domain domain.dom -insertlist <listname> <filename>
```

Kick

Kicks (starts) the POST service. This means that POST will start sending any currently waiting outbound mail.

Kicking removes the entries in the current POST queue and rebuilds the queue from the mail messages located in the Out directory. Therefore, if there is a large number of messages in the Out directory, you might notice a lot of disk activity up to 12 seconds after the kick is issued.

Syntax

```
mail -k
```

Examples

```
C:\Gordano\Bin>mail -kick  
Kicked POST at Mon, 30 Sep 2002 11:28:41
```

List of Users

Writes a list of all the users in the Registry to a file. This file can be used with the load (-load) and update (-update) options. This command requires the -domain command

Syntax

```
mail -domain <domain name> -listusers <filename>
```

Load

Replaces the contents of the Registry with those of the POP script file.



Users which are in the Registry are removed unless they are also in the file.

The format of lines in the file should be:

```
mailbox username [-[password]]
```

There are three ways to define passwords:

- -password — the leading hyphen means any existing password will be replaced by the new password.
- password — if there is no leading hyphen and the user already has a password but has changed it using the Password Server, their new password will not be overwritten.

- "-" — entering just a hyphen means use the password from the NT User Database.

Syntax

mail -load <filename>

Example

Consider the following POP script file:

```
C:\mail>type test.txt
inbox.mbx postmaster pwd
inbox.mbx username1 pwd
inbox.mbx username2 pwd
```

To load the script enter:

```
C:\Gordano\bin\mail -transfer postmaster@company.dom -domain
domain.dom -load test.txt -update
```

This produces the following output if user username1 already exists and user username2 is new:

```
C:\Gordano\bin\mail -transfer postmaster@company.dom -domain
domain.dom -load test.txt -update
Reading details from file...
About to update Registry with new values...
username = postmaster, domainId = 1
username = username1, domainId = 1
/

Update completed...
New users:    1
Changed:     1
Unchanged:   0
Removed:     0
```



*The -update option effects the changes immediately. You do not have to stop and restart the services.
If a user has a forwarding account, this is not upset by the loading process - it remains a forwarding account and is updated if required*



The transfer option must be used to reassign any vCards from accounts that are deleted. See "Transfer" on page 19.

Load Share

Load sharing allows multiple servers to cater for the mail requirements of a single large domain. This provides the capabilities for Gordano messaging systems to cope with 100,000's of users. As it is a complex subject it is dealt with separately. See "Load Sharing" on page 24.

Logon

Can be used to logon and run commands where a administrative logon is required. Usually used with other options on the same command line, such as -insertlist or -im.

Syntax

```
mail -logon <account> <password>
```

Max Wait

Sets the maximum wait time for any option (except -wait) that waits for an action to complete. A wait time of 0 means infinity (that is, wait for the action to finish). This is usually used before another option on the same command line.



-wait MAIL is an exception — it returns immediately.

Syntax

```
mail -maxwait <time_in_seconds>
```

Example

To wait a maximum of 600 seconds for POST to complete, you would use this command line:

```
C:\Gordano\bin\mail -maxwait 600 -update POST
```

MX Lookup

Lists the MX records for a given host. For more details of how MX records are used, see "How is the Mail Server Found?" in the *NTMail Administrator's Guide*.

Syntax

```
mail -mx <hostname>
```

Example

```
C:\Gordano\Bin>mail -mx gordano.com
```

Number of entries 2

```
mail.gordano.com 10 62.172.232.100
```

```
gate05.gordano.com 20 62.172.232.71
```

Post

Posts a file to a given destination using a given mail server.

Syntax

```
mail -post <filename> <to>[ <hostname>[ <subject> [ <from>]]]
```

Where the parameters listed above are defined as follows:

Option	Description
filename	Name of the file you wish to post to another destination.
to	The full e-mail address of the person to receive the file.
host-name	Name of the mail server if it's not the local machine.
subject	Subject line of the message. If you want to include spaces in the subject line, you must enclose the whole option in quotes (as illustrated in the second example below).
from	The user who sent the mail. If no From clause is entered, the mail will be sent from <> and so lost if it is not delivered, also the intended recipient will not be able to reply to your mail.

Examples

The minimum needed to send a file would be:

```
C:\Gordano\bin\mail -post file.txt david@company.co.uk
```

A fully specified send operation would be:

```
MAIL -post file.txt david post.company.dom "The time of day" brian@test.dom
```

Purge DNS Cache

Clears the DNS Cache. This may be necessary if you have changed a DNS configuration but the information is still being held in cache — running **mail -purgecache** clears the cache and forces the genuine DNS settings to be re-read.

Syntax

```
mail -purgecache
```

Query Server

Queries a Gordano server for its status. If it is running on this host and no password has been defined in SmtpData, "Mail -query" displays a set of statistics about the progress of messages.

You can query other GMS servers if you know their hostname (giving a password if required).

Syntax

mail -query

Example

```
C:\Gordano\Bin>mail -query mail.company.dom
```

```
GMS SMTP Service
```

```
=====
```

GMS Setup characteristics:

Started Mon, 30 Sep 2002 14:05:36 +0100

Current time Mon, 30 Sep 2002 15:53:20 +0100

Version 8.00.3075

Local maildrops 1655 (17 aliases)

Number Threads SMTP 20, POP 20 & POST 20

Local Machine name mail.company.dom

GMS Mail users unlimited

GMS Comm users 3000

GMS WebMail users 1000

GMS Anti-Virus 8

GMS Anti-Spam enabled

SMTP statistics:

Messages processed 116

Messages in progress 1 of 20 inbound

Up time 1 hour

Run Script

Runs the given MML script carrying out any actions indicated within the script. See the MML Programmers Guide for more information.

Syntax

mail -runscript <mmlscript>

Sleep

Makes Mail.exe wait for n seconds. This is usually used before another option on the same command line.

Syntax

mail -sleep <time_in_seconds>

Example

To wait 30 seconds and then kick the POST service, you would use this command line:

```
C:\Gordano\bin\mail -sleep 30 -kick
```

Sortlist

The sortlist parameter is one of the following and is used with other commands on the same command line:

Order	Description
	Unsorted, the order in which users joined the list.
D	Domain. The output list is sorted into ascending domain names.
A	Alphabetical. The list is sorted by user name.

Syntax

mail -sortlist D

Start Gordano services

Starts the six services which are part of the Gordano product suite.

Syntax

mail -start

Stop Gordano services

Stops the six services which are part of the Gordano product suite.

Syntax

mail -stop

Transfer

When using the load option you need to specify an account that vCards can be transferred to from any accounts deleted to ensure this information is not lost.

If you do not want it transferred to a particular user set up a special user to receive these and transfer them all to that user.

Syntax

```
mail -transfer postmaster@domain.dom .....
```

Update (Reread Registry)

Tells one or all of the services to re-read their values from the Registry. Mail.exe will wait for all the services to report that they have updated.

Syntax

```
mail -update [IMAP|LIST|POST|POP|SMTP|WWW|ALL]
```

Example

To tell POP to reload, use this command:

```
C:\Gordano\bin\mail -update POP
```

Verbose

Tells Mail.exe to output the name and start time of each operation. This is useful for checking that a batch file runs correctly.

Syntax

```
mail -verbose
```

Version

Displays the version number of the Mail.exe program.

Syntax

```
mail -version
```

Wait

Makes Mail.exe wait until one of four actions is complete — POST, SMTP, BOTH or MAIL. The options are:

- POST — wait on the POST server.
- SMTP — wait on the SMTP server.
- BOTH — wait on both SMTP and POST.
- MAIL — wait for new mail to appear in the outbound mail directory.

Syntax

```
mail -wait [POST|SMTP|BOTH|MAIL]
```

Example

By default Mail.exe only waits for 10 minutes, so if you want a script to continue when mail becomes available for sending, you must use this command:

C:\Gordano\bin\mail -maxwait 0 -wait MAIL

Watch

Displays four sets of mail statistics that are updated once a second. Each set of figures has two values: the number of messages in the last second and the total since the service was started. The values have a maximum size of 232 and 'clock' back to zero if this is exceeded.

Syntax

mail -watch [i|p|s|t]

The letter after the "d" indicates the type of statistics to display:

Option	Description
-watch i	Display IMAP statistics
-watch p	Display POP statistics.
-watch s	Display SMTP statistics.
-watch t	Display POST statistics.

These values can also be viewed using the Performance Monitor — see "Performance Monitor (Perfmon)" on page 37.

Dump Current Configuration

Tells Mail.exe to dump the current Gordano setup into the text file setup.txt in the root directory, user, profile and database files may optionally be included. All Registry keys, file locations and other system parameters are written to the file, which is very useful if you have to contact Support at Gordano Ltd.

Syntax

mail -y[r|rn|rP|A|Z|R|RZ|RnZ|RZn|P|D|B]

The -y option switches can be used to save/restore extra server settings such as additional files like autoresponders etc. These are:

- yr — interactive restore setup from file.
- yrn — restores setup from file in quiet mode
- yrP — interactive restore setup from file created by porting tool.
- yA — appends a zip file to the setup.txt file containing all user, profile and database files. This can be restored later.
- yZ — extracts the zip file (if present) containing user, profile and database files from setup.txt and saves it to the current directory.
- yR — restores the setup.txt that has previously been saved with the -y option.

- yRZ — as for -yR but also extracts the contents of the zip file containing user, profile and database files into the new Gordano directory structure.
- yRnZ - as 'yRZ' but in quiet mode. Can also be run as 'yRZn'
- yP - appends a zip file to the setup.txt file containing the profile files.
- yD - appends a zip file to the setup.txt file containing the MySQL database files
- yB - appends a zip file to the setup.txt file containing the profile and database files

3.3 Mail.exe Reported Error Codes

The following error codes may be reported by the **mail** command:

Error	Description
0	No Error.
2	Key has expired.
3	Key is not valid for this machine.
4	Key length is incorrect.
5	Key validation has failed.
6	NTMail is not installed on this machine.
7	Key has not been installed.
9	MX lookup failed: no resolution for specified address.
11	Incorrect version of NTMail installed.
14	Winsock failed to initialise.
18	Error reading Registry.
22	One or more Domain Name Server (DNS) servers could not be reached.
24	No such event, for example the POST service is not running.
32	DNS reported an error in the format of the query.
33	DNS was unable to process this enquiry due to an internal problem.
34	The domain requested does not exist.
35	The DNS does not support MX record resolution.
36	The DNS refuses to perform the specified operation for policy reasons. For example, a name server may not wish to provide the information to the particular client.
37	The name server reported an undocumented error.
38	The record returned by the name server had an error in the authority record.
39, 40, 41	Unable to contact or connect to the name server. This could apply to any of the names servers specified in the NT network configuration.
46	No such user defined in Registry.
49	No file specified or found.
55	Bad list definition.

Error	Description
56	Failed to copy specified file.
57	Unknown order.
58	Could not open specified list file.
60	Timed out waiting for event.
61	Bad command format, for example a missing or badly formed parameter.
62	Network is down.
63	No records for specified host.
64	No data for specified host.
65	SMTP service does not allow queries/password.
66	Could not resolve address of server.
68	Access to mail server denied.
69	Failed at HELO command.
70	Failed at MAIL command.
71	Failed at RCPT command.
72	Failed at DATA command.

3.4 Load Sharing



Any customer using this utility should be aware that it will delete users and files from their system. They should take a full backup of their server before performing the load sharing split. No recovery options are provided. If all servers to be used in the array are clean, i.e. do not have users existing under the Gordano products installation, load sharing may be setup from the Domains > Load Sharing page of the GUI.

Load sharing allows multiple servers to cater for the mail requirements of a single large domain. This provides the capabilities for Gordano mail servers to cope with 100,000's of users. As it is a complex subject it is dealt with separately in the section immediately following this one.

The conversion utility is part of the mail.exe program already shipped with the Gordano install. A new command argument -loadshare is provided. It will take no additional options. The domain to be split will be specified by the -domain option. The -verbose option will turn on verbose mode. A minimum command line would be mail -bdomain.dom -ls

1. Only one domain may be split amongst servers. A separate licence is required for each server in the load sharing array.

2. Machines other than the original server are assumed to be blank but with a full GMS Mail or GMS WebMail key. Thus a setup.txt can be loaded onto them using the existing mechanism.
3. Users in virtual domains based on the source domain will remain on the source machine.
4. Administrative users will not be moved.
5. All services must be stopped before running this conversion program.
6. If the primary server is not listed in the rules file then a default rule with a "*" in it for the primary server will be added to the start of all rules files.

The program will prompt for the following additional information:

1. The name of the rules file. This file should contain rules (in order) for all machines in the load sharing array including the source machine.
2. The name of the source machine as identified in the rules file.
3. Confirmation as to whether this is a dry run or the real conversion. The default will be to do a dry run only.
4. The program will give a big warning prompt before doing a real conversion. The default will be not to continue.

The output of the program will be the following:

1. A full setup.txt for each destination machine. This setup.txt will contain a zip file which must be restored. No keys will be dumped in the setup.txt so that the keys on the destination machine will always be preserved even if they are a demo key.
2. The zip file in the setup.txt for each destination machine will contain all the files from the base directory of the source machine (i.e. global filters etc will be transferred). The contents of the MML directory will also be transferred. It will also contain all the files from the domain directory - all mailboxes, plans etc. will be transferred.
3. An SMPTServers.txt file will be produced for each machine. This will be contained in the zip file.
4. The source machine will be deemed the primary server and the postmaster account will be left on it. A postmaster account will be created on all target machines with administrator rights.
5. Any users being transferred to another machine will be deleted from the destination machine.
6. A postmaster user will be created on each destination machine with full access rights.
7. A report of any warning or errors encountered will be produced. This report will give the total number of users on each machine. In a dry run situation step 5 will be ignored.

The the following will need to be done on the source machine:

1. Setup load sharing as the primary machine

The following will need to be done on the destination machines:

1. Setup load sharing as a secondary machine
2. Add a postmaster account with administrator rights
3. Update user access list with postmaster

Step By Step Guide to Setting Up Load Sharing

The following steps should be taken to perform a load sharing conversion for the domain "domain.dom":

1. Decide how the domains should be split. In this example we shall split over three machines "main", "server1", "server2".
2. Decide which users will be on which on which machines e.g. initial letter "a" to "j" on "main" etc.
3. Install a fresh copy on the Gordano products on "server1" and "server2" and apply keys - note all keys must be different. If they have an installation on the machines already then it will be lost.
4. On each machine make sure that the machines "main.domain.dom", "server1.domain.dom" and "server2.domain.dom" can be resolved from the Gordano server. You may need to reconfigure your DNS server or Hosts file to do this.
5. On the source machine "main" create a file in the Gordano base directory called "ls.txt". Edit this file and add the following lines:
[a-j]* main.domain.dom
[0-9]* main.domain.dom
[k-p]* server1.domain.dom
[q-z]* server2.domain.dom
6. The above rules should cover all eventualities, if not you may need to add more. See the on-line help for definitions of the wildcard matching rules.
7. Start a DOS command prompt.
8. Run the Gordano "mail" program to create a setup.txt for your machine. This is needed in case anything goes wrong. The command should be "mail -yA".
9. Run the Gordano "mail" program to create the load sharing array.
Run the command "mail -domain domain.dom -loadshare".
10. You will be prompted for the name of the file containing the load sharing rules. This is "ls.txt".
11. You will be prompted for the name of the source server in these rules. This is "main.domain.dom".
12. You will now see a warning screen. We will perform a dry run initially to check all is well. Just hit return.

13. The domain conversion program now takes place. Any errors or warnings that will occur in the full conversion will be shown. You can either deal with these now or after the conversion. If a user with now matching rule is found then you should return to step 5 and add additional rules.
14. You will now see the file ls.out in the base directory, this file contains the output of the mail program for later reference.
15. Now run the conversion program again. Follow steps 9 and 10 again.
This time at the warning screen enter "YES".
Another warning screen is shown. Enter "YES" again.
The domain conversion will now take place for real.
16. Two files will be created in the Gordano base directory named server1.domain.dom.txt and server2.domain.dom.txt. Copy "server1.domain.dom.txt" to server1 and "server2.domain.dom.txt" to server2.
17. On server1 and server2 load these files using the mail.exe program. The command is "mail -yRZ<filename>".
18. Log onto each of main, server1 and server2 via a web browser. You may need to perform the following corrections:
Modify the local domain names to remove the "*.domain.dom" entry.
Modify the IP Address of domain.dom on server1 and server2 to use the correct value.
19. Ensure that the MX records for "domain.dom" point to "main.domain.dom", "server1.domain.dom" and "server2.domain.dom".
20. The domain conversion is now complete. You may wish to modify some of the settings in the "Load Sharing" tab. All users mail has been transferred.

3.5 AddUser.exe

Adduser.exe is a command line tool that lets you interact with Gordano mail servers by adding and deleting domains, adding and deleting users and retrieving and changing user variables. It can be run directly from the command line processing a single command passed to it as an argument, interactively where you step inside the program and can issue one command after another without calling adduser.exe each time or in batch mode where it reads its commands directly from a text file. For security each command requires an administrator's email address and password which can be specified in the command line.

Add Alias

Add an alias called 'alias' to an existing account 'user' in the domain 'domain'

Syntax

aa adminemail adminpassword user domain alias

Example

C:\Gordano\bin\adduser aa postmaster@company.dom adminpassword user domain alias

Add Domain

Add a domain of the specified type to a Gordano mail server, if the domain is to be a virtual domain then the base domain it should be associated with and the postfix must also be specified.

Syntax

ad adminemail adminpassword domain type [param1 [param2]]

specify the IP Address in param1 for a Full Domain

e.g. ad company.com 1 1.2.3.4

specify 'baseDomain' in param1 and 'postFix' in param2 for a Virtual Domain

e.g. ad postmaster@company.dom adminpassword company.dom 8 base.dom company

Domain Type	Value
Full Domain	1
Virtual Domain	8
POP Domain	2
Robot Domain	4

Example

C:\Gordano\bin\adduser ad postmaster@company.dom adminpassword domain 1

Command: ad postmaster@company.dom adminpassword domain type [param1 [param2]]

Add User

Add a user to the specified domain. If the user already exists the password will not be changed.

Syntax

au adminemail adminpassword user domain password [forced]

Example

```
C:\Gordano\bin\adduser au postmaster@company.dom adminpassword user  
domain password
```

Delete Alias

Removes the alias 'alias' from the account 'user' in domain 'domain'.

Syntax

```
da adminemail adminpassword user domain alias
```

Example

```
C:\Gordano\bin\adduser da postmaster@company.dom adminpassword user  
domain alias
```

Delete Domain

Will remove an existing domain from a Gordano mail server.

Delete the domain 'domain' and transfer vcards to user 'tuser' in domain 'tdomain'

Syntax

```
dd adminemail adminpassword domain tuser tdomain
```

Example

```
C:\Gordano\bin\adduser dd postmaster@company.dom adminpassword vcard  
user@company.dom domain
```

Delete User

Removes a user from the specified domain.

Syntax

```
du adminemail adminpassword user domain tuser
```

Example

```
C:\Gordano\bin\adduser du postmaster@company.dom adminpassword user  
domain vcarduser@company.dom
```



The tuser option must be used to reassign any vCards from the account to ensure the information is not lost.

Set Variable

This option can be used to set any of the variables for Gordano products whether at the System , Domain or User level. The last part of the variable path must contain two back slashes i.e. for the system level \\<variable>, for the domain level

<domain>\\<variable> and for the user level they are split by the user name<domain>\<user>\<variable>.

Syntax

se adminemail adminpassword variable value

Example

```
C:\Gordano\bin\adduser se postmaster@company.dom adminpassword
\poplog 278
C:\Gordano\bin\adduser se postmaster@company.dom adminpassword
domain.dom\\CompanyName Gordano
C:\Gordano\bin\adduser se postmaster@company.dom adminpassword
domain.dom\user\password mypassword
```

Get Variable

This option can be used to get any of the variables for Gordano products whether at the System , Domain or User level. The last part of the variable path must contain two back slashes i.e. for the system level \\<variable>, for the domain level <domain>\\<variable> and for the user level they are split by the user name<domain>\<user>\<variable>.

Syntax

ge adminemail adminpassword variable

Example

```
C:\Gordano\bin\adduser ge postmaster@company.dom adminpassword
\poplog
C:\Gordano\bin\adduser ge postmaster@company.dom adminpassword
domain.dom\\CompanyName
C:\Gordano\bin\adduser ge postmaster@company.dom adminpassword
domain.dom\user\password
```

Interactive Mode

Once in interactive mode any of the commands listed above may be run directly from the command prompt, without having to call adduser each time. Any of the commands outlined above may be called. To leave interactive mode, type "quit" and press enter.

Syntax

i

Example

```
i
ad adminemail adminpassword domain 1
au adminemail adminpassword user domain password
aa adminemail adminpassword user domain alias
quit
```

Batch Mode

Batch mode is very similar to Interactive Mode except that all of the commands are held in a pre written file. The commands should be written one per line and written exactly as if you were in interactive mode, including a "quit" command as the last line of the file.

Syntax

f filename

Example File

```
ad adminemail adminpassword domain 1
au adminemail adminpassword user domain password
aa adminemail adminpassword user domain alias
quit
```

3.6 Admin Applet

The Admin Applet is provided to cover for situations where you are not able to access the WWW Interface to make administrative changes for one reason or another. It has a number of options that can allow you to regain access to the system.

Adding an Administrator

If you are unable to gain administrative access to the Gordano WWW GUI, it is possible to create a new user or set the permissions for an existing user. To do this

Domain

Select the domain to create the user in from the drop down menu.

Username

Enter the name for the user you want to set permissions for, this may be either a new or existing user.

Password

Enter a password for this user

Confirm Password

Confirm the password was entered correctly by entering it a second time.

Permissions

Select all the permissions you would like this user to have on the Gordano Server, the options are Domain, System, Log and Anti-Spam, one or all of the permissions may be set.

Once completed click on the **Add User** button to enter the user into the configuration. You should now be able to log on using the username and password given above.

Admin Server port

If the Gordano WWW Service refuses to start up correctly a possible cause could be a port conflict with another service running on the server. If this is the case then the only option is to change the port being used by one or other of the service. This option makes it particularly easy to change the port number for the WWW service, simply enter the new port number in the area provided and click on the **Update** button then restart the WWW service.

3.7 Gordano Inspector

Introduction

Gordano Inspector helps you maintain e-mail on your server. It lets you view the contents of any domain or mailbox and perform various “housekeeping” duties, such as renaming a mailbox or deleting a message from a mailbox.



Gordano Inspector is an Option Pack utility that you can download free from our Download page, or using our FTP site.

Getting started

When Gordano Inspector first starts a dialog box appears, telling you that the program is analysing your mail server. This will take some time if you have lots of accounts. Once it finishes a window like that shown below appears, graphically displaying your computer name and local host, all the domains, including POP domains, present on your system, and all the mailboxes within each domain:



Each folder represents a domain so, if you only have a single domain configured on the server, only one folder will be displayed. To display all the local mailboxes, select the domain folder. Note that a mailbox is not shown if no e-mail has ever been sent to that user. Selecting a mailbox displays a message summary where individual messages can be deleted or edited.

In a multi-domain version of a Gordano server all the additional domains will also be displayed. Any POP domain accounts are shown as green folders and selecting such a folder displays its contents directly. POP domain folders do not contain any

mailboxes, of course, they only contain the messages in that POP domain.

To view the mailboxes within a domain, just click on the domain folder.

Gordano Inspector also displays the statistics for mailboxes and messages. To view the messages within a mailbox, click the mailbox.

If you right-click on a folder, mailbox or e-mail message, the shortcut menu is displayed. This is an alternative to using the drop-down menus described below.



The index file (.idx) associated with a mailbox is updated automatically by Gordano Inspector, so you do not have to edit or delete anything in the index yourself.

The Action menu

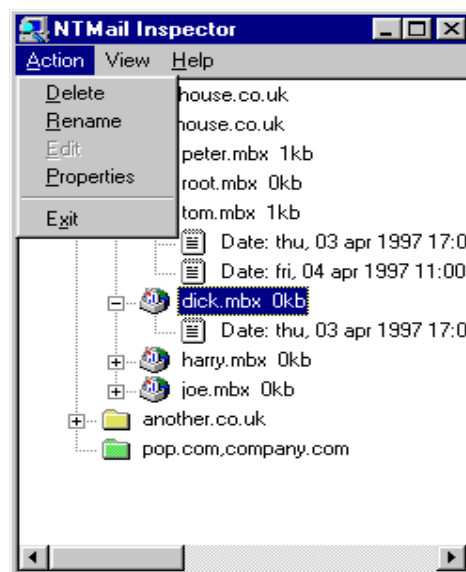
Your Gordano Inspector options depend on the item you select:

- Domain — you can just view its properties.
 - Mailbox — you can delete this, rename it or view its properties.
 - Message — you can delete this, edit it or view its properties.
- When you try to delete a message, Gordano Inspector sometimes reports that its mailbox is busy. This is normal and means that your mail server is currently using the mailbox; re-try after a short wait.

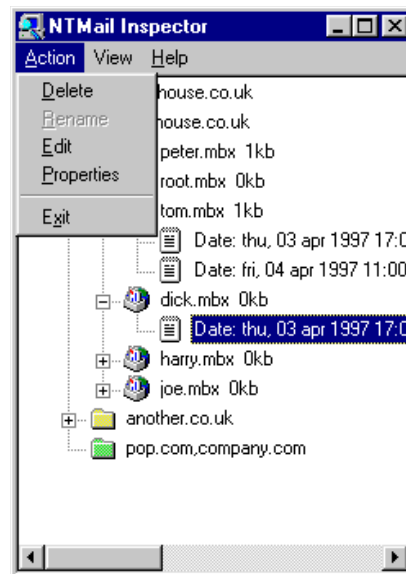
The following picture shows the Action menu options for a mailbox:



If you rename a mailbox, the user will no longer be able to access the e-mail in it.



The following screenshot shows the menu options available for a single message within a mailbox.



The View menu

The View menu lets you expand and collapse the graphical display to display either all of the domain folders, mailboxes and messages or just the computer name and local host.

The View menu also lets you refresh the graphical display so you can use Gordano Inspector while your server is running. If the information displayed is out of date, click on **Refresh**. The dialog that appeared when you first started Gordano Inspector will appear, telling you that your Gordano server is being analysed. When the analysis finishes the display will have been updated with any new messages, mailboxes or domains that have arrived or been added.

The Help menu

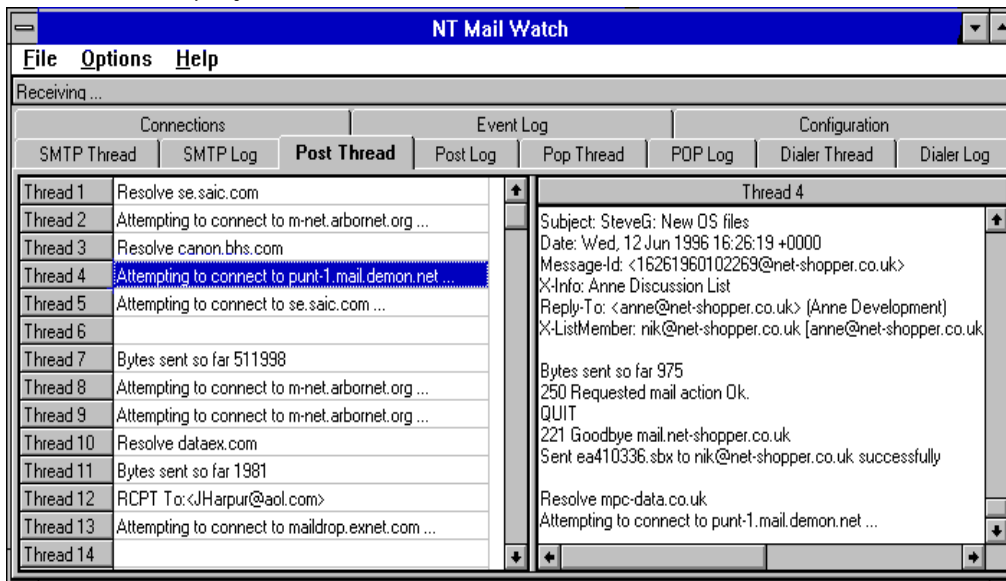
The Help menu lets you display version information for the utility.

3.8 Mailbox Monitor

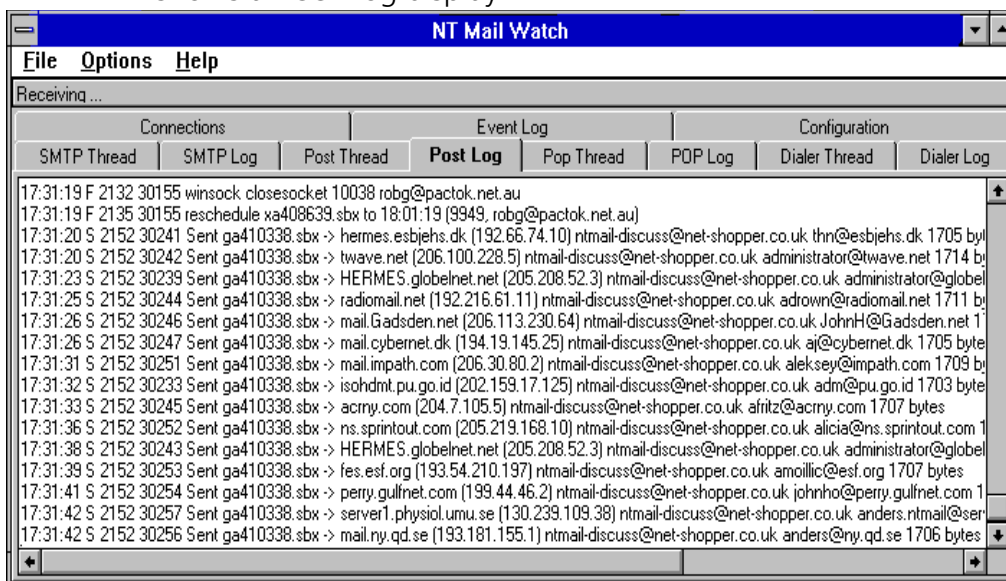
Mailmon.exe is a powerful tool which monitors the running of the mail server — thread allocation and the sending and receipt of e-mail. It's like having a dynamic view of the log files (see "Log Contents" on page 301 for more information about log files). Once started, the monitor usually takes up to 10 minutes to activate. This delay has been introduced to reduce the load on the mail server.

There are three sets of pages, as follows:

- SMTP, POST and POP Threads. These three pages show you what each thread is doing. This example shows a POST Thread display:



- SMTP, POST and POP Logs. These three pages display all the log activities, including protocol and error messages, if these are activated on the Configuration page. The following example shows a POST log display:



- An Event Log shows the current status of the Monitor and what it is doing.

The Configuration page lets you select:

- The Port to Monitor.
- Monitor on Startup — whether to monitor the services on startup of the program.
- Monitor SMTP, Monitor POST, Monitor POP — which of the three services to monitor.
- Fill Logs From Top — whether the most recent event is shown at the top of the display.
- Max Log Lines — the maximum number of lines to display in logs.
- Toolbar on Startup — whether to display the Monitor Toolbar when the program starts.

The drop-down menus are:

- File — lets you change the Print Setup.
- Options — lets you turn the monitor On/off and Clear All the Logs.

3.9 Performance Monitor (Perfmon)

Performance Monitor is a Windows NT utility which lets administrators check on the progress of all the services running on the system.

Operation

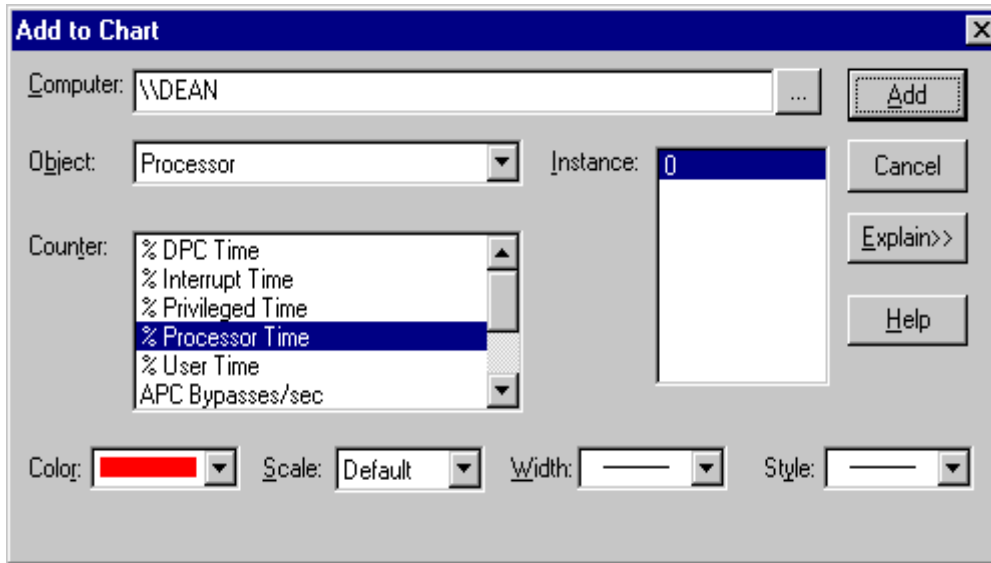
You can use Performance Monitor:

- To trace abnormal memory usage.
- To look for memory leaks (memory not released cleanly).

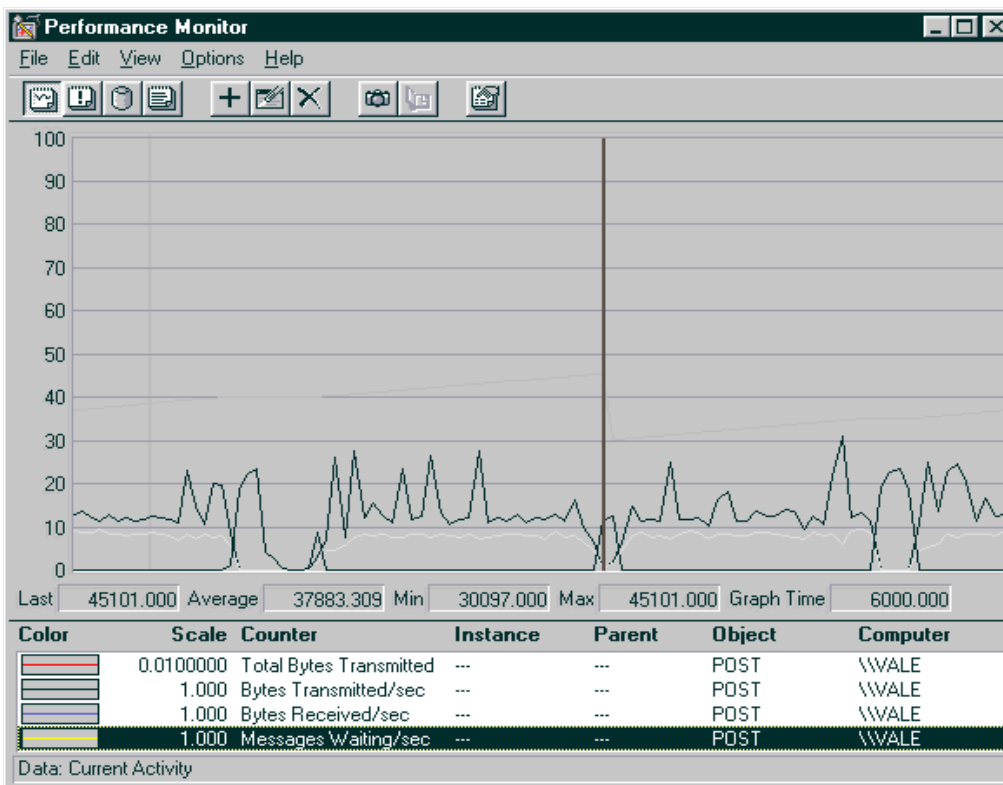
To start Performance Monitor:

1. Type **perfmon** on the command line. You'll see an empty graph.

- From the Edit menu, choose Add to Chart in the dialog shown below and select the statistics you want to view.



- This example shows monitoring of the Total Messages Transmitted by the POST server:



After some time your display may resemble that shown above. Here the POST server has been run at maximum capacity with some 15,000 messages being posted (out of a total of 50,000 waiting). This shows how the POST server runs under such a large load - you can see how it checks for new messages, sends them, then looks for more to send.

Each process in your Gordano server has its own "Object" — SMTP, POST and POP. The entries shown above — Total Bytes Transmitted, Bytes Received/Sec, etc. — and all other parameters are described in more detail in the following sections. LIST is described in the *GMS Communication Server Guide*.

SMTP parameters

The SMTP Object Type handles the GMS SMTP Server on your system. In addition there are some entries for the list server which are included here for completeness:

Name	Description
Messages In/Sec	The number of messages that have completed in the last second. If a message is delivered to two destinations, this counter only increases by one.
NTMail/sec	The number of messages that were sent to an account in the last second. Thus, a message sent to two POP accounts increases this counter by two.
GLCommunicator/sec	The number of messages sent to the list server in the last second.
TDomain/sec	The number of messages sent by one of the additional domains supported by this server in the last second
POP/sec	The number of messages delivered to a POP account in the last second.
Forward/sec	The number of messages forwarded in the last second.
Robot/sec	The number of messages sent to the executable or robot account in the last second.
Info/sec	The number of messages sent to the Autoresponder or Info account in the last second.
Fax/sec	The number of messages sent though the FAX gateway in the last second.
Messages to Null/sec	The number of messages sent to the NULL account in the last second. These messages may be stored in the BadMes directory.
Messages Inbound/sec	The number of messages delivered to the local domain or any domain controlled by the mail server in the last second.
Messages Outbound/sec	The number of messages that are to be forwarded to another mail server received in the last second.
Message Bytes Received/sec	The number of bytes in messages that passed though the server in the last second.
Bytes Transmitted/sec	The number of bytes written to TCP/IP in the last second.
Bytes Received/sec	The number of bytes received from TCP/IP in the last second.

Name	Description
List Help Requests/sec	The number of help messages sent by the list server in the last second.
List Post Requests/sec	The number of messages that have been sent to a list controlled by the list server in the last second.
List Post Rejected/sec	The number of messages sent to a list that have been rejected in the last second. The message may have been rejected because the sender is not allowed to post to the list.
List Post Help/sec	The number of help messages sent in response to an attempt to send a message to a list in the last second. Help messages are sent because the user has sent a message with "subscribe", "unsubscribe" etc. to the list rather than the list manager.
List Post Too Big/sec	The number of messages that exceeded the maximum message size for this list in the last second. These messages are automatically returned to the sender.
List Post Reposted/sec	The number of messages which have already been to this list in the last second.
List Manager/sec	The number of messages processed by the list manager in the last second.
List Join Requests/sec	The number of join requests that have been processed in the last second.
List Join Rejected/sec	The number of join requests that have been rejected in the last second because the sender is not allowed to join the list.
List Join Failed/sec	The number of joins that have failed in the last second due to the sender already being a member of the list.
List Leave Requests/sec	The number of leave requests received per second

POST parameters

The POST “Object Type” handles the GMS POST Server on your system. The parameters are as follows:

Name	Description
POST	The Gordano POST Server sends mail out to its final destination.
Messages Waiting/sec	The change in the number of messages waiting to be sent (per second).
Messages Transmitted/sec	The number of messages sent in the last second.
Total Messages Transmitted	The total number of messages transmitted.
Messages In Progress/sec	The number of messages currently being transmitted.
Messages Failed/sec	The number of messages that failed in the last second.
Total Messages Failed	The total number of messages that failed to be posted and have been returned to their sender.
Messages Returned/sec	The number of messages returned in the last second.
Total Messages Returned	The total number of messages returned.
Message Bytes Transmitted/sec	The number of message data bytes transmitted in the last second.
Bytes Transmitted/sec	The total number of bytes transmitted in the last second.
Bytes Received/sec	The total number of bytes received in the last second.
Total Message Bytes Transmitted	The total number of message bytes transmitted.
Total Bytes Transmitted	The total number of bytes transmitted.
Total Bytes Received	The total number of bytes received.
Connections Made/sec	The number of connections made to other services in the last second. Each connection to a remote SMTP and DNS server is included.
Total Connections Made	The total number of connections made.
DNS Requests/sec	The number of DNS requests made in the last second.
Total DNS Requests	The total number of DNS requests made.

POP parameters

The POP “Object Type” handles the GMS POP Server on your system:

Name	Description
POP	The Gordano POP Server (including finger and password servers).
Client Connection/sec	The number of POP clients that have connected to the POP server in the last second.
Failed Authorisations/sec	The number of password validations that failed in the last second.
Mailbox Message Reads/sec	The number of messages read out of mailboxes in the last second.
Mailbox Message Failed Reads/sec	The number of attempts to read a mailbox that failed in the last second.
Mailbox Busy/sec	The number of times a mailbox was reported as busy in the last second.
Mailbox Update/sec	The number of mailboxes updated in the last second.
Mailbox Failed Update/sec	The number of mailbox updates that failed in the last second.
Finger Requests/sec	The number of requests made to the Finger server in the last second.
Finger Failures/sec	The number of finger requests that failed to find a user in the last second.
Logins/sec	The number of attempts to change a password using the password server in the last second.
Failed Logins/sec	The number of attempts to change a password that failed in the last second.
Passwords Changed/sec	The number of passwords changed in the last second.
Message Bytes Sent/sec	The number of message bytes send to clients in the last second.
Bytes Transmitted/sec	The total number of protocol and message bytes that the POP server transmitted in the last second.
Bytes Received/sec	The total number of bytes that the POP server received in the last second. These will all be commands.
Total Client Connection	The total number of connections made to the POP server.

Name	Description
Total Failed Authorisations	The total number of failed authorisations at the POP server.
Total Mailbox Message Reads	The total number of mailbox messages read.
Total Mailbox Message Failed Reads	The total number of failed reads of mailbox messages.
Total Mailbox Busy	The total number of mailbox busy responses.
Total Mailbox Update	The total number of mailbox updates.
Total Mailbox Failed Update	The total number of failed updates for mailboxes.
Total Finger Requests	The total number of Finger requests made to the POP server.
Total Finger Failures	The total number of failed Finger requests.
Total Logins	The total number of user logins made to the POP server.
Total Failed Logins	The total number of failed login attempts.
Total Passwords Changed	The total password change attempts.
Total Message Bytes Received	The total number of message bytes received.
Total Bytes Transmitted	The total number of bytes transmitted.
Total Bytes Received	The total number of bytes received.

3.10 Regedt32

Windows NT stores its configuration information in a database (the Registry) organized in a tree format. The Registry Editor Regedt32.exe lets you inspect and modify the Registry. Use it to change the parameters listed in the Registry which you cannot set from Gordano's user interface (see the following chapter).



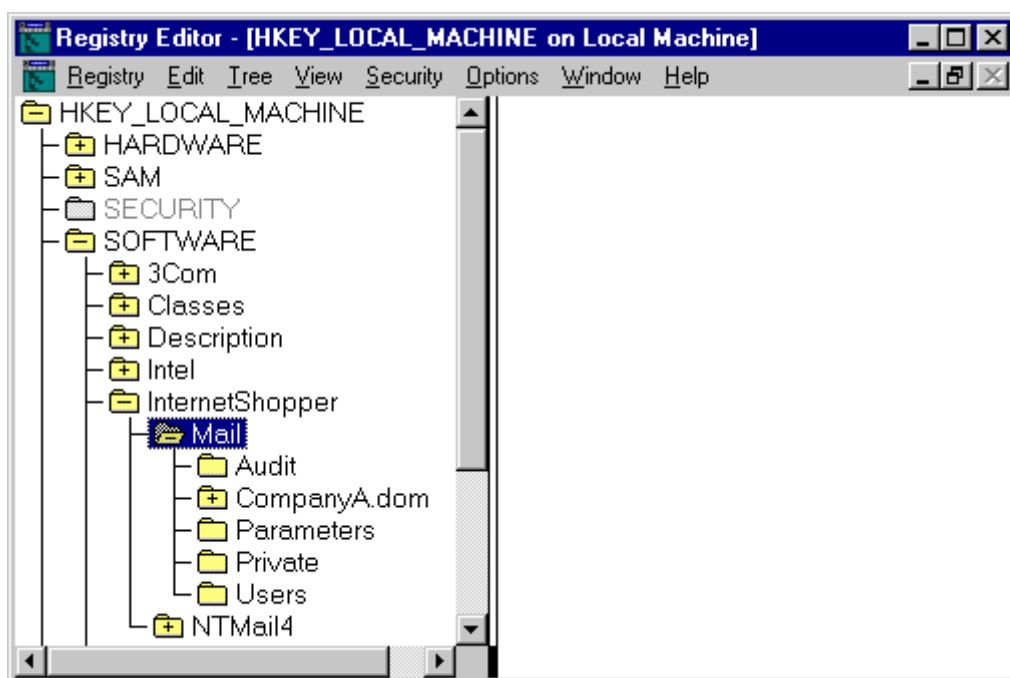
Use Regedt32.exe, which supports multi-string Registry entries, not any other version of the Registry editor.

You may break your system if you make incorrect changes, so take care. For more instructions, open Regedt32 and select Help>Contents.

There are several ways to run the Registry Editor:

- Click Start, Run and type **regedt32** in the Open box.
- In Windows NT Explorer, double-click Regedt32.exe in the %SystemRoot%/System32 folder.
- In a Command Prompt window, type **regedt32**.

Folders represent keys in the Registry and are shown in the tree view in the left-hand pane. In the right-hand pane, the values of that key are displayed. Double-click a value to open an editing dialog box.



In this example, company.dom is the only domain on the system. The other entries at the same level are described in detail in the following chapter.

Use the Add Key, Add Value and Delete options under the Edit menu to add or change your Registry entries.

3.11 Ping and Traceroute

These tools help you check connectivity between your server and another.

Ping

Use ping to test connectivity between your server and one or more servers. You can run it from a command prompt as described in this section.



You can also run ping from Gordano's Web interface.

Syntax

```
ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL]
[-v TOS] [-r count] [-s count] [[-j host-list] |
[-k host-list]] [-w timeout] destination-list
```

Description

Ping sends packets called Echo Requests to the specified host(s) asking them to reply. It then tells you whether the replies have been received.

Its options are:

Option	Description
-t	Ping the specified host until interrupted.
-a	Resolve addresses to hostnames.
-n count	Number of echo requests to send.
-l size	Send buffer size.
-f	Set Don't Fragment flag in packet.
-i TTL	Time To Live.
-v TOS	Type Of Service.
-r count	Record route for count hops.
-s count	Timestamp for count hops.
-j host-list	Loose source route along host-list.
-k host-list	Strict source route along host-list.
-w timeout	Timeout in milliseconds to wait for each reply.

Traceroute (Tracert)

Use Traceroute to find the route packets take to a destination. You can run it from a command prompt as described in this section. Its name is abbreviated to tracert.



You can also run traceroute from Gordano's Web interface.

Syntax

```
tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] target_name
```

Description

Tracert determines the route taken to a destination by sending Internet Control Message Protocol (ICMP) echo packets with varying Time-To-Live (TTL) values to the destination. Each router along the path must decrement the TTL on a packet by at least one before forwarding it, so the TTL is effectively a hop count. When the TTL on a packet reaches 0, the router should send back an ICMP Time Exceeded message to the source server.

Tracert determines the route by sending the first echo packet with a TTL of one and incrementing the TTL by one on each subsequent transmission until the target responds or the maximum TTL is reached. The route is determined by examining the ICMP Time Exceeded messages sent back by intermediate routers.



Some routers silently drop packets with expired time-to-live (TTLs) and these will be invisible to tracert.

The options are:

Option	Description
-d	Do not resolve addresses to hostnames.
-h maximum_hops	Maximum number of hops to search for target.
-j host-list	Loose source route along host-list.
-w <timeout>	Wait <timeout> milliseconds for each reply.

3.12 Nslookup

This NT tool has many options, but you'll probably use it to look up MX records when you have a problem.

To use it this way, open a command prompt and type:

```
nslookup -querytype=MX -<computer wanted> -[server]
```

Where:

- -computer wanted — is the computer you want to see records for.
- -server — is the DNS name server. Either specify the server you want to see the MX records on or leave the parameter blank (just type a hyphen) to use the default DNS name server.



For a description of other Nslookup options, use the Windows NT help.

3.13 Finger

Use Finger to display information about a user on a specified system running the Finger service. Finger is only available if the TCP/IP protocol has been installed.

Syntax

```
finger [-l] [user]@computer [...]
```

Description

Output varies depending on the remote system. The options are:

Options	Description
-l	Displays information in long list format
user	Specifies the user you want information about. Omit the user parameter to display information about all users on the specified computer.
@ computer	Specifies the server on the remote system whose users you want information about.

4 Configuration — The Registry

You can set up most of the Gordano features and options from the user interface. Use this section if you need to configure the other options. To do this, use Regedt32 to change parameters listed in the Registry.



Only use Regedt32.exe, which supports multi-string Registry entries, not another version of the Registry editor.

This section describes:

- How the Registry works.
- The three types of Registry parameter you are most likely to edit — global, domain-specific and user. All these parameters and their meanings, defaults, etc. are covered.
- Private parameters which you'll probably not need to change.

The chapters which follow describe the Registry parameters for GMS Communication Server, GMS WebMail, GMS WapMail, GMS Anti-Spam and GMS Anti-Virus.

4.1 Structure

Registry structure (Windows)

All the Registry entries for Gordano products are stored under this Registry key:

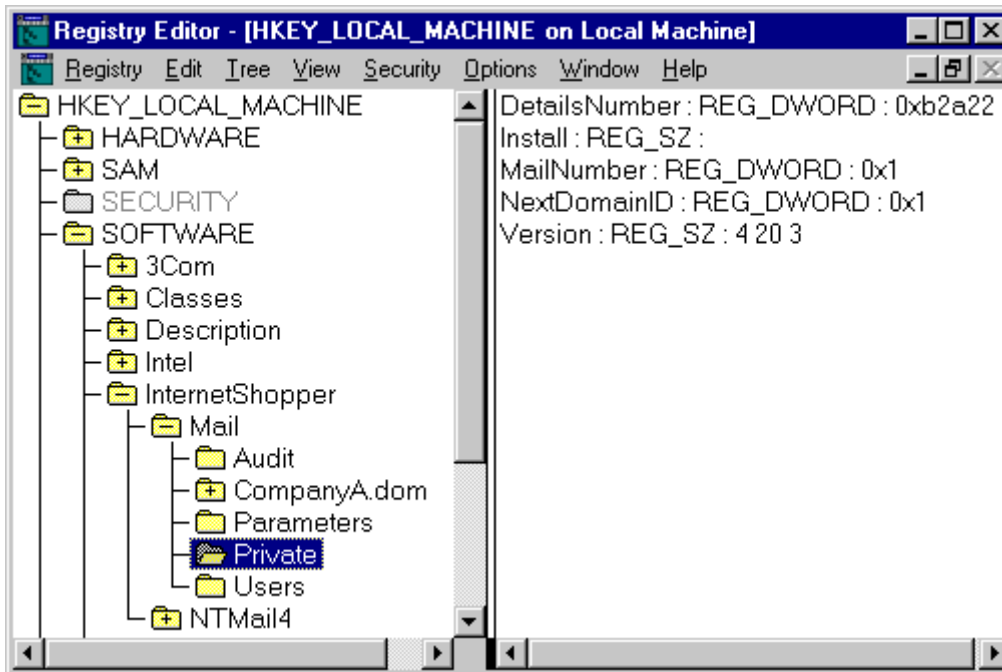
HKEY_LOCAL_MACHINE/Software/InternetShopper/Mail



You can restrict access to the keys. Using Regedt32, select the InternetShopper key, then use the Security, Permissions options in the menu bar.

Under this Registry key there are two sets of definitions - the global Registry parameters and the domain-specific parameters. In this

example, there's just one domain, CompanyA.dom, and the Private key is selected:



Configuration Database structure (Unix)

All the Configuration Database entries for GMS are stored in hidden .reg files stored in the Gordano directory structure. That is in the Mail directory and its sub-directories

Under both structures there are three sets of definitions - The global parameters, domain specific parameters and user parameters.

Global, domain-specific and user parameters

There are three types of parameter:

- Global — affect the overall operation of Gordano Products. These are stored under the key Mail\Users in an un-named Registry Value, which appears as <No Name> in Regedt32.
- Domain-specific — each domain has three mandatory parameters and many optional parameters. In a multi-domain system, only the values which can change on a per-domain basis are duplicated under the domain's own key.

The three mandatory parameters are held under the key matching the specific domain name. These are:

- Domain id — this is Gordano product-specific, not an NT identifier.
- Domain type — full, virtual, POP, robot or DLL (held as a number).
- User count — number of users in the domain.

All the other parameters for the domain are held under the Users key for the specific domain name. For example, the parameters for domain test.dom would be found in Mail\test.dom\Users. The value holding the parameters is unnamed so appears as <No Name> in Regedt32.

- User — All the parameters for each user that have been created within a domain. These normally appear as values with names matching the name of the user.

How values are stored in the Registry or Configuration database

All Gordano parameters are stored in MULTI_SZ type Registry values, one setting per line. All UNIX parameters are stored in space separated lists in the .reg files.

Details used for NT SAM users

Adding users from a Windows NT SAM database does not change the Registry, but if you modify a user account later, the change does affect the Registry.

Warnings

The following list summarises precautions you must take:



Do not:

Delete the user Postmaster.

Change any parameter from its default without good reason — the default values give maximum efficiency in most circumstances.

Use a Registry editor other than Regedt32.

Entering IP addresses

There are several ways of inputting multiple IP addresses to Gordano products, as shown below. In all of these, the letters a to e represent numbers in the range 0 to 255:

- a.b.c.d — a specific IP address, for example 194.194.194.194.
- a.b.c.* — all IP addresses beginning a.b.c. For example, 194.194.194.* gives addresses in the range 194.194.194.0 to 194.194.194.255.
- a.b.c.d-e — a range of IP addresses from d to e. For example, 194.194.192-194.* gives addresses in the range 194.194.192.0 to 194.194.194.255
- a.b.c.d/n — means use the first *n* bits. For example, 194.194.194.194/22 gives addresses in the range 194.194.192.0 to 194.194.195.255.

Similarly, 194.194.194.194/16 gives IP addresses in the B Class range 194.194.0.0 to 194.194.255.255.

- !a.b.c.d — the "!" at the beginning of the address means NOT.
For example, !194.194.194.194 means not 194.194.194.194.

4.2 The parameters and their meanings

This section describes all the global, domain-specific and user parameters. To list all the parameters from within the administration interface, choose Support> System Variables or Support>Domain Variables.

Global parameters

The global parameters which control the overall operation of Gordano products are under this key:

HKEY_LOCAL_MACHINE\Software\InternetShopper\Mail\Users\<No name>

The table which follows lists the parameters. Note the following:

- If you use Gordano's user interface to change one of these values *back to the default value*, the parameter is automatically deleted from the Registry.
- Values which have a range of 0 - 1 are Boolean: 0=false (disabled), 1=true (enabled).
- Values which have a range shown as "0-" or "1-" do not have a specific upper limit.

The parameters are listed in the following table:

The remainder of this section describes each parameter in turn.

Name	Range	Recommended value
AccessIP		
AccessIPDenied	0 - 1	0
AdminBackgroundBackdrop		#008282
AdminMenuTitleColour		#00007f
Administrator		
AlertMaxEntries		
AlertPort	1 - 65535	8081
AlertsRequired	1 - 65535	1021
AllowAdminGuiServer	0 - 1	1
AllowAPOLogin	0 - 2	0
AllowCollaborationServer	0 - 1	1
AllowDailyStats	0 - 1	0
AllowEditScripts	0 - 1	1
AllowEditVariables	0 - 1	0
AllowEtrn	0 - 3	1
AllowedRCPTLimit	1 - 500	100
AllowedRCPTLimitMes		See text
AllowESMTP	0 - 1	1
AllowFaxDLL	0 - 1	1

Name	Range	Recommended value
AllowFingerServer	0 - 1	1
AllowHelp	0 - 1	1
AllowIMServer	0 - 1	1
AllowLookup	0 - 1	1
AllowManuals	0 - 1	1
AllowNTInfo	0 - 1	1
AllowNTPassword	0 - 1	1
AllowPasswordServer	0 - 1	1
AllowProxyServer	0 - 1	1
AllowReturnReceipt	0 - 1	1
AllowReverseProxyServer	0 - 1	1
AllowScriptServer	0 - 1	1
AllowSSLProxyServer	0 - 1	1
AllowUserGUIServer	0 - 1	1
AllowVRFY	0 - 2	0
AllowWebMailGUIServer	0 - 1	1
AllowWWWServer	0 - 1	0
AllowXTND	0 - 1	0
AutoPortHost		
BlockingIO	0 - 127	0
CollaborationTcpBlockSize	0-64	0
DeliveryDLL		
DeliveryScript		DeliveryScript.mml
DialupActions		
DialupLog	1 - 65535	18
DialupRequested	0 - 1	0
DisplaySupport	0 - 1	1
DNS Aliases		
DNSCacheDir		
DNSCacheExpiry		24
DNSCacheFlush		5
DNSCacheSize		1024
DNSCacheZeroTTL		1
DNSDomain		
DNSHost		
DNSMaxCNameDepth	1--100	10
DNSMinNumDots		1
DNSOptions	0 - 224	1

Name	Range	Recommended value
DNSPort	1 - 65535	53
DNSQueryOptimise	0 - 1	1
DNSServers		
DNSTimeClient	1-	30
DNSUseHOSTSFile	0 - 1	1
EnforceCRLF	0 - 2	1
EnforceDSN	0 - 1	1
EnforceReturnPath		
EnforceRFC822Header	0 - 1	0
ESMTP8BitMIME	0 - 1	1
ESMTPAuth	0 - 3	0
ESMTPDeliveryStatus	0 - 1	1
ESMTPEnhancedStatusCodes	0 - 1	1
ESMTPPipelining	0 - 1	0
ESMTPRestart	0 - 1	0
ESMTPSize	0-	10
FaxDLL		
FAXPageLength		2340
FAXPageToLength	0 - 1	0
FAXPageWidth		1728
FAXPriority	0 - 3	0
FilterScript		FilterScript.mml
FingerPort	1 - 66535	79
FolderDisconnectIdle	0-2	0
IMAPFetchBufferSize	0-64	32
IMAPLog		18
MAPLogonDelay		0
IMAPMaxConnections	1 - 255	20
IMAPPort	1 - 66535	143
IMAPTcpBlockSize	0-64	0
IMAPThreads	1 - 256	20
IMAPTimeClient		300
IncomingDLL		
IncomingScript		IncomingScript.mml
LocalIP		See text
LogAllMessages	0 - 1	1
LogAutoDelete		0
LogDeleteAfter		5

Name	Range	Recommended value
LogEmailAfter		0
LogonDelay		0
LogZipAfter		0
MailerMessage		
MailLogDLLLocation		
MaxAccountSize	0-	0
MaxBadCommand		100
MaxBadMesDirSize		5,342,880 bytes
MaxHopCount	1-50	17
MaxHTTPDataLength		10240
MaxLineLen		0
MaxLogDirSize		10 MB
MaxMailboxSize	0-	0
MaxMessageSize	0-	0
MaxMessagesPerIPAddress		
MaxMessagesPerIPAddressMes		
MaxMessagesPerUser		
MaxMessagesPerUserMes		
MaxNumberFilesInCache		
MaxPasswordLetterOccur		
MinDiskSpaceRequired		
MinPasswordLen		5
MMLErrorToSupport	0 - 1	1
MSMailAddressFix	0 - 1	0
NameServers		
NextCookie		
NextNtListCookie		
NTComputer		
NTDomains		
OutDir		basedir\out
OutgoingDLL		
OutgoingScript		OutgoingScript.mml
PasswordPolicy	0-7	0
PasswordPort	1 - 65535	106
POPBandwidth		0
POPReetrBufferSize	0-64	32
POPLog	1 - 65535	279
POPLogonDelay		0

Name	Range	Recommended value
POPMaxConnections	1 - 256	1
POPPort	1 - 65535	110
POPSaveOnDisconnect	0 - 1	0
POPTcpBlockSize	0-64	0
POPThreads	1 - 256	20
POPTimeClient		300
POSTBandwidth		0
POSTLog	1 - 65535	791
PostTcpBlockSize	0-64	0
PostmasterStats	0 - 1	0
POSTMaxContLine		0
PostRetryTime	0 - 10080	12
PostReturnTime	1 - 744 hour	72
POSTServers		See text
POSTSSL	0 - 2	0
POSTThreads	1 - 256	20
POSTTimeSend		1200
PostWarningTime		24 hours
ProductsInstalled	0 - 63	
ProxyCacheSize		0
ProxyCacheSizeMax		32768
ProxyPurgeAge		7
ProxyPurgeAgeLast		
ProxyPurgeMethod	0 - 1	1
ProxyRASAccName		
ProxyRASAccName2		
ProxyRASBackup		
ProxyRASPassword		
ProxyRASPassword2		
ProxyRASPhoneNumber		
ProxyRASPhoneNumber2		
ProxyRASSelected	0 - 1	
QuarantineAdminPort		8000
QuarantineAdminServer		
RASAccName		
RASAccName2		
RASDisconnectDelay		2 minutes

Name	Range	Recommended value
RASMaxScheduleTimeout		
RASMinScheduleTimeout		
RASPassword		
RASPassword2		
RASPhoneNumber		
RASPhoneNumber2		
RASProxyDisconnectDelay		
RcptLimit	1 - 600	100
RedirectFile		See text
ReturnReceiptAllow	0 - 1	1
RFC1123Return	0 - 1	1
SaveBadMes	0 - 1	1
SaveSetupFiles	0 - 1	0
SaveSetupSendTo		
ShowRealHost	0 - 1	1
SMTPAuth	0 - 3	0
SMTPCheckAccept	0 - 1	0
SMTPCheckFrom	0 - 1	0
SMTPConnectScript		SMTPConnectScript.mml
SMTPCorrectFrom	0 - 1	0
SMTPDATAScript		See text
SMTPDLL		
SMTPEHLOScript		EHLOScript.mml
SMTPEOMScript		EOMScript.mml
SMTPHELOScript		HELOScript.mml
SMTPKickPOST	0 - 1	1
SMTPLog	1 - 65535	219
SMTPLogonMsg		
SMTPMAILScript		MAILScript.mml
SMTPMaxConnections		0
SMTPPortReceive	1 - 65535	25
SMTPPortSend	1 - 65535	25
SMTPRCPTScript		RCPTScript.mml
SMTPResolveHostname	0 - 1	1
SMTPScript	0 - 1	1
SMTPSSL	0 - 2	0
SMTPThreads	1 - 256	20
SMTPTimeReceive		1200

Name	Range	Recommended value
SQLConnectionMode	0 - 2	1
StatisticsMessage	0 - 1	0
StatPassword		
TimeZoneName		GMT+0000BST
TNEFGetBufferSize	0-64	32
TNEFTcpBlockSize	0-64	0
TRAPToSupport	0 - 1	1
TreatAsLocal		
UserAccessList		
UserDLLLocation		None
WatchHostname		localhost
WatchPort	1 - 65535	22200
WebMailAccessIP		
WebMailAccessIPDenied	0 - 1	0
WebMailAutoDisplayNameFormat	0-2	0
WebMailLog	1 - 65535	18
WWWAdminMMLPort	1 - 65535	8000
WWWCacheDir		
WWWConnectionRetryAttempts		5
WWWConnectionRetryTimeOut		500
WWWCUList		
WWWDefaultPageName		index.htm
WWWErrorPageName		error.htm
WWWGetBufferSize	0-64	32
WWWFtpAllowed	0 - 1	1
WWWFtpExpress	0 - 1	1
WWWHostname		
WWWLog	1 - 65535	18
WWWMaxConnections	0 - 256	0
WWWMaxSessions	1 - 9999	1000
WWWMIMETypes		
WWWMMLDefaultPageName		index.mml
WWWMMLErrorPageName		error.mml
WWWMMLRoot	1 - 65535	
WWWMMLScriptPort		8025
WWWPassword		
WWWPort	1 - 65535	80
WWWProxyPort	1 - 65535	8080

Name	Range	Recommended value
WWWRequestLogDir		<basedir>WWWLog
WWWRequestLogTypes	0 - 2	0
WWWResolveHostname	0 - 1	1
WWWSessionTimeOut		600
WWWShareSessionsAcrossPorts	0 - 1	1
WWWTcpBlockSize	0-64	0
WWWThreads	1 - 256	32
WWWTimeClient		300 secs
WWWUserMMLDefaultPageName		index.mml
WWWUserMMLDir		gordano\mml\usr
WWWUserMMLPort	1 -65535	8888
WWWUseSessionCookies	0 - 1	1
WWWUseSessionIPAddress	0 - 1	1

AccessIP

A list of IP addresses that are allowed access to the Configuration Pages (GUI). This is ignored if AccessIPDenied is set to 1.

AccessIPDenied

If set to the default 0 then only addresses listed in AccessIP will be allowed access to the Configuration Pages (GUI)

AdminBackgroundBackdrop

The background color displayed in the configuration screens. Values are based on the standard method of displaying colours in HTML, i.e. red, white, etc. For more control you can use an RGB set as well as for example #000000.

AdminMenuTitleColour

The color of the menu title bar displayed in the configuration screens. Values are based on the standard method of displaying colours in HTML, i.e. red, white, etc. For more control you can use an RGB set as well as for example #000000.

Administrator

Determines the email address to be used as the administrator address in the event that the postmaster account is not found.

AlertMaxEntries

The maximum number of entries to show in the Alert table at any one time.

AlertPort Range 1 - 65535, default = 8081

The port that the server uses for processing Alerts

AlertsRequired Range 1 - 65535, default = 1021

The range of alerts that you are interested in seeing.

AllowAdminGuiServer Range 0 - 1, default = 1

Enables or disables the administration interface. Normally this should never be changed as you will require access to the administration interface from time to time.

AllowAPOPlogin Range 0 - 2, default = 0

Disables, allows or forces APOP login. This encrypts passwords, making it more difficult to hack into the system by "sniffing" passwords from TCP packets transferred to/from your server. Possible values are:

- 0 — disables APOP login.
- 1 — lets the POP3 server announce that it can accept APOP encrypted passwords.
- 2 — makes APOP login mandatory. If you set this, ensure that all your subscribing POP clients support APOP.



APOP passwords cannot be used with NT User Database accounts.

AllowCollaborationServer Range 0 - 1, default = 1

Determines whether or not the GMS Collaboration Server is allowed to run. Disabling this option will stop Microsoft Outlook clients being able to use the advanced features provided by the GMS Collaboration Server.

AllowDailyStats Range 0 - 1, default = 0.

If this is set to 1, a message is sent to Gordano Ltd. at midnight every night, indicating the number of messages sent and received by the server over the past 24 hours.

AllowEditScripts Range 0 - 1, default = 0.

Enables/disables editing of scripts on the server.

AllowEditVariables Range 0 - 1, default = 0.

Enables/disables editing of variables on the server.

AllowEtrn Range 0 - 7, default = 2.

Enables/disables the ETRN SMTP command; see "ETRN" on page 199. The values are as follows:

Bit	Value	Meaning
0	1	requires the password in plain text.
1	2	requires the password to be encoded, as in APOP. (NTMETRN takes care of this for you.)
2	4	allows ETRN to work with dynamic IP addresses. To make this work, set a password and specify "*" as the destination server in the Sending Rules. The mail is dequeued to the IP address which sent the command.

For example to allow all three options set AllowEtrn=7. To disallow ETRN completely set AllowEtrn=0.

AllowedRCPTLimit Range 1 - 500, Default = 100.

The maximum number of RCPT clauses allowed in one SMTP transaction. GMS Anti-Spam lets you specify numbers for different addresses, to produce an entry like this:

123.89.89.89:12 123.19.24.88:36

AllowedRCPTLimitMes

The message returned to a transaction failing the RCPT Limit test. The default is "442 - Too many RCPT clauses".

AllowESMTP Range 0 - 1, default = 1.

Enables/disables Enhanced SMTP commands. Setting this to 1 enables the VRFY, DSN, Enhanced Status Codes, ETRN, Size and 8Bit-MIME commands. Other ESMTP commands are disabled by default, and must be enabled independently.

AllowFaxDLL Range 0 - 1, default = 1

Enables or disables the ability to configure the Fax DLL.

AllowFingerServer Range 0 - 1, default = 1.

If this is set to 1, the POP service runs a finger server. This server lets users **finger** the server to find out how much mail is waiting for a particular user, and also view that user's plan, if they have one.

AllowHelp Range 0 - 1, default = 1

Enables or disables display of the help icon within the administration interface.

AllowIMServer Range 0 - 1, default = 1

Enables or disables the GMS Instant Messenger Server. Note that disabling the GMS Instant Messenger Server will also stop access to other functions such as the sending of alerts via IM to filter actions, etc.

AllowLookup Range 0 - 1, default = 1.

Enables/disables lookups on the system. Enabling lookups lets people connecting to the Web Configuration server look up your company information — Company Name, Telephone Number and Fax Number — if you entered these.

In addition to this:

- Individual domains have their own setting.
- Each user can decide individually whether to allow lookups on themselves.

AllowManuals Range 0 - 1, default = 1.

Enables/disables display of PDF manuals using the Manuals button on the Help bar.

AllowNTInfo Range 0 - 1, default = 1.

Enables/disables use of NT SAM user profiles. If you are using user profiles for your NT SAM users the mail server will attempt to save the users mailbox in the users home directory. You may need to set the relevant Share rights for this directory. If this option is disabled the server will simply use the normal mailbox location.

AllowNTPassword Range 0 - 1, default = 1

Allows accounts set up within the GMS Administration interface to use their NT SAM passwords during authentication. Note that this is not the same as enabling use of the NT SAM for user authentication, this is for password purposes only.

AllowPasswordServer Range 0 - 1, default = 1.

If this is set to 1, the POP service runs a password server that lets Eudora users log in and change their password. If a user changes their password and prefixes it with a dash, the encrypted version is written back into the Registry.

AllowProxyServer Range 0 - 1, default = 1.

Enables/disables the proxy server.

AllowReturnReceipt Range 0 - 1, default = 1.

If this is set to 1 the SMTP server can send acknowledgement of the receipt of mail messages.

AllowReverseProxyServer Range 0 - 1, default = 1

Enables or disables the Reverse Proxy Server.

AllowScriptServer Range 0 - 1, default = 1

Enables or disables the Script Server.

AllowSSLProxyServer Range 0 - 1, default = 1

Enables or disables the SSL Proxy Server.

AllowUserGUIServer Range 0 - 1, default = 1

Enables or disables the User GUI Server. This is used if you wish to develop your own MML interface and provide it to your users.

AllowVRFY Range 0 - 2, default = 0.

Enables/disables use of the SMTP VRFY command. This command lets external servers check that an e-mail account actually exists on your server. A setting of 1 enables VRFY for a single account per transaction, a setting of 2 allows the use of the wildcard * to obtain a list of all users.

AllowWebMailGUIServer Range 0 - 1, default = 1

Enables or disables the WebMail Server interface.

AllowWWWServer Range 0 - 1, default = 0.

Enables/disables the Web server.

AllowXTND Range 0 - 1, default = 0.

Enables/disables use of the ESMTP XTND command.

AutoportHost

The fully qualified name or IP address of the server that you are porting from.

BlockingIO Range 0 - 6, default = 0

Defines whether TCP calls should use non-blocking IO as opposed to the default blocking mode. This should only be enabled if instructed to do so by Gordano support. The values are a bitmask with each bit affecting an individual GMS Service.

Bit	Service
0	POP
1	POST
2	SMTP

Bit	Service
5	IMAP
6	WWW

CollaborationTcpBlockSize Range 0-64, default 0

The size of TCP block that IMAP transmits data in, a value of 0 indicates the default settings for your TCP/IP stack will be used. We would recommend that if you change this setting you stay with multiples of 8 as this seems to give the best throughput. Changes require a server restart.

DeliveryDLL

Full pathname of the DLL used to act on delivery of mail.

DeliveryScript Default = DeliveryScript.mml.

Name of the Delivery Script used to act on delivery of mail.

DialupActions

A space-separated list of names of actions which can be linked to a dial-up schedule. Actions are set up by users.

DialupLog Range 1 - 65535, default =18

The level of logging for the dialup process.

DialupRequested Range 0 - 1, default = 0.

This is automatically set to 1 when dial-up is requested during installation.

DisplaySupport Range 0 - 1, default = 0.

Enables/disables display of the Support menus.

DNSAliases

A space-separated list of aliases.

DNSCacheDir

The directory where the DNS cache is held. The mail server stores DNS entries in a cache and when it transmits mail it checks this DNS cache to see if it has already looked up the DNS information it needs. Setting up a DNS Cache will greatly speed up DNS requests for frequently requested domains.

DNSCacheExpiry Default = 24.

The number of hours before an item is flushed from the cache. Do not make this too long or out of date information may be retrieved from the cache.



You can set the expiry and flush times to any value but you must be aware that DNS records are constantly changing and, if the times are set too high, it is likely that the DNS records for a domain will have changed before the cache is refreshed and the mail server will be trying to deliver e-mail to the wrong host

DNSCacheFlush Default = 5.

The number of days before the DNS cache is completely flushed to remove any potentially out of date information. If the DNS server needed for the MX lookup is not available, the previous MX record information can be stored until the cache expiry time is reached.

DNSCacheSize Default = 1024

The size in KB allowed for the DNS cache. You should not normally have to change this from the default. If the mail server cannot service a DNS request from the cache, it automatically contacts the specified DNS Servers to do so.

DNSCacheZeroTTL Default = 1.

The cache time to live in hours.

DNSDomain

The NT Domain used for DNS requests.

DNSHost

The host to use for DNS MX backup identification.

DNSMaxCNameDepth Range 1 -100, default = 10.

A CName is really an alias. In addition to its A record for mail.domain.name, a machine can have many associated CNames, such as WWW, server, machine, etc. Also, one CName can point to another. The depth parameter controls the number of CNames examined during NS lookup.

Reasons to change this:

- High — setting this high really slows down DNS lookups for any domain that has a lot of CNames set for a server.
- Low — setting this low speeds up DNS lookups.

DNSMinNumDots Default = 1.

The minimum number of dots in a domain name.

DNSOptions Range 0 - 224, default = 0.

Sets the options for DNS Lookups, by default the mail server will use UDP only for DNS lookups. If you are having problems with lookups truncating then enable the option "UDP then TCP if data truncated"

DNSPort Range 1 - 65535, default = 53.

The port that the mail server uses for DNS when resolving MX records.



Changing the DNSPort entry causes MX lookups to fail unless you have a DNS server which supports the new port number.

DNSQueryOptimise Range 0 - 1, default = 1.

Enables/disables optimization of DNS requests.

DNSServers

A space-separated list of DNS servers. If the list is empty, the servers entered in the system TCP/IP configuration are used. If more than one server is defined, a query is sent to each server in turn to spread the load.

DNSTimeClient Default = 30.

Timeout for DNS requests in seconds.

Reasons to change:

- High — if you set this high the server takes a long time to return if the DNS server is down.
- Low — if you set this low the server returns more quickly, but do not change it from the default unless you have problems.

DNSUseHOSTSFile Range 0 - 1, default = 1.

Enables/disables use of the hosts file for DNS requests. If this is enabled, an attempt will be made to resolve names using the local HOSTS file before attempting to resolve them using DNS.

EnforceCRLF Range 0 - 2, default = 1.

A value of 0 allows messages with mis matched CR or LF characters to enter the server unhindered. A value of 1 will attempt to fix any mis matches so that the end user only ever receives messages to their mailbox containing CR and LF pairs. A value of 2 will cause any messages containing mis matched CR or LF characters to be rejected.

EnforceDSN Range 0 - 1, default = 1.

Enforces use of the Delivery Status Notification ESMTP command.

EnforceReturnPath

Forces a return path header to be inserted into the headers of incoming

emails if it is not already present.

EnforceRFC822Header Range 0 - 8, default = 6.

This controls treatment of headers:

- 0 — takes no action with non-RFC822 headers.
- 1 — rejects messages whose headers do not conform to RFC822.
- 2 — inserts a From: clause if this is necessary.
- 4 — inserts a To: clause if this is necessary.
- 8 — always inserts a To: clause.

ESMTP8BitMIME Range 0 - 1, default = 1.

Enables/disables the 8BitMIME ESMTP command; see “8BitMIME” on page 198.

ESMTPAuth Range 0 - 3, default = 0.

Enables/disables the Auth ESMTP command; see “Auth” on page 198.

ESMTPDeliveryStatus Range 0 - 1, default = 1.

Enables/disables the Delivery Status Notification ESMTP command, which requests that the mail server confirm that a transaction was completed as desired. For more details, see “Delivery Status Notification (DSN)” on page 198.

ESMTPEnhancedStatusCodes Range 0 - 1, default = 1.

Enables/disables the Enhanced Status Codes ESMTP command, which gives precise error codes relating to the delivery of mail. These are only delivered to servers issuing the EHLO command to indicate that they understand ESMTP — all other servers receive the standard response codes. For full details, see “Enhanced Status Codes” on page 199.

ESMTPPipeLining Range 0 - 1, default = 0.

Enables/disables the Pipelining ESMTP command; see “Pipelining” on page 200. This setting and ESMTPRestart are mutually exclusive.

ESMTPRestart Range 0 - 1, default = 0.

Enables/disables the Restart ESMTP command, allowing failed SMTP transactions to be restarted from the point that they failed. See “Restart” on page 200. This setting and ESMTPPipelining are mutually exclusive.

SMTPSize Default = 10MB.

Limits the size of messages accepted for a particular domain. This saves bandwidth because messages exceeding the value are rejected before they ever reach your server. If you set MaxMessageSize, set SMTPSize to the same level. SMTPSize can be used on its own but only works with servers that support ESMTP; see “Size” on page 201.

Reasons to change:

- High — set this high to allow large messages to be accepted by your server. (A setting of 0 allows unlimited message sizes).
- Low — set this low if you want to limit the maximum message size accepted by your server.

FaxDLL

The full path of the DLL used to handle faxes.

FAXPageLength Default = 2340.

The fax page length, measured in pixels (A4 = 2340).

FAXPageToLength Range 0 - 1, default = 0.

If this is set to 1 the length of the fax page is reduced to fit the message.

FAXPageWidth Default = 1728.

Sets the fax page width, measured in pixels (A4 = 1728).

FAXPriority Range 0 - 3, default = 0.

The priority of the fax conversion, from 0 (lowest) to 3 (highest):

- 3 — stops the machine for about two seconds per fax page, while the conversion is done.
- 0 — does the conversion as a ‘background’ process when nothing else is waiting. This value is recommended.

FilterScript Default = FilterScript.mml.

Name of the script used to act on filtering of mail.

FingerPort Range 1 - 65535, default = 79.

The port allocated for the Finger server.

FolderDisconnectIdle Range 0-2, default=0

The Blackberry Internet Service has recently changed its behavior by permanently connecting to IMAP accounts using the IMAP IDLE facility. This has the effect of permanently locking out other protocols, such as POP and WebMail from accessing any message store held open by the Blackberry service.

Setting FolderDisconnectIdle to a value of 1 will silently close any IMAP connections that are in the IDLE state allowing the other protocols access without any effect on the Blackberry Internet Service.

Setting FolderDisconnectIdle to a value of 2 will silently close any IMAP connections no matter what state they are in. This will likely invoke an error in the IMAP client, but may be useful in situations where an IMAP client has been left logged in to the account but access is needed from outside the client location, most likely via WebMail.

IMAP itself is a multiple access enabled service so is not affected.

IMAPFetchBufferSize Range 0 - 64, default 32

In Version 12 of GMS IMAP buffers data in 32K blocks and then sends the whole block at once. In normal situations this is fine but there are circumstances where this may need to be tuned due to some networks seeing "delayed ACKs" at the packet level

We have tried various buffering options to see what the effect is and there is no simple answer. For some 8K buffers were best but for us we found 32K buffers were best. Unfortunately this is just trial and error.

Using odd buffer sizes can allow you to tune the performance between TCP transmission and processor usage. For instance selecting a 32K buffer size allows the packet to be transmitted very quickly but with a corresponding increase in CPU usage. Using a 10K buffer size slows down TCP transmission considerably but also provides significant reductions in CPU usage.

IMAPLog Range 1 - 65535, default = 18.

The level of logging for IMAP. The higher the number, the more information which is logged.

Reasons to change:

- High — set this high if you are having trouble with the IMAP service and want to see what is happening.
- Low — if you are not experiencing any problems with IMAP set this low, otherwise very large log files will be created.

IMAPLogonDelay

This is used in protection against dictionary attacks. IMAPLogonDelay specifies the increment used when increasing delays between failed attempts to logon.

IMAPMaxConnections Range 1 - 255.

The number of simultaneous IMAP connections allowed from a single host.

Reasons to change:

- High — set this high if you have heavy users of IMAP and need more than 20 simultaneous connections.
- Low — helps prevent denial of service attacks eating up the available IMAP threads.

IMAPPort Range 1 - 65535, default = 143.

The port the IMAP server listens on for the IMAP protocol.



Changing from the default values can prevent other services accessing the port.

Changes do not take effect until you stop and then restart the service.

IMAPTcpBlockSize Range 0-64, default 0

The size of TCP block that IMAP transmits data in, a value of 0 indicates the default settings for your TCP/IP stack will be used. We would recommend that if you change this setting you stay with multiples of 8 as this seems to give the best throughput. Changes require a server restart.

IMAPThreads Range 1 - 1000, default = 20.

The maximum number of threads allowed for the IMAP server. The value is ignored if it is not between 8 and 1000. This value can be set to a maximum of 255 via the GUI.



The number of threads controls how many simultaneous transactions the service can handle, so affects the performance of the mail server. As a general rule, the more threads you enable the more memory your server must have available.

IMAPTimeClient Default = 300.

The IMAP client inactivity timer value in seconds.

Reasons to change:

- High — setting this high means the server takes a long time to automatically expire threads that are not doing anything.

- Low — set this lower if you experience problems with IMAP threads being eaten up, for example in a denial of service attack

IncomingDLL

The full path of a DLL used to process incoming mail.

IncomingScript Default = IncomingScript.mml.

The name of the script used to act on incoming mail.

IPAllowed Default = *.

A space-separated list of banned IP addresses. Any messages from these addresses are immediately rejected. The list must begin with an asterisk (*), and each address must be preceded by an exclamation mark (!).

IPAllowedMes

The message returned to a transaction failing the IPAllowed test. The default is "542 - Your server has been banned from this server".

LocalIP

A space-separated list of IP addresses to be treated as local when mail relay is disabled. These computers will still be able to send external mail through the mail server.

The default is to treat as local a block of Class C addresses based on the IP address of the mail server's network card. You may want to amend this, for instance, if you only have a partial Class C address block or have more than one Class C address block.

LogAllMessages Default = 0

Logs all messages that are relayed through your system in the gordano\meslog directory.

LogAutoDelete Default = 0.

If disk space falls below this number of bytes at midnight, all the log files are deleted.

LogDeleteAfter Default = 5.

The number of days after which logs are automatically deleted.

LogEmailAfter Default = 0.

The number of days after which the logs are mailed to the given address.

LogonDelay Default = 0.

Sets a delay between the moment of connection and the mail server's first response. This is useful for POP clients which might be too slow to accept the response without a pause in the protocol. Enter the delay time in milliseconds. The default value, 0, disables the delay.

LogZipAfter Default = 0.

The number of days after which the log files are zipped up. The default value, 0, means never zip the files.

MailerMessage

The message which appears in the "X-mailer" line in a message header. This is normally the name "Gordano Messaging Suite " and its version number.

MailLogDLLLocation

The pathname of the DLL which may operate on logs once they are in the MesLog directory.

MaxAccountSize Range 0 - , default = 0.

The maximum disk space allocated to any one user for all their files, their inbox and any other mailboxes, etc. Mail received after this quota is exceeded is temporarily rejected.

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.
- Low — set it low if you are short of disk space and want to restrict the amount available to each user.

MaxBadCommand Default = 100.

The maximum number of bad commands allowed in one transaction. The Gordano server automatically disconnects when this number is reached.

Reasons to change:

- High — remote servers may sometimes send bad or incomprehensible commands to your server. The higher this setting, the more of these will be accepted before disconnection.
- Low — setting this to a low figure prevents your server threads being tied up talking to another server that sends lots of bad or incomprehensible commands to your server.

MaxBadMesDirSize Range is 0 - available space, default = 5MB.

The maximum size of the BadMes directory in bytes. If the total size of all the files in the directory exceeds this value at midnight, an

e-mail message is sent to the administrator telling them that it is filling up. A value of 0 means send no warning.

Reasons to change:

- High — saves the largest possible history of bad messages sent to your server.
- Low — conserves disk space.

MaxHopCount Range 1 - 50, default = 17.

The maximum number of hops a message is allowed to perform before being returned to sender. A message gains a "hop" every time it is transmitted from one server to another. A high number allows a large number of hops between mail servers but could cause excessive traffic on an incorrectly configured mail system (that is, internally bouncing mail or autoresponder e-mail). The default is 17 hops.

Reasons to change:

- High — you may need to set this higher if a message has to take a circuitous route to reach its destination.
- Low — prevents mail looping round and round, for example in the case of misconfigured DNS settings.

MaxHTTPDataLength

The maximum for data passed over http. Set to a lower figure to stop DOS attacks via large HTTP requests and POSTs.

MaxLineLen Default = 0.

The maximum number of characters on a line allowed in a mail message. Gordano servers can cope with line lengths of any size, so the only reason to change this is if you are passing mail on to other systems that cannot.

MaxLogDirSize Range is to available disk space, default = 10

MB.

The maximum size of the Log directory in KB. If the total size of all the files in the directory exceeds this value at midnight, an e-mail message is sent to the administrator, telling them that it is filling up. A value of 0 means send no warning.

Reasons to change:

- High — saves the maximum history of traffic to/from your server.
- Low — conserves disk space.

MaxMailboxSize Range 0 - , default = 0.

The maximum size of an inbox on the system in KB. (MaxAccountSize governs total disk space available to the user.) If it's zero or

non-existent there is no limit. If this value is exceeded at midnight, an e-mail message is sent to the administrator and any e-mail to the mailbox is rejected with the message '453 Try again later'.

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.
- Low — set it low to conserve disk space, restricting the amount available to each user.

MaxMessageSize Range 0 - , default = 0 (no limit).

The maximum size of *incoming* message that will be accepted for any user on the server, in KB. If the message exceeds this limit it will be permanently rejected and a message returned to the sender informing them of this. Use this to prevent people sending excessively large messages.



If you set MaxMessageSize, set SMTPSize to the same level.

Reasons to change:

- High — set this high if you have lots of disk space.
- Low — set it low if you are short of disk space and want to restrict the message size that can be received by each user, or if you are short of bandwidth.

MaxMessagesPerIPAddress

The maximum number of messages allowed from any remote IP address in any 24 hour period. This helps prevent Spam.

Reasons to change:

- High — set this high if you regularly receive a lot of messages from the same IP addresses.
- Low — set it low to prevent any computer flooding your server with messages.

MaxMessagesPerIPAddressMes

The message returned to any transaction failing the MaxMessagesPerIPAddress test. The default is "453 Too many messages from you today".

MaxMessagesPerUser

The maximum number of messages allowed per user per day. This helps prevent Spam.

Reasons to change:

- High — set this high if you want a fairly lenient limit on the number of messages accepted for each user in any 24 hour period.

- Low — set it low if you want to restrict the number of messages any user can receive in any 24 hour period.

MaxMessagesPerUserMes

The default message returned to any transaction failing the MaxMessagesPerUser test. The default is "453 Too many messages to this user today".

MaxNumberFilesInCache

The maximum number of files which can be held in the proxy cache. Once this is reached, old files are deleted as required to make room for new files.

Reasons to change:

- High — saves on activity on your line. If you are big users of the Web, you may want to set this to a fairly high number and rely on the other cache expiry options to control how long files are cached for.
- Low — if you set this too low and a live page on the Web changes, you may not realise this since the copy held in cache will be used instead.

MaxPasswordLetterOccur Range 0 - 1, default = 0.

The maximum number of times any particular character can appear in a users password.

MinDiskSpaceRequired default = 10.

Specifies the minimum amount of free disk space required by an installation of Gordano products.

MinPasswordLen default = 5.

Specifies a minimum length for all passwords on the system.

MMLErrorToSupport Range 0 - 1, default = 1.

If this option is enabled all MML errors reported will be automatically forwarded back to Gordano Limited so they can be assessed and corrected.

MSMailAddressFix Range 0 - 1, default = 0.

This corrects e-mail addresses for MSMail gateway. If set to 1, it makes the Gordano server check for the account NULL or an empty e-mail address in the protocol and replace it with the 'From:' or 'To:' clause of the message itself.



While this option gives MSMail users the warm feeling that they know where the message came from, it is potentially dangerous because the To: clause may not contain the real destination address of the e-mail.

NameServers

A list of space-separated dotted-decimal addresses of DNS servers to use. If this parameter is specified, those defined by NT are ignored.

NextCookie

Used to track the tips/quotes which appear in the Help bar, if enabled.

NextNtListCookie

Controls which is the next cookie displayed in the help bar in the GLCommunicator GUI. These are taken from ntlistcookies.txt in the basedir

NTComputer

The name of the computer holding the NT SAM User Database that the Gordano server should use for validation of users and passwords. If this entry is not defined, the local machine is used for validation.

NTDomains

A space-separated list of domain names for user and password validation on Windows NT. If no domain is defined, the server asks Windows NT which domain the user belongs to.

OutgoingDLL

The full path of the DLL used to process outgoing messages.

OutgoingScript

The script used to act on outgoing mail. The default is OutgoingScript.mml.

PasswordPolicy

Bit map defining the password policy for all users:

1 = password must have a digit.

2 = password must have a letter.

4 = password must have a symbol.

PasswordPort Range 1 - 65535, default = 106.

The port the password server listens on for password request changes.

POPBandwidth Default = 0.

The bandwidth available for POP users. The default of 0 means use all available bandwidth.

Reasons to change:

- High — you may want to set this to a very high value, close to the capacity of your connection.
- Low — set it to a lower figure if you want to ensure that some of your bandwidth is always preserved, for example for browsing the Web.

POPLog Range 1 - 65535, default = 279.

Level of logging for the POP service.

Reasons to change:

- High — set this high if you have trouble with the POP service and want to see what is happening.
- Low — if you do not experience any problems with POP set this low, otherwise very large log files will be created.

POPLogonDelay

This is used in protection against dictionary attacks. POPLogonDelay specifies the increment used when increasing delays between failed attempts to logon.

POPMaxConnections Range 1 - 256, default = 1.

The maximum number of simultaneous POP connections allowed from a single IP address.

Reasons to change:

- High — set this high if you have many users accessing your server from the same IP address, that is, if they are coming through a proxy or firewall.
- Low — this helps prevent denial of service attacks on POP, as normally you would only expect a single connection from any one IP address.

POPPort Range 1 - 65535, default = 110.

The port number the POP server listens on for the POP3 protocol.

POPReetrBufferSize Range 0 - 64, default 32

In Version 12 of GMS POP buffers data in 32K blocks and then sends the whole block at once. In normal situations this is fine but there are circumstances where this may need to be tuned due to some networks seeing "delayed ACKs" at the packet level

We have tried various buffering options to see what the effect is and there is no simple answer. For some 8K buffers were best but

for us we found 32K buffers were best. Unfortunately this is just trial and error.

Using odd buffer sizes can allow you to tune the performance between TCP transmission and processor usage. For instance selecting a 32K buffer size allows the packet to be transmitted very quickly but with a corresponding increase in CPU usage. Using a 10K buffer size slows down TCP transmission considerably but also provides significant reductions in CPU usage.

POPTcpBlockSize Range 0-64, default 0

The size of TCP block that POP transmits data in, a value of 0 indicates the default settings for your TCP/IP stack will be used. We would recommend that if you change this setting you stay with multiples of 8 as this seems to give the best throughput. Changes require a server restart.

POPSaveOnDisconnect Range 0 - 1, default = 0.

If this is set on, when a disconnect occurs the user's mailbox is treated in the same way as if they had quit. That is, messages they have read are deleted.

POPThreads Range 1 - 256, default = 20.

The maximum number of threads allowed for the POP server. The number of threads used can be surprisingly low unless there are lots of users on slow (for example, dial-up) connections.



The number of threads controls how many simultaneous transactions the service can handle, so affects the performance of the mail server. As a general rule, the more threads you enable the more memory your server must have available.

Reasons to change:

- High — set this high if you have many users and plenty of bandwidth between the clients and the server.
- Low — set it low if you do not have a lot of memory available.

POPTimeClient Default = 300.

The time in seconds the POP server gives POP clients before automatically logging them off due to inactivity.

Reasons to change:

- High — set this high if your users experience timeouts downloading their mail (these are more likely if they get a lot of large mail messages). If this is the case, your users should also increase the timeouts in their client software.
- Low — helps prevent POP threads being used up for long periods of time, so prevents denial of service attacks.

POSTBandwidth Default = 0.

The bandwidth available for POST users. The default of 0 means use all available bandwidth.

Reasons to change:

- High — if you want to set this to a very high value, close to the capacity of your connection, but not use all available bandwidth.
- Low — set this to a low figure if you want to ensure that some of your bandwidth is always preserved, for example for Web browsing.

POSTLog Range 1 - 65535, default = 791.

The level of logging for post service.

Reasons to change:

- High — set this high if you are having trouble with the POST service and want to see what is happening.
- Low — set this low if you do not experience any problems with POST, otherwise very large log files will be created.

PostMasterStats Range 0 - 1, default = 0.

Enables/disables the sending of the statistics message to the post-master at midnight.

POSTMaxContLine Default = 0.

The maximum number of responses to a command which POST will accept. The default of 0 means there's no limit.

Reasons to change:

- High — set this high only if you have a known problem with a remote SMTP server.
- Low — helps prevent your post thread being tied up for a long period of time by a remote SMTP server, for example in the case of *tar pitting* attacks.

Tar pitting occurs when you post to a remote SMTP server, and it responds to the POST server commands very slowly, tying up your POST threads.

It normally takes the form of them sending multi-line SMTP responses with one line being sent every minute or two. This could go on for hours, but setting PostMaxContLine ends it after the set period.

OutDir Default = basedir\out.

Defines the directory where post queues messages waiting to be posted outside the server.

PostRetryTime Range 0 - 10080, default = 12.

The delay in minutes between attempts at sending e-mail messages to servers that could not be reached on a previous attempt because of, for example, a DNS or mail server failure. If this is set to zero, POST never checks to see if there is mail waiting to be sent and you must either use SMTP to tell POST about new mail or use 'MAIL -k'. The upper limit, 10080 minutes, is seven days.



*This value cannot be changed from 0 to a definite time (or vice versa) by the 'MAIL -u' command. To effect this change, you **must** stop and restart the server.*

Do not set this on if a dial-up connection is being used.

Reasons to change:

- High — if you have a very busy server you may want to increase this to reduce server activity.
- Low — if all your mail is urgent you can reduce this value to make sure that any mail that fails to be delivered is retried again fairly quickly. We do not recommend setting a value lower than five minutes. Be careful if you have a busy server as this can increase server activity dramatically.

PostReturnTime Range 1 - 744, default = 72.

The time in hours after which mail is returned to a user if it cannot be delivered.

Reasons to change:

- High — increase this if you do not have a very busy server and do not mind how long it takes mail to fail completely. Bear in mind, though, that mail will be retried according to the PostRetryTime for the duration of this period.
- Low — set this low if all your mail is urgent and you want to know as soon as possible if it fails. Do not make this interval too short or e-mail will be returned to a sender before a temporary DNS or mail server fault can be rectified. We do not recommend values under 24 hours as it can take at least this time for mail to be delivered over the Internet — if the message is very urgent e-mail is not the best way to send it!

POSTServers Default = postservers.txt.

Pathname of a file containing a list of post servers. This tells the POST service how to send the mail. The file is space-, colon- or semi-colon-separated and lower entries in the file override earlier entries.

Use of this parameter depends on the type of connection:

- Permanent connection — use an entry like the following:

* mail.isp 25 10 0 password name

Where:

- * — means for mail to all domains.
- mail.isp — is the ISP's server.
- 25 — is the port number.
- 10 — is the retry time in minutes.
- 0 — is the SSL setting (0 - don't use SSL, 1 - use SSL and fall back to normal if SSL fails, 2 - enforce use of SSL)
- password — encrypted password if authentication to server is required.
- name — account name if authentication to server is required.
- Dial-up connection — use an entry like the following:

* mail.isp 25 0 0 password name

Where the entries are as for a permanent connection, except for the retry value of 0, meaning do not retry.

- Local domains — requires a double entry like the following:

domain 127.0.0.1 25 10 0

* domain

Where the port number and retry parameters are as for a permanent connection. The second entry covers any subdomains.

PostSSL Range 0 - 2, default = 0

Determines whether or not Post uses secure socket layer.

0 - don't use SSL

1 - use SSL and fall back to normal if SSL fails

2 - enforce use of SSL

Individual post queues can have their own SSL setting which overrides the POSTSSL setting see "POSTServers Default = postservers.txt." on page 81.

POSTTcpBlockSize Range 0-64, default 0

The size of TCP block that POST transmits data in, a value of 0 indicates the default settings for your TCP/IP stack will be used. We would recommend that if you change this setting you stay with multiples of 8 as this seems to give the best throughput. Changes require a server restart.

POSTThreads Range 1 - 256, default = 20.

The maximum number of threads allocated for the POST server.



The number of threads controls how many simultaneous transactions the service can handle, so affects the performance of the mail server. As a general rule, the more threads you enable the more memory your server must have available.

Reasons to change:

- High — set this high if you have plenty of bandwidth and a large number of users who send a lot of mail.
- Low — set it low if you do not have a lot of memory available to you, otherwise keep the default.

POSTTimeSend Default = 1200.

The time in seconds the POST server waits for an SMTP server to respond (measured to the nearest 10 seconds). The default, 1200, is 20 minutes.

Reasons to change:

- High — set this high if you see timeouts when you send mail.
- Low — set this low if you find that threads are being held idle by the remote SMTP servers. Reducing the value slightly may help.

PostWarningTime Default = 24.

The time in hours after which a warning is generated and sent to the original sender of an e-mail if it has not been delivered during this period.

Reasons to change:

- High — increase this if you do not have a very busy server and do not mind how long it takes mail to generate a warning.
- Low — set it low if you want to be warned quickly if mail is not sent out. We do not recommend a setting of less than four hours as it is not unusual for mail to take at least this long to be delivered. This could be for any number of reasons, such as connectivity to the remote server being unavailable for a period.

ProductsInstalled

A bit map showing which Gordano products have been installed

Bit	Value	Meaning
0	1	GMS Mail
1	2	List
2	4	GMS Anti-Spam
3	8	GMS Anti-Virus

Bit	Value	Meaning
0	1	GMS Mail
4	16	Proxy
5	32	GMS WebMail

ProxyCacheSize

Displays the size of the proxy cache. This value should not be edited and is for information only. See ProxyCacheSizeMax to control the size of the cache.

ProxyCacheSizeMax Default = 32768.

The maximum size of the Web proxy server cache.

Reasons to change:

- High — set this high if you do a lot of Web browsing to the same sites and want to reduce the use of your Internet connection.
- Low — set it low to save disk space and to stop saving too many files in the cache. When the maximum cache size is reached, the oldest files are replaced first.

ProxyPurgeAge Default = 7.

The number of days files are stored in the Web proxy server before purging.

Reasons to change:

- High — set this high if you do a lot of web browsing to the same sites and want to reduce the use of your Internet connection.
- Low — set it low to save disk space and to stop saving too many files in the cache, especially if the pages change frequently.

ProxyPurgeAgeLast

The date and time the files were last purged.

ProxyPurgeMethod Range 0 - 1, default = 1.

The method of purging the Web Proxy cache, one of these:

- 0 — by date cached.
- 1 — by date last used.

ProxyRASAccName

Proxy RAS primary account name. The primary account is used for contacting an ISP.

ProxyRASAccName2

Proxy RAS backup account name. The backup account is used for e-mail.

ProxyRASBackup

The proxy RAS backup.

ProxyRASPassword

The password for the primary Proxy RAS account.

ProxyRASPassword2

The password for the backup Proxy RAS account.

ProxyRASPhoneNumber

The telephone number of the primary Proxy RAS account.

ProxyRASPhoneNumber2

The telephone number of the backup Proxy RAS account.

ProxyRASSelected Range 0 - 1, default = 0.

Enables/disables proxy RAS.

QuarantineAdminPort

The port used to access a users Quarantine folder (Vanguard).

QuarantineAdminServer

The fully qualified name (or the IP address) of the server (Vanguard). By default this uses the IP address of the server, so will need changed if installed on a private IP Address.

RASAccName

The RAS primary account name. This account is used for contacting an ISP.

RASAccName2

The RAS backup account name. This account is used for e-mail.

RASDisconnectDelay Default = 120.

The period in seconds the connection is kept open for after sending and receiving of e-mail finishes.

RASMinScheduleTimeout

The minimum time a dial-up connection is kept open for in seconds.

Specify a value if your ISP's SMTP only transmits e-mail to you after you have been logged on for a certain time. Set a minimum time which is greater than the period the ISP uses.

RASMaxScheduleTimeout

The maximum time a dial-up connection is kept open for in seconds.

Set a maximum connection time if you want to use this as a fail-safe. This is useful as it prevents a faulty connection hanging indefinitely.

RASPassword

The password for the primary RAS account.

RASPassword2

The password for the backup RAS account.

RASPhoneNumber

The telephone number of the primary RAS account.

RASPhoneNumber2

The telephone number of the backup RAS account.

RASProxyDisconnectDelay Default = 120.

The period in seconds the connection is kept open for after sending and receiving of e-mail finishes.

RCPTLimit Range 1 - 600, default = 100.

The maximum number of RCPT clauses that a remote POSTing server can use. The RFC maximum limit is 100. Junk e-mail (Spam) often uses a large To: clause to pass on its messages, so limiting this parameter reduces this potential problem. If you have GMS Anti-Spam installed, this setting can be overridden on a per-domain basis.

Reasons to change:

- High — there are not many reasons to set this it high, except that the RFC recommended setting is 100. We recommend a setting of 5 or 10.
- Low — setting this to a low number prevents Spammers using multiple RCPT clauses to send messages through your server.

RedirectFile Default = redirectfile.txt.

The name of the SMTP redirect file in the BaseDir. This file should contain a list of message redirection commands specifying whether messages coming from or going to certain addresses should be

redirected to another Gordano account. For more details see in the *GMS Administrator's Guide*.

ReturnReceiptAllow Range 0 - 1, default = 1.

This allows receipts to be returned automatically, if this is requested.

RFC1123Return Range 0 - 1, default = 1.

Enables/disables the returning of mail according to RFC1123.

SaveBadMes Range 0 - 1, default = 1.

If this is set to 1, the Gordano server saves messages sent to the list server to NULL@ or from NULL. These are placed in the directory \BadMes, if this exists. A text file describing the problem is also saved.

SaveSetupFiles Range 0 - 1, default = 0.

Enables/disables saving of the recovery file.

SaveSetupSendTo

The address which the recovery file is sent to.

ShowRealHost Range 0 - 1, default = 1.

Enables/disables the display of the real host name in SMTP log.

SMTPAuth Range 0 - 3, default = 0.

Enables/disables the Auth SMTP command; see "Auth" on page 198.

SMTPCheckAccept Range 0 - 1, default = 0.

If enabled this option will prevent SMTP from responding on any IP's that are not configured for use by a domain in the Gordano products. If you would like to over-ride this option for a particular domain you can do so by adding the other IP addresses that SMTP should answer on to the list of aliases for the domain in question.

SMTPCheckFrom Range 0 - 1, default = 0.

If set on, the From clause is checked for a match. If there's no match, the message is rejected.

SMTPConnectScript Default = SMTPConnectScript.mml.

Name of script used to act on a connection to the SMTP port.

SMTPCorrectFrom Range 0 - 1, default = 0.

If this is enabled it checks that SMTP's MAIL FROM address is the same as the From: header in the message itself. It corrects the malformed SMTP FROM clause.

SMTPDATAScript Default = DATAScript.mml.

The script used to act at the DATA stage of an SMTP transaction.

SMTPDLL

The full path of the DLL used to act on SMTP transactions.

SMTPEHLOScript Default = EHLOScript.mml.

The script which acts on the ESMTP EHLO command.

SMTPHELOScript Default = HELOScript.mml.

The script which acts on the SMTP HELO command.

SMTPEOMScript Default = EOMScript.mml.

The script which acts on the End Of Message indicator in an SMTP transaction.

SMTPKickPOST Range 0 - 1, default = 1.

A flag to tell SMTP to signal POST when outbound mail is available. A value of 0 indicates that POST should not be kicked when new outbound mail arrives.

SMTPLog Range 1 - 65535, default = 219.

The level of SMTP Logging.

Reasons to change:

- High — set this high if you are having trouble with the SMTP service and want to see what is happening.
- Low — set this low if you do not experience any problems with SMTP, otherwise very large log files will be created.

SMTPLogonMsg

Allows for the inclusion of a message as part of the SMTP logon banner. Each line of the message should be preceded by "220-", followed immediately by the start of the message — there must be no space between the hyphen and the first word. An example might be:

```
220-Unsolicited commercial email will be rejected
220-Contact postmaster@domain.dom if rejected in error
```

SMTPMAILScript Default = MAILScript.mml.

The script used to act on the SMTP MAIL clause.

SMTPMaxConnections Default = 0.

The maximum number of simultaneous SMTP connections available from any IP address. The default of zero means no limit.

Reasons to change:

- High — set this high if you have many SMTP connections to your server from the same IP address, that is, they are coming through a proxy or firewall.
- Low — this helps prevent denial of service attacks on SMTP. Normally you would only expect a single connection from any one IP Address. (Many servers will open a new thread for each message they are sending, so if a remote host is sending 200 messages to your server they could conceivably use up 200 threads for the period that the messages are being transferred.)

SMTPPortReceive Range 1- 65535, default = 25.

The port the SMTP server uses to accept e-mail. The Internet Standard is to use port 25, but you may wish to run another mail server on the same machine and make it direct mail to the Gordano server at this new port.

SMTPPortSend Range 1 - 65535, default = 25.

The port the Gordano server uses to send all mail to destination servers. This affects all the mail the server sends. In general, you would only use this definition if you knew all mail was going to a specific port on a machine defined by the parameter PostServers.

SMTPRCPTScript Default = RCPTScript.mml.

The script used to act on the SMTP RCPT clause.

SMTPResolveHostname Range 0 - 1, default = 1.

If this is present and set to 0, the SMTP server does not attempt to resolve the name of any client sending mail. This is especially useful when Gordano servers are used in a network where there is no DNS server until a dial-up connection is made to the Internet. The default action is to try to resolve the name of the host sending the mail message.

SMTPScript Range 0 - 1, default = 1.

Enables/disables the use of the SMTP scripts SMTPConnectScript, SMTPDATAScript, SMTPPEHLOScript, SMTPHELOScript, SMTPPEOMScript, SMTPMAILScript and SMTPRCPTScript.

SMTPSSL Range 0 - 2, default = 0

Determines whether or not SMTP uses secure socket layer.

0 - don't use SSL

1 - use SSL and fall back to normal if SSL fails

2 - enforce use of SSL

SMTPThreads Range 1 - 256, default = 20.

Maximum number of threads for SMTP.



The number of threads controls how many simultaneous transactions the service can handle, so affects the performance of the mail server. As a general rule, the more threads you enable the more memory your server must have available.

Reasons to change:

- High — set this high if you have a large number of users who send a lot of mail and plenty of bandwidth.
- Low — set it low if you do not have a lot of memory available to you, otherwise leave at default.

SMTPTimeReceive Default = 1200.

The time in seconds the SMTP server waits before assuming that any server connecting to it has gone (measured to the nearest 10 seconds). The RFC specifies a time of about 20 minutes and this is the default.

Reasons to change:

- High — set this high if you get a lot of connections from really slow servers or clients, but note that increasing it does leave you open to denial of service attacks.
- Low — set it low if you do not have any problems with SMTP reception and feel this is safe. We do not recommend a value less than 10 minutes, but it should really be left at the default to comply with the RFC.

SQLConnectionMode

Defines how often Gordano products connect/disconnect to the integral MySQL server.

The 3 options are:

- 0 - Open/close SQL connection for every database transaction
- 1 - Open/close SQL connection for every WWW connection. i.e. The SQL connection is opened at the start of a WWW connection and closed at the end of the WWW connection. Multiple database transactions may take place in between.

- 2 - Open/close SQL connection on a WWW session basis. i.e. The SQL connection is opened when a session is created and closed when the session is destroyed.

Setting a value of 2 is the fastest but leaves the SQL connection open for the duration of the session. This can be used for installations with smaller numbers of users. The default setting is 1.

StatisticsMessage

Enables/Disables the sending of the statistics message to the postmaster at midnight

StatPassword

A string telling SMTP whether or not to accept the STAT command. This command shows statistics on the SMTP server usage and is issued after logging on to port 25 via telnet. Data include the number of accounts on the system, server uptime, etc.

The possible values are:

- '+' — the server displays a set of statistics when the command 'MAIL -q' is used. This is the default.
- '-' — SMTP rejects the command.
- Any other value — read as a password that must be specified to display the statistics. This lets you check the statistics but deny others access. The password can be encrypted.

TimeZoneName Default = GMT+0000BST.

Defines the time zone message that is written in time stamps. If this is not defined, GMT is used. Any number of letters can be used, but there should always be a plus or minus followed by four digits and a daylight saving string.

For example, the default, GMT+0000BST, causes GMT to be used during the winter months and BST for Daylight Saving Time.



The changeover time is as defined by the Windows NT operating system.

TNEFGetBufferSize Range 0 - 64, default 32

In Version 12 of GMS Collaboration buffers TNEF data in 32K blocks and then sends the whole block at once. In normal situations this is fine but there are circumstances where this may need to be tuned due to some networks seeing "delayed ACKs" at the packet level

We have tried various buffering options to see what the effect is and there is no simple answer. For some 8K buffers were best but for us we found 32K buffers were best. Unfortunately this is just trial and error.

Using odd buffer sizes can allow you to tune the performance between TCP transmission and processor usage. For instance selecting a 32K buffer size allows the packet to be transmitted very quickly but with a corresponding increase in CPU usage. Using a 10K buffer size slows down TCP transmission considerably but also provides significant reductions in CPU usage.

TRAPToSupport Range 0 - 1, default = 1.

Enables/disables the automatic sending of traps to Support.

TreatAsLocal

A space-separated list of Domain Names to treat as local. From version 7 this information is stored in the treataslocal.txt file rather than in the registry.

UserAccessList

A colon (:) separated list of users with specific access rights to the mail server over and above that of the normal user, in the form domain.name\account.

UserDLLLocation Default = none.

The location and name of a security DLL (see "User Authentication DLLs" on page 256). The specified DLL is used to validate unknown users.

WatchHostname Default = localhost.

The name of the machine that runs the Mailbox Monitor program; see "Mailbox Monitor" on page 36. If no name is supplied, the local machine is used.

Each of the Gordano services attempts to connect to the Mailbox Monitor every five minutes to send details of its activities.



After an 'update' to the services, it may take up to five minutes for the connection to be re-established.

WatchPort Range 1 - 65535, default = 22200.

The port that the Mailbox Monitor listens on.

WebMailAccessIP

A list of IP addresses that are allowed access to the GMS WebMail. This is ignored if WebMailAccessIPDenied is set to 1.

WebMailAccessIPDenied

If set to the default 0 then only addresses listed in WebMailAccessIP will be allowed access to the GMS WebMail.

WebMailAutoDisplayNameFormat Range 0-2, default=0

Uses the First Name and Last Name entries to autofill the Display Name of the Address Book entry when creating Address Book entries.

WebMailLog Range 1 - 65535, default =18

The level of logging for GMS WebMail.

WWWAdminMMLPort Range 1 - 65535, default = 8000.

The port used by the Web configuration server.

WWWCacheDir

The directory to be used by the Web Proxy service.

WWWConnectionRetryAttempts Default = 5.

The number of times the Web Proxy Service will attempt to connect to a site before failing.

WWWConnectionRetryTimeout Default = 500.

The period that the proxy waits for before retrying a failed connection in milliseconds.

WWWCUList

Used internally by the WWW service to display the status of each of the Gordano services. Do not edit.

WWWDefaultPageName Default = index.htm.

The default Web page.

WWWErrorPageName Default = error.htm.

The Web error page that is returned when an action cannot be completed, for example a page cannot be found.

WWWFtpAllowed Range 0 - 1, default = 1.

Enables/disables use of proxy FTP through the Web proxy server.

WWWFtpExpress Range 0 - 1, default = 1.

Enables/disables use of non-buffered FTP retrieval. This should be on.

WWWGetBufferSize Range 0 - 64, default 32

In Version 12 of GMS the WWW service buffers data in 32K blocks and then sends the whole block at once. In normal situations this is

fine but there are circumstances where this may need to be tuned due to some networks seeing "delayed ACKs" at the packet level

We have tried various buffering options to see what the effect is and there is no simple answer. For some 8K buffers were best but for us we found 32K buffers were best. Unfortunately this is just trial and error.

Using odd buffer sizes can allow you to tune the performance between TCP transmission and processor usage. For instance selecting a 32K buffer size allows the packet to be transmitted very quickly but with a corresponding increase in CPU usage. Using a 10K buffer size slows down TCP transmission considerably but also provides significant reductions in CPU usage.

WWWUseSessionCookies Range 0 - 1, default = 1.

Enables/disables use of cookies to allocate each user their own session id. If you do not use session cookies then 2 users coming from the same IP address will inherit each others access rights. If you log on to the WWW interface and are thrownback out as soon as you click on a button it could well be that you have cookies disabled in your browser.

WWWUseSessionIPAddress Range 0 - 1, default = 1.

Use this combined with the previous parameter to allocate each user their own session id. It should be enabled.

WWWHostname

Host name for WWW service.

WWWLog Range 1 - 65535, default = 18.

The level of logging for the WWW Service.

Reasons to change:

- High — set this high if you are having trouble with the WWW service and want to see what is happening.
- Low — set this low if you do not experience any problems with WWW, otherwise very large log files will be created.

WWWMaxConnections Range 1 - 256, default = 0.

The maximum number of simultaneous connections allowed to the Web server from a particular host.

Reasons to change:

- High — set this high if you have many Web connections to your server from the same IP address, that is, if they are coming through a proxy or firewall.

- Low — helps prevent denial of service attacks on the Web server. Normally you would only expect a single connection from any one IP address.

WWWMaxSessions

Specifies the number of simultaneous sessions available. This limit is implemented to stop systems using too much memory. The value can be set from 1 to 9999. Default is 1000. Each session requires that memory is allocated for it, therefore the greater number of sessions the greater amount of memory is required.

WWWMIMETypes

A space-separated list of MIME types.

WWWMMLDefaultPageName Default = index.mml.

Default page for the script server.

WWWMMLErrorPage Default = error.mml.

The default error page for the script server.

WWWMMLRoot

The path to the root of the script server.

WWWMMLScriptPort Range 1 - 65535, default = 8025.

The port used by SMTP, POST and MAIL to ask for scripts to be executed.

WWWPassword

The password required to access the WWW service.

WWWPort Range 1 - 65535, default = 80.

The port used by the Web server to listen for connections.

WWWProxyPort Range 1 - 65535, default = 8080.

The port used by the Proxy server.

WWWRequestLogTypes

IIS logging, the proprietary Microsoft log format. This is similar to CLF logging, with each request logged with a comma-separated line.

WWWRequestLogDir Default = <basedir>WWWLog.

The directory where the log files are written.

Log file names take the format xxYYMMDD.LOG, where xx is one of the following (for details of these formats, see the description of WWWRequestLogTypes):

- "IN" - IIS logs.
- "W3" - W3C logs.
- "CL" - CLF logs.

WWWResolveHostname Range 0 - 1, default = 1.

Allows the Web server to resolve the name of connecting hosts.

WWWSessionTimeout Default = 600.

The timeout for Web clients in seconds.

WWWShareSessionAcrossPorts Range 0 - 1, default = 1.

If this option is enabled then users logging in to MML on one port will be able to transfer to another port while maintaining the session information.

WWWTcpBlockSize Range 0-64, default 0

The size of TCP block that the WWW service transmits data in, a value of 0 indicates the default settings for your TCP/IP stack will be used. We would recommend that if you change this setting you stay with multiples of 8 as this seems to give the best throughput. Changes require a server restart.

WWWThreads Range 1 - 256, default = 32.

The number of threads available for Web server connections.



The number of threads controls how many simultaneous transactions the service can handle, so affects the performance of the mail server. As a general rule, the more threads you enable the more memory your server must have available.

Reasons to change:

- High — set this high if you have a large number of users accessing your server through the Web interface.
- Low — set it low if you do not have a lot of memory available.

WWWTimeClient Default = 300.

The time in seconds the Web server should give clients before automatically logging them off the system (due to inactivity).

Reasons to change:

- High — set this high if your users experience timeouts accessing the Web interface.
- Low — for security reasons you may want to keep this setting low. For example, if an administrator logs on and forgets to log

off again when they finish, another user coming from the same IP address would be able to gain administrator rights to the mail server. Reducing the timeout reduces the chance of this happening.

WWWUserMMLDefaultPageName Default = index.mml.

The default page name for any accesses made to user written MML pages. This will be returned if a directory is accessed and no file name is passed as part of the URL.

WWWUserMMLDir Default = gordano\mml\usr.

The base directory for all user generated MML.

WWWUserMMLPort Range 1 - 65535, default = 8888.

The port that the customer interface to MML runs on, this is used for access to any user generated MML placed in the base-dir\mml\usr directory.

WWWUseSessionCookies

Enables/disables cookies to maintain session information. Using this method of maintaining sessions is satisfactory provided of course that the user has not disabled the use of cookies in their browser, if they have the logon will be refused. It is recommended that WWWUseSessionIPAddress is also enabled.

WWWUseSessionIPAddress

Maintains sessions by noting the user's IP address. Care should be taken when selecting this option on its own, as a second connection from the same IP address will over-ride the settings of the first connection. This could occur for example if your users connect through a proxy server. It is recommended that WWWUseSessionCookies is also enabled.

Domain-specific parameters

The domain-specific parameters can be defined separately for each domain supported by a Gordano mail server. Each additional domain stores the values in a different location, under this key:

HKEY_LOCAL_MACHINE\Software\InternetShopper\
Mail\<domain-name>\Users\<No Name>



Do not change any parameter from its default without good reason — the default values give maximum efficiency in most circumstances.

The table which follows lists the parameters. Note the following:

- If you use Gordano's user interface to change one of these values *back to the default value*, the parameter is automatically deleted from the Registry.
- Values which have a range of 0 - 1 are Boolean: 0 = false (disabled), 1 = true (enabled).

Name	Range	Recommended value
AdminBackgroundBackdrop		#008282
AdminMenuTitleColour		#00007f
AllowedSenderIP		
AllowedSenderIPMes		See text
AllowLookup	0 - 1	1
BaseDir		
BaseDomain		
BypassAuthIP		
BypassAuthIPMes		
CompanyFax		
CompanyName		
CompanyTel		
Date created		
DomainType		
FAXHandler		root
FooterFile		
IPAddress		
LocalDomainNames		
LogAllMessages	0 - 1	0
LogDeleteAfter		5
LogEmailAfter		0
LogZipAfter		0
MaxAccounts	0-	0
MaxAccountSize	0-	0
MaxMailboxSize	0-	0
MaxMessageSize	0-	0
MaxMessageSizeMes		See text
MaxMessageSizeOut		0
MaxMessageSizeOutMes		See text
NTComputer		
NTDomains		
NullAccount		Null
POPBandwidth		0
PostFix		

Name	Range	Recommended value
PostFixesFile		
PostMaster		postmaster
QuarantineAdminPort		8000
QuarantineAdminServer		
QuotaMessageFile		
Script		script.mml
SMTPLoadSharingFile		
SMTPLoadSharingPrimaryServer		
SMTPLoadSharingEnabled	0 - 1	0
UnknownUserAction	T,U,R or N	Uroot
UseNTDatabase	0 - 3	0
UserMessage		
WebMailAutoDisplayNameFormat	0-2	0
WWWBaseDir		

The parameters are as follows:

AdminBackgroundBackdrop

The background color displayed in the configuration screens. Values are based on the standard method of displaying colours in HTML, i.e. red, white, etc. For more control you can use an RGB colour for example #000000.

AdminMenuTitleColour

The color of the menu title bar displayed in the configuration screens. Values are based on the standard method of displaying colours in HTML, i.e. red, white, etc. For more control you can use an RGB colour for example #000000.

AllowedSenderIP

Specifies a range of IP addresses that are allowed to send e-mail messages using a From address belonging to this domain.

AllowedSenderIPMes

The message returned to anyone failing the AllowedSenderIP test. The default is "553 You are not allowed to send e-mail via this server".

AllowLookup Range 0 - 1, default = 1.

Enables/disables lookups on the current domain. Enabling lookups lets people connecting to the Web Configuration Server look up

your company information — Company Name, Telephone Number and Fax Number, if you entered these. In addition, each user in the domain decides individually whether to allow lookups on themselves.

BaseDir

Defines a directory where all the domain specific files are stored including the user's directories. If you are happy for your domain information to reside in the default directory structure then this need not be set. This variable is set in a slightly different location to the other domain registry variables. i.e:

```
HKEY_LOCAL_MACHINE\Software\InternetShopper\Mail\<domain-name>
```

BaseDomain

Specifies the base domain that a virtual domain runs under.

BypassAuthIP

A space-separated list of IP addresses that can bypass the Auth command; see "Auth" on page 198.

BypassAuthIPMes

The message sent to addresses which cannot bypass the Auth command. The default is "553 You are not allowed to send e-mail via this server".

CompanyFax

The fax number of your company, displayed when anyone performs a lookup on an account under the domain using the "Find" option.

CompanyName

The name of your company, displayed when anyone performs a lookup on an account under the domain using the "Find" option.

CompanyTel

The Telephone Number of your company, displayed whenever anyone performs a lookup on an account under the domain using the "Find" option.

DateCreated

The date and time when the domain was created.

DomainType

The type of domain: 1 = full, 2 = POP, 4 = robot, 8 = virtual, 16 = DLL.

FAXHandler Default = root.

The account that any error information regarding faxes should go to.

FooterFile

A file which is inserted at the end of every message received via SMTP. This is inserted directly or, if the message is a multi-part MIME message, as a MIME section.

IPAddress

The IP address that is being used by the additional domains for POP collection and SMTP delivery. Only one IP address is allowed per full domain. If any POP Client connects to the machine at an undefined IP address, it is uncertain which domain they will reach.

LocalDomainNames

The default domain followed by the domains to accept mail for. The first entry is taken to be the default domain name and is added to any e-mail address where no domain name has been specified. This could be a machine or domain name. For example, Demon users in the UK have a name of 'xxx.demon.co.uk' so their domain is 'demon.co.uk'. However they need to set this value to 'xxx.demon.co.uk' otherwise the SMTP service will assume any mail to any other 'demon.co.uk' was (in fact) local to this machine (it would probably go straight to the root account).

Up to 200 domain names can be specified, separated by colons, spaces or semi-colons. For example 'test.dom elephant.dom' would accept mail for the two different domains.

LogAllMessages Range 0 - 1, default = 0.

If this is enabled, a copy of all messages passing through the mail server is saved (in the MesLog directory).

LogDeleteAfter Default = 5.

The number of days after which the domain logs are automatically deleted.

LogEmailAfter Default = 0.

The number of days after which the domain logs are mailed to the given address.

LogZipAfter Default = 0.

The number of days after which the domain log files are zipped up. The default value, 0, means never zip the files.

MaxAccounts Range 0 - , default = 0.

The maximum number of accounts that can be set up under the current domain. The default, 0, means no limit.

MaxAccountSize Range 0 - , default = 0.

The maximum disk space in Kb for each account. The default, 0, means no limit.

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.
- Low — set it low if you are short of disk space and want to restrict the amount available to each user.

MaxMailboxSize Range 0 - , default = 0.

The maximum size of the user's inbox.mbx in the domain in KB. If it's zero or non-existent there is no limit. If the limit is exceeded, a message is sent to the administrator at midnight and any e-mail to the mailbox is rejected with the message "453 Try again later".

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.
- Low — set it low if you are short of disk space and want to restrict the amount available to each user.

MaxMessageSize Range 0 - , default = 0.

The maximum size in KB an *incoming* message is limited to. You can use this to prevent people sending you excessively large messages. The default, 0, means no limit.

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.
- Low — set it low if you are short of disk space and want to restrict the amount available to each user.

MaxMessageSizeInMes

The error message returned to an SMTP transaction failing the MaxMessageSize test. The default is "553 Exceeded maximum inbound message size".

MaxMessageSizeOut Range 0 - , default = 0.

The maximum size in KB of an outgoing mail message from the current domain. The default, 0, means no limit.

MaxMessageSizeOutMes

The error message returned to an SMTP transaction exceeding the MaxMessageSizeOut. The default is "553 Exceeded maximum outbound message size".

NTComputer

The name of the computer holding the NT SAM User Database that should be used for validation of users and passwords. If this entry is not defined, the local machine is used for validation.

NTDomains

A space-separated list of domain names for user and password validation on Windows NT. If no domain is defined, the server asks Windows NT which domain the user belongs to.

NullAccount Default = NULL.

The account that acts as the null account.

PostFix

A qualifier to be added to the user information of incoming e-mail messages. This is only used with Virtual Domains; for details, see "Virtual domains" in the *GMS Administrator's Guide*.

PostFixesFile

This file is associated with a base domain that has virtual domains running under it. It contains a reference to each of the postfixes for these virtual domains.

PostMaster Default = postmaster.

The name of the user account to send e-mail queries to.

QuarantineAdminPort

The port used to access a users Quarantine folder (Vanguard).

QuarantineAdminServer

The fully qualified name (or the IP address) of the server (Vanguard). By default this uses the IP address of the server, so will need changed if installed on a private IP Address.

QuotaMessageFile

Contains the text to be sent to users when their mailbox is reaching capacity, warning them that if they do not reduce the size of their mailbox it will soon stop accepting mail. The warning is sent when the mailbox reaches 90% of its allowed capacity.

Script Default = script.mml.

A domain-specific script that is used to act on e-mail arriving at the domain.

SMTPLoadSharingFile

The name of the file that contains the loadsharing rules and configuration. See "Load Share" on page 15.

SMTPLoadSharingPrimaryServer

The name of the server that acts as the primary server in the Load Sharing array.

SMTPLoadSharingEnabled Range 0 - 1. Default = 0.

Enables this machine to become part of a load sharing array.

UnknownUserAction Range - T, U, R or N. Default = U.

Tells SMTP what it is to do with mail if there is no local mail drop defined for the destination name of the mail. There are four options:

- **T<user>** - transfer the message to a user account without giving any diagnostic message.
- **U<user>** - The server accepts the message, generates a fault report and adds the e-mail message as a MIME attachment. It then sends this to the named account.
- **R<machine_name>** - send all mail to the machine specified. This is often used when the Gordano server is acting as a firewall and passes all mail on to an internal server.
- **N** - No action (reject the mail). The sending server returns the message to its originator.

UseNTDatabase Range 0 - 3, default = 0.

Enables/disables use of the NT Database for verifying users. The values are:

- 0 = disabled
- 1 = Lookup in any group
- 2 = Lookup in Local groups only
- 3 = Lookup Global groups only

Enabling via the GUI sets the value to 1, this means GMS will look through all the groups on the server itself starting with the local groups. Setting the value to 3 means that GMS will bypass the Local groups altogether and go straight to Global groups, resulting in fewer lookups for users as normally the local groups are always checked first. If the NT SAM is on a remote PDC then the users will be in Global groups.

UserMessage

The welcome message automatically sent to every new user account you create in this domain..

WebMailAutoDisplayNameFormat Range 0-2, default=0

Uses the First Name and Last Name entries to autofill the Display Name of the Address Book entry when creating Address Book entries.

WWWBaseDir

The base directory for serving/saving web pages. If WWWBaseDir is an absolute pathname, for example "f:\www", that path is used explicitly. If it is relative, for example "server_root", it will resolve it as relative to the domain directory.

User parameters

In each full domain there is a key called Users, under which all the users are held. Each user has their own entry.

The key is:

```
HKEY_LOCAL_MACHINE\Software\InternetShopper\Mail\<domain-
name>\Users\<User-name>
```

The name of the entry is ignored, but we recommend that you set the name of the entry to the user name or their first alias, allowing the Registry editor to display all the entries in alphabetical order. (This is done automatically when users are added through the Gordano user interface.)

Each entry follows this format (though is all on one line):

```
Name postmaster
Type REG_MULTI_SZ
Data type=3
  DateCreated=1998-11-22 14:08:06
  Password=dgfe433
  Aliases=root,hostmaster
```



Do not change any parameter from its default without good reason — the default values give maximum efficiency in most circumstances.

There may be more parameters than shown in this example. The full set is listed in this table and described after it:

Name	Range	Default
AccessFile		
AccessRights		
Aliases		
AliasOf		
AllowEditScript	0 - 1	0
AllowLookup	0 - 1	1
AutoResponder		auto.txt
Date created		
DLL		
DLLSelected		
Executable		
Fax		
FaxCompanyName		
FaxFont		Lucinda console
FaxTemplate		template.fax
Forward		

Name	Range	Default
FullName		
Hidden	0 - 1	0
LogonDelay		0
MailBox		
MaxAccountSize		0
MaxMailboxSize		0
MaxMessageSize		0
Moved		
Password		NONE
PasswordChangeDate		
Plan		
Position		
ResponseInterval		
Telephone		
Type		2
STDIn	0 - 1	0
STDOut	0 - 2	0
UseStdIn		0
UseStdOut		
WebMailAutoDisplayNameFormat	0-2	0
WWWTimeOut		300

The parameters are described in detail below.

AccessRights

The access rights the user has, made by summing any of the following components:

Bit	Value	Meaning
0	1	Not Used
1	2	Domain Access
2	4	Support Access
3	8	GMS Anti-Spam Access
4	16	System Access
5	32	Log Access
6	64	Guest Access

AccessFile

The fax access file.

Aliases

A list of up to 20 aliases that will receive mail into this mailbox.

AliasOf

The full user name which this user name is an alias of.

AllowEditScript Range 0 - 1, default = 0.

Enables/disables the user's ability to edit scripts on the server.

AllowLookup Range 0 - 1, default = 1.

Enables/disables lookups on the user using the "Find" option. Enabling lookups lets people connecting to the Web Configuration Server look up the user's company information — Company Name, Telephone Number and Fax Number — if they have entered these.

In addition to this parameter:

- There is a global setting for the server.
- Individual domains have their own setting.

AutoResponder default=auto.txt

Determines the file to be returned as the result of an autoresponse to a sender being generated. If this is a binary file it will be MIME encoded and included in a mail message

DateCreated

The date and time when the user was created.

DLL

Name of a DLL used by the account. This is required, for example, by fax accounts.

DLLselected

When you put a DLL file in Gordano's bin directory, the name appears in the Gordano user interface's list of available DLLs. This parameter shows which DLL the user selected.

Executable

For a robot account, this field stores the executable command (its full filename). Any flags or options that follow the executable name will be passed to the executable as the usual argc, argv values. The message that fired the executable may be delivered on stdin and an outgoing message may be written to stdout. For example:

<executable name>=c:\bin\dump.exe

Fax

The full name of the owner of the account

FaxCompanyName

The name of the company sending the FAX message. This is only used by the fax server. It is optional and up to 60 characters long.

FaxFont Default = Lucinda Console.

This sets the font to be used when converting e-mail to FAX messages. It is optional.

FaxTemplate Default = template.fax.

The template file for a fax account, this controls what's passed from e-mail to fax.

Forward

The full e-mail address to which mail is redirected. Up to 20 comma-separated addresses may be included. If no domain is specified, the mail is forwarded to the default domain (that is, it is assumed to be local), but this use is not recommended - use aliases instead.

For example:

Forward=mark@another.company.dom,

FullName

The full name of the owner of the account.

Hidden

Enable/Disable lookups of the account

LogonDelay

This is used in protection against dictionary attacks. LogonDelay specifies the increment used when increasing delays between failed attempts to logon.

MailBox Default = Inbox.mbx.

Lets users specify a mailbox other than Inbox.mbx as their default mailbox.

MaxAccSize Default = 0.

The maximum disk space allocated to a user for all their files, mailboxes, etc.

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.

- Low — set it low if you are short of disk space and want to restrict the amount available to this user.

MaxMailboxSize Default = 0.

The maximum size of the user's inbox on the system in KB. (MaxAcctSize governs total disk space available to the user.) If it's zero or non-existent there is no limit. If this value is exceeded at midnight, an e-mail message is sent to the administrator and any e-mail to the mailbox is rejected with the message '453 Try again later'.

Reasons to change:

- High — set this high if you have lots of disk space, or leave it at 0 for no restrictions.
- Low — set it low if you are short of disk space and want to restrict the amount available to this user.

MaxMessageSize Default = 0 (no limit).

If this is present and has a value other than zero, the maximum text of an *incoming* message is limited to the given number of KB. Use this to prevent people sending excessively large messages.

Reasons to change:

- High — set this high if you have lots of disk space.
- Low — set it low if you are short of disk space and want to restrict the message size that can be received by this user, or if you are short of bandwidth.

Moved

An alternative email address that will be returned at the SMTP stage to any host sending mail to the current account. Mail for the account will be refused.

Password Default = no password.

The user password, held in encoded form.

PasswordChangeDate

The date this user's password was last changed.

Plan

The path of a file holding the user's details.

POPBandwidth Default = 0.

The bandwidth available for POP users. The default of 0 means use all available bandwidth.

Reasons to change:

- High — you may want to set this to a very high value, close to the capacity of your connection.
- Low — set it to a lower figure if you want to ensure that some of your bandwidth is always preserved, for example for browsing the Web.

Position

The position within the company of the owner of the account.

ResponseRate

The interval in days at which an autoresponder sends its message. 0 means send the message whenever a message is received. Using a non-zero value stops the autoresponder sending large numbers of messages.

StdIn Range 0 - 1, default = 0.

If set to an executable account, determines whether the mail message is passed to the executable program.

StdOut Range 0 - 2, default = 0.

If set to an executable account, determines whether the mail message is read back from the robot. A setting of 2 means that the server will not report an error if the executable fails to pass a message back.

Telephone

The telephone number of the user of the account. For a fax account, this is the number to send faxes to.

Type

The user type, made by summing any of the following components:

Bit	Value	Meaning
0	1	Alias
1	2	Gordano Database
2	4	Forward
3	8	List
4	16	SuperList
5	32	List Control Account (ListManager)
6	64	Mum DLL
7	128	Autoresponder
8	256	Moved
9	512	Robot / executable
10	1024	Template

Bit	Value	Meaning
0	1	Alias
11	2048	Virtual Pop Account
12	4096	Mail Manager
13	8192	Translate (a special case)
14	16384	ListAll Account
15	32768	NT Database
16	65536	Script
17	131072	User DLL
18	262144	Mailbox enabled
29	268435456	List Command Account
30	536820912	Deleted

For example, a standard mail account held in the Registry with no aliases would have a type of 262146, the decimal equivalent of bit 1 + bit 18. If this account also had an alias it would be 262147, created by adding bits 0, 1 and 18.

Further variables are required to identify some of the bit information correctly. For example a POP account that had aliases would need a second account set up, named <alias-account>. This would need:

- Bit 0 of Type set to 1.
- Its AliasOf variable set to "user".

The first POP account's Aliases variable would need to be set to <alias-account>.



Note that bits 1,6,11 & 15 are those which are used for user authentication. e.g. Type=294912 (bit 15,18) authenticates against the NTSAM database and enables the mailbox

UseSTDIn

Sends the contents of the message to the robot

UseSTDOut

The Gordano server expects to receive a message back from the robot.

WebMailAutoDisplayNameFormat Range 0-2, default=0

Uses the First Name and Last Name entries to autofill the Display Name of the Address Book entry when creating Address Book entries.

WWWTimeOut Default = 300.

The number of seconds that a client can remain connected to the Web Configuration Server without any activity before the connection times out.

Reasons to change:

- High — set this high if users are experiencing timeouts accessing the Web interface.
- Low — for security reasons you may want to keep this setting low. If, for example, an administrator logs on but forgets to log off again, another user coming from the same IP address could gain administrator rights to the mail server unless the session had timed out. Reducing the period reduces the chances of this happening.

Private parameters

The following parameters are all stored under the mail\private registry key. Take great care when editing any of the parameters in this section:

Name	Range	Default
BadMessageNumber		
DetailsNumber		
Install		
FaxNumber		
ListNumber		
MailNumber		
NListVersion		
POSTMaxWaiting		5000
POSTMinWaiting		2000
POSTRescanDelay	1-	120
POSTRescanFilesWaiting	1-	100
SaveBadMes		
UpdateSQLConfiguration	0 -31	0
Version		

The parameters are as follows:



Do not change any parameter from its default without good reason — the default values give maximum efficiency in most circumstances.

BadMessageNumber

The number of bad messages arriving at the server

DetailsNumber

The number of times the details message has been sent out to Gordano Limited.

Install

This entry provides details on the status of your last installation or upgrade.

FaxNumber

The number of the last FAX message sent out bound by SMTP. This increments as the mail is sent so tells you exactly how many fax messages have been sent by the Gordano server.

ListNumber

This is the number of messages processed by the list server.

MailNumber

This is the number of the last message sent externally by SMTP. It increments as the mail is sent and tells you exactly how many messages have been sent by the Gordano server.

NTListVersion

The current release of NTLsit being run on the server.

POSTMaxWaiting Default = 5000.

The maximum number of messages that POST will internally queue for sending. This lets you tune the performance of the POST service.

If there are more messages than this waiting for delivery, they are added to the queue later. POST will next scan the Out directory when the number of messages to send is less than POSTMinWaiting (see below).

Reasons to change:

- High — The more messages that can be queued by POST at one time, the more efficient it will be as it does not have to go back and read from disk all the time. Make sure you have plenty of memory available before increasing this setting.
- Low — Low memory or a slow machine may require that this setting is lowered to improve efficiency.



The right balance between POSTMaxWaiting and POSTMinWaiting is normally achieved through trial and error. The defaults have been set to allow the optimum performance across the broadest machine base possible.

POSTMinWaiting Default = 2000.

Allows tuning of the POST service. When the number of messages in the POST internal queue is less than this value, POST starts scanning the directory for any extra messages to place in the queue.

Reasons to change:

- High — Setting this too high can cause POST to rescan the queue well before it finishes posting all the mail it has already queued internally. This results in decreased performance.
- Low — If set too low then the queue may never be rescanned. If POSTMaxWaiting is set to a high figure and the number of messages that have failed to be delivered on the first pass is higher than POSTMinWaiting, the queue will not be rescanned until the number falls below this limit.

POSTRescanDelay Default = 120.

Improves POST's performance by reducing the frequency of the call to scan for new files. If all the POST threads are in use and the number of files waiting to be posted is greater than set by POSTRescanFilesWaiting, the POST queue rescan is delayed by this amount (in seconds). Otherwise a rescan takes place every 10 seconds.

Reasons to change:

- High — If the server is particularly busy you may want to delay the rescan even though the POSTRescanFilesWaiting limit has been reached. This gives POST more time to complete its current internal queue.
- Low — There is no reason to set this lower than the default.

POSTRescanFilesWaiting Default = 100.

This variable improves the POST performance by reducing the frequency of the call to scan for new files. It works in conjunction with POSTRescanDelay - refer to that entry for details.

Reasons to change:

- High — Reduces the frequency of the call to rescan the queue. If a lot of messages are waiting to go out you do not want to rescan the queue too frequently.
- Low — Unless for some reason you want the queues rescanned frequently, there should be no need to change this from the default.

SaveBadMes

Number of next bad message to be stored in BadMes directory.

UpdateSQLConfiguration

Used during upgrades to show the progress of updating the integrated mysql database with new information. If this value is present after an upgrade it suggests part of the upgrade has failed. You

should contact Gordano support at the earliest opportunity to resolve the problem.

Bit	Value	Meaning
0	1	Import files
1	2	Remove existing grants
2	4	Create GLWebMail Grant
3	8	Add calendar tables
4	16	Add messenger tables

Version

A three digit version number, in the form 6.01.02. This is used by the configuration software when updating the Registry from an old version of the software.

System Parameters

There are also a number of registry entries stored under the mail\parameters registry key.

Name	Range	Default
BaseDir		C:\Gordano\
BinStats	0-1	0
Email1		
Email2		
Email3		
MaxThreads	1-1000	255
MSBugFix	0 - 1	0
NumAllocUnits	16-2048	1000
NumBins	0-1024	260
NumWWWAllocUnits	16-2048	1000
NumWWWBins	0-1024	256
RegisterSupport	0 - 1	1

BaseDir

The base directory for the Gordano installation.

BinStats

Dumps memory allocation statistics to the log file on service shut-down

NumAllocUnits

Number of memory units to request from the OS as required. The value given to this variable should be at least equal to the value of **NumBins**.

NumBins

The number of memory allocation "bins" available for each service.

NumWWWAllocUnits

Number of memory units to request from the OS as required. The value given to this variable should be at least equal to the value of **NumBins**.

NumWWWBins

The number of memory allocation "bins" available for each service.

Email1,Email2,Email3

These contain the 3 email addresses that Gordano should respond to when you contact support@gordano.com

MaxThreads

Sets the maximum number of threads that may be allocated to each of the Gordano Services. This is a useful setting for those installations that are running on servers with small amounts of available memory.

MSBugFix

This is a special variable which you can enter as a DWORD set to 1. This fixes a bug in Windows 2000 that causes problems when shutting down SMTP. The Gordano server calls WSACleanup as part of the shutdown procedure. This function call never returns showing the process is stuck way down in the depths of the OS. Setting MSBugFix to 1 circumvents the WSACleanup call and allows SMTP to shut down.

RegisterSupport

If set to 1 the addresses in email1, email2 and email3 will be registered with Gordano support.

5 GMS Communication Server Configuration Database Parameters

The section entitled “Configuration — The Registry” on page 49 described the Gordano general configuration parameters and how to change them. This section describes the parameters which are used by GMS Communication Server.

The GMS Communication Server and NTList Version 3 parameters are listed separately — consult the appropriate section for your GMS Communication Server software. There’s also a section showing the parameter changes if you update from Version 3 to the latest version; see page 151.

5.1 Introduction

The parameters are divided into two groups:

- List variables — variables specific to each list.
- User variables — variables specific to each user of a list.
- Domain variables — variables applying to all lists in the domain.
- Global — variables applying to the whole system.

The Version 3 parameters are divided into four groups:

- Global variables — variables applying to the whole system.
- Private — variables applying to the whole system, but which you should not change unless you are sure of the effect.
- Domain — variables specific to each domain.
- User — variables specific to each user.

If you are upgrading from Version 3 to the latest version, there’s a summary of changes between the two on page 151.

This example shows the configuration database entry for a list called test-list:

```
Testlist : REG_MULTI_SZ :TYPE=8
DateCreated=1999-03-11 17:29:01
Password=%D$Lan*3o
listmanager=listman
listowner=user@domain.dom
```



If you do edit the Registry, only use Regedt32.exe, not another version of the Registry editor. Regedt32.exe supports multi-string Registry entries.

Note the following:

- If you use the Gordano user interface to change any value *back to the default value*, the parameter is automatically deleted from the configuration database.

- Values which have a range of 0-1 are Boolean: 0=false (disabled), 1=true (enabled).
- Values which have a range shown as "0-" or "1-" do not have a specific upper limit.

5.2 GMS Communication Server Parameters

This section describes the GMS Communication Server parameters. If you are still using NTList Version 3, skip to the next section.

Membership file format

The list membership file is stored in the directory Users\<listname>. It is named member.txt (or member.V3 if you have just upgraded from version 3 and the file has not yet been accessed).

Each member of a list has a single line in the list membership file. The line has the following format:

domain\username,properties,variables

The file is organised in domain, user order. The parameters are:

- Type — a 32 bit number indicating the membership properties. For a list of values, see “Properties” on page 149.
- Variables — a list of all the variables for this particular person, stored similarly to HTML strings. That is, + is a space, &+ is a "+" sign, & is a separator and && is an ampersand.



Variables that are no longer in the RequiredFields or OptionalFields lists may still be present. This happens when the variables are removed from the RequiredFields and OptionalFields lists.

This example shows the entries for three users, one of whom has left the list:

```
company.dom\kath,8,datejoined=2002-04-19+11:22:44  
company.dom\fred,0,datejoined=2002-04-19+11:22:44&fordays=60  
companyA.dom\joe,1,datejoined=2002-04-19+11:22:44&dateleft=2003-03-  
29+14:55:24&leftreason=1
```

In this example:

- Kath's properties value is 8, indicating that she is an active member of the list and receives posts as digests in the list's default text form. Joe's is 1, since he has left the list.
- Fred has two variables — the date he joined and the number of days his membership lasts.
- Joe has been a member of the list but has left at his own request.

List parameters

The parameters are listed in the following table and described in detail after it.

Name	Range	Recommended Value
AcceptWords		
ArchiveBackgroundColor		
ArchiveBaseURL		
ArchiveGifURL		default: localhost:8000/i/archiver
ArchivePassword		
ArchiveToHTML	0 - 1	0
ArchiveToMML	0 - 1	1
ArchiveWallpaperURL		
ArchiveWWWDirectory		
BadMessageAction	1-3, 5-9	6
BadMessageAddress		
BadMessageLimit		5
BadMessageResetFrequency		7
BadMessageResetStartDate		
BandwidthUsed		
BannedFromAddressFile		ban_addr.txt.
BannedMessageAction	1 - 5	2
BannedMessageAddress		
BannedSubjectFile		ban_sub.txt.
BounceAttachmentPosts	0 - 1	0
CheckMXOnJoin	0 - 1	0
CmdCmdAsSubject	0 - 1	0
ConfirmExpires		96 hours.
DailyCommands		
DailyPosts		
DataSourceName		listname#domain
DBType	1 - 4	1 (member.txt file).
DefaultAck		1
DefaultDigest	0 - 1	0
DefaultDigestType	1 - 3	2 (MIME.)
DefaultShow	0 - 1	1
DigestSizeLimit	0 -	0
DigestTimeHour	0 - 23	0 (midnight)
DirAccess	1 - 12	5
DirAccessFile		diraccess.txt

Name	Range	Recommended Value
DirPassword		
DirWords		
DisableMemberLookup	0 - 1	0
Display	Y/N	Y
DSNPassword		
DSNUser		
EndMsgCmdString		
FailedPostAccess	0 - 1	0
ForDefault		0
ForWords		
GetAccess	1 - 12	5
GetAccessFile		getaccess.txt
GetPassword		
GetWords		
HelpMsg		help.txt
HelpMsgPeriod		1
HelpWords		
JoinAccess	See text	3
JoinAccessFile		joinaccess.txt
JoinConfirmationMsg		joinconfirm.txt
JoinConfirmRequired	0 - 1	0
JoinMsg		join.txt
JoinPassword		
JoinWords		
LastPost		
LeaveAccess	4 - 13	3
LeaveConfirmationMsg		leaveconfirm.txt
LeaveConfirmRequired	0 - 1	0
LeaveInSubject	0 - 1	0
LeaveMsg		leave.txt
LeaveWords		
ListAccess	1 - 12	1
ListAccessFile		listaccess.txt
ListDescription		
ListManager		
ListMessage		
ListNameInSubject		
ListOwner		

Name	Range	Recommended Value
ListPassword		
ListServerOneFile	0 - 3	3
ListSizeInform	0 - 1	1
ListStatus	0 - 2	0
ListSummary		
ListWords		
LurkerStartDate		
MaxDailyPosts		0 (no limit).
MaxListMembers		
MaxListMsgSize		0 (no limit).
MaxMessageExceededMsg		maxmsgexceeded.txt
MaxToAddresses		5
MemberAddress		
MemberConfirmDays		0 (no limit)
MemberConfirmExpires		96 (hours)
MemberConfirmMessage		memberconfirm.txt
MemberExpireMsg		expire.txt
MemberExpireWarning		0
MemberExpireWarningMsg		warning.txt
MemberFile		Default is member.txt if DBType is 1 and member.mdb if it's 2 or 3.
MemberModeratorAddress		
MemberModeratorJoinNotify	0 - 1	0
MemberModeratorLeaveNotify	0 - 1	0
MemberModeratorPassword		
ModerateMemberExpires		96 hours.
ModeratePostExpires		96 hours.
ModerateRequestExpires		96 hours.
MsgFileName		
NumAdverts		
NumberFilesSent		
NumberJoined		
NumberLeft		
NumberPosted		
NumberReturned		
ODBCTableName		members.
OnlyJoinOnce	0 - 1	0

Name	Range	Recommended Value
OptionalFields		Digest, Show, Ack, MyPassword, Digest_Type, Suspended.
Password		
PasswordWords		
PostAccess	See text	3
PostAccessFile		postaccess.txt
PostAllCmdModerators	0 - 1	0
PostAllPostModerators	0 - 1	0
PostCmdAsLineOne	0 - 2	0
PostCmdAsSubject	0 - 2	0
PostHelpAsLineOne	0 - 2	0
PostHelpAsSubject	0 - 2	0
PostInsertFile		
PostModeratorAddress		
PostModeratorPassword		
PostWords		
ProcessMML	0 - 1	0
ProcessMMLAtDigest		
Public	Y/N	Y
PurgeFrequency		7
PurgeStartDate		
PutAccess	See text	1
PutAccessFile		putaccess.txt
PutPassword		
PutWords		
QuietWords		
ReadOnlyAction	1, 2, 5	2
ReadOnlyAddress		
RejoinLurkerDays		0 (no limit)
RejoinLurkerExpires		96 (hours)
RejoinLurkerMsg		rejoinlurker.txt
RemoveBinaryAttachments	0 - 1	0
RemoveHeaderFields		
ReportAction	1, 4, 5	4
ReportAddress		
RequestModeratorAddress		
RequestModeratorPassword		
RequiredFields		

Name	Range	Recommended Value
ResponseInsertFile		
ResumeAccess	1 - 13	3
ResumeAccessFile		resumeaccess.txt
ResumePassword		
SendHelpMsg	0 - 1	1
SendJoinMsg	0 - 1	1
SendLeaveMsg	0 - 1	1
SendStatistics		1
SetAccess	1 - 12	5
SetAccessFile		setaccess.txt
SetPassword		
SetWords		
StatusAccess	1 - 12	5
StatusAccessFile		statusaccess.txt
StatusPassword		
StatusWords		
StopWWWAction	0 - 1	0
SuperListEntries		
SuspendAccess	1- 13	3
SuspendAccessFile		suspendaccess.txt
SuspendPassword		
TotalMessageSize		
WildcardMatch		
ZombiePurgeFrequency		7
ZombiePurgeStartDate		

Values of Access parameters

Access to commands can be limited by the following parameters — DirAccess, GetAccess, JoinAccess, LeaveAccess, ListAccess, PostAccess PutAccess, Resume Access, SetAccess, StatusAccess and SuspendAccess. These are described in detail in the following pages.

This table shows available values for these parameters.

Value	Description
1	Post/Command not allowed from anyone.
2	Post/Command allowed from anyone for themselves.
3	Post/Command allowed from anyone for anyone.
4	Post/Command allowed from members of list for anyone.
5	Post/Command allowed from members of list for themselves.
6	Post/Command allowed for anyone from addresses in the file named in the list variable <cmd>AccessFile, where <cmd> is the command e.g JoinAccessFile. The full pathname may be specified as long as the drive letter is included.
7	Post/Command allowed for themselves from addresses in the file named in the list variable <cmd>AccessFile, where <cmd> is the command e.g JoinAccessFile.
8	Post/Command allowed from anyone for anyone if they include the line: password <postaccesspassword> as the first line of their message. When posting to a list, this line will be removed from the message before it is sent to all the list members <i>providing</i> the password line is the first line of text in the message.
9	Post/Command allowed from anyone for themselves. A password will be authenticated.
10	Post/Command allowed from members of list for anyone. A password will be authenticated.
11	Post/Command allowed from members of list for themselves. A password will be authenticated.
12	Post/Command allowed from anyone whose e-mail address matches the wildcard. '?' represents any one character and '*' represents any sequence of characters. The wildcard definition can contain more than one '*'.

- 13 Post/Command passed to moderator for moderation. All commands apart from **join** must be for a member of the list. Only one moderation request can be outstanding. If a command is being moderated or being processed for user acceptance then new commands requiring moderation will be rejected.
- 14 As 13 but a password is required from the moderator. The password line will be removed from the message before it is sent to all the list members *providing* the password line is the first line of text in the message.

Value	Description
15	Used in PostAccess only. The post is passed to the moderator if the message is not from a member of the list.
16	As 15 but a password is required from the moderator. The password line will be removed from the message before it is sent to all the list members <i>providing</i> the password line is the first line of text in the message.

The list parameters are as follows:

AcceptWords

A space-separated list of synonyms for the **accept** command.

ArchiveBackgroundColor

The background colour for archive Web pages, if you use standalone HTML files rather than GMS Communication Server archive format.

ArchiveBaseURL

The Base URL for the Web site that will be used to display the pages of standalone HTML files, if you use these.

ArchiveGifURL Default = localhost:8000/i/archiver.

The Base URL for the GIF files stored on the Web site that is used to display the archive pages.

ArchivePassword

The encrypted password used for archive commands. (If it's defined the command must have a password.)

ArchiveToHTML Range 0 - 1, default = 0.

Enables/disables HTML archiving.

ArchiveToMML Range 0 - 1, default = 1.

Enables/disables MML archiving.

ArchiveWallpaperURL

The URL of the directory you store the wallpaper files in.

ArchiveWWWDirectory

This is required if ArchiveToHTML or ArchiveToMML is set.

BadMessageAction Range 1 - 3, 5 - 9, default = 6.

The action required on returned messages from the list, one of these:

- 1 — do nothing.
- 2 — send to list owner.
- 3 — send to post moderator.
- 5 — forward to address specified by BadMessageAddress.
- 6 — suspend (see below) and ignore the message.
- 7 — suspend (see below) and send to list owner.
- 8 — suspend (see below) and send to post moderator.
- 9 — suspend (see below) and forward to address specified by BadMessageAddress.

If the value is in the range 6 to 9 the account is only suspended if the number of bad messages exceeds BadMessageLimit. However, if there is no X-ListMember clause in the returned message, the message is either ignored, forwarded to the owner or forwarded to the BadMessageAddress.

BadMessageAddress

The address to be used if BadMessageAction is 5 or 9.

BadMessageLimit default = 5.

If BadMessageAction is 6 or 9, the member is suspended once this number of messages has been returned.

BadMessageResetFrequency Default = 7.

The number of days after which to automatically reset NumBadMsgs. A value of zero means do not reset it.

BadMessageResetStartDate

The date when the first reset of NumBadMsgs will take place. Subsequent resets are based on this date.

BandwidthUsed

Equals TotalMessageSize x NumberOfPosts which gives the number of bytes per day.

BannedFromAddressFile Default = ban_addr.txt.

The name of the file containing an address (which can include wildcards) per line which is banned from posting messages or issuing commands.

BannedMessageAction Range is 1- 5, default = 2.

The action to be taken with a message containing a banned subject, one of these:

- 1 — do nothing.
- 2 — send to listowner.
- 3 — send to post moderator.
- 4 — return to sender.
- 5 — forward to address specified by BannedMessageAddress.

BannedMessageAddress

The address to send the banned message to if BannedMessageAction is 5.

BannedSubjectFile Default = ban_sub.txt.

The file listing banned subjects, one per line. (These can include wildcards.) This stops messages which include these subjects in their subject header being posted to a list.

BounceAttachmentPosts Range is 0 - 1, default = 0.

If this is set to 1 any posted message with an attachment is returned to its sender.

CheckMXOnJoin Range is 0 - 1, default = 0.

If this is enabled, when a **join** request is received from a user GMS Communication Server performs a lookup check on the MX record of that user's server.

CmdCmdAsSubject Range 0 or 1, default = 0.

If set to 1 the subject of a message to the control account will be parsed for a valid command. If one isn't found the body of the message will be parsed.

ConfirmExpires Default = 96.

The length of time (in hours) in which a command must be confirmed. If this time elapses the command is rejected.

DailyCommands

The number of commands in a day.

DailyPosts

The number of posts since either startup or midnight on the previous day, whichever is latest.

DataSourceName

The data source name to use if DBType is not 1.

DBType Range is 1 - 4, default = 1.

Member database type:

- 1 — V4_File.
- 2 — ODBC, a standard ODBC database.
- 3 — ODBC_RO, a read-only ODBC database.
- 4 — CUSTOM_ODBC, a standard ODBC database that allows you to amend the SQL queries used by GMS Communicator to suit your needs/database software.

DefaultAck Default = 1.

The default value of a member's ACK value. Also used for non-member posts and posts to a superlist.

DefaultDigest Range is 0 - 1, default = 0.

This is used if there's no member value (see the member parameter Properties). A value of 1 means send a digest.

DefaultDigestType Range 1 - 3, default = 2.

The digest type used for a user when no user level variable is defined:

- 1 — text.
- 2 — MIME.
- 3 — index.

DefaultShow Range is 0 - 1, default = 1.

If this is enabled, the sender's address is shown when they post and also in response to the **list** command. This is used when there's no member value (see the member parameter Properties) or for a post to a superlist.

DigestSizeLimit Range is 0 - , default = 0.

If this is non-zero the digest is sent when it reaches this size (in KB).

DigestTimeHour Range 0 - 23, default = 0 (midnight).

The time required for a digest posting to start.

DirAccess Range 1 - 12, default = 5.

The security access value for the **dir** command.

DirAccessFile Default = diraccess.txt.

The name of the file listing addresses allowed to issue **dir** if the access value is 6 or 7.

DirPassword

The encrypted password for **dir** if the access value is 8 or 9.

DirWords

A space-separated list of synonyms for the **dir** command. This includes **index**.

DisableMemberLookup Range 0 - 1, default = 0.

If set to 1 this disables lookup of a message's from address. This generally improves performance but means that whether the poster's address is displayed in the posted message is based on the list's default variable DefaultShow rather than the user's variable Show. The RejoinLurker function is also disabled if set to 1.

Display Range Y or N, default = Y.

If set to "Y" the lists description will be displayed on Gordano's web site.

DSNPassword

The password to be used with DataSourceName.

DSNUser

The username to be used with DataSourceName.

EndCmdMsgStr

Text identifying the end of a command message. This may be a signature.

FailedPostAccess Range 0-1, default = 0.

If set to 1 this determines that any post that fails security will be sent to the Post Moderator for moderation.

ForDefault Default = 0.

The default time in days that someone remains joined to the list. If it's set to zero, the person is joined until they decide to leave. If the FOR clause is specified in the join request and the message is not from the Member Moderator, the lower of the two lengths is used. If the message is from the Member Moderator, the FOR clause length is used.

ForWords

A space-separated list of synonyms for the **for** Member Moderator command.

GetAccess Range is 1 - 12, default = 5.

The security access value for the **get** command.

GetAccessFile Default = getaccess.txt.

The name of the file listing addresses allowed to issue **get** if its access value is 6 or 7.

GetPassword

The encrypted password for **get** if the access value is 8 or 9.

GetWords

A space-separated list of synonyms for the **get** command.

HelpMsg Default = help.txt.

The file containing the help message for the list.

HelpMsgPeriod Default = 1.

The number of days since the last help message was sent.

HelpWords

A space-separated list of synonyms for the **help** command.

JoinAccess Range is 1 - 4, 6 - 10, 12 and 14. Default = 3.

The security access value for the **join** command.

JoinAccessFile Default = joinaccess.txt.

Filename with list of addresses allowed to issue this command if its access value is 6 or 7.

JoinConfirmationMsg Default = joinconfirm.txt.

The name of the file holding the join confirmation message.

JoinConfirmRequired Range 0-1, default = 0.

If confirmation is required by anyone who joins the list, this is set to 1.

JoinMsg Default = join.txt.

The name of the file holding the join message.

JoinPassword

The encrypted password used for **join** if the access value is 8 or 9.

JoinWords

A space-separated list of synonyms for the **join** command. The list includes **subscribe** and **sub**.

LastPost

The date and time at which the last post was received by the list.

LeaveAccess Range is 3 - 13, default = 3.

The security access value for the **leave** command.

LeaveConfirmationMsg Default = leaveconfirm.txt.

The name of the file holding the leave confirmation message.

LeaveConfirmRequired Range 0 - 1, default = 0.

This is set to 1 if confirmation is required by anyone leaving the list.

LeaveInSubject Range 0-1, default = 0.

If set to 1 then we treat a replied-to post with a subject including any synonym of the leave command as a leave request. The member that is considered as requiring removal is the member record that the original post was sent to and not necessarily the 'from' address. If this is enabled the 'Reply-To' header field will not be inserted by the InsertHeaderFile functionality.

LeaveMsg Default = leave.txt.

The name of the file holding the leave message.

LeavePassword

The encrypted password used for **leave** if its access value is 8 or 9.

LeaveWords

A space-separated list of synonyms for the **leave** command. The list includes **unsubscribe** and **unsub**.

ListAccess Range 1 - 14, default = 1.

The security access for the **list** command.

ListAccessFile Default = listaccess.txt.

The name of the file listing addresses allowed to issue the **list** command if its access value is 6 or 7.

ListDescription

A five to ten line description of what the list is for.

ListManager

List Control Account. This is not needed for Superlist or Domain (PopAll) lists.

ListMessage

Added as 'X-Info' header field of posted messages if defined.

ListNameInSubject

String to be inserted in the subject of every message posted.

ListOwner

The name of the list owner.

ListPassword

The encrypted password used for **list** if its access value is 8 or 9.

ListServerOneFile Range 0 - 3, default = 3.

One of the following (also see the parameter ProcessMML):

- 0 — each posted message has an idx and mbx file.
- 1 — one idx and one mbx file is used for all posted messages.
- 2 — one idx and mbx file is used for all posted messages per domain.
- 3 — either 0,1 or 2 is selected automatically.

ListSizeInform Range 0 - 1, default = 1.

If this is enabled, an ack message gives the number of members a message was posted to.

ListStatus Range 0 - 2, default = 0.

The state of the list:

- 0 — active.
- 1 — stopped.
- 2 — paused.

If a list is stopped mail is returned. If it's paused mail is queued.

ListSummary

A short description of the list.

ListWords

A space-separated list of synonyms for the **list** command.

LurkerStartDate

The date at which checking for lurkers starts.

MaxDailyPosts Default = 0 (no limit).

Maximum posts allowed per day. Once this limit is reached, posted messages are returned.

MaxListMembers

Maximum number of members permitted for the list. There is no limit, however this variable will not allow a number of members greater than the license limit.

MaxListMsgSize

The posted message size limit in bytes (0 means no limit). Posted messages are rejected if they exceed than this. If this is not defined, the domain variable MaxMessageSize is used.

MaxMessageExceededMsg

The name of a file containing the message sent when a posted message exceeds the list limit (MaxListMsgSize)

MaxToAddresses Default = 5.

If this number is exceeded, a message is not posted.

MemberAddress

If the list server is stopped during a post this contains the members address to be continued from on restart.

MemberConfirmDays Default = 0.

The length of time in days before a member will be sent a message to confirm their wish to remain a member of the list. No reply by the member will result in their removal. A value of 0 means no message is sent.

MemberConfirmExpires Default = 96

The length of time, in hours, that the member must reply in before they are removed as a member of the list. See MemberConfirmDays and MemberConfirmMsg.

MemberConfirmMsg Default = memberconfirm.txt

The name of the file containing the membership confirmation request message.

MemberExpireMsg Default = expire.txt.

The name of the file holding the membership expiry message.

MemberExpireWarning Default = 0.

The length of time (days) before a membership expiry when a warning of expiry is sent. A value of 0 means no warning is sent.

MemberExpireWarningMsg Default = warning.txt.

The name of the file holding the membership expiry warning message.

MemberFile

Name of member file or database file. It must not be *.v3 or *.new. This parameter is ignored if DataSourceName is given.

The default is member.txt if DBType is 1 and member.mdb if it's 2 or 3.

MemberModeratorAddress

The e-mail address of the member moderator (moderates the **join**, **leave**, **suspend** and **resume** commands).

MemberModeratorJoinNotify Range 0 - 1, default = 0.

If this is enabled, a message is sent to the Member Moderator each time someone joins the list.

MemberModeratorLeaveNotify Range 0 - 1, default = 0.

If this is enabled, a message is sent to the Member Moderator each time someone is removed from/leaves the list.

MemberModeratorPassword

The encrypted password of the Member Moderator. Case-sensitive comparison is performed.

ModerateMemberExpires Default = 96 hours.

The length of time (hours) the Member Moderator must respond in before a request to **join**, **leave**, **suspend** or **resume** is rejected.

ModeratePostExpires Default = 96 hours.

The length of time (hours) the Post Moderator must respond in before the posted message is rejected.

ModerateRequestExpires Default = 96 hours.

The length of time (hours) the Request Moderator must respond in before the posted message is rejected.

MsgFilename

If the list server is stopped during a post this contains the filename of the post to continue with on restart.

NumAdverts

The number of adverts in the advert file.

NumberFilesSent

Number of files sent in response to the GET command since startup.

NumberJoined

The number of people who have joined the list since it was started.

NumberLeft

The number of people who have left the list since it was started.

NumberPosted

A 64-bit number value containing the number of messages which have been posted to the list by all its members since it was started.

NumberReturned

Number of bounced messages to the return address since startup.

ODBCTableName Default = members.

Member table to use with an ODBC database.

OnlyJoinOnce Range 0 - 1, default = 0.

If enabled, this only allows a list member to join the list once. If they attempt to rejoin the list (once left), a message is sent saying that they cannot rejoin.

OptionalFields

This lists the fields which users can optionally supply with the **join** command. You can add other fields, for example, "Date of birth". The default variables for OptionalFields are:

Default field	Value	Meaning
Digest	On/off	Send digests rather than all posts.
Show	On/off	Allow member's email address to be displayed to others when: <ul style="list-style-type: none"> • They post to the list. • Someone requests a list of all members of the list.
ACK	On/off	Send member an acknowledgment when they post to a list. If ON/OFF missing default ON.
MyPassword	<value>	A password the member must use to modify anything to do with them on the list. This value is used with the Password moderator for commands with access type 10 or 11.
DIGEST_TYPE	<value>	Send digest as Text, MIME or Index.
Suspended	On/off	Suspend postings.

Password

Lists password.

PasswordWords

A space-separated list of synonyms for the moderator command **password**. These must not include MYPASSWORD.

PostAccess Range is 3 - 16 for a basic list.

The security access for posting messages to a list. The default for a basic list is 3.

Possible values for a superlist are: 3, 8, 9, 12 and 13 - 16. The default for a superlist is 3.

PostAccessFile Default = postaccess.txt.

The name of the file holding listing addresses allowed to **post** if the access value is 6 or 7.

PostAllCmdModerators Range 0 - 1, default = 0

If you have more than one command moderator and would like them all to receive the message containing a command set this to 1. This is useful if a moderator is likely to be away for a period of time as it allows one of the others to approve the request. If more

than one moderator approves a request the action will still only be carried out once. If this option is set to 0 messages will be sent to the moderators in rotation.

PostAllPostModerators Range 0 - 1, default = 0

If you have more than one post moderator and would like them all to receive the message set this to 1. This is useful if a moderator is likely to be away for a period of time as it allows one of the others to approve the message. If more than one moderator approves a message the message will still only be posted once. If this option is set to 0 messages will be sent to the moderators in rotation.

PostCmdAsLineOne Range 0-2, default = 0.

The action to be taken if a command (not HELP) is detected on the first line of a post.

- If set to 0, then post the message to the list of members.
- If set to 1, then return the message to the sender
- If set to 2, then if CmdCmdAsSubject = 1 process the command else return the message to the sender.

PostCmdAsSubject Range 0-2, default = 0.

The action to be taken if a command (not HELP) is detected in the subject of the post.

- If set to 0, then post the message to the list of members.
- If set to 1, then return the message to the sender
- If set to 2, then if CmdCmdAsSubject = 1 process the command else return the message to the sender.

PostHelpAsSubject Range 0-2, default = 0.

The action to be taken if HELP is detected in the subject of the post.

- If set to 0, then post the message to the list of members.
- If set to 1, then return the message to the sender
- If set to 2, then if CmdCmdAsSubject = 1 process the command else return the message to the sender.

PostHelpAsLineOne 0-2, default = 0.

The action to be taken if HELP is detected on the first line of a post.

- If set to 0, then post the message to the list of members.
- If set to 1, then return the message to the sender
- if set to 2, then if CmdCmdAsSubject = 1 process the command else return the message to the sender.

PostInsertFile

Name of a text file to be inserted in every message posted. Header fields are also specified.

PostModeratorAddress

The address of the Post Moderator.

PostModeratorPassword

The encrypted password of the Post Moderator.

PostWords

A space-separated list of synonyms for **post**.

ProcessMML Range 0 - 2, default = 0.

This is not supported for superlists. It's one of the following:

- 0 — if ProcessMML is set to 1 or 2 it overrides ListServerOneFile with the value of 0. Setting 0 stops this.
- 1 — messages containing MML scripts are processed using a member's variables as parameters to the script.
- 2 — an MML is only allowed in the header file to be inserted and any MML in the posted message will be removed.

ProcessMMLAtDigest

Stores the value of ProcessMML when the digest was last sent. Any change between ProcessMMLAtDigest and ProcessMML will force the digest to be automatically sent.

Public Range Y or N, default = Y.

Whether the list is open to the public or not. This will be noted if the list's description is to be displayed on Gordano's web site. See Display.

PurgeFrequency Default = 7.

The time in days after which to purge all entries that have never achieved membership and members who have left. A value of zero means do not purge.

PurgeStartDate

The date when the first purge will take place. Subsequent purges are at intervals after this date.

PutAccess Range is 1, 2, 5, 7, 9 11 and 12 - 14. Default = 1.

The security access value for the **put** command.

PutAccessFile Default = putaccess.txt.

The file which lists addresses allowed to issue **put** if the PutAccess value is 7.

PutPassword

The encrypted password for **put** if its access value is 8 or 9.

PutWords

A space-separated list of synonyms for **put**.

QuietWords

A space-separated list of synonyms for the **quiet** Moderator command.

ReadOnlyAction Range is 1, 2, 5. Default = 2.

The action to take for commands that need to write to the database when its type is ODBC_RO (read only):

- 1 — do nothing.
- 2 — send to list owner.
- 5 — forward to address specified by BannedMessageAddress.

ReadOnlyAddress

If DBType is 3 (ODBC_RO) and ReadOnlyAction is 5, this is the address to forward commands to.

RejoinLurkerDays Default = 0.

If a member hasn't posted a message to the list in this many days then a message is sent to them from the list asking them to confirm their wish to remain as a member. If no reply is received within RejoinLurkerExpires hours then the member is removed from the list. No confirmation request message is sent if the value is 0.

RejoinLurkerExpires Default = 96.

See RejoinLurkerDays .

RejoinLurkerMsg Default = rejoinlurker.txt

The name of the file containing the lurker confirmation request message.

RemoveBinaryAttachments Range 0 - 1, default = 0.

If this is enabled any non-text parts of a multipart message are removed before posting.

RemoveHeaderFields

If this is enabled all the header fields except the From, To, Subject and Date fields are removed from a posted message.

ReportAction Range is 1, 4, 5. Default = 4.

The action taken with the transaction reports for a list, one of these:

- 1 — do nothing.
- 4 — return to sender.
- 5 — forward to address specified by ReportAddress.

ReportAddress

If ReportAction is 5 this is the address to send the report to.

RequestModeratorAddress

The e-mail address of the request moderator (moderates all user commands other than **join**, **leave**, **suspend** and **resume**).

RequestModeratorPassword

The encrypted password of the Request Moderator. The comparison is case-sensitive.

RequiredFields

This lists the fields which users must supply with the **join** command. You can add other fields, for example, "Date of birth". Password is not allowed — use MyPassword if a user must specify their own password.

ResponseInsertFile

Name of a text file to be inserted in every command response message that doesn't already have a configuration file. Header fields are also specified.

ResumeAccess Range is 1 - 14, default = 3.

The security access value for the **resume** command.

ResumeAccessFile Default = resumeaccess.txt.

The file which lists addresses allowed to issue **resume** if its access value is 6 or 7.

ResumePassword

The encrypted password for **resume** if its access value is 8 or 9.

SendJoinMsg Range 0 or 1, default = 1

If set to 1 then a Join message will be sent to the member.

SendHelpMsg Range 0 or 1, default = 1

If set to 1 then a Help message will be sent to the member.

SendLeaveMsg Range 0 or 1, default = 1

If set to 1 then a Leave message will be sent to the member.

SendStatistics Range 0 - 1, default = 1.

If this is set to one, statistics are sent daily to stats@GMS Communication Server.co.uk.

SetAccess Range is 1 - 14, default = 5.

The security access for the **set** command.

SetAccessFile Default = setaccess.txt.

The file which lists addresses allowed to issue **set** if its access value is 6 or 7.

SetPassword

The encrypted password for **set** if its access value is 8 or 9.

SetWords

A space-separated list of synonyms for the **set** command.

StatusAccess Range is 1 - 14, default = 5.

The security access for the **status** command.

StatusAccessFile Default = statusaccess.txt.

The file which lists addresses allowed to issue **status** if its access value is 6 or 7.

StatusPassword

The encrypted password used for **status** if its access value is 8 or 9.

StatusWords

A space-separated list of synonyms for **status**. The list includes **stat** and **show**.

StopWWWAction

When set to 1 no membership functions can be configured via a web browser. i.e. users cannot join/leave/post/get etc using the GMS Communicator administration interface. They will instead have to carry out these functions via email or a custom web interface. The list owner and administrators will however still be able to carry out these functions using the administration interface. The default (0) is to allow membership functions.

SuperListEntries

If the list is a superlist, this is either the name of a file containing lists or a comma-separated list of lists.

SuspendAccess Range is 1 - 14, default = 3.

The security access for the **suspend** command.

SuspendAccessFile Default = suspendaccess.txt.

The file which lists addresses allowed to issue **suspend** if its access value is 6 or 7.

SuspendPassword

The encrypted password for **suspend** if its access value is 8 or 9.

TotalMessageSize

Number of bytes of all messages posted to the list. Reset daily.

WildCardMatch

The * can be replaced by any command that can have an access value of 12 (see above). This is the wildcard address that must be matched if any of the command access values is 12. If this is not defined when there is a command access value of 12, that command will fail for all addresses.

ZombiePurgeFrequency Default = 7.

The number of days between automatic purges of entries for expired members. A value of zero means do not purge.

ZombiePurgeStartDate

The date when the first purge of entries for expired members will take place. Subsequent purges are based on this date.

Global parameters

Name	Range	Recommended Value
ListFileLimit		999
ListLog		18
ListMaxAddresses		100
ListMaxOutFiles		50000
ListPOSTThreads		20
ListServerPassword		
ListSizeResponse	0 - 1	1
ListThreads		4

ListFileLimit

Maximum number of messages per list file.

ListLog

Sets level of logging for the List Server.

ListMaxAddresses

Maximum number of addresses per list file.

ListMaxOutFiles

Maximum number of list files in the out directory.

ListPOSTThreads

Number of threads allocated to list posting.

ListServerPassword

Password required to access the List Server.

ListSizeResponse

Enables/Disables inclusion of list size information in the returned list message. Enabled by default.

ListThreads

Number of threads allocated to the list service.

MinDiskSpaceRequired

If the disk space available on the server falls below this level, list processing stops until more space becomes available. In the meantime, messages are held in a queue.

User parameters

There are a number of pre-defined user variables which may be found in a member record in the member database. Each member record also has a "properties" value which is a bit map of several user variables (e.g Digest).

These parameters are listed and detailed below:

Value	Range	Recommended
AcceptCommand		
AcceptVals		
DateConfirmSent		
DateJoined		
DateLeft		
ForDays		
ForDaysStartDate		
HelpSentTime		
LastConfirmed		
LastPosted		
LeftReason	0 - 99	
LurkerMsgSent		
ModerationDate		
ModFile		
MyPassword		
Name		
NumBadMsgs		
Organization		
Properties		
SuspendDays		
UserConfirm		

The parameters are described in detail below.

AcceptCommand

The command awaiting user confirmation.

AcceptVals

A list of command parameters for the command being moderated.

DateConfirmSent

If this list requires members to confirm their membership, the date that the confirmation request was sent to the user.

DateJoined

The date that the person joined the list. If the join has not taken effect yet, the date that the request was received.

DateLeft

The date that the user was removed from the list. This is required if the user leaves the list. It is kept until either old members are purged or the member rejoins.

ForDays

If defined, this is the number of days before the person should automatically be removed from the list.

HelpSentTime

The last time help was sent to member. Help can only be sent once in 24 hours.

LastConfirmed

Date member last confirmed their membership.

LastPosted

Date of members last post - used for removing Lurkers.

LeftReason Range 0 - 99.

The reason why the member was removed from/left the list:

- 0 — their subscription expired so they were automatically removed.
- 1 — they issued a **leave** command.
- 2 — they were banned by a moderator.
- 3 — the BadMessageLimit for returned posts was reached.
- 4 — Lurker did not rejoin on our request.
- 5 — member did not reply to member confirm request.
- 99 — member removed using the configuration interface.

This is kept until either old members are purged or the member rejoins.

ModerationDate

Date used for expiry of moderation for commands other than **join** and **leave**.

ModFile

The file containing the file being moderated.

MyPassword

A password that a member must use to modify anything to do with them on the list. This is used for commands that have access types 10 and 11.

Name

This is taken from the **join** message header unless it's also in the RequiredFields or OptionalFields parameter lists.

NumBadMsgs

The number of returned messages for the member. If the list's BadMessageAction is 6 or 9, the user will be suspended when the number of returned messages reaches the BadMessageLimit.

Organization

This is taken from the **join** message header unless it's also in the RequiredFields or OptionalFields parameter lists.

Properties

A binary number with the values listed in this table:

Bit	Meaning
bit0 - bit2	MEMBER_TYPE_MEMBER = 0 MEMBER_TYPE_OLD = 1 MEMBER_TYPE_BANNED = 2 MEMBER_TYPE_BEING_MODERATED = 3 MEMBER_TYPE_CONFIRM = 4 (join confirm only) MEMBER_TYPE_JOIN_EXPIRED = 5 MEMBER_TYPE_ZOMBIE = 6
bit3	DIGEST has been set by member.
bit4	DIGEST_TYPE has been set by member.
bit5	SHOW has been set by member.
bit6	ACK has been set by member.
bit7 - bit8	Digest type: DIGEST_OFF = 0 TEXT_DIGEST_TYPE = 1 MIME_DIGEST_TYPE = 2 INDEX_DIGEST_TYPE = 3
bit9	SHOW
bit10	ACK — send acknowledgements.
bit11	This is only set by configuration.
bit12	Suspended.

SuspendDays

The number of days for which to suspend posting of messages to the member.

Udil

Identifier used for matching moderation and user confirmation responses.

UserConfirm

Set to 1 if the command is awaiting user confirmation.

5.3 Parameter Changes Between Versions 3 and 6

If you have upgraded from NTList Version 3 to GMS Communication Server, this section tells you the main parameter changes.

Version 3 Allow parameters

The names of these parameters have changed, with the word "Allow" being replaced by "Access":

- GetAllow is now GetAccess.
- JoinAllow is now JoinAccess.
- ListAllow is now ListAccess.
- PostAllow is now PostAccess.
- PutAllow is now PutAccess.

There are also new parameters of this type called SetAccess, SuspendAccess, etc.

V3 name	V6 values	To change to Version 6:
A	3	
B	3	Set DefaultAck to "0". (This is only valid for PostAllow.)
Filename	6, 7	Set <cmd>AccessFile to the filename. For example, set JoinAccessFile = allowjoin.txt.
M	4, 5	
N	1	
Pwd	8	Set <cmd>Password to the password (for example, for join set JoinPassword = pass1).
R	13	This depends on the command: Join — set MemberModerator to ListOwner and MemberModeratorPassword to password. Post — set PostModerator to ListOwner and PostModeratorPassword to password. List, Get and Put — set CommandModerator to ListOwner and CommandModeratorPassword to password.
S	2	
W	12	Set <cmd>wildcard to wildcarded address. For example, joinwildcard=*@test.dom.

Other parameters which need changing are described below.

ListAllAccount

If ListAllAccount is set and it matches the list name, do the following:

1. Set the MUM_LIST_ALL bit for the list account.
2. Remove the ListAllAccount variable.

SuperAllow

If SuperAllow is set to one, do the following:

1. Set the MUM_SUPER_LIST bit in the account type for this list.
2. Remove the SuperAllow parameter.

ListReturnAction

The variable name has changed from ListReturnAction (domain parameter) to BadMessageAction (list parameter). Its possible values have also changed:

- N is now 1 (do nothing).
- R and S are now both 6 (suspend and do nothing).

ListMsgSize

The variable name has changed from ListMsgSize (user parameter) to MaxListMsgSize (list parameter).

HeaderFile

The variable name has changed from HeaderFile (user parameter) to PostInsertFile (list parameter).

6 Registry Parameters GMS WebMail Professional

Global Variables

Name	Range	Recommended value
AllowWebMailGUIServer		
PreloadImages	0-1	1
PreloadImagesThreads	0-255	4
PreloadImagesThreadsTimeout		1
PreloadImagesTimeout		10
WebMailAccessIP		
WebMailAccessIPDenied		
WebMailAllowUserRunExecutablehtml	0 - 1	0
WebMailCustomButton1URL		
WebMailCustomButton1Alt		
WebMailLog		
WebMailLogoAlt		
WebMailLogoUrl		
WWWWebMailMMLPort		9000
WWWWebMailMMLSSLPort		
WebMailScript	0 - 1	1
WWWWebMailSessionTimeout		240

AllowWebMailGUIServer

Determines whether GMS WebMail is permitted on the system (default yes)

PreloadImages

Preloads all of the images required by WebMail so a complete page is displayed to the user after logging in.

PreloadImagesThreads

Specifies the number of threads to use to load the images above. A value of 4 means four images will be loaded at a time.

PreloadImagesThreadsTimeout

Time in seconds to allow each individual image to load.

PreloadImagesTimeout

Time in seconds to allow the above preload to take place. If this is exceeded then we just proceed to the main page and the images are loaded as usual.

WebMailAccessIP

Specifies a range of IP addresses that are allowed access to the WebMail server. Default allows all IP addresses. Wild cards may be used.

WebMailAccessIPDenied

Specifies a range of IP addresses that are denied access to the WebMail server. Default allows all IP addresses. Wild cards may be used.

WebMailAllowUserRunExecutablehtml

Determines whether any executable content in the body of an email message can be run, i.e. Java, JavaScript, ActiveX, etc. (default = 0).

WebMailCustomButton1URL

Custom variable defining the URL assigned to Custom Button 1 in WebMail.

In addition this variable can be defined as WebMailCustomButton2URL and WebMailCustomButton3URL to allow for three custom buttons to be utilized.

WebMailCustomButton1Alt

Custom variable defining the text displayed when the mouse is held over the Custom Button 1 in WebMail.

In addition this variable can be defined as WebMailCustomButton2Alt and WebMailCustomButton3Alt to allow for three custom buttons to be utilized.

WebMailLog

Specifies the level of logging set for WebMail.

WebMailLogoAlt

Defines the ALT tag for custom logos. Default none. The domain parameter takes precedence over the global setting.

WebMailLogoUrl

Defines the URL which custom logos are hyper linked to. Default none. The domain parameter takes precedence over the global setting.

WWWWebMailMMLPort

This determines the port for WebMail only sessions - i.e. the port through which users go directly to the WebMail signon screen (default port 9000)

WWWWebMailMMLSSLPort

This determines the port used by the GMS WebMail session only over SSL (no default)

WebMailScript

Allows WebMail scripting, i.e filter scripts, independently from the any settings configured under NTMail (default 1 = On).

WWWWebMailSessionTimeout

Specifies the time period of inactivity in minutes before the user will be logged out (default 240).

Domain Variables

Name	Range	Recommended value
PreloadImages	0-1	1
PreloadImagesThreads	0-255	4
PreloadImagesThreadsTimeout		1
PreloadImagesTimeout		10
WebMailAccessIP		
WebMailAccessIPDenied		
WebMailLogoAlt		
WebMailLogoUrl		
WebMailUseDefaultAccessIP		

PreloadImages

Preloads all of the images required by WebMail so a complete page is displayed to the user after logging in.

PreloadImagesThreads

Specifies the number of threads to use to load the images above. A value of 4 means four images will be loaded at a time.

PreloadImagesThreadsTimeout

Time in seconds to allow each individual image to load.

PreloadImagesTimeout

Time in seconds to allow the above pre-load to take place. If this is exceeded then we just proceed to the main page and the images are loaded as usual.

WebMailAccessIP

Specifies a range of IP addresses that are allowed access to the WebMail server. Default allows all IP addresses. Wild cards may be used.

WebMailAccessIPDenied

Specifies a range of IP addresses that are denied access to the WebMail server. Default allows all IP addresses. Wild cards may be used.

WebMailLogoAlt

Defines the ALT tag for custom logos. Default none. The domain parameter takes precedence over the global setting.

WebMailLogoUrl

Defines the URL which custom logos are hyper linked to. Default none. The domain parameter takes precedence over the global setting.

WebMailUseDefaultAccessIP

If this is set to "N" for a given domain, then the specific values of WebmailAccessIP and WebmailAccessIPDenied that have been defined for the domain will be used. Otherwise the system settings will be used.

User Variables

Name	Range	Recommended value
AccessRights		
DefaultPersonality		Signon personality
DocumentStoreCapacity		20
Language		en-us
MailAutoRefresh		0
PreloadImages	0-1	1
PreloadImagesThreads	0-255	4
PreloadImagesThreadsTimeout		1
PreloadImagesTimeout		10
RevisionsHighWater		
WebMailAccessRights		
WebMailAddressHistoryCount	0 - 20	0
WebMailAddressHistory		
WebMail AllowCustomisation	0 - 1	0
WebmailAttachDropdown	0 - 1	0
WebmailAutoPersonality	0 - 1	0
WebmailBackgroundBackdrop		
WebmailBackgroundForm		
WebmailBackgroundFrameBorder		
WebmailBackgroundMail		
WebmailBackgroundMailHeader		
WebmailBackgroundMailList		
WebMailBackgroudMenuBar		
WebmailBackgroundMessagebar		
WebmailBackgroundNavigator		
WebmailDateFormat		yyyy-mm-dd
WebMailDateSelectorPosition	0 - 2	
WebMailDefaultTemplate		
WebMailDraftsMailboxName		Drafts
WebMailHeaderWindowHeight		70 (Netscape), 60 (others)
WebMailKeepTrash	0 - 1	1
WebMailMaxForwardText		0
WebMailMeetingGranularity		
WebMailMessageReadFont		Arial
WebMailMessageReadFontSize	1 - 5	2
WebMailMessageWriteFont		Arial
WebMailMessageWriteFontSize	1 - 5	2

Name	Range	Recommended value
WebMailMsgEncoding		None
WebMailNotifyNewMail	0 - 1	0
WebMailPopCheck Interval		15
WebMailPurgePending		
WebmailReadWindow	0 - 1	1
WebmailReadPreviewWindow	0 - 1	1
WebMailReplyIndentText		>
WebMailRunExecutableHTML	0 - 1	0
WebMailSaveSentMessages	0 - 1	1
WebMailSentMailboxName		Sent
WebMailSelectorAppearance		
WebMailSelectorWindowHeight		225
WebMailShowLinks	0 - 1	1
WebMailShowRealNames	0 - 1	0
WebMailSmallMenu	0 - 1	0
WebMailTimeOut		10
WebMailTrashMailboxName		Trash
WebMailWeekStartday		

AccessRights

Determines the specific Profile configuration rights applied to the user. This value is from the Profile applied to the user.

By default this value is represented as 65535.

Bit	Default Value	Meaning
1	2	Manage domains
3	8	Manage GMS Anti-Spam and GMS Anti-Virus for all domains
4	16	Manage complete system
5	32	Manage logs for all domains

DefaultPersonality

The default personality used when composing messages.

DocumentStoreCapacity

Default 20Mb

The maximum size allowed for any users document store in Mb.

Language

Language to use. Default: en-us

MailAutoRefresh default 0 (off)

The interval in minutes at which GMS WebMail checks for new mail.

PreloadImages

Pre-loads all of the images required by WebMail so a complete page is displayed to the user after logging in.

PreloadImagesThreads

Specifies the number of threads to use to load the images above. A value of 4 means four images will be loaded at a time.

PreloadImagesThreadsTimeout

Time in seconds to allow each individual image to load.

PreloadImagesTimeout

Time in seconds to allow the above pre-load to take place. If this is exceeded then we just proceed to the main page and the images are loaded as usual.

RevisionsHighWater

The maximum number of revisions to be stored for any document in a users document folder.

WebMailAccessRights

Determines the specific access rights applied to the user. This value is from the Profile applied to the user.

By default this value is represented as 65535. Therefore the first 16 bits are set by default.

Bit	Default Value	Meaning
0	1	Allow WebMail Personalities
1	2	Allow WebMail Servers
2	4	Allow WebMail filters
3	8	Allow WebMail address books
4	16	Deny WebMail domain address book
5	32	Allow WebMail system address book
6	64	Deny sharing address book entries with allusers
7	128	Deny sharing address book entries with allusers
8	256	Deny manage domain address book

9	512	Deny manage domain address book entries
10	1024	Deny manage system address books
11	2048	Deny manage system address books entries
12	4096	Allow WebMail local address book
13	8192	Allow WebMail calendar
14	16384	Allow WebMail attach vCards
15	32768	Allow WebMail shared Calendars
16	65536	Allow WebMail SMS gateway
17	131072	Allow WebMail pager gateway
18	262144	Allow transitory email addresses
19	524288	Allow user selected presence image
20	1048576	Allow user presence image
21	2097152	Allow include image
22	4194304	Allow Local Personalities Only
23	8388608	Allow Auto Add Junk
24	16777216	Allow External Spam Filter
25	33554432	Allow Auto False Negative Report
26	67108864	Allow Auto Add Mail

WebMailAddressHistoryCount default 0 (No history)

The number of addresses to store in the history dropdown. A maximum of 20 addresses can be stored.

WebMailAddressHistory

The current address history (list of most recently used addresses).

WebMailAllowCustomisation

If set to 1 the user is allowed to change the look of the GMS WebMail client by altering background colours and patterns.

WebmailAttachDropdown

Attachments shown in a dropdown or a list. Default: 0 (list)

WebmailAutoPersonality

Match personality to reply address. Default: 0 (no)

WebmailBackgroundBackdrop

The pattern or image displayed behind dialog boxes such as the preferences dialog.

WebmailBackgroundForm

The background color of the dialog boxes used to change GMS WebMail settings.

WebmailBackgroundFrameBorder

The color of table borders where used in the GMS WebMail client.

WebmailBackgroundMail

The color used as a background to the contents of a message.

WebmailBackgroundMailHeader

The color used as a background to the message header bar between the message listing and message contents.

WebmailBackgroundMailList

The color used as a background to the list of messages contained in a folder.

WebMailBackgroudMenuBar

The background color of the menu bar at the top of the GMS WebMail client.

WebmailBackgroundMessagebar

The background colour of the status bar at the bottom of the page. This is the bar where any notification or error messages are usually displayed.

WebmailBackgroundNavigator

The background colour of the tree menu used to navigate around the GMS WebMail folders and configuration options.

WebmailDateFormat

Date format for message listing. Default: yyyy-mm-dd

WebMailDateSelectorPosition

Defines the position of the WebMail Calendar. Default - 0.
0 - Bottom, 1 - Not displayed, 2 -Top

WebMailDefaultTemplate

The default template to use when composing messages.

WebMailDraftsMailboxName

The name of the folder in which to save draft messages. Default is "Drafts".

WebMailEnabledAccessRights

Determines the specific access rights applied to the user. This value is from the Profile applied to the user.

By default this value is represented as 65535. Therefore the first 16 bits are set by default.

Bit	Default Value	Meaning
0	1	Allow Instant Messenger
1	2	Allow WebMail HTML email
2	4	Enable Instant Messenger launch on Login
3	8	Allow GMS Collaboration
4	16	Allow changes to FreeBusy settings
5	32	Allow report mail as spam
6	64	Allow user search
7	128	Allow Anti-spam filter
8	256	Allow sending of external email
9	512	Allow receiving of external email
10	1024	Allow sending of external email if in shared address book
11	2048	Allow receiving of external email if in shared address book
12	4096	Allow sending of external email if in personal address book
13	8192	Allow receiving of external email if in personal address book
14	16384	Allow CC of email in WebMail
15	32768	Allow BCC of email if in WebMail
16	65535	Allow Documents

WebMailHeaderWindowHeight

Height of the message header summary window. Default: 70 (Netscape) or 60 (all other browsers)

WebMailKeepTrash default 1

Whether to delete trash messages at signoff. 1 keep trash, 0 delete trash.

WebMailMaxForwardText default 0 (unlimited)

Maximum size in Bytes of text to include in forwarded messages.

WebMailMeetingGranularity

Determines the time increments, in minutes, available when setting Calendar Events. Default is 15

WebMailMessageReadFont

Font face to use when displaying a message. Default is Arial, alternative is Courier.

WebMailMessageReadFontSize Range(1 - 5) default 2

Font size to use when displaying messages.

WebMailMessageWriteFont

Font face to use when composing messages. Default is Arial, alternative is courier.

WebMailMessageWriteFontSize Range(1 - 5) default 2

Font size to use when composing messages.

WebMailMsgEncoding

Message encoding for sent messages. Default: None

WebMailNotifyNewMail

If set to 1 a pop-up notification will appear in the foreground of the user's desktop whenever a new message arrives at their account. This only happens if the user is logged in to webmail at the time.

WebMailPopCheckInterval default 15

Interval in minutes at which to perform pop downloads for server entries.

WebMailPurgePending

Defines the actions taken on messages in the Quarantine folder when the user signs off. Values are separated by a colon and are in the form of x:y:z where
x = the number of days after which messages should be purged.(default - 1)
y = determines if all messages from the same user should be purged (0 - No, 1 - Yes)
z = determines if the address should be added to the blocklist address book (0 - No, 1 - Yes)

WebmailReadWindow

Read mail in another window. Default: 1 (yes)

WebmailReadPreviewWindow

Read mail in preview pane. Default: 1 (yes)

WebMailReplyIndentText default >

Line prefix to use for forwarded text.

WebMailRunExecutableHTML default 0

Allow/disallow executable in launched messages. i.e. Java script, Java, embeds. 0 disallowed, 1 allowed.

WebMailSaveSentMessages default 1

Save/don't save copies of all sent messages. 0 don't save, 1 save.

WebMailSentMailboxName

The name of the folder in which to save copies of sent messages. Default is "Sent".

WebMailSelectorAppearance

Layout of message selector.

WebMailSelectorWindowHeight

Height of message selector. Default: 225

WebMailShowLinks default 1

Highlight ftp, http, and mailto links when reading messages. 0 don't highlight, 1 highlight,

WebMailShowRealNames default 0

Use real name portion of email address in mail selector list. 0 don't use real names, 1 use real names.

WebMailSmallMenu default 0

Use compact version of top menu bar. 0 Don't use compact menu, 1 use compact menu.

WebMailTimeOut default 10 Minutes

Number of minutes inactivity before GMS WebMail session times out.

WebMailTrashMailboxName

The name of the folder in which deleted messages are stored. The default is "Trash".

WebMailWeekStartday

Determines the whether the WebMail Calendar weeks start on Sunday or Monday. Default - M

7 Registry Parameters GMS Mobile.

The following user parameters apply to the Mobile client available with GMS WebMail. This is a simplified client designed for use by WAP phones

User Variables

Name	Range	Recommended value
WMXMaxAttachments		1
WMXMsgIndexRange		10
WMXFolderIndexRange		10
WMXDisplayForwardText	0 - 1	0
WMXHelpDisabled	0 - 1	0
WMXMainMenuLink	0 - 1	0
WMXEmptyTrash	0 - 1	1
WMXMailAckText		"Your message has been received by <email address>"

WMXMaxAttachments

Number of attachments to allow when composing mail

Default: 1

WMXMsgIndexRange

Number of messages to display in message listing

Default: 10

WMXFolderIndexRange

Number of folders to display in folder listing

Default: 10

WMXDisplayForwardText

Display original message text when forwarding or replying

Default: 0 (no)

WMXHelpDisabled

Disable help link

Default: 0 (no)

WMXMainMenuLink

Show shortcut link to main menu

Default: 0 (no)

WMXEmptyTrash

Empty trash folder at signoff

Default: 1 (yes)

WMXMailAckText

Message acknowledgement text

Default: "Your message has been received by <email address>"

8 Collaboration Server Registry Parameters

The section entitled "Configuration — The Registry" on page 49 described the Gordano general Registry parameters and how to change these. This section describes the Registry parameters which are used by Collaboration Server.

8.1 Introduction

The GMS Collaboration server is designed to allow Microsoft Outlook access to the Address Books, Calendaring and Scheduling functions of the GMS Server. In the current release sharing is not supported, all other aspects of MS Outlook are fully supported.



If you do edit the Registry, only use Regedt32.exe, not another version of the Registry editor. Regedt32.exe supports multi-string Registry entries.

Note the following:

- If you use the Gordano user interface to change any value *back to the default value*, the parameter is automatically deleted from the Registry.
- Values which have a range of 0-1 are Boolean: 0 = false (disabled), 1 = true (enabled).
- Values which have a range shown as "0-" or "1-" do not have a specific upper limit.

8.2 Parameters

Global Parameters

The parameters are as follows:

Name	Range	Recommended Value
CollaborationMaxConnections-PerIP	1 - 255	
CollaborationCheckAccept	0 - 1	0
CollaborationLogonDelay		0
CollaborationLogonTimeout		120
CollaborationMaxConnections	0 - 255	0
CollaborationPort	1 - 65535	8376
CollaborationRecvThreads	0 - 255	10
CollaborationSessionTimeout	1 -	90
CollaborationSSLPort	1 - 65535	8377
CollaborationThreads	1 - 255	20
CollLog	1 - 65535	4114
FreeBusyAuth	0 - 1	
FreeBusyAuthTime	1 -	3600
FreeBusyLocalIPAllowed	0 - 1	1
FreeBusyPublish	0 - 1	1
FreeBusyPublishPeriod	1 -	61
FreeBusyPublishPrivate	0 - 1	1

CollaborationMaxConnectionsPerIP Range 0 - 255, default =

0

The maximum number of simultaneous connections allowed from a single IP address. The default setting allows unlimited connections.

CollaborationCheckAccept Range 0 - 1, default = 0

If enabled this option will prevent the Collaboration Server from responding on any IP's that are not configured for use by a domain in the Gordano products. If you would like to over-ride this option for a particular domain you can do so by adding the other IP addresses that the Collaboration Server should answer on to the list of aliases for the domain in question.

CollaborationLogonDelay Range 0 -

This is used in protection against dictionary attacks. LogonDelay specifies the increment used when increasing delays between failed attempts to logon.

CollaborationLogonTimeout Range 0 -

This defines the timeout applied to Logon Failures where the client does not automatically disconnect.

CollaborationMaxConnections Range 0 - 255, default = 0

The maximum number of simultaneous connections available to the Collaboration Server from any IP address. The default of zero means no limit.

Reasons to change:

- High — set this high if you have many connections to your server from the same IP address, that is, they are coming through a proxy or firewall.
- Low — this helps prevent denial of service attacks on the Collaboration Server. Normally you would only expect a single connection from any one IP Address.

CollaborationPort Range 1 - 65535, default = 8376

The port the Collaboration Server listens on.

CollaborationRecvThreads Range 1 - 255, default = 10

The Collaboration Receive threads are the threads that handle all of the protocol transactions between the client and the server. These should be maintained at an approximate ratio of 0.5 the available Collaboration threads.

CollaborationSessionTimeout Range 1 -, default = 900

Sets the idle timeout for connections to the Collaboration Server. By default this is set to 900 seconds.

CollaborationSSLPort Range 1 - 65535, default = 8377

The port the collaboration server listens on for SSL connections.

CollaborationThreads Range 1 - 255, default = 20

The Collaboration Threads are the ones that actually do the work on the server itself, they do not talk directly to the client machines. These threads should be maintained at approximately double the level of the CollaborationRecvThreads

CollLog Range 1 - 65535, default = 4114

Sets the level of Logging for the Collaboration Server. The logs are written to the gordano\logs directory and are indicated by filenames starting with.

FreeBusyAuth Range 0 - 1, default = 0

Defines whether or not authentication to the POP or IMAP services is required in order to use the free/busy server.

FreeBusyAuthTime Range 1 - , default = 3600

Defines the length of time in seconds that authentication to the POP or IMAP server allows access to free/busy information for.

FreeBusyLocalIPAllowed Range 0 - 1, default = 1

Controls whether or not any connections from an IP Address covered by the LocalIP range is allowed un-authenticated access to the free/busy server.

FreeBusyPublish Range 0 - 1, default = 0

Enables or disables the publishing of free/busy information.

FreeBusyPublishPeriod Range 1 - , default = 61

The length of time in days that free/busy information is published for.

FreeBusyPublishPrivate Range 0 - 1, default = 0

If enabled then any private entries in the users calendar are published, these are automatically marked as busy.

Domain Parameters

There are no domain level parameters

User Parameters

FreeBusyPublish	0 - 1	1
FreeBusyPublishPeriod	1 -	61
FreeBusyPublishPrivate	0 - 1	1

FreeBusyPublish Range 0 - 1, default = 0

Enables or disables the publishing of free/busy information.

FreeBusyPublishPeriod Range 1 - , default = 61

The length of time in days that free/busy information is published for.

FreeBusyPublishPrivate Range 0 - 1, default = 0

If enabled then any private entries in the users calendar are published, these are automatically marked as busy.

9 Registry Parameters GMS WAPMail

Global/Domain Variables

Name	Range	Recommended value
WapMailAllowed	0 - 1	1
WapMailMainTitle		GMS WapMail
WapMailMOTD		Welcome
WapMailSplash		GMS WapMail.bmp
WapMailSplashTime		20

WapMailAllowed Range 0 - 1 default 1

Enables/disables the use of WAP, enabled by default.

WapMailMainTitle

The title for main menu and signon pages. Default is "GMS WapMail".

WapMailMOTD

Text used as a welcome message, in addition to new message count. Default is "Welcome"

WapMailSplash

You can set a registry entry to control this at any of the System, Domain or User levels, it would need an entry in the relevant MULTI_STRING

WapMailSplash=<p></p>

If you want to use your own gif file rather than ours it should be placed in the MML directory.

WapMailSplashTime

Duration of splash screen in 1/10 seconds. Default is 20

User Variables

Name	Range	Recommended value
WapFolderIndexRange		10
WapMailAckText		
WapMailAllowed	0 - 1	1
WapMailMainTitle		GMS WapMail
WapMailMOTD		Welcome
WapMailMsgIndexRange		10
WapMailMsgPageSize		20

WapFolderIndexRange

Number of folders to display in folder listing. Default: 10

WapMailAckText

Text used in acknowledging receipt of a message. Default is "Your message has been received by user@domain.dom"

WapMailAllowed Range 0 - 1 default 1

Enables/disables the use of WAP, enabled by default.

WapMailMainTitle

The title for main menu and signon pages. Default is "GMS WapMail". This variable can only be set by an administrator.

WapMailMOTD

Text used as a welcome message, in addition to new message count. Default is "Welcome". This variable can only be set by an administrator.

WapMailMsgIndexRange

The number of messages to show at a time when listing folder contents. Default is 10.

WapMailMsgPageSize

The number of lines to show at a time when displaying message text. The default is 20.

10 GMS Anti-Spam Registry Parameters

The section entitled "Configuration — The Registry" on page 49 described the Gordano general Registry parameters and how to change these. This section describes the Registry parameters which are used only by GMS Anti-Spam.

10.1 Introduction

The parameters are divided into two groups:

- Global variables - variables which apply to the whole system.
- Domain-specific variables — variables which apply to a single domain.

All parameters and their meanings, defaults, etc. are covered.



If you do edit the Registry, only use Regedt32.exe, not another version of the Registry editor. Regedt32.exe supports multi-string Registry entries.

Note the following:

- If you use the Gordano user interface to change any value *back to the default value*, the parameter is automatically deleted from the Registry.
- Using GMS Anti-Spam adds extra variables to some parameters which have already appeared in the standard Gordano parameter list. These are marked in the tables as "GMS Anti-Spam add-on".
- Values which have a range of 0-1 are Boolean: 0 = false (disabled), 1 = true (enabled).
- Values which have a range shown as "0-" or "1-" do not have a specific upper limit.

10.2 Parameters

This section lists all the global and domain-specific GMS Anti-Spam parameters.

Global parameters

The parameters listed in this table apply to the whole system. They are described in detail after the table.

Name	Range	Recommended Value
AIIPActivate	0 -1	0
AIIPAverageMultiplier		10
AIIPCheckActive	0 -1	0
AIIPFailureMes		
AIIPRequiredSamples		7
AIIPRunningAverageMin		25
AIMAILActivate	0 -1	0
AIMAILAverageMultiplier		10
AIMAILCheckActive	0 -1	0
AIMAILFailureMes		See text.
AIMAILRequiredSamples		7
AIMAILRunningAverageMin		25
AIRCPTActivate	0 -1	0
AIRCPTAverageMultiplier		10
AIRCPTCheckActive	0 -1	0
AIRCPTFailureMes		See text.
AIRCPTRequiredSamples		7
AIRCPTRunningAverageMin		25
AllowedRCPTLimit		GMS Anti-Spam add-on
BypassAuthIP		comma separated IP addresses
IMAPMaxConnectionsPerIP	1-255	
IPAllowed		*
IPAllowedMes		See text
MaxMessagesPerIPAddress		
MaxMessagesPerIPAddress-Mes		See text
MaxMessagesPerUser		
MaxMessagesPerUserMes		See text
OnlyAcceptLocal	0-2	2
OnlyAcceptLocalMes		See text
POPAuthSMTP	0 - 1	0

Name	Range	Recommended Value
POPAuthSMTPTime	0 -	3600
POPMaxConnectionsPerIP	1-255	1
DNSBLServers		DNSblservers.txt
DNSBLServersMes		See text
DNSBLServerMode	0 - 2	1
RejectAttachmentExt		
RejectAttachmentExtOptions	0 - 7	
Script		filterscript.mml
SMTPScript	0 - 1	0
SMTPMaxConnectionsPerIP	1 - 255	255
VerifyHostname		GMS Anti-Spam add-on
VerifyHostnameMes		GMS Anti-Spam add-on
VerifyIncoming		
VerifyIncomingMes		See text
VerifyOutgoing		
VerifyOutgoingMes		See text



For a full description of how the GMS Anti-Spam AI parameters work, see the GMS Anti-Spam Guide.

The global parameters are as follows:

AIIPActivate Range 0 - 1, default = 0.

Enables/disables Artificial Intelligence (AI) checking of the number of connections from a given IP Address.

AIIPAverageMultiplier Default = 10.

Multiplier applied to the running average before the AI function in GMS Anti-Spam starts rejecting mail

AIIPCheckActive Range 0 - 1, default = 0.

Enables/disables AI checking of IP Addresses.

AIIPFailureMes

The message returned to a transaction failing the AI test. The default is "453 Exceeded IP count - please try later".

AIIPRequiredSamples Default = 7.

The number of days that the GMS Anti-Spam AI samples mail server traffic to build up a profile of your e-mail throughput.

AIIPRunningAverageMin Default = 25.

The value that is automatically given to new IP addresses that AI has not had a chance to build up a profile against

AIMAILActivate Range 0 - 1, default = 0.

Enables/disables AI checking of the MAIL Clause

AIMAILAverageMultiplier Default = 10.

The multiplier applied to the running average before the AI function in GMS Anti-Spam starts rejecting mail

AIMAILCheckActive Range 0 - 1, default = 0.

Enables/disables AI checking of the MAIL clause.

AIMAILFailureMes

The message returned to a transaction failing the AI test. The default is "453 Exceeded MAIL count - please try later".

AIMAILRequiredSamples Default = 7.

The number of days for which GMS Anti-Spam AI samples mail server traffic to build up a profile of your e-mail throughput.

AIMAILRunningAverageMin Default = 25.

The value that is automatically given to new MAIL clauses that AI has not had a chance to build up a profile against.

AIRCPTActivate Range 0 - 1, default = 0.

Enables/disables AI checking of the RCPT Clause.

AIRCPTAverageMultiplier Default = 10.

Multiplier applied to the running average before the AI function in GMS Anti-Spam starts rejecting mail.

AIRCPTCheckActive Default = 0.

Enables/disables AI checking of the RCPT Clause.

AIRCPTFailureMes

The message returned to a transaction failing the AI test. Default is "453 Exceeded RCPT count - please try later".

AIRCPTRequiredSamples Default = 7.

The number of days for which GMS Anti-Spam AI samples mail server traffic to build up a profile of your e-mail throughput.

AIRCPTRunningAverageMin Default = 25.

The value that is automatically given to new RCPT clauses that AI has not had a chance to build up a profile against.

AllowedRCPTLimit Default = 100.

This is a general Gordano parameter, limiting the maximum number of RCPT clauses allowed in one SMTP transaction. GMS Anti-Spam lets you specify different numbers for different addresses to produce an entry like this:

```
123.89.89.89:12 123.19.24.88:36
```

BypassAuthIP

You can enable a type of "whitelist" by using the new "BypassAuthIP" variable. It will allow the IPs listed to pass all GMS Anti-Spam checks. The IP addresses should be entered in a comma separated list.

IMAPMaxConnectionsPerIP Range 1 - 255

The number of simultaneous IMAP connections allowed from a single IP address, this overrides the global IMAPMaxConnections setting..

IPAllowed Default = *.

A space-separated list of banned IP addresses. Messages from these addresses are automatically rejected. The list must begin with an asterisk (*), and each address must be preceded by an exclamation mark (!).

IPAllowedMes

The message returned to a transaction failing the IPAllowed test. The default is "542 - Your server has been banned from this server".

MaxMessagesPerIPAddress

Defines the maximum number of messages that can be received to any one IP address on the server in a 24 hour period.

MaxMessagesPerIPAddressMes

The message returned when the MaxMessagesPerIPAddress limit is reached. This is a temporary failure message.

MaxMessagesPerUser

Defines the maximum number of messages that can be received for any individual user in a 24 hour period.

MaxMessagesPerUserMes

The message returned when the MaxMessagesPerUser limit is reached. This is a temporary failure message.

OnlyAcceptLocal Range 0 - 2, default = 2.

Enables/disables mail relay (see the parameter LocalIP). Values are:

- 0 — allow relay.
- 1 — at least one of the mail and RCPT clauses must be local.
- 2 — ignore mail, all RCPT clauses must be local.

OnlyAcceptLocalMes

The message returned to any transaction failing the OnlyAcceptLocal test. The default is "542 Mail relay not allowed at this site".

POPAuthSMTP Range 0 - 2, default = 2.

A successful POP logon from a non-local client adds that client to the list of allowed IP addresses for relay. If this parameter is enabled, relay for authenticated POP3 users is always allowed.

The address expires after the period given by POPAuthSMTPTime.

POPAuthSMTPTime Range 0 - , default = 3600.

The expiry time in seconds after which relay for an authenticated POP3 user expires.

POPMaxConnectionsPerIP

The maximum number of simultaneous connections to the POP service from a single IP address.

DNSBLServers

A list of DNSBL Servers, their IP response address and the action to take on a match. Held one entry per line in the file dnsblservers.txt.

DNSBLServersMes

The message returned to a transaction failing the DNSBL check. The default is "542 - Mail not accepted from server in DNSBL".

DNSBLServerMode Range 0 - 2, default = 1

Determines where in an SMTP transaction that checks are made against the list of DNSBL servers.

- 0 — disabled.

- 1 — check on connection.
- 2 — check on receipt of the MAIL SMTP protocol clause.

RejectAttachmentExt

A comma separated list of file extensions to be rejected. For example RejectAttachmentExt=vbs will reject all attachments with the extension .vbs

RejectAttachmentExtOptions Range 0 - 7, default = 0

This is a BitMask enabling further options for Attachment blocking, specifically relating to zip files.

- Bit 1 — Reject all password protected zip files
- Bit 2 — Reject standard banned extensions inside a zip file.
- Bit 3 — Reject zip files within zip files

Script

The name of the script to run for content filtering in GMS Anti-Spam.

SMTPScript

Enables the processing of scripts during receipt of messages via smtp.

SMTPMaxConnectionsPerIP

The maximum number of simultaneous connections to the SMTP service from a single IP address.

VerifyHostname Range 0 - 2, default = 0.

GMS Anti-Spam gives more options for this standard Gordano parameter:

- 0 — do nothing.
- 1 — do reverse lookup on the sending server.
- 2 — use raw IP address or the result of reverse lookup in logs.

VerifyHostnameMes

The message returned if a message fails the reverse lookup on the sending server. The default is "542 - Quoted name does not match IP address".

VerifyIncoming Range 0 - 1, default = 0.

If enabled, performs reverse MX lookup on FROM e-mail address.

VerifyIncomingMes

The message returned if a message fails the reverse lookup on its FROM address. The default is "542 - From: address does not have MX/A record".

VerifyOutgoing Range 0 - 1, default = 0.

If enabled, performs reverse MX lookup on RCPT e-mail address.

VerifyOutgoingMes

The message returned if a message fails the reverse lookup on its RCPT address. The default is "552 - From: address does not have MX/A record".

Domain-specific parameters

Name	Range	Recommended Value
AllowedSenderIP		
AllowedSenderIPMes		
ByPassWordCheckIP		
MaxMessageSizeOut		
RestrictedWordFile		
RestrictedWordFileEnabled	0-1	0
RestrictedWordFileMes		See text
RestrictedWordFileReturn	0-1	1
RestrictedWordFileSave	0-1	1
RestrictedWordMode	0 - unlimited	0

AllowedSenderIP

Defines the IP addresses that are allowed to send mail purporting to be from your domain.

AllowedSenderIPMes

The error returned to any server that breaks the above rule.

BypassWordCheckIP

A space-separated list of IP Addresses. Mail from these addresses is not checked for restricted words.

RestrictedWordFile

The pathname of a file in each domain which is used to check for barred or censored key words in an incoming e-mail message. The

key words or phrases are defined in this "filter file" named in the Registry entry, one word per line. For example the file:

```
naughty  
disgusting phrase
```

would reject an e-mail message which contains "naughty" or "disgusting phrase" in its message body. A message will be sent from the local Postmaster, with the original message attached, informing the sender that their message has been returned.

RestrictedWordFileEnabled Range 0 - 1, default = 0.

Enables/disables use of the restricted word file.

RestrictedWordFileMes

The message returned to the sender of a message which fails the Restricted Word test. The default is "500 Phrase in e-mail not acceptable".

RestrictedWordFileReturn Range 0 - 1, default = 1.

Enables/disables returning of the message that failed the Restricted Word test.

RestrictedWordFileSave Range 0 - 1, default = 1.

Enables/disables saving of messages that fail the Restricted Word test.

RestrictedWordMode Range 0 - unlimited, default = 0.

Specifies which part of a mail message is checked against the restricted word file. 0 checks both the header and the message body, 1 checks only the header and 2 checks only the body. Any figure above 2 will mean the entered number of lines will be checked against the restricted word file.

11 Virus Scanner Registry Parameters

The section entitled "Configuration — The Registry" on page 49 described the Gordano general Registry parameters and how to change these. This section describes the Registry parameters which are used by Virus Scanner.

11.1 Introduction

GMS allows configuration of the Virus Scanner on a per domain basis.



If you do edit the Registry, only use Regedt32.exe, not another version of the Registry editor. Regedt32.exe supports multi-string Registry entries.

Note the following:

- If you use the Gordano user interface to change any value *back to the default value*, the parameter is automatically deleted from the Registry.
- Values which have a range of 0-1 are Boolean: 0 = false (disabled), 1 = true (enabled).
- Values which have a range shown as "0-" or "1-" do not have a specific upper limit.

11.2 Parameters

Global Parameters

The parameters are as follows:

Name	Range	Recommended Value
VScanAction		This is a bitmap with a range of
RedirectMail	1	1-4, each bit has the meaning
RejectMail	2	indicated
ReturnMail	4	
VScanActivate	0 - 2	1
VScanAlert		This is a bitmap with a range of
AlertPostmaster	1	1-4, each bit has the meaning
AlertSender	2	indicated.
AlertUser	4	
VScanInline	0 - 1	0
VScanLog		
VScanPostmasterAcc		postmaster@domain.name
VScanProgram	0 - 1	0
VScanRedirectAcc		
VScanReturnText		
VScanTimeoutl	0 -	10,000
VScanVendor		

VScanAction

Specifies the action to take with messages containing attachments that fail the virus check.

VScanActivate Range 0 - 2, default = 1.

Enables/disables the Virus Scanner.

- 0 - Disables the scanner
- 1 - Enables the scanner
- 2 - Disables disinfection when viruses are detected.

VScanAlert

Specifies who to alert if an incoming mail message is found to contain a virus.

VScanInline

Allows or disallows the scanning of the body of messages in addition to any attachments. The default is 0 disallowed. The value 1 allows the scanning of the message body.

VScanLog

The pathname of the file used for Virus Scanner logging.

VScanPostmasterAcc Default is postmaster@domain.name.

The e-mail address of the person alerts are sent to.

VScanProgram

The Virus Scanner program to use. Set this from the user interface, not using Regedt32.exe, as only those programs displayed on the Virus>Select Scanner page will be supported.

VScanRedirectAcc

The account which any files found to contain a virus will be redirected to.

VScanReturnText

The text message, if any, appended to the returned mail.

VScanTimeout Range 0 - , default = 10,000.

If you use Norton (NAV NT v2.0), you may need to set a Timeout value for it to work correctly. The default value is 10,000 but you may need to either increase or decrease this time depending on the speed of your server. Try increasing/decreasing it in increments of 2000 until you find a setting that suits your setup.

VScanVendor

The Virus Scanner program vendor. Set this from the user interface, not using Regedt32.exe, as only those vendors displayed on the Virus>Select Scanner page will be supported.

Domain Parameters

The parameters are as follows:

Name	Range	Recommended Value
VScanAction		This is a bitmap with a range of
RedirectMail	1	1-4, each bit has the meaning
RejectMail	2	indicated
ReturnMail	4	
VScanActivate	0 -2	0
VScanAlert		This is a bitmap with a range of
AlertPostmaster	1	1-4, each bit has the meaning
AlertSender	2	indicated.
AlertUser	4	
VScanRedirectAcc		
VScanReturnText		

VScanAction

Specifies the action to take with messages containing attachments that fail the virus check.

VScanActivate Range 0 - 2, default = 0.

There are three options as to how a virus scanner is used under a domain

- 0 — inherit the system settings
- 1 — Use specific domain settings
- 2 — Disable the virus scanner for this domain

VScanAlert

Specifies who to alert if an incoming mail message is found to contain a virus.

VScanRedirectAcc

The account which any files found to contain a virus will be redirected to.

VScanReturnText

The text message, if any, appended to the returned mail.

12 Registry Parameters GMS Windows IM Client

The Windows IM client variables are all stored under the user space in the registry, HKEY_CURRENT_USER/Software/Gordano/GMS Instant Messenger

If system administrators wish to set any of these variables automatically at installation time they can do so by taking advantage of the additional facilities provided via the MSI installation options.

Variables

Name	Range	Recommended value
EmailClient		
EmailClientCommand		
EmailClientWindowName		
EmailMode	0-1	0
LogoFile		
LogoLink		
InstallDir		

EmailClient Range 0-1, default 0

The registered name of the email client to be used if EmailMode is set to a value of 1. May be set to any of the clients registered on the machine or to GMS WebMail.

EmailClientCommand

A custom launch command to be used when the value of EmailClient has been set to GMS WebMail. The command takes the format `http://server:port?param1=value¶m2=value`, the logon username and password for GMS WebMail is automatically added.

This setting can also be used to launch any other type of executable using the form `c:\full_path\application.exe`

EmailClientWindowName

Specifies the name of the window created by running a custom command. Providing the window name allows the same window to be re-used each time the client is launched rather than using a new window.

EmailMode Range 0-1, default 0

A value of 0 will use the default email client registered on the workstation running the IM client. If set to a value of 1 you must specify the client to be used in the EmailClient variable.

LogoFile

Specifies a path to a bitmap file to replace the logo in the top right hand corner of the main IM client window.

LogoLink

Specifies a custom link URL opened when clicking on the Logo in the main IM client window.

InstallDir

This value is added by the installation script so that MS Outlook knows where to find the IM client executable file.

13 Directory Structure

This section describes the directory structure a Gordano installation creates under the location specified by the installer. This location is specified in the Registry entry 'BaseDir'.

This section shows:

- Directories in the global Gordano structure.
- The directories created for each domain.
- How to change the directory structure.

13.1 Global Directories

This table shows the Gordano directories which are not user-specific:

Directory	Function
\$BaseDir\$	The directory Gordano products are installed in.
Backup	Used to save old files if the system is upgraded.
Bin	Where Gordano products keep all their program files.
Cache	Files cached by the proxy server
<Domain 1>	First domain.
<Domain 2>	Second domain, if any. Other domains follow this.
Install	Any files that the installation program could not copy to the correct place as they were in use.
ListIn	Contains all List postings prior to them being processed. The subdirectory Done holds the processed messages.
ListMan	Contains all messages sent to the List Manager prior to them being processed. The subdirectory Done holds the processed messages.
Log	Contains all mail logs (debug etc.)
MesLog	Used if Message Logging is enabled.
MML	Contains user written MML scripts, timed events, etc. Also contains these manuals..
MySQL	Contains GLWebMail Address book and vCard tables and data.
Out	Contains outbound mail waiting to be sent, held in sub-directories according to domain name.
Profiles	Contains the profiles index file and profiles sub folders detailing profile information

Directory	Function
\$BaseDir\$	The directory Gordano products are installed in.
SpamMes	Contains files saved because they broke the JUCE rules.
Temp	Used as a buffer for incoming messages.

The directories are described in turn below (for the domain directories, see the section which follows this).

Backup

This directory only exists if you have used the SETUP program to carry out an upgrade. It contains details of the last installation.

Bin

Contains the Gordano applications, service applications, DLLs and other files used by Gordano products.

ListIn

Contains all List postings before they are processed. The subdirectory Done contains messages processed by the list manager.

ListMan

Contains all messages sent to the List Manager before they are processed. The subdirectory Done contains messages processed by the list manager.

Log

This directory contains several types of log file, including one for each Gordano service. For details of their names, see "Log File names" on page 303. For details of log file contents, see "Log Contents" on page 301.

MesLog

If message logging is enabled, the base directory holds e-mail messages relayed through the server.

MML

This contains user-written .MML files and also any .HTM files used to customise the interface.

MySQL

Contains the GLWebMail address book and vCard data and tables

Out

There are two types of file in this directory:

- External mail — if a user sends mail to user@externaldomain, a file is written to a subdirectory under Out named after the external domain.
- List mail — if batch processing is enabled for the list, the delivery files will be written to the base of the out directory

initially and post will try and deliver them from there. Once it has passed through all the addresses once any that failed to be delivered will then be fed to the domain directories for onward processing.

Profiles

This contains index.txt listing the profile sub directories and the domains they refer to.

SpamMes

If the options which let you save Spam are enabled, the messages are saved here.

Temp

This acts as a buffer so that incoming mail is not lost. Incoming mail is stored here until its transmission ends successfully, at which time it is moved to the user directory.



The MySQL folder will be present on installs that do not specify GLWeb-Mail. This ensures the upgrade to this product can be completed at a later date without the need for the installation of additional files

13.2 Domain Directory Structure

For each additional full domain that you set up, a directory structure is created using the name of the domain as a subdirectory, attached to the Gordano base directory.

This table shows directories which are domain-specific. These are subdirectories of the directory <domain>.dom. This structure is repeated for each domain set up:

Directory	Function
<domain>.com	The first domain set up
BadMes	Holds all bad messages to the domain.
MesLog	Message logs for the domain.
Users	Contains subdirectories for each user under the domain test.com.
User 1	Contains the user's mailboxes and configuration files.

.... User n	Further user directories.
WWW	Holds the htm pages to deliver a traditional web site for the domain.

The directories are as follows:

BadMes

If this directory exists and the Registry entry 'SaveBadMes' does not exist or is non zero, it will contain the messages that have been sent to NULL. Each message will be accompanied by a .TXT file explaining why the message was ignored.

MesLog

If this directory exists and LogAllMessages is defined and non zero (that is, logging is enabled), all the messages accepted by SMTP will be recorded in a file in this directory. At midnight a new log file is started. The directory contains .idx and .mbx files.

Users

This directory contains a subdirectory for each user in the domain. This contains files of these types:

- .idx — an index for the mailbox file.
- .mbx — the mailbox.
- fax — fax files.
- auto — autoresponder message.
- plan — text document containing the user's plan details (if any have been set up).
- list manager files etc., depending on the account.

WWW

Gordano messaging products can also act as a traditional web server running on port 80 (the standard WWW port), all the files to be delivered are held in this directory and any sub-directories.

13.3 Profiles Directory Structure

The directory holds an index.txt file referencing the sub directories and their data to the relevant domains on the server.

The sub directories structure is as follows:

- Directories denoted by a number contain a Base Profile for a domain. Directory 0 contains the System Base Profile. These profiles are created by default when a domain is created.

- Directories denoted by letters are profiles that have been manually created.

Each directory contains 2 files:

- Userlist.txt — contains the user names to which the profile applies
- Variables.txt — contains the specific rights that have been assigned to the profile

13.4 Changing the Directory Structure

You can set up two entries which are used to save domain and user files to locations other than the main Gordano basedir.

- For users you can set up an entry called homedir.
- For a domain you can set up a directory basedir.

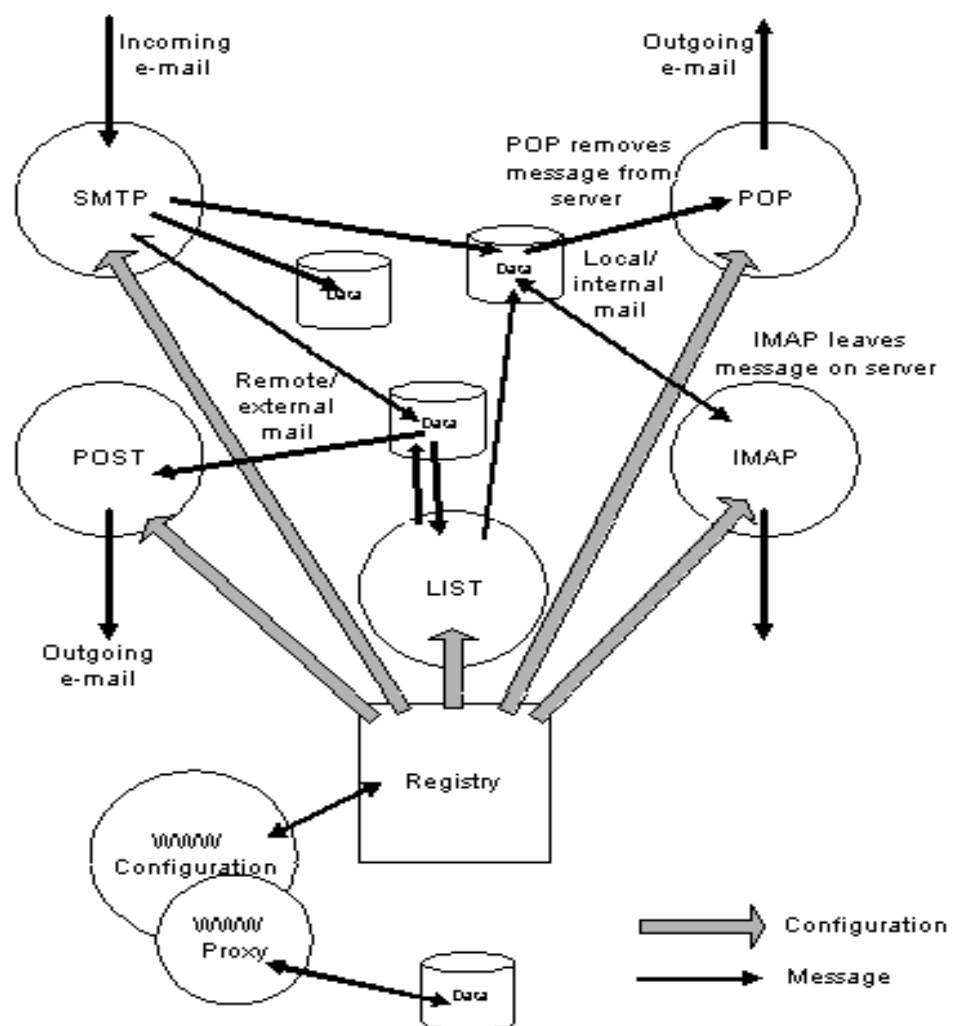
14 Services

This section describes the Gordano services, as follows:

- Overview of the interactions between POST, POP, SMTP, IMAP, LIST and WWW.
- Enhanced SMTP (ESMTP) — available features are 8BitMIME, AUTH, Delivery Status Notification (DSN), Enhanced Status Codes, ETRN, Pipelining and Restart, Size, VRFY and XTND.
- The switches used by POST, POP, SMTP, IMAP and LIST.

14.1 Interactions between Services

The following diagram summarises the way e-mail and configuration data is passed between services.



The components shown are:

- POP server — lets a POP client collect mail from a Gordano server.
- POST server — uses SMTP to post mail to non-local domains.

- IMAP server — lets an IMAP client collect mail from a Gordano server.
- SMTP server — accepts mail from the Internet and from local mail clients.
- LIST server — provides list services. The Gordano list server, GMS Communication Server, is described in the *GMS Communication Server Guide*.
- WWW Configuration server — allows configuration of Gordano products using a Web browser.
- Registry — NT's means of storing system information.

14.2 ESMTP (Enhanced SMTP)

ESMTP offers a set of extensions to SMTP. These are enabled/disabled using Incoming > ESMTP or global Registry parameters like ESMTPetrn; see page 62. The extensions are described below.

8BitMIME

The sender uses this command to announce that it supports higher bit ASCII transmission — the transmission of MIME messages over SMTP, allowing for a content body consisting of a MIME message containing arbitrary octet-aligned material.

Auth

Use Auth to set up authenticated SMTP transactions. Gordano mail servers support three types of authenticated SMTP, set by the value of the command ESMTPAuth, as follows:

- 1 = LOGON authentication. This uses a unique value generated in the POP server sign-on message to encrypt the password.
- 2 = MD5 authentication. This uses a unique string appended to the AUTH=MD5 line to encrypt the password.
- 3 = CRAM-MD5 authentication.
- 4 = PLAIN authentication. This users plain text.

LOGON and MD5 work in similar ways to POP's PASS and APOP mechanisms respectively (except that all data is binhex encoded). Both require the client to logon to the server - i.e. the client must know a username and password.

Delivery Status Notification (DSN)

This option asks the Gordano mail server to confirm that a transaction was completed as desired. The keyword associated with this ESMTP extension is DSN.

Enhanced Status Codes

Enhanced Status Codes give precise error codes relating to the delivery of mail. These are only delivered to servers issuing the EHLO command to indicate that they understand ESMTP; all other servers receive the standard response codes. The keyword associated with this ESMTP extension is ENHANCEDSTATUSCODES.

For example, when delivery fails due to a routing failure, using standard SMTP the server might return "550 delivery failure" while using enhanced SMTP the server might return "550 5.4.3 delivery failure". The additional 5.4.3 included in the reply indicates that delivery failed due to a routing server failure; for details, see "Enhanced SMTP error codes" on page 312.

ETRN

ETRN, also known as QSND, is an SMTP command specifically designed to allow integration with dial-up mail servers. A dial-up mail server can connect to the Gordano mail server and issue the ETRN command to force all the e-mail for it server to be posted out. The keyword associated with this ESMTP extension is ETRN. For details of the Registry parameter, see page 62.

The standard ETRN specified in RFC 821 is not secure as anyone issuing the ETRN command for a domain can retrieve all mail for that domain. For this reason the Gordano server requires an extra parameter containing a password to be passed to it.

Setting up an ETRN queue

To set up an ETRN queue for a domain within your Gordano server:

1. Do not set up the domain within the server.
2. Ensure that the Gordano server will accept mail for this domain by adding the domain to the "Allow Relay for..." section.
3. Add an entry to your Sending Rules as follows:

```
domain.dom server.domain.dom 25 0
```

This means send all mail for domain.dom to server.domain.dom on port 25, but only send it when requested.



The domain server.domain.dom must have a valid A record in DNS.

Using ETRN with a GMS server as the client.

When using a Gordano server to retrieve mail via SMTP over a dial-up connection, you can use the utility NTMEtrn.exe provided as part of the GMS Accessory Pack.

ETRN queues can also be password-protected by adding an extra parameter to the relevant line in the Postservers.txt file as follows:

DOMAIN.DOM SERVER.DOMAIN.DOM 25 0 PASSWORD

In the above example the Retry time for the domain is set to 0 (zero) so the Gordano server does not try to send mail itself, but always waits to receive the ETRN command.

Pipelining

A sending server can use pipelining to send all the messages it has to a receiver in one burst over a single SMTP connection, without sending a Reset command after each message. It does not wait for a response from the remote server — once all the commands have been issued, the remote server issues all its responses at once.

With pipelining, if a user on a remote server sends a message to multiple users on your server, this uses only a single SMTP connection with one MAIL FROM clause and multiple RCPT clauses and consequently the message body is only transmitted once. This reduces the time it takes to send the messages.

It looks like this (S: = server, C: = client):

```
S: <wait for open connection>
C: <open connection to server>
S: 220 companyB.dom SMTP service ready
C: EHLO xyz.companyA.dom
S: 250-innosoft.com
S: 250 PIPELINING
C: MAIL FROM:<joe@xyz.companyA.dom>
C: RCPT TO:<ned@companyB.dom>
C: RCPT TO:<dan@companyB.dom>
C: RCPT TO:<liz@companyB.dom>
C: DATA
S: 250 sender <joe@xyz.companyA.dom> OK
S: 250 recipient <ned@companyB.dom> OK
S: 250 recipient <dan@companyB.dom> OK
S: 250 recipient <liz@companyB.dom> OK
S: 354 enter mail, end with line containing only "."
...
C: .
C: QUIT
S: 250 message sent
S: 221 goodbye
```

The keyword associated with this ESMTP extension is PIPELINING.



Pipelining and Restart are alternatives.

Restart

This is also known as Checkpoint. If a connection is dropped part way through the transmission of a mail message, on reconnection the Restart command from the sender gives the receiver the option

of continuing from the point it had reached, rather than starting at the beginning. This is particularly useful in the transmission of large mail messages.

The keyword associated with this ESMTP extension is CHECKPOINT.

Size

The Size command lets a restriction be placed on the total size of messages accepted for a particular domain. The keyword associated with this ESMTP extension is SIZE n, where n is the size of the message in bytes. The sending server uses the command to state that it has a message of the specified size for the receiving server. The receiver replies to either accept or reject the message.



This command is not particularly useful at the moment due to the number of other mail servers that lie about the size of the message.

VRFY

The VRFY command is used to verify a user name. It allows external servers to check that an e-mail account actually exists on your server. The keyword associated with this ESMTP extension is VRFY and the argument is a character string specifying a user name. The response may include the full name of the user and must include their mailbox.

XTND

XTND supports two elements, XMIT and XLIST.

XMIT

XMIT is used to send mail via POP servers rather than SMTP, as follows:

1. User sends xtndxmit.
2. POP replies "+OK Send message".
3. User sends the message, a block of data terminated by CRLF.CRLF, to the POP server.
4. POP server takes the message and passes it on for delivery by POST.

XLIST

XLIST is used to list message headers. It can operate in several modes:

- Get all headers.
- Get headers matching the given clause (Received, To, etc.).

- Get the header for a specific message ID. This uses Xlist's last parameter (n), which is optional. For example, to list the header of message 7 it would use **xtndxlist clause 7**.

14.3 POST, POP, SMTP, IMAP, LIST and WWW Switches

Each of these services supports the following switches or parameters:

Switch	Description
-d	Displays the computer's name. This name is used by the key software and is required when you register your software.
-h	Lists all the parameters available to the service (as in this table).
-i	Install — adds the service to the service manager and creates default entries in the Registry, if they do not already exist. If it's a new version of the service, it replaces the existing version with this. If you already have version 2.00 or later, the entries are automatically updated.
-r	Explains how to register to use the service.
-s	Displays the installed status of the software. This reads the key and indicates what is allowed.
-u	Uninstalls the service from the service manager.
-v	Displays the version number.

Do not run any of the programs with no switches. They will appear to hang. If you do this accidentally, use CTRL+C to stop the program.

15 Customising the User Interface

You can alter four aspects of the Gordano Web Interface:

- the header and footer of each page.
- the first page.
- the Support pages.

This section describes how to do this.

15.1 Background on Page Structure

Each of the Gordano configuration pages can be branded by the simple addition of 4 html files to the MML directory. These html files will be embedded inside the normal GUI pages and displayed as part of them.

15.2 Replacing Pages

To replace the pages, just create the relevant file in the \$BaseDir\$/MML directory. The Web Configuration server picks up the new information immediately and displays it.

It is important that these pages do not have any HTML error, for example uncompleted tables, since this may make your Gordano products unconfigurable. The four files have these names:

File	Function
\$BaseDir\$/MML/Head.htm	HTML code to be displayed at the top of each page.
BaseDir\$/MML/Domain/Head.htm	HTML code to be displayed at the top of each page but only for users of "Domain".
\$BaseDir\$/MML/Foot.htm	HTML code to be displayed at the bottom of each page.
\$BaseDir\$/MML/Domain/Foot.htm	HTML code to be displayed at the bottom of each page but only for users of "Domain".
\$BaseDir\$/MML/First.htm	HTML code which will be displayed in the "work space" once the user has logged on to the Gordano interface.
\$BaseDir\$/MML/Support.htm	HTML page which will be displayed when the user selects the Support menu option.
\$BaseDir\$/MML/Domain/Support.htm	HTML page which will be displayed when the user selects the Support menu option specifically for "Domain".

16 Redirect, Template and Postservers Files

This section describes three types of file used by a Gordano mail server:

- The Redirect file used for smart routing.
- Postservers.txt — the file POST may use when it sends outgoing mail. The entries list servers and are known as the *Sending rules*.
- Template files — used by autoresponders.

16.1 The Redirect File

Accounts to be redirected are listed in a Redirect file using a notation that allows entire domains to be included using wildcards. The Redirect file is an ASCII text file located in the Gordano BaseDir directory and named by the Registry parameter RedirectFile. It contains a list of message redirection commands specifying the actions to take for messages coming from or going to certain e-mail or IP addresses. To set up smart routing, see "Advanced Management" in the *GMS Administrator's Guide*.

Syntax

Each entry in the redirect file consists of a line which takes this form:

```
<from addr> <from IP> <to addr> <to IP> <action> <result>
```

where:

- From addr — is the MAIL clause. This can be the wildcard '*'.



You can also use partial wildcards, for example **@domain.dom* means all users at *domain.dom*. Similarly, *123.123.123.** means any IP address in the Class C block *123.123.123.0* to *123.123.123.255*.

- From IP — is the remote IP address. This can be the wildcard '*'.
- To addr — is the RCPT clause. This can be the wildcard '*'.
- To IP — is the local IP address. This can be the wildcard '*'.
- Action — is a "T" followed by one of these combinations:
 - F <message> = Refuse with message given (see <result>).
 - W <message> = Retry later with message given (see <result>).
 - R<account> = Redirect to the specified account (see <result>).
 - I = Don't take any action on a match (ignore/do not redirect).
- Result — is one of the following:
 - The message to return, entered as a string contained within quotes ("). This will begin either 4xx or 5xx (the following example shows one of each of these). A retry later message must have the code 4xx and a refusal message must begin with the code 5xx.

The final two digits of this number depend on the meaning of the message. In the example shown below, a refusal would contain the message "500 mail rejected".

- The account redirected mail is sent to. This address must be local. Set it to NULL to discard messages, otherwise any type of account can be specified.



This feature is very powerful — an entry such as "` * * * T R NULL`" would trash every message passing through the server! Take great care.*

The following example shows one action of each form, in the order that they're listed above:

```
* 194.194.231.231 * * T F "500 Mail rejected"
dean@test.dom * * * T W "400 Mail retry later"
* * user1@test.dom * T R postmaster
trouble@* T I
```

This would:

- Reject mail from the address 194.194.231.231, sending back a message "500 mail rejected".
- Temporarily reject mail from the user dean@test.dom, sending Dean a message "400 try later".
- Redirect all mail to user1@test.dom to the postmaster.
- Ignore all mail from users named *trouble* in any domain.

Operation

When the Gordano server receives a message it checks the protocol arguments MAIL and RCPT and the From and To IP addresses. If a match is found in the redirect file, the content of the RCPT argument is replaced by the redirected account name. The mail message then proceeds through the matching routines.

The whole redirect file is parsed for every message. This means entries at the start of the file can be overridden by later entries, as in the following example that deletes all messages from a particular domain, except the postmaster's:

```
# trash mail from company.dom (except postmaster's)

*@company.dom * * * T R NULL
postmaster@company.dom * * * T I
```



Mail from the postmaster is allowed so that any query as to why the domain has been banned can be answered.

Redirection can be used to set up the distribution of event messages to any account. For example, the entries:

```
* * info@company1.dom * T R info1@company1.dom
* * info@company2.dom * T R info2
```

allow two different information messages to be sent to the Information accounts of two different domains (providing the autoresponder accounts info1 and info2 have been setup).

You can also catch messages coming from a single address and redirect them to a user such as the postmaster, as in the following example:

```
# redirect messages from problem user to postmaster

trouble@xxx.dom * * * T R postmaster@company1.dom
```


16.2 Postservers.txt

When POST sends outgoing mail, it works through the list of servers defined in the `postservers.txt` file (known as the *Sending rules*). For details of setting this up using the Gordano user interface, see "Advanced Management" in the *GMS Administrator's Guide*.

Each line in this file has four entries:

- Target address — a "*" means mail for all domains.
 - Server — for a permanent or dial-up connection, this should be the name of the ISP's server.
 - Port — the port number (25 is the default).
 - Retry time — the retry time in minutes. For a dial-up system, this should be 0, meaning "do not retry"
-
- If a "*" is found, POST resolves the message's domain name using MX records and attempts to send the mail to that machine.
 - If any other entry is found, POST attempts to send the message to the given A record.

POST reads the `Postservers.txt` file from top to bottom to find the next entry in the server list. This means that later entries override those at the top of the file. For example, with a file containing the two lines shown below, mail to all domains is resolved by MX records through DNS, except for mail to `DOMAIN.DOM`, which is always sent to `SERVER.DOMAIN.DOM`:

```
* * 25 12
DOMAIN.DOM SERVER.DOMAIN.DOM 25 12
```

For a permanent connection you would use a record like this:

```
* mail.isp 25 10
```

Where:

- * — means mail for all domains.
- mail.isp — is the ISP's server.
- 25 — is the port number (25 is the default).
- 10 — is the retry time in minutes.

For a dial-up connection you would use a record like this:

```
* mail.isp 25 0
```

Where the entries are as for a permanent connection, except for the retry value of 0, meaning "do not retry".

16.3 Template File Format

When a mail message arrives at a mailbox that has been configured as an autoresponder or FAX account, the Gordano server finds the file to be returned. For autoresponder details, see the *GMS Administrator's Guide*.

The server scans through the file looking for special command strings. These are lines that begin with a "-" character or variables like %variable%. When a line starting with "-" is found, the text following the "-" is added to (or used to replace) the equivalent line of the mail message header. For example:

-Subject: Some subject

will change the subject of the message returned by the autoresponder message to always read "Some subject". In the same way:

-Sender: <email_address>

will change the sender of the message returned by the autoresponder message to the given e-mail address.

Use this option with extreme care — autoresponder outputs normally have a sender of "null" to avoid messages bouncing back and forth. If you change this field, make sure that the output message is never addressed to another autoresponder. The scan continues until the line is encountered that does not start with a "-".

In addition, any message file can have the following fields:

- %subject% — the message subject.
- %from% — the complete From clause.
- %from-email% — e-mail address in the From clause.
- %from-user% — if this is in the From clause, the user's name, otherwise their e-mail address.
- %to%
- %to-email% — the e-mail address in the To clause.
- %to-user% — if this is in the To clause, the user's name, otherwise their e-mail address.
- %date% — the time stamp on the message.
- %cc% — any CC names the message has.
- %message% — the entire message body.
- %header% — the entire message header.

These fields are replaced by the appropriate string from the incoming message.

For example the following information account message file:

-From: null@company.dom (support)
-Subject: Re: %subject%

Many thanks for your e-mail message about %subject%.
Messages will be answered in strict rotation.

would cause the server to respond with the message:

From: null@company.dom (support)
Subject: Re: Fried tomatoes
To: another@company.dom
Date: Fri, 6 Oct 95 08:16:17 +0000 (GMT)

Many thanks for your e-mail message about Fried tomatoes. Messages will
be answered in strict rotation.

17 SharePoint

Microsoft SharePoint is a collection of products and software elements that include collaboration functions, process management modules, search modules and a document-management platform. SharePoint can be used to host web sites that access shared workspaces, information stores and documents, as well as user defined applications such as wikis and blogs. All users can manipulate proprietary controls called "web parts" and interact with content such as lists and document libraries.

To enhance the level of collaboration available to users of the Gordano Messaging Suite we have provided a Web Part that directly integrates GMS into any SharePoint installation. This allows full integration of calendars, address books and other PIM data, allowing both individual and group user accounts to synchronize and manipulate their GMS data directly within the SharePoint interface. The GMS web parts can be configured to display address books and calendars for either a user's own account or for a shared account that the user has permissions to view.

The GMS Web Parts package may be installed with Sharepoint Services 2.0 or 3.0. On the client, the web parts are intended for use with Internet Explorer 6.0 or later and Mozilla Firefox 2.0 or later.

17.1 Installation

To install the GMS Web Parts you will need to copy the file GmsWebParts.CAB from the distribution zip file into a temporary folder on the SharePoint server. Open a command prompt window and change the current directory to the folder containing the cab file. Use the SharePoint stsadm utility to install the web part package using the following command;

```
stsadm -o addwppack -globalinstall -force -filename GmsWebParts.CAB
```

Before the GMS Web Part will become available in SharePoint you will need to restart the IIS instance on the server. The simplest method of doing this is to use iisreset from the same command prompt window. restart IIS;

```
iisreset
```

17.2 Configuration

The GMS Web Parts need to be configured from within SharePoint. Open Sharepoint and navigate to the page to which you want to add a web part. Click on the modify my/shared page link in top-right corner and select the option for add web part and then the option for browse within the pop-up menu. The newly added GMS web parts (Calendar, Contacts, Tasks and Notes) will be visible

within the Virtual server gallery. Select the required web part and drag it on to the page at the desired location.

Once added to the page, the web parts will initially display the following message;

One or more web part properties have not yet been configured or have been set to invalid values.

This message will persist until the properties for the web part have been correctly initialised. To do this, click on the "Open the tool pane" link and expand the GMS Properties section. All GMS Web parts have the same set of core properties (although they may of course be set to different values). Once you have set the web part properties, click on **Apply** or **OK** to commit the changes. Note that it may be necessary to close and restart the browser session before the changed properties become effective.



Please note that the Sharepoint administration interface is optimised for use with Internet Explorer. We would advise you to use Internet Explorer for the initial installation and configuration of the GMS Web Part package since this will more closely correspond to the steps described below.

GMS Collaboration Server

Enter the server name or IP address of the server on which the GMS Collaboration Server is running. Typically this will be the address of your GMS mail server with the port number 8376 (which is the default port for the collaboration server). e.g.

<http://mail.example.com:8376/>.

Rows per Page

This option defines the number of lines of data visible within each page of the web part when displaying data in list mode. (Note that this property will have no significance for the calendar web part)

Container Name

The Container Name option allows you to define the GMS data folder to be displayed/maintained. A value of personal will correspond to the top-level My Addresses or My Calendar folder, although you may enter the name of any sub-folders which you have created. If you enter the name of a sub-folder, it should not be prefixed with the name of any parent folder. For example, the following table shows how the the folders and sub-folders of a simple GMS address book structure should be referenced within the corresponding GMS Contacts web parts.

GMS Folder Name	Web Part Container Name
My Addresses	Personal

GMS Folder Name	Web Part Container Name
My Addresses/Suppliers	Suppliers
My Addresses/Dealers	Dealers

GMS Login

This should be the email address of the the GMS user whose account credentials will be used to access the data. Typically this will be the owner of the data container which is to be accessed. However, in the case of access to a shared container it will be the account to which access rights have been granted and the original owner of the cotainer must then be saved in the **Data Access Account** property.

GMS Password

This should be set to the password corresponding to the GMS account defined in the **GMS Login** property.

Data Access Account

When accessing a shared folder, this property should be set to the account name of the user who owns the container to be accessed. When accessing non-shared containers this property should be left empty.

17.3 Calendar Specific Properties

The Calendar Web Part includes some additional properties which must also be configured before use;

GMS Timezone Offset

GMS internally stores all dates and times in GMT. To enable the calendar to adjust these times into a local time zone, this property maintains the number of whole minutes which will be added to any GMT time in order to resolve to your local time zone. This property may be set to a negative value for those Time Zones which are ahead of GMT.

18 Gizmos

A Gizmo provides a means of adding functionality to GMS WebMail by interfacing with external data sources. This is more commonly known as mashup technology. A mashup is a small web application that allows the combination of data from more than one source into a single integrated tool, a good example is the use of data from Google Maps to add location information to an address book entry (an example of this is included).

As a starting point we have provided a number of Gizmos within GMS WebMail, further examples are available within the GMS Option Pack. These examples have been designed to show what can be achieved using Gizmo technology. The functionality of each is described below along with an indication of the effect on GMS WebMail.

Twitter

Implements the Twitter API allowing you to see posts from other Twitter users who you are following. It also allows you to post your own Twitter updates and to reply to return comments to others updates. Twitter commands can also be issued as part of you update.

- How to add a simple menu button
- How to add an item to the Tree
- How to add an item to the Panel
- How to enable Gizmo configuration screens
- How to integrate with an external API

RSS Reader

Implements an RSS Reader within the GMS interface. A configuration screen is included allowing you to subscribe to your chosen RSS Feeds.

- How to add a simple menu button
- How to add an item to the Tree
- How to add an item to the Panel
- How to enable Gizmo configuration screens
- How to integrate with multiple external sites using AJAX

Mimic

Allows sorting of the GMS tree in a similar manner to Outlook, i.e. it mimics Outlooks behavior.

- How to monitor the panel for changes

Archive

Moves a selected message or messages to a folder called "Archive".

- How to add a simple menu button.
- How to over-ride existing menu buttons.

Dictionary

Allows you to look up a highlighted word in a dictionary. Uses the babylon.com dictionary and displays the result as an alert.

- How to integrate with an external site using AJAX.
- How to add a simply menu button.

Google Translate

Translates highlighted text from English to French.

- How to integrate with an external site using AJAX.
- How to add a simply menu button.

Empty Folder

This button only appears when either the Trash or Quarantine folders are selected - allows a single click empty of that folder.

- How to add menu buttons.
- How to refresh the message selector.

Hide buttons

Allows you to remove (or re-add) buttons from the menu that aren't commonly used.

- How to hide menu buttons.
- How to show hidden menu buttons.

Compose

Displays/Hides features within the Compose window, such as the BCC and Attachments fields. There are two examples of this Gizmo, one using GreaseMonkey code and the other using jQuery code.

- How to extend GMS WebMail functionality.
- How to add/remove functionality from a window.

Mark Mail

Buttons to mark a message as read or unread in webmail.

- How to add menu buttons.
- How to extend GMS WebMail functionality.
- How to refresh the message selector.

UK Postcodes

Where a UK postcode is shown in an address book entry, clicking on this Gizmo will show a map of the location via a mashup with Google Maps.

- How to modify what is displayed in the preview pane

Skype

Detects a Skype contact address and gives an option to start a skype call or chat session.

- How to add menu buttons that are enabled\disabled for each addressbook entry displayed

Whitelist

This will only appear when the Quarantine folder is selected - it takes the currently selected message and moves it to the Inbox, and adds the senders email address to the whitelist address book.

- How to add menu buttons.
- How to extend GMS WebMail functionality.
- How to refresh the message selector.

Mouse Gestures

This is an "invisible" Gizmo which allows navigation through message lists using mouse gestures. You need to right click with the mouse to enable the gestures which work as follows

- **Up** - moves up a message
- **Down** - moves down a message
- **Right** - replies to all recipients of a message, or opens tasks, notes, etc.
- **Left** - deletes a message



Only the administrator can add new Gizmos to GMS WebMail.

Break

Monitors usage patterns within WebMail and forces users to take a break, useful from a Health & Safety perspective. This is a good example of an Admin Gizmo, which the end user has no control over.

- How to monitor usage.
- How to extend WebMail functionality.
- How to enable an Admin Gizmo.

Compose Quiz

Interacts with the Send button when composing an email. If you try and send an email within the configured period you have to complete a short quiz before the message will be sent. Uses can include preventing the sending of messages when you may not be fully alert, or can be used educationally for school pupils.

- How to monitor usage.
- How to extend WebMail functionality.
- How to use HTML Layers.

18.1 What are Gizmos

A Gizmo is simply a javascript file containing the javascript to create an object with a set of predefined prototypes. The code required to produce a Gizmo is straightforward and is easily producible by those with even minimal javascript experience.

The code required to produce a Gizmo is similar to that used in Greasemonkey scripts, and these scripts can be easily amended to produce Gizmos. This opens up a large external development community that can be used to leverage the power of Gizmos. At the last count there were over 18,000 Greasemonkey scripts available, not all of which would be suitable for use within GMS.

Greasemonkey is a Firefox extension that allows you to customize the way webpages look and function. Once the scripts are adapted for use within GMS they become browser independent and you are not restricted to using Firefox. All of the currently available Greasemonkey scripts can be obtained from the web site <http://userscripts.org>.

The following information combined with the example Gizmos contained in the GMS Option Pack should give anyone with some javascript and/or MML experience a good starting point from which to produce their own Gizmos.

If preferred it is also possible to use jQuery to produce Gizmos. Microsoft have recently announced the inclusion of jQuery within Visual Studio .Net. Some differences will be apparent using jQuery and these are explained later in this chapter.

18.2 Gizmo Functionality

Naming Conventions

The Gizmo object name and the name of the underlying javascript script file are closely related. For instance if a file called `zoom.js` implements a plugin then the name of the Javascript object should be `gmsPlugin_zoom` and an object constructor with zero arguments should exist:

```
function gmsPlugin_zoom()
```

```
{  
    // Whatever the object needs.  
}
```

Display Name

A Gizmo may also have a unique display name. By default the plugin name is used e.g. in the above the display name is "zoom". Adding a line at the start of the JS file as shown below will change the name displayed to the user to "Magnifier".

```
// Name:Magnifier
```

Description

Each Gizmo can be given a short description which will be displayed to the user along with the Display Name when they select Gizmo's in the GMS WebMail interface.

```
// Description: Short description of Gizmo
```

External Scripts

A Gizmo may also require that additional scripts be pulled into the system. These may be included by adding a line at the start of the JS file like:

```
// ExternalScripts: http://download.skype.com/share/skypebuttons/js/  
skypeCheck.js
```

Due care should be taken when accessing external scripts to ensure that no security issues arise.

Administration Gizmos

Administration Gizmos are Gizmos which end users do not have any control over, i.e. they can not turn them on or off using the WebMail Gizmo configuration pages.

```
//Type: Value
```

Value may be one of Admin, User, or All.

Access Rights

Access to a Gizmo may be restricted to certain levels of users. Access is restricted by adding a line at the start of the JS file like:

```
// AccessRights: ALL
```

Valid access restrictions are: ALL, SYSTEM, DOMAIN, <wildcard email address> and multiple access rights may be included within the line as follows

```
// AccessRights: SYSTEM DOMAIN postmaster@*
```

Javascript File

Some Gizmos such as the Twitter and RSS Reader examples above may need specific configuration. This option enables loading of

such a configuration screen by adding a line at the start of the JS file like:

```
//ConfigurationRequired: Yes
```

The loading of a configuration screen is implemented via the ConfigLoad API.

HTTP Object

The following non-standard features have been added to the AJAX request object.

Feature	Details
timeout	timeout in seconds (default 60)
account	account for sites requiring HTTP Auth
password	password for sites requiring HTTP Auth
uniqueid	unique id, this will be passed back in the callbacks.
ontimeout	handler called on timeout

Installation

Copy the Gizmo into the <BaseDir>\mml\webmail\plugins directory. The plugin will be listed in the plugin configuration. Note that all Gizmos are available to all users, although individual users may not have sufficient access rights to use them.

18.3 Object Methods

The following object methods may be defined. If defined the objects should return either true or false to indicate if the Gizmo wishes to be notified of changes to that type.

```
handleemail()  
handleaddressbook()  
handlecalendar()  
handletasks()  
handlenotes()
```

The following object method must be defined and is called when an item is selected.

```
previewload()
```



The preview pane is still loaded in the background even if the display of the preview pane is disabled.

The following object methods are optional, and are called when the relevant item is loaded.

```
configload()  
treeload()  
menuload()  
panelload()  
windowload()
```

Finally there is a method which allows the Gizmo to perform a one time initialization. This is required as the Gizmo constructor is not able to call the inbuilt functions to obtain the Gizmo configuration information.

```
init()
```

18.4 Helper Methods

A number of methods are provided to help developers navigate around the GMS WebMail system in a simple abstract way regardless of underlying changes.

General APIs

General

<code>getType()</code>	Gets the type of the current preview. values are: EMAIL, ADDRESSBOOK, CALENDAR, NOTES, TASKS
<code>getContentWindow()</code>	Returns the content window. This is the right hand pane in which Gizmos can display their content.
<code>getViewerWindow()</code>	Gets a windows handle to the overall preview viewer window.
<code>getSummaryWindow()</code>	Gets a windows handle to the summary iframe in the viewer window
<code>getViewerWindowText()</code>	Gets the viewer window content in text format
<code>getSummaryWindowText()</code>	Gets the summary window content in text format
<code>getViewerWindowHTML()</code>	Gets the viewer window content in HTML format
<code>getSummaryWindowHTML()</code>	Gets the summary window content in HTML format
<code>getViewerWindowSelectedText()</code>	Gets the currently selected text in the viewer window
<code>getSummaryWindowSelectedText()</code>	Gets the currently selected text in the summary window
<code>getAccountName()</code>	Returns the logged on users email address
<code>getAttributeText(id)</code>	Search for an element with this id in the summary or viewer windows and return the text content
<code>getAttributeHTML(id)</code>	Search for an HTML element with this id in the summary or viewer windows and return the HTML content
<code>log(message)</code>	Log a message. This is logged to the server log and also (if Firebug if installed on Firefox) to the javascript console
<code>getConfigurationValue(key, defaultValue [, encrypt])</code>	Retrieve Gizmo specific configuration. Configuration values are persisted on the server and are available between sessions. Optionally deals with encrypted values.

Current Selection Values

Current Selection and Values

<code>getCurrentOption()</code>	Get the currently selected option on the tree
<code>getCurrentValue()</code>	Get the currently selected value in the grid or calendar
<code>getPreviousValue(startValue)</code>	Get the previous value in the grid starting at <code>startValue</code> . <code>startValue</code> is optional and the current selected value is unused if it is not provided. This is not available in the Calendar
<code>getNextValue(startValue)</code>	Get the next value in the grid starting at <code>startValue</code> . <code>startValue</code> is optional and the current selected value is unused if it is not provided. This is not available in the Calendar
<code>getAllValues()</code>	Get all values in the grid. This is not available in the Calendar
<code>getSelectedValues()</code>	Get the currently selected values in the grid or calendar
<code>getColumnValue(value, column)</code>	Return the content of a column for a value in the grid. Not available in the Calendar
<code>getLanguage()</code>	Return the configured language, i.e. en-us.
<code>removeAllValues()</code>	Remove all the values in the grid. Not available in the Calendar
<code>removeValues(values)</code>	Remove the given values from the grid or calendar. <code>values</code> is optional, if it is not provided then the current selection is used.
<code>setUpdatedValues(type, values)</code>	Inform GMS that you have modified the state of certain values. This allows for GMS to refresh internal caches
<code>selectValue(value)</code>	Select a value in the grid or calendar. This will cause it to be loaded in the preview pane
<code>setViewerWindowHTML(html)</code>	Sets the HTML in the viewer window
<code>setSummaryWindowHTML(HTML)</code>	Sets the contents of the summary window
<code>setViewerWindowHTML(HTML)</code>	Sets the contents of the viewer window, requires reset of <code>setSummaryWindpw</code> value
<code>setConfigurationValue(key, value [, encrypt])</code>	Set Gizmo specific configuration. Configuration values are persisted on the server and are available between sessions. Optionally encrypts the value before writing to the registry.

Current Selection and Values

<code>getCurrentOption()</code>	Get the currently selected option on the tree
<code>getCurrentValue()</code>	Get the currently selected value in the grid or calendar
<code>getPreviousValue(startValue)</code>	Get the previous value in the grid starting at startValue. startValue is optional and the current selected value is unused if it is not provided. This is not available in the Calendar
<code>refresh</code>	Refresh the grid or calendar content
<code>reload</code>	Reloads the Gizmo. Currently causes a full browser refresh. Should be called when changes, such as to the configuration, require a complete reload of the Gizmo.

Menu APIs**Menus**

<code>registerMenuCommand(text, shortcut, tip, image, action, enabled, callbackFunction [, menu [, win]])</code>	Add a custom button to the GMS menu. Note that buttons must be added each time a preview loads
<code>addMenuDivider([menu [, win]])</code>	Add a divider to the GMS menu
<code>clickMenuCommand(action [, menu [, win]])</code>	Click handler for a menu button
<code>existMenuCommand(action [, menu [, win]])</code>	Checks whether a button exists
<code>removeMenuCommand(action [, menu [, win]])</code>	Remove a command from the GMS menu
<code>hideMenuCommand(action [, menu [, win]])</code>	Hide a command in the GMS menu. Note the button is still active and can be accessed via keyboard shortcuts
<code>showMenuCommand(action [, menu [, win]])</code>	Re-show a hidden button
<code>enableMenuCommand(action [, menu [, win]])</code>	Enable a menu button
<code>disableMenuCommand(action [, menu [, win]])</code>	Disable a menu button

menu defaults to "context". It may be set to one of "command", "admin" or "context". It controls which menu a button is added to.

win defaults to the main GMS window. The window returned by `openCustomWindow` or `getCurrentWindow` should be used for windows launched by GMS.

Panel APIs

Panels

<code>getCurrentPanel()</code>	Returns the currently selected panel
<code>monitorPanelButton(panel, interested)</code>	Register to be informed of panel button switches. Any interested Gizmos will receive a <code>treeLoad</code> callback. If any panel button is monitored then the "folders" panel button is also monitored.
<code>addPanelButton(text, shortcut, image, panel)</code>	Adds a panel button. The button is automatically monitored.
<code>clickPanelButton(panel)</code>	Simulates a click on the panel button
<code>existPanelButton(panel)</code>	Checks that a panel button exists
<code>removePanelButton(panel)</code>	Removes a panel button
<code>setPanelDefaultTreeNode(panel, node)</code>	Sets the default tree node for a panel

Tree APIs

Trees

<code>addTreeNodeNext(text, image, action, panel, callbackFunction [, next])</code>	Adds a tree node immediately after the "next" tree node at the same level. Only actioned if the panel is the currently selected panel or the "folders" panel is selected
<code>addTreeNodeChild(text, image, action, panel, callbackFunction [, parent])</code>	Adds a tree node as a child of the parent node. Only actioned if the panel is the currently selected panel or the "folders" panel is selected
<code>existTreeNode(node)</code>	Checks if a tree node exists
<code>closeTreeNode(node)</code>	Closes a tree node
<code>removeTreeNode(node)</code>	Removes a tree node
<code>getCurrentOption</code>	No change
<code>getSelectedTreeNode</code>	Returns the selected node
<code>setSelectedTreeNode()</code>	Sets the selected node
<code>sortTreeNode(node)</code>	Sort the sub tree down from a node

The default value for next is "GMS Webmail.Notes" or "GMS WebMail.Mail" or "GMS Webmail". The first existing node is used.

The default value for parent is "GMS Webmail".

Content APIs

Contents

<code>loadCustomContent(title, html, customContentCallback, gmsStyle)</code>	Loads custom content in the content window. <code>customContentCallback</code> will be called once the window has loaded.
<code>insertCustomContent(id, html[, win])</code>	Inserts the html at element with id "id"
<code>insertCustomButton(id, text, action, callbackFunction [, win])</code>	Inserts a standard HTML button at element with id "id". When the button is clicked the <code>callbackFunction</code> function is called
<code>setFieldValue(id, value [, win])</code>	Sets the contents of an input field
<code>getFieldValue(id [, win])</code>	Gets the contents of an input field
<code>loadExternalContent(url)</code>	Loads the content specified by url into the content window

If `gmsStyle` is true then standard GMS dialogs in the current theme are used to wrap the html.

`win` defaults to the content window.

Window APIs

Windows

<code>openCustomWindow(windowCallback, callbackParameters, height, width [, target [, options]])</code>	Launches a custom window. The callback is called with the <code>callbackParameters</code> to load content into the window.
<code>openExternalWindow(url, height, width[, target, options])</code>	Opens the external url in a window
<code>getCurrentWindow</code>	Returns the current window

The options are passed to `window.open`.

`target` defaults to "_blank"

Ajax Requests

AJAX Requests

<code>xmlHttpRequest(details)</code>	Invoke an XML HTTP request. Requests may be made to any server. See also HTTP Object above.
--------------------------------------	---

The following may be setup in the object. Please see the Greasemonkey documentation for more information.

```
url
method
headers
data
status
statusText
responseHeaders
responseText
```

onreadystatechange
onload
onerror



Note: details is an object containing information about the request and response. It is compatible with the Greasemonkey GM_xmlHttpRequest details object.

Greasemonkey helpers

Greasemonkey helpers

GM_log	Wrapper to the log method
GM_setValue	Wrapper to the setConfigurationValue method
GM_getValue	Wrapper to the getConfigurationValue method
GM_registerMenuCommand	Wrapper to the registerMenuCommand method
GM_xmlHttpRequest	Wrapper to the xmlhttpRequest method

jQuery

jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.

By default GMS ships with jQuery 1.2.6. If you wish to use a later version of the jQuery library then you simply need to copy it to the <basedir>MML\webmail\common directory and a variable at either the System, Domain or User level. Normally you would just set it at the System level so everyone uses the new library. So to use jQuery 1.2.8 you would copy the file jQuery-1.2.8.js into place, then set WebMailjQueryVersion=1.2.8

The jQuery library and is based around the concept of "selectors". To access elements you would call something like \$("div") and it will return all the "div" elements in the current document. From a Gizmo perspective this can lead to issues as the "current document" is not the document you normally want to use. For instance, the example jQuery **Compose** Gizmo needs to act on the current compose window. To solve this we use "context" which causes the selector to work in another window or in just part of a document.

To use "context" requires getting the current window handle and then using this for further operations similar to the example shown.

```
var w = this.getCurrentWindow();
// Standard Gizmo code

$("#to", w).hide();
// Hide fields called "to" in the window "w".
```

This approach works with all core jQuery functionality. Using core functionality should be straightforward.

The use of jQuery plugins can be more problematic as you need to ensure that you both set the context correctly, and also maintain the context for later use.

There are two example jQuery based Gizmos included in the GMS Accessory Pack.

Compose

Displays/Hides features within the Compose window, such as the BCC and Attachments fields. There are two examples of this Gizmo, one using GreaseMonkey code and the other using jQuery code.

- How to extend GMS WebMail functionality.
- How to add/remove functionality from a window.

Mouse Gestures

This is an "invisible" Gizmo which allows navigation through message lists using mouse gestures. It provides an example of how to use a jQuery plugin, in this instance jGesture. You need to right click with the mouse to enable the gestures which work as follows

- **Up** - moves up a message
- **Down** - moves down a message
- **Right** - replies to all recipients of a message, or opens tasks, notes, etc.
- **Left** - deletes a message
- **Up + Right** - replies to a message
- **Up + Left** - forwards a message
- **Down + Right** - prints a message
- **Down + Left** - opens a message
- **Rotate Clockwise** - composes a new message
- **Rotate Anti-Clockwise** - refreshes the message list

18.5 Sharing Gizmos

If you design any Gizmos you particularly like, and would like to share those Gizmos with others please feel free to do so. Alternatively please contact us and we will make them available to other GMS users via our web site.

19 Themes

Themes allow the administrator to completely change the look and feel of the GMS WebMail Professional interface, i.e. it is now fully "skinable". There are two distinct types of theme that can be developed, a full theme where all of the colours and images are changed, or a partial theme which uses the existing images but changes the colours to suit a corporate scheme.

Individual users can select which theme they wish to use from the Preferences > Appearance option within GMS WebMail. The themes they can choose from are limited to those enabled by system administrators. GMS WebMail ships with two themes enabled by default, a blue theme and a silver theme. These are also available within the GMS Option Pack to provide a starting point for any themes you may wish to develop yourself. Examples of each are shown below.

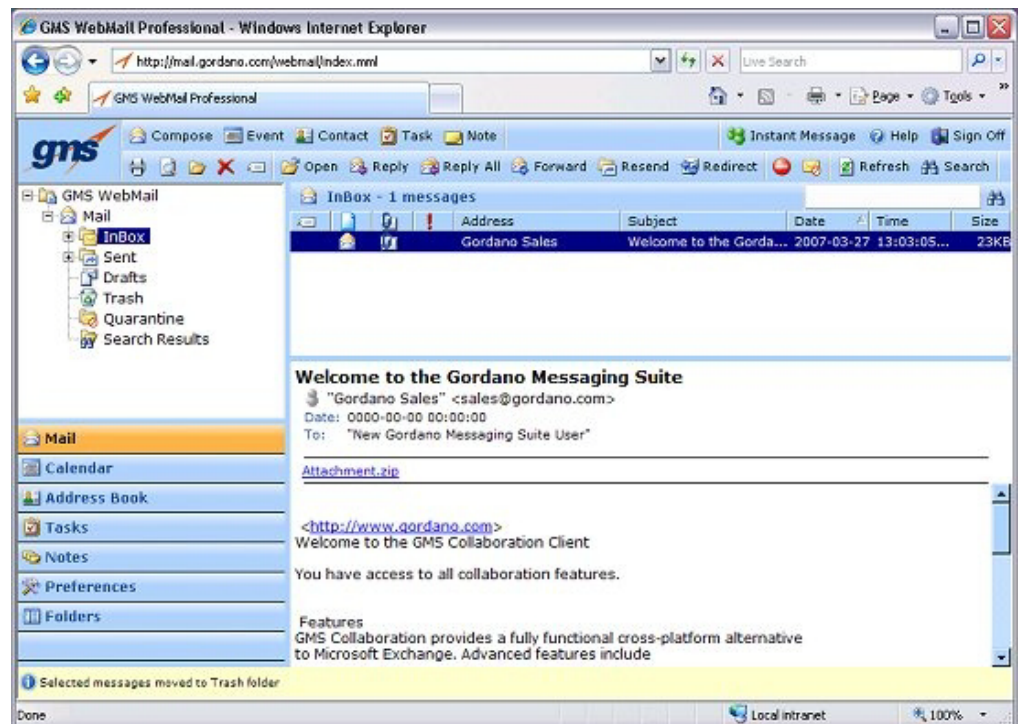


FIGURE 1. Blue theme

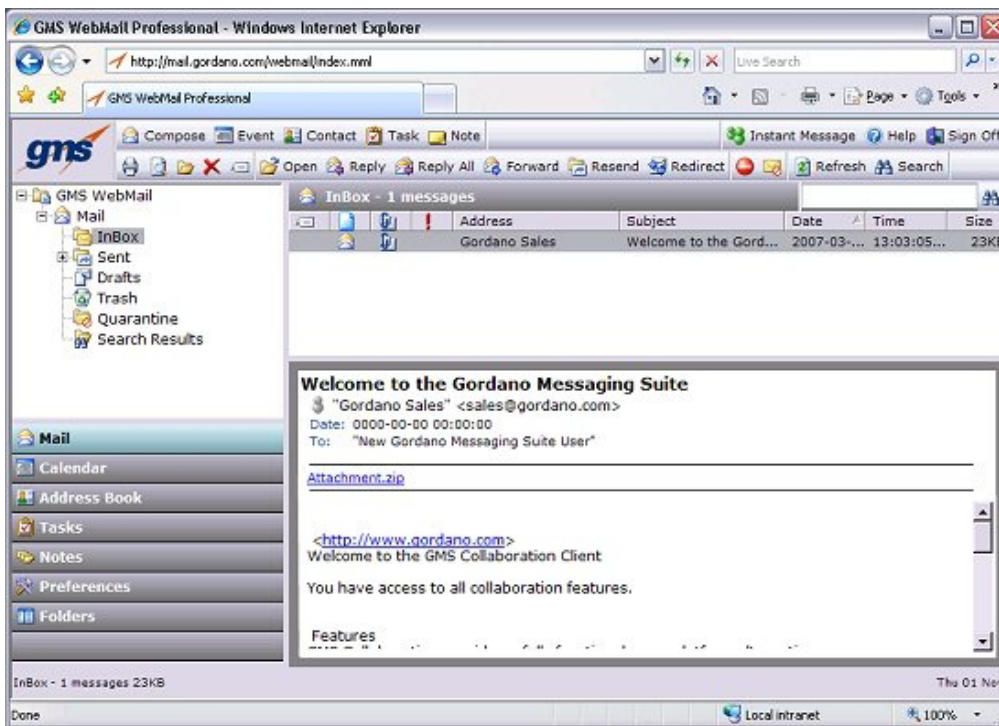


FIGURE 2. Silver Theme

19.1 What is a Theme

A theme consists of three constituent parts as follows:

1. **Options file**
Each theme is controlled by a file called options.txt. This file provides the theme name, some image size information, image type information (e.g. gif or png) and where the images are located.
2. **Cascading Style Sheets**
A complete set of CSS files, which control the layout and color scheme.
3. **Images**
An "i" directory containing all your images – both enabled and disabled forms of each image are required.



Note: By specifying the image location it is possible to re-use the images from an existing theme within your own theme.

Themes can be one of two types - either full or partial;

A full theme includes all CSS style definitions, and all images - this means you need your own images to replace every single image in the interface. We acknowledge that very few people would ever want or need to do this, so you can also create a partial theme.

A partial theme re-uses the standard image set, but allows you to control colours, fonts, backgrounds, etc. There is an example of a

partial theme in the GMS Option Pack - if you unpack this you will find it in the mml\themes folder within the zip.

19.2 Adding Custom Themes

Adding a custom theme to GMS is straightforward and consists of the following steps.

1. Create a "themes" directory in <BaseDir>\mml
2. Create a directory for your theme (call it anything you wish but "admin" and "webmail" are reserved)
3. Copy your theme files into the directory, they will be immediately available for use

To help you get started with themes we have provided to examples in the GMS Option Pack, simply copy these files from the Option Pack zip file into a directory of your own choosing under the mml\themes directory. Modifying any of the supplied themes is very simple - if you have any experience of writing HTML using Cascading Style Sheets (CSS) you will find updating a theme very easy indeed. Even if you don't have experience, you will easily see how to change the colors associated with a theme.

In the mml\themes directory you will find the sub-directory folder created above which will contain your custom theme. Open this directory, and edit the file "options.txt" to rename the theme to something appropriate - the theme name is the first item in the file - the only other items in this file you might want to edit are some of the base colours.

Also within this directory you will find a number of CSS files. There are separate CSS files to control each element of the interface. So for example, the folder tree is controlled by "tree.css" - in addition where browsers behave differently, you might also have an "msietree.css" and a "firefoxtree.css" - these contain only the items that are specific to that browser.

A simple way to change all the colours is to open all the files in a text editor at once (we find Textpad particularly useful) - then do a find and replace in all the open files replacing a particular colour with the colour of your choice,

You can go further than this by changing some or all of the background images, but you can get a nice effect simply by changing the colours.



We would recommend using the Firefox browser when amending the CSS files, in conjunction with the plugin called "Firebug". This allows you to "inspect" an item by hovering over it with the mouse, so you can view the code - so for example if you hover over the compose button, you see:

```
<td id="leftminimenuButton1" class="menubuttonout" title="Compose a new message" style="-moz-user-select: none;">
```

which tells you that you can use the class "menubuttonout" to control the look of this button. Menubuttonout is found in the menu.css file.

19.3 Sharing Themes

If you design any themes you particularly like, and would like to share those themes with others please feel free to do so. Alternatively please contact us and we will make them available to other GMS users via our web site.

20 Robots

Incoming mail messages can be used to run NT console application programs, sometimes called robots or executables. This is extremely useful if you want to trigger a program remotely by sending e-mail to the appropriate address. When a message is received the specified program starts.

Two distinct types of robot are available for Gordano products:

- Domain robot — started by an e-mail message arriving at any address in a domain.
- User robot — only started for e-mail messages arriving at the particular user account. They have no effect on e-mail arriving for any other user in that or other domains.

The robot:

- Can duplicate the result up to 100 times.
- Can use all e-mail addresses in the To:, CC: and Bcc: clauses.
- Removes the BCC: clause before the message is sent.
- May or may not accept/deliver the message.
- Can be written in C, C++, Java, Perl, VB, etc.

20.1 Domain Robots

A typical use of a domain robot might be to take all mail arriving for any user at a particular domain and rewrite it so that it is forwarded on to the same user but at a completely different domain.

20.2 User Robots

A typical use of a user robot might be to archive the postings to a list so that they can be displayed on a Web page, allowing interested parties can search or browse through past postings to the list.

20.3 Accepting the E-mail Message

To make a robot accept e-mail, use the stdin/stdout parameters:

- Use stdin = 1 means send to robot.
- Use stdout = 1 means expect message.

These parameters are described in the Registry section.

20.4 Sending an E-mail Message

To, CC, BCC: clauses can be used. Up to 100 destinations can be requested. The BCC: clause will be removed by the Gordano server.

20.5 Example Code

The example program shown below, Dumpmail.exe, shows the basic technique for collecting the mail message from standard input while details about the message are read from a set of environment variables.

The end of the message can be found by a looking for CRLF.CRLF, or checking when CONTENT_LENGTH characters have been read from stdin.

The parameters which can be used are as follows:

Variable	Description
CONTENT_LENGTH	The number of bytes in the mail message. Note that the length reported has been adjusted to overcome some inconsistencies within the MS libraries. Therefore, we recommend that any robot you use looks for a single line with a single period on it to indicate the end of a message.
FROM	The e-mail address extracted from the MAIL clause of the message envelope. This is the result after Gordano's 'name unmangling' routines have tried to fix whatever was sent by the remote mail server.
HOSTNAME_LOCAL	The name of the host where this message was delivered. This is particularly useful if the robot has to distinguish between e-mail arriving on a multi-homed mail server.
HOSTNAME_REMOTE	The name of the remote machine which sent the mail message. If reverse lookup has been switched off, this will be the IP address of the remote machine in square brackets
IPADDR_LOCAL	The IP address of the server the mail has been delivered.
IPADDR_REMOTE	The IP address of the remote mail server which sent the message in dotted decimal format.
LINES	Number of lines in the message.
MAIL_SERVER	Name of the mail server. For NTMail this is 'NTMail'.
MAIL_VERSION	Version number of the mail server in the format 6.02.02
SUBJECT	The subject of the message.
TO	The destination of the e-mail message as defined by its RCPT. If the robot is a domain robot or is used for one or more mail drops, this variable may be used to work out where this message was sent.

This listing shows the Dumpmail.exe:

```

/* Program:      DUMPMAIL                      */
/* Author:       Brian Dorricott                */
/* Date:         11th January 1996              */
/* Copyright:(C)opyright 1996-1999, Gordano Ltd */

#include <windows.h>    /* needed for all Windows apps */
#include <winsock.h>

#include <stdio.h>      /* for sprintf                */
#include <string.h>     /* for strlen                 */
#include <stdarg.h>     /* For va_start etc          */
#include <stdlib.h>

#define CRLF           "\15\12"
#define CR             '\15'
#define LF             '\12'
#define MAX_LINE       4096

main(int argc, char *argv)
{
    FILE *File;
    char *buf;
    BOOL done = FALSE;
    char line[ MAX_LINE ];

    // Lets read the input and write it to a file
    // along with all the environmental strings

    File = fopen( "testfile.txt", "w" );
    if ( File != NULL )
    {
        // Write environemntal variables to file
        //
        buf = GetEnvironmentStrings();
        while (*buf != '\0')
        {
            fprintf(File, "%s%s", buf, CRLF);
            buf += strlen(buf)+1;
        }
        fprintf(File, "%s%s", CRLF, CRLF);

        // Now read the message from stdin until reach
        // line which is single dot (end of message).
        //
        while ( !done )
        {
            fgets( line, MAX_LINE, stdin );
            while (*(line + strlen(line) - 1) == '\r' ||
                *(line + strlen(line) - 1) == '\n' )
                *(line + strlen(line) - 1) = '\0';
            if ( strcmp( line, "." ) == 0 )
                done = TRUE;
            if ( File != NULL && !done )
                fprintf( File, "%s\n", line );
        }
        fclose( File );
    }

    //
    // Now create the message to send back.

    printf ("From:Robot <Robot@company.com>%s", CRLF);

```

```
printf( "To:\n" %s\n" <%s>%s", getenv( "FROM"),
        getenv( "FROM" ), CRLF );
printf( "Subject: Re: %s%s",getenv
        ( "SUBJECT" ),CRLF );
printf( " %s%s", CRLF, CRLF );
printf( " Environmental variables (interesting)
        :%s%s%s", CRLF, CRLF, CRLF );
buf = GetEnvironmentStrings();
while (*buf != '\0')
{
    printf("%s%s", buf, CRLF);
    buf += strlen(buf)+1;
}
printf( " %s%s", CRLF, CRLF );
printf( " end of new message.%s", CRLF );

//
// A single dot = the end of the message.
//
printf( " %s.%s", CRLF, CRLF );

return 0;
}
```

21 DLLs

This section describes DLLs, which are also referred to as executables.

The available DLL types are as follows (note that the first three types all use the same three files):

- Account DLLs — use maildll.c, maildll.h and mailfc.c.
- Domain DLLs — use maildll.c, maildll.h and mailfc.c.
- Transmission DLLs (Incoming, Outgoing & Delivery DLLs) — these all use maildll.c, maildll.h and mailfc.c.
- SMTP Clause DLL — uses smtpdll.c and smtpdll.h.
- User authentication DLL — uses userdll.c.
- Message parsing and log DLLs.
- External Folder Manager DLL

These are described in turn below.

The following DLLs are **not** described in this version of the manual:

- MML Additional Command DLL.
- MUM DLL.

21.1 Why use a DLL?

A DLL is more efficient than a robot.

21.2 Available DLLs

Transmission DLLs

There are three types of transmission DLL:

- Incoming DLLs — affect all e-mail entering the system and act on each RCPT clause in turn before the message body is processed.
- Outgoing DLLs — act on all e-mail leaving the system, just before the message is delivered to the Out directory. This would, for instance, allow you to stop mail being sent to particular addresses.
- Delivery DLLs — act on all e-mail entering the system after the message has been processed and just prior to the mail being delivered to the intended account. At this stage it is too late to operate on the contents of the message itself, although you can still redirect the message to another account.

maildll.c

```
/*
 * (C)opyright 1995-2015 Gordano Ltd.
 * All rights reserved.
 * No copies of this source may be made without express
 * written permission of Gordano Ltd, Unit 1, Kenn Road, Kenn
 * Clevedon, North Somerset, BS21 6UW, UK
 */

#pragma warning ( disable : 4001 4115 4201 4209 4214 4514 )

#include <windows.h>
#include <winbase.h>
#include <stdio.h>

#pragma warning ( disable : 4001 4115 4201 4209 4214 4514 )

#include "mailifc.h"

#include "maildll.h"

#define MAX_LINE 1024

#define S_SUCCESS 0
#define CONF_LOG_STATS 0x000010

#define VER_MAJOR 4
#define VER_MINOR 20
#define VER_TRIVIAL 1

#define MailDLLVersion (((VER_MAJOR*100)+VER_MINOR)*10000+VER_TRIVIAL)

const char *Copyright = "Copyright Gordano Limited";

static init = FALSE;
static char BaseDir[ MAX_FILENAME ];
static char Name[ MAX_FILENAME ];
static char Description[ MAX_FILENAME ];

void Initialise( HANDLE hMod )
{
    if ( !init )
    {
        HKEY hKey;
        DWORD size;
        wchar_t Line[ MAX_LINE ];
        char *Buffer;

        // sample processing to get info from resources and registry

        GetModuleFileName( hMod, Name, sizeof( Name ) );
        if ( GetFileVersionInfo( Name, 0, sizeof( Line ), (void *)Line ))
        {
            VerQueryValue( Line, TEXT( "\\StringFileInfo\\080904b0\\FileDescription" ), (void **)&Buffer, &size);
            strcpy( Description, Buffer );
        }
        if ( RegOpenKey(HKEY_LOCAL_MACHINE, "Software\\InternetShopper\\Mail\\Parameters", &hKey) == ERROR_SUCCESS)
        {
            int linelen = sizeof( BaseDir );
            DWORD type_of_entry = REG_SZ;
```



```

        *BaseDir = '\0';
        RegQueryValueEx(hKey, "BaseDir", 0, &type_of_entry, (unsigned
            char *)BaseDir, &linelen );
        if ( *BaseDir != '\0' )
        {
            int length = strlen( BaseDir );

            if ( *(BaseDir + length - 1) != '\\' )
                strcat( BaseDir, "\\" );
        }

        RegCloseKey( hKey );
    }

    init = TRUE;
}

BOOL __stdcall DllMain( HMODULE hmod, ULONG Reason, LPVOID Reserved )
{
    if ( Reason == DLL_PROCESS_ATTACH )
    {
        Initialise( hmod );
    }
    else if ( Reason == DLL_THREAD_ATTACH )
    {
    }
    else if ( Reason == DLL_THREAD_DETACH )
    {
    }
    else if ( Reason == DLL_PROCESS_DETACH )
    {
    }

    return( TRUE );
}

int FAR PASCAL GetExtensionVersion( MAIL_VERSION *data )
{
    int status = S_SUCCESS;
    char Line[ MAX_LINE ];

    Initialise( NULL );

    // log call to SMTP log with id 3997
    sprintf( Line, "GetExtensionVersion called in %s - %s (%d)", Name,
        Description, status );
    WriteToLog( data->context, CONF_LOG_STATS, 3997, Line );

    // initialise dll id parameters
    data->version = MailDLLVersion;
    strcpy( data->description, Description );
    strcpy( data->copyright, Copyright );

    return( status );
}

MAIL_RETURN FAR PASCAL MailExtension( MAIL_CONTEXT *data )
{
    MAIL_RETURN status = MAIL_DLL_SUCCESS;
    char Line[ MAX_LINE ];
    char user[ MAX_EMAIL_ADDRESS ];

```

```
    char domain[ MAX_EMAIL_ADDRESS ];
    char Variable[ MAX_LINE ];
    char Value[ MAX_LINE ];
    char *sep;

    // crudely interpret address
    *domain = '\0';
    strcpy( user, data->destination );
    sep = strchr( user, '@' );
    if ( sep )
    {
        *sep = '\0';
        strcpy( domain, sep + 1 );
    }

    // do something with the mail message - just log in this case

    sprintf( Line, "MailExtension called in %s (%d)", Name, status );
    WriteToLog( data->context, CONF_LOG_STATS, 3998, Line );

    // get global variable
    GetVariable(data->context, "\\Administrator", Value, sizeof(
        Value));

    // get user variable
    sprintf( Variable, "%s\\%s\\DateCreated", domain, user );
    GetVariable( data->context, Variable, Value, sizeof( Value ) );

    // get domain variable
    sprintf( Variable, "%s\\PostMaster", domain );
    GetVariable( data->context, Variable, Value, sizeof( Value ) );

    // return message to sender
    DeliverMessage(data->context,data->messageFile,Value,data->source,
        data->depth );

    return( status );
}

int FAR PASCAL ConfigureExtension( MAIL_CONFIG *data )
{
    int status = S_SUCCESS;
    char Line[ MAX_LINE ];

    // log call to SMTP log with id 3999
    sprintf(Line, "ConfigureExtension called in %s (%d)", Name, status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}
```

maildll.h

```

/*
 * (C)opyright 1995-2015 Gordano Ltd.
 * All rights reserved.
 * No copies of this source may be made without express
 * written permission of Gordano Ltd, Unit 1, Kenn Road, Kenn
 * Clevedon, North Somerset, BS21 6UW, UK
 */

#ifdef __cplusplus
extern "C" {
#endif

#define MAX_MAIL_DESCRIPTION 1024

typedef void *MAIL_HANDLE;

typedef enum
{
    MAIL_TYPE_INCOMING = 1,
    MAIL_TYPE_OUTGOING = 2,
    MAIL_TYPE_DELIVERY = 4,
    MAIL_TYPE_ROBOT = 8,
    MAIL_TYPE_DOMAIN = 16
} MAIL_TYPE;

typedef struct mail_version_struct
{
    DWORD version;                // dll version id - see
                                // MailDLLVersion
    MAIL_TYPE dllType;            // dll type
    MAIL_HANDLE context;
    char description[ MAX_MAIL_DESCRIPTION ]; // dll description text
    char copyright[ MAX_MAIL_DESCRIPTION ]; // dll copyright text
} MAIL_VERSION;

typedef enum
{
    MAIL_DLL_SUCCESS= 0,
    MAIL_DLL_PROCESSED= 1,
    MAIL_DLL_REJECT= 2
} MAIL_RETURN;

typedef struct mail_context_struct
{
    DWORD size;
    DWORD version;
    MAIL_HANDLE context;
    int depth;
    char messageFile[ MAX_FILENAME ];
    char source[ MAX_EMAIL_ADDRESS ];
    char destination[ MAX_EMAIL_ADDRESS ];
} MAIL_CONTEXT;

typedef enum
{
    MAIL_CONFIG_ACTION_NEW_USER= 1,    // unused
    MAIL_CONFIG_ACTION_EDIT_USER= 2,   // unused
    MAIL_CONFIG_ACTION_DELETE_USER= 4, // unused
    MAIL_CONFIG_ACTION_INSTALL= 8,
    MAIL_CONFIG_ACTION_UNINSTALL = 16
} MAIL_CONFIG_ACTION;

```

```
typedef struct mail_config_struct
{
    DWORD size;
    DWORD version;
    MAIL_CONFIG_ACTION action;
    MAIL_HANDLE context;
    char user[ MAX_EMAIL_ADDRESS ];
    char domain[ MAX_EMAIL_ADDRESS ];
} MAIL_CONFIG;

#ifdef __cplusplus
}
#endif
```

mailfc.h

```

/*
 * (C)opyright 1995-2006 Gordano Ltd.
 * All rights reserved.
 * No copies of this source may be made without express
 * written permission of Gordano Ltd, 18 Kenn Road,
 * Clevedon, North Somerset, BS21 6EL, UK
 */

#ifdef __cplusplus
extern "C" {
#endif

#define MAX_EMAIL_ADDRESS 256
#define MAX_FILENAME 256

typedef struct _mail_dll_ifc_struct
{
    int (*deliverMessage)( const char *messageFile, const char *mailFrom, const char *rcptTo, int
        depth );
    int (*writeToLog)( int status, int logID, const char *message );
    int (Reserved0)( const char *messageFileIn, const char *template File, const char
        *messageFileOut );
    int (Reserved1)( const char *variableName, const char *value );
    int (*getVariable)( const char *variableName, char *buffer, size_t size );
    int (Reserved2)( const char *variableName, const char *buffer );
    int (Reserved3)( void *buffer, size_t size );
    int (Reserved4)( void *buffer, size_t size );
    int (Reserved5)( const char *mesLog, const char *messageFile, const char *mailFrom, const char
        *rcptTo, const char *from, const char *to);
    int (Reserved6)( const char *address, const char *password);
    int (Reserved7)( const char *messageFile, const char *mailFrom, const char *rcptTo, const char
        *queue );
    int (Reserved8)( const char *address, const char *filename, char *fullFilename, size_t size );
    int (Reserved9)( int type, const char *message );
} MAIL_IFC;

#define DeliverMessage(a,b,c,d,e)((MAIL_IFC *)TlsGetValue((DWORD)a))-
    >deliverMessage((b),(c),(d),(e))
#define WriteToLog(a,b,c,d)((MAIL_IFC *)TlsGetValue((DWORD)a))->write
    ToLog((b),(c),(d))
#define GetVariable(a,b,c,d)((MAIL_IFC *)TlsGetValue((DWORD)a))->get
    Variable((b),(c),(d))

#ifdef __cplusplus
}
#endif

```

SMTP DLLs

SMTP DLLs act on all e-mail entering the system during the course of a normal mail transaction. The DLLs can act on any stage of the SMTP protocol. There are two files, smtpdll.h and smtpdll.c.

smtpdll.h

```
/*
 * Copyright :: 1995-2015 (C)opyright Gordano Ltd
 * All rights reserved. No copies of this source may be
 * made without express written permission of:
 * written permission of Gordano Ltd, Unit 1, Kenn Road, Kenn
 * Clevedon, North Somerset, BS21 6UW, UK
 */

#ifdef __cplusplus
extern "C" {
#endif

#define MAX_SMTP_DESCRIPTION 1024
#define MAX_ESMTP_LEN 1024
#define MAX_PROTOCOL_MES_LEN 80

typedef void *SMTP_HANDLE;

typedef struct smtp_version_struct
{
    DWORD version;
    SMTP_HANDLE context;
    char description[ MAX_SMTP_DESCRIPTION ];
    char copyright[ MAX_SMTP_DESCRIPTION ];
} SMTP_VERSION;

typedef enum
{
    SMTP_DLL_SUCCESS= 0,
    SMTP_DLL_PROCESSED= 1,
    SMTP_DLL_REJECT = 2
} SMTP_RETURN;

typedef struct smtp_info_struct
{
    DWORD size;
    DWORD version;
    SMTP_HANDLE context;
    int depth;
    int ESMTPError;
    struct in_addr ClientIPAddress;
    char Client[ MAX_EMAIL_ADDRESS ];
    struct in_addr ServerIPAddress;
    char Server[ MAX_EMAIL_ADDRESS ];
    char ESMTPOptions[ MAX_ESMTP_LEN ];
    char Filename[ MAX_FILENAME ];
    char From[ MAX_EMAIL_ADDRESS ];
    char Rcpt[ MAX_EMAIL_ADDRESS ];
    char AllRcpts[ MAX_STRING ];
    char Response[ MAX_PROTOCOL_MES_LEN ];
    char Mailbox[ MAX_MAILBOX_NAME ];
    char *Msg;
} SMTP_INFO;

typedef enum
```

```
{
    SMTP_CONFIG_ACTION_NEW_USER= 1,    // unused
    SMTP_CONFIG_ACTION_EDIT_USER= 2,    // unused
    SMTP_CONFIG_ACTION_DELETE_USER= 4,  // unused
    SMTP_CONFIG_ACTION_INSTALL= 8,
    SMTP_CONFIG_ACTION_UNINSTALL = 16
} SMTP_CONFIG_ACTION;

typedef struct smtp_config_struct
{
    DWORD size;
    DWORD version;
    SMTP_CONFIG_ACTION action;
    SMTP_HANDLE context;
    char user[ MAX_EMAIL_ADDRESS ];
    char domain[ MAX_EMAIL_ADDRESS ];
} SMTP_CONFIG;

#ifdef __cplusplus
}
#endif
```

smtpdll.c

```
/*
 * (C)copyright 1995-2015 Gordano Ltd. All rights reserved.
 * No copies of this source may be made without express
 * written permission of Gordano Ltd, Unit 1, Kenn Road, Kenn
 * Clevedon, North Somerset, BS21 6UW, UK
 */

#pragma warning ( disable : 4001 4115 4201 4209 4214 4514 )

#include <windows.h>
#include <winbase.h>

#pragma warning ( disable : 4001 4115 4201 4209 4214 4514 )

#include "mailfc.h"
#include "smtpdll.h"

#define MAX_LINE 1024
#define S_SUCCESS 0
#define CONF_LOG_STATS 0x000010

const char *Copyright = "Copyright Gordano Limited";

#define VER_MAJOR 4
#define VER_MINOR 20
#define VER_TRIVIAL 1

#define SMTPDLLVersion (((VER_MAJOR*100)+VER_MINOR)*10000+VER_TRIVIAL)

static init = FALSE;
static char BaseDir[ MAX_FILENAME ];
static char Name[ MAX_FILENAME ];
static char Description[ MAX_FILENAME ];

void Initialise( HMODULE hMod )
{
    if ( !init )
    {
        HKEY hKey;
        wchar_t Line[ MAX_LINE ];
        unsigned int size = sizeof( Line );
        char *Buffer;
        GetModuleFileName( hMod, Name, sizeof( Name ) );
        if(GetFileVersionInfo(Name,0,sizeof(Line),(void *)Line))
        {
            VerQueryValue(Line, TEXT("\\StringFileInfo\\
080904b0\\FileDescription"),(void **)&Buffer,&size);
            strcpy( Description, Buffer );
        }
        else
        {
            size = GetLastError();
        }

        *BaseDir = '\\0';
        if ( RegCreateKey( HKEY_LOCAL_MACHINE, "Software\\Inter
netShopper\\Mail\\Parameters", &hKey) == ERROR_SUCCESS )
        {
            DWORD linelen = sizeof( BaseDir );
            DWORD type_of_entry = REG_SZ;
```



```

        RegQueryValueEx( hKey, "BaseDir", 0, &type_of_entry,
            (unsigned char *)BaseDir, &linelen );
        if ( *(BaseDir + strlen( BaseDir ) - 1) != '\\' )
            strcat( BaseDir, "\\" );

        RegCloseKey( hKey );
    }

    init = TRUE;
}

}

BOOL __stdcall DllMain(HMODULE hmod,ULONG Reason,LPVOID Reserved)
{
    if ( Reason == DLL_PROCESS_ATTACH )
    {
        Initialise( hmod );
    }
    else if ( Reason == DLL_THREAD_ATTACH )
    {
    }
    else if ( Reason == DLL_THREAD_DETACH )
    {
    }
    else if ( Reason == DLL_PROCESS_DETACH )
    {
    }

    return( TRUE );
}

int FAR PASCAL GetSMTPExtensionVersion( SMTP_VERSION *data )
{
    int status = S_SUCCESS;
    char Line[ MAX_LINE ];

    Initialise( NULL );

    wsprintf( Line, "GetSMTPExtensionVersion called in %s - %s (%d)",
        Name, Description, status );
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    data->version = SMTPDLLVersion;
    strcpy( data->description, Description );
    strcpy( data->copyright, Copyright );

    return( status );
}

int FAR PASCAL ConfigureSMTPExtension( SMTP_CONFIG *data )
{
    int status = S_SUCCESS;
    char Line[ MAX_LINE ];

    Initialise( NULL );

    wsprintf( Line, "ConfigureSMTPExtension called in %s (%d)",Name,
        status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}

```

```

// called on initial connection prior to returning logon banner
SMTP_RETURN FAR PASCAL SMTPConnection( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];

    Initialise( NULL );

    wsprintf( Line, " %d SMTPConnection called in %s (%d)", id, Name,
              status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}

// example showing creation of private ESMTP extension
SMTP_RETURN FAR PASCAL SMTP EHlo( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];

    wsprintf( Line, " %d SMTP EHlo called in %s (%d)", id, Name, status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    if ( *(data->ESMTPOptions) != '\0' )
        strcat( data->ESMTPOptions, ",X-NTMAILEXTENSION" );

    return( status );
}

SMTP_RETURN FAR PASCAL SMTP Helo( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];

    wsprintf( Line, " %d SMTP Helo called in %s (%d)", id, Name, status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}

// example showing simple processing of data in MAIL command
SMTP_RETURN FAR PASCAL SMTP Mail( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];
    char user[ MAX_LINE ];
    char domain[ MAX_LINE ];
    char Variable[ MAX_LINE ];
    char VariableName[ MAX_LINE ];
    char *sep;

    wsprintf( Line, " %d SMTP Mail called in %s (%d)", id, Name, status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    strcpy( Line, data->From );
    sep = strchr( Line, '@' );
    if ( sep )
    {
        *sep = '\0';
        strcpy( domain, sep + 1 );
        sep = domain;
        while ( *sep && ( *sep == '.' || isalnum( *sep ) ) )

```

```

        sep++;
        *sep = '\0';

        sep = Line + strlen( Line ) - 1;
        while ( sep > Line && ( *sep == '.' || isalnum( *sep ) ) )
            sep--;
        strcpy( user, sep );

        *Variable = '\0';
        strcpy( VariableName, "\\SMTPLog" );
        GetVariable( data->context,VariableName,Variable,sizeof
            (Variable));
        if ( *Variable == '\0' )
            wsprintf(Line," %d SMTPMail %s not found",id,VariableName);
        else
            wsprintf(Line," %d SMTPMail %s %d",id,VariableName,atoi
                ( Variable));
        WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

        *Variable = '\0';
        wsprintf( VariableName, " %s\\WPAddress", domain );
        GetVariable( data->context,VariableName,Variable,sizeof
            ( Variable));
        if ( *Variable == '\0' )
            wsprintf(Line," %d SMTPMail %s not found",id,VariableName);
        else
            wsprintf(Line," %d SMTPMail %s %s",id,VariableName,
                Variable);
        WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

        *Variable = '\0';
        wsprintf( VariableName, " %s\\%s\\MailBox", domain, user );
        GetVariable(data->context,VariableName,Variable,sizeof
            (Variable));
        if ( *Variable == '\0' )
            wsprintf(Line," %d SMTPMail %s not found",id,user,domain);
        else
            wsprintf(Line," %d SMTPMail user %s@%s local - %s",id,user,
                domain,Variable);
        WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

        *Variable = '\0';
        wsprintf( VariableName, " %s\\%s\\DateCreated", domain, user );
        GetVariable(data->context,VariableName,Variable,sizeof
            (Variable));
        if ( *Variable == '\0' )
            wsprintf(Line," %d SMTPMail user %s@%s not local",id,user,
                domain);
        else
            wsprintf(Line," %d SMTPMail user %s@%s local - created %s",
                id, user,domain,Variable);
    }
    else
    {
        wsprintf(Line," %d SMTPMail invalid address \" %s\" ",id,
            data->From );
    }
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}

```

```
// example showing simple processing of data in RCPT command
SMTP_RETURN FAR PASCAL SMTPRcpt( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];
    char user[ MAX_LINE ];
    char domain[ MAX_LINE ];
    char Variable[ MAX_LINE ];
    char VariableName[ MAX_LINE ];
    char *sep;

    wsprintf( Line, " %d SMTPRcpt called in %s (%d)", id, Name, status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    strcpy( Line, data->Rcpt );
    sep = strchr( Line, '@' );
    if ( sep )
    {
        *sep = '\0';
        strcpy( domain, sep + 1 );
        sep = domain;
        while ( *sep && ( *sep == '.' || isalnum( *sep ) ) )
            sep++;
        *sep = '\0';

        sep = Line + strlen( Line ) - 1;
        while ( sep > Line && ( *sep == '.' || isalnum( *sep ) ) )
            sep--;
        strcpy( user, sep );

        *Variable = '\0';
        wsprintf( VariableName, " %s\\%s\\DateCreated", domain, user );
        GetVariable(data->context,VariableName,Variable,sizeof
            (Variable));
        if ( *Variable == '\0' )
            wsprintf(Line," %d SMTPRcpt user %s@%s not local",id,user,
                domain);
        else
            wsprintf(Line," %d SMTPRcpt user %s@%s local - created %s",
                id,user,domain,Variable);
    }
    else
    {
        wsprintf(Line," %d SMTPRcpt invalid address \" %s\\ \"",id,
            data->From);
    }
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}
```

Listing continues on next page.....

```
// called after DATA received but before data is sent
SMTP_RETURN FAR PASCAL SMTPDataBefore( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];

    wsprintf(Line," %d SMTPDataBefore called in %s (%d)",id,Name,status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}

// called after terminating CRLF.CRLF
SMTP_RETURN FAR PASCAL SMTPDataAfter( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];

    wsprintf(Line," %d SMTPDataAfter called in %s (%d)",id,Name,status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}

// called once for each recipient
SMTP_RETURN FAR PASCAL SMTPDataAfterEachRcpt( int id, SMTP_INFO *data )
{
    SMTP_RETURN status = SMTP_DLL_SUCCESS;
    char Line[ MAX_LINE ];

    wsprintf( Line," %d SMTPDataAfterEachRcpt called in %s (%d)",id,Name,status);
    WriteToLog( data->context, CONF_LOG_STATS, 3999, Line );

    return( status );
}
```

User Authentication DLLs

Gordano provides a way of replacing the authentication of user accounts within its message delivery mechanism.

To use your own authentication method:

1. Create a DLL as described below.
2. Place the location of the DLL in the Registry entry:
Software/InternetShopper/Mail/Users/UserDLLLocation.

The DLL should have the following entry points:

Entry point	Description
EntryFn	The entry function for the DLL. This is required for any DLL that is written
ListUserDomain	takes the domain name and returns a list of users and the number of users A non-zero return value will be treated as an error, processed via UserGetErrorMessage to return an error message string and printed to failure logs
ListUserFree	Takes the user list generated by ListUserDomain and frees it.
VerifyUser	Given the username and the domain name, this tells the Gordano server what to do with the e-mail message. This routine is used by the SMTP, POP3 and IMAP4 servers. There are three possible actions: 0 — The user is local (i.e. the POP3 account is on this machine) 1 — This mail account does not exist on this machine. Send the mail to the /Out directory for delivery again. Note that there must be some special delivery rule or DNS setup to make this option work, otherwise the mail is delivered back to the Gordano server again. 2— Unknown user. If the VerifyUser query returns S-SUCCESS and type=2 SMTP will issue a temporary failure notice rather than a permanent one.
VerifyPassword	Check the password and return 0 (SUCCESS) if it's correct. Any non-zero value causes the user to be rejected. This routine is used by the POP3 and IMAP4 servers.
GetMailboxName	Define where the mail message should be placed. If this file already exists, the message will be appended and terminated by a full stop (.). an .idx file is also created with an additional line with three parameters — the byte number this message starts at, a unique string and zero (unread). This routine is used by the SMTP, POP3 and IMAP4 servers.

ChangePassword	If the password server has been enabled, this routine may be called to allow a password to be changed.
GetErrorMessage	Ignored.

This is a listing of the userdll.c file:

```

/*
 * (C)opyright 1995-2015 Gordano Ltd.
 * All rights reserved.
 * No copies of this source may be made without express
 * written permission of Gordano Ltd, Unit 1, Kenn Road, Kenn
 * Clevedon, North Somerset, BS21 6UW, UK

#pragma warning (disable : 4001 4115 4201 4209 4214 4514 )

#include <windows.h>
#include <winbase.h>
#include <stdio.h>

#pragma warning ( disable : 4001 4115 4201 4209 4214 4514 )

#define S_SUCCESS 0

BOOL __stdcall DllMain( HMODULE hmod, ULONG Reason, LPVOID Reserved )
{
    if ( Reason == DLL_PROCESS_ATTACH )
    {
    }
    else if ( Reason == DLL_THREAD_ATTACH )
    {
    }
    else if ( Reason == DLL_THREAD_DETACH )
    {
    }
    else if ( Reason == DLL_PROCESS_DETACH )
    {
    }

    return( TRUE );
}

int __cdecl ListUserDomain(char *domain, /* in */
                          char **users, /* out */
                          int *numUsers); /* out */
/*domain = domain name. e.g "test.dom".
 users = colon separated user names. E.g. "test1:test2:test3".
 numUsers = number of users in the list. 3 for the above example.*/
{
    int status = S_SUCCESS;

    return ( status );
}

int __cdecl ListUserFree (char *users); /* in */
/*users = buffer returned from ListUserDomain().*/
{
    int status = S_SUCCESS;

    return ( status );
}

int __cdecl VerifyUser (char *user, /* in */
                       char *domain, /* in */
                       int *type); /* out */
/*user = user name.
```



```

domain = domain name.
type = type of account. Set to 0 if user exists, 2 if the user is
unknown.*/

```

```

    int status = S_SUCCESS;

    // accept any account
    *type = 0;

    return( status );
}

```

```

int __cdecl VerifyPassword (char *user, /* in */
                           char *domain, /* in */
                           char *password); /* in */
/*user = user name.
domain = domain name.
password = the users password.*/
{
    int status = S_SUCCESS;

    // accept any password

    return( status );
}

```

```

int __cdecl GetMailboxName (char *user, /* in */
                           char *domain, /* in */
                           char *mailbox); /* out */
/*user = user name.
domain = domain name.
mailbox = the users inbox name. e.g. "InBox.mbx".*/
{
    int status = S_SUCCESS;

    // create default mailbox filename

    strcpy( filename, "User.mbx" );

    return( status );
}

```

```

int __cdecl ChangePassword (char *user, /* in */
                           char *domain, /* in */
                           char *oldpassword, /* in */
                           char *newpassword); /* in */
/*user = user name.
domain = domain name.
oldpassword = the users old password.
newpassword = the users new password.*/
{
    int status = 12;

    // return error - password cant be changed

    return( status );
}

```

```

int __cdecl GetErrorMessage (int error, /* in */

```

```
        char *message); /* out */
/*error = error code returned from one of the above functions.
 message = buffer into which error string is written.*/
{
    int status = S_SUCCESS;

    // create error message

    sprintf( msg, "User DLL error %d", error );

    return( status );
}

/*Use a buffer size of 10240 bytes for all output strings.
 A return code of 0 indicates success and -1 indicates failure.*/
```

Extenal Folder Manager (EFM) DLL

An API has been developed which allows complete control over GMS message stores and allows message stores to be populated on-the-fly from external applications. To use your own message store rather than the default GMS one, or to integrate with external applications such as Archiving Systems or Unified Messaging Systems you will need to use the following method:

1. Create a DLL as described below.
2. Place the location of the DLL in the Registry entry:
Software/InternetShopper/Mail/Users/EFMDLL.

Once configured the DLL will be loaded by all GMS services on startup.

The DLL consists of an abstract set of APIs to allow mail to be stored in external folders. Mail messages are handled individually and stored in individual files. The DLL can choose whether to store mail for a specific user or not. If the DLL does not store it then the native GMS folder library will be used. This will allow for different domains to store their data locally or externally. It will also allow message logs to be stored in different locations. This will transform the interface from being simply a storage interface to also being an archiving\auditing interface.

Types

All external folder access is performed using an external folder handle which is returned by the open API and should be destroyed by the close API. A number of helper functions are necessary to extract information from the handle.

EFM_SESSION_HANDLE is a generic pointer to some structure.

```
typedef void *EFM_SESSION_HANDLE;
```

EFM_LOCK_HANDLE is a generic pointer to some structure.

```
typedef void *EFM_LOCK_HANDLE;
```

EFM_FOLDER_HANDLE is a generic pointer to some structure.

```
typedef void *EFM_FOLDER_HANDLE;
```

Folder indexes are returned in a list of EFM_MESSAGE_LIST structures:

```
typedef struct efm_message_list_struct
{
    int number;
    char UDIL[];
    int UID[];
    int size;
    char flags[];
    char event[],

    struct efm_message_list_struct *next;
} EFM_MESSAGE_LIST;
```

Folder lists and subscribed folders are returned in a list of EFM_FOLDER_LIST structures:

```
typedef struct efm_folder_list_struct
{
    char *foldername;
    int attributes;

    struct efm_folder_list_struct *next;
} EFM_FOLDER_LIST;
```

The following attributes may be set:

1. NOSELECT - no select folders may not be selected, deleted etc.
2. NOINFERIORS - no inferior folders may be created

Date\Time Format

All date\time strings must be in RFC format and in UTC (GMT), for example:

Wed, 7 Jul 2015 10:12:19 GMT

Status Flags

All status flags must be passed as strings using standard GMS format. The following flags are standard:

READ
ANSWERED
FLAGGED
DELETED
DRAFT
RECENT
EXPUNGED

All other flags are "non-standard".

The EXPUNGED flag is a special case flag passed by to GMS to indicate that a message has been expunged. Only EXPUNGED messages should be purged using EFM_PurgeFolder. Messages will always be moved from the DELETED state to the EXPUNGED state prior to purging.

All flags passed by GMS to the external folder API should be passed back in the same manner. Specifically, all non-standard flags should be preserved unless modified by GMS. Standard flags may be modified but update notifications should be generated (see later).

Message File Format

Messages are transferred one message per file. The file format used is the standard GMS MBX file format but with a single message per MBX file.

This format of the MBX file is an "off the wire format" from SMTP. It contains the trailing CRLF.CRLF sequence used in SMTP to indicate the end of the message. This has a number of implications:

1. Lines starting with a '.' Character are escaped. See RFC 2821 Transparency.
2. POP uses the same transparency rules (RFC 1939)
3. IMAP does not use transparency. The IMAP server will handle this.
4. Message sizes should all include the escaped lines. Sizes in IMAP will be adjusted accordingly.
5. Message sizes should include the end of message trailing dot.

Functions

All functions will take an "id" parameter as the first parameter. This will be a unique thread/session identifier as defined by the calling service.

Error Codes

All functions will return an integer as their return value. This will be a status indicator.

0 - SUCCESS

Non-zero - FAILURE

Initialization

A function is provided to initialize the EFM library. This should perform any one time initialization of the library.

```
extern int EFMConfigure(int id
                        void *logMes,
                        void *logHandle );
```

A similar function is provided to close down the library

```
extern int EFMClose(int id,
                    int serviceType);
```

Sessions

Some systems may require a session to be created to allow folder access. APIs are provided to create and close sessions:

```
extern int EFMCreateSession(int id,
                           char *user, char *domain,
                           int serviceType,
                           EFM_SESSION_HANDLE *sessionHandle);
```

```
extern int EFMCloseSession(int id,
                          EFM_SESSION_HANDLE *sessionHandle);
```

User and domain should be NULL or empty strings to indicate global or domain sessions.

Session Keep Alive

An API is provided to handle NOOP, CHECK and IDLE commands.

```
extern int EFMKeepAlive(int id,
                       EFM_SESSION_HANDLE sessionHandle);
```

Message Functions

An API is required to load the given message into a local file. The returned filename should either be an absolute path or a path relative to the GMS installation directory.

```
extern int EFMGetMessage(int id,
                        EFM_FOLDER_HANDLE handle,
                        int uid,
                        char *filename, size_t size);
```

An API is required to mark the message as no longer needed. This will allow the DLL to free of resources and delete any temporary files prior to folder close.

```
extern int EFMCloseMessage(int id,
                          EFM_FOLDER_HANDLE handle,
                          int uid,
                          char *filename);
```

An API is provided to copy messages. This will allow the DLL to perform more efficient copies and moves than the EFMStoreMessage will allow.

```
extern int EFMCopyMessages(int id,
                          EFM_FOLDER_HANDLE handle,
                          char *foldername,
                          EFM_MESSAGE_LIST *messageIdList);
```

The messageIdList is a list of messages to be copied. The UID and UDIL of the stored message are returned in the same list. The folder list should be closed with EFMCloseFolderList. This API should either copy all messages or none of them. The message meta data (e.g. statuses) should also be copied.

An API is provided to control access to expunged messages prior to final deletion. Some EFM implementations may support this, others may not.

```
extern int EFMExpungedMessageAccessAllowed(int id,
                                           EFM_SESSION_HANDLE sessionHandle,
                                           BOOL *allowed)
```

Folder Helper Functions

The following helper functions should be available. Note that not all functions may be relevant to a particular instance of an external storage DLL. In this case, meaningful defaults should be returned.

The EFMIsExternalFolder is used as the basis of all future storage requests for this account\folder. No EFM calls will be made unless this API returns TRUE.

```
extern int EFMIsExternalFolder(int id,
                              char *user, char *domain,
                              int serviceType,
                              BOOL *external);

extern int EFMGetFolderUIDValidity(int id,
                                  EFM_SESSION_HANDLE sessionHandle,
                                  char *foldername,
                                  int *uid);
```

```
extern int EFMGetFolderNextUID(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    int *uid);
```

```
extern int EFMGetFolderSize(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    int *size);
```

```
extern int EFMGetFolderModified(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    char *modifiedStr, int size);
```

```
extern int EFMFolderCountMessagesOfType (int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    char *includeFlags,  
    char *excludeFlags,  
    int *count);
```

Folder Locking

GMS has a concept of client locks (to stop simultaneous access from IMAP, POP and WebMail) and service locks (used to prevent all access to folders during critical operations). DLLs may provide their own locking or allow GMS to perform locking. The external client and service locks will be called as and when necessary by GMS.

```
extern int EFMGMSClientFolderLockRequired(int id,  
    BOOL *required);
```

```
extern int EFMGMSServiceFolderLockRequired(int id,  
    BOOL *required);
```

```
extern int EFMClientFolderLock(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    EFM_LOCK_HANDLE *lockHandle);
```

```
extern int EFMServiceFolderLock(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    EFM_LOCK_HANDLE *lockHandle);
```

```
extern int EFMCloseFolderLock(int id,  
    EFM_LOCK_HANDLE *lockHandle);
```

```
extern int EFMDoesClientFolderLockExist(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    BOOL *exists);
```

```
extern int EFMDoesServiceFolderLockExist(int id,  
    EFM_SESSION_HANDLE sessionHandle,
```

```
char *foldername,  
BOOL *exists);
```

Folder Exists

An API is provided to check folder existence. Note that folders with the noselect flag on them are deemed not to exist.

```
extern int EFMExistFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    BOOL *exists);
```

Folder Create

An API is provided to create folders. If the folder exists then an EXISTS error should be returned. If the parent folder(s) do not exist then they should be created.

```
extern int EFMCreateFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername);
```

Folder Delete

An API is provided to delete folders. If the folder does not exist then an NO_FOLDER error should be returned. The API should not remove sub-folders if they exist. Please note that deleting a folder does not affect the folder subscription.

```
extern int EFMDeleteFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername);
```

Folder List

An API is provided to return a full list of folders. Wildcard searches are handled by the IMAP server.

```
extern int EFMGetFolderList(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    EFM_FOLDER_LIST **list);
```

An API is also provided to allow the DLL to free off the folder list once GMS has used it.

```
extern int EFMCloseFolderList(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    EFM_FOLDER_LIST **list);
```

Folder Rename

An API is provided to rename folders. This API will be responsible for renaming all of the inferior folders. This is to allow optimizations in the rename to occur. Note that renaming a folder (except Inbox) causes all sub-folders to be renamed. Note also that renaming a folder does not affect the folder subscription.

```
extern int EFMRenameFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    char *newfoldername,
```



```
BOOL renameSubFolders);
```

Folder Subscribe

An API is provided to subscribe to a folder.

```
extern int EFMSubscribeFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername);
```

An API is also provided to unsubscribe from a folder.

```
extern int EFMUnsubscribeFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername);
```

Folder List Subscribed

An API is provided to return a list of subscribed folders.

```
extern int EFMGetSubscribedFolderList(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    EFM_FOLDER_LIST **list);
```

An API is also provided to allow the DLL to free off the folder list once GMS has used it.

```
extern int EFMCloseSubscribedFolderList(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    EFM_FOLDER_LIST **list);
```

Folder Open

A folder is always opened using the following API: If a failure is returned then returned handle must be NULL.

```
extern int EFMOpenFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    char *foldername,  
    int flags,  
    EFM_FOLDER_HANDLE *folderhandle);
```

The flags parameter is designed to allow different types of open to be performed:

1. FULL
2. INFORMATION

This is to allow GMS to inform the external folder DLL that a folder has been opened more than once. No new folder handle is required but any open counts should be updated.

Message Index

The folder open API is required to return a list of messages in the folder. This list should be a linked list of messages. The list will be translated to the GMS index list. Note that both deleted (but not purged) and non-deleted messages should be returned. The GMS servers will interpret these appropriately depending on the service requirements.

```
extern int EFMGetFolderIndex(int id,  
    EFM_FOLDER_HANDLE folderhandle,  
    EFM_MESSAGE_LIST **messageIndex);
```

An API is also provided to allow the DLL to free off the messageIndex once GMS has interpreted it.

```
extern int EFMCloseFolderIndex(int id,  
    EFM_FOLDER_HANDLE folderhandle,  
    EFM_MESSAGE_LIST **messageIndex);
```

Folder Close

A folder is always closed using the following API. GMS will attempt to close all folders it opens. The functions should de-allocate all resources (including temporary files) associated with the external folder handle. The handle should be set to NULL on return. This function will not flush any un-committed changes to the database. This should be explicitly done with a FolderCommit.

```
extern int EFMCloseFolder(int id,  
    EFM_SESSION_HANDLE sessionHandle,  
    EFM_FOLDER_HANDLE *folderhandle,  
    int flags);
```

The flags parameter is designed to allow different types of close to be performed:

1. FULL
2. INFORMATION

This is to allow closure of multiply opened folders.

Please note that GMS may not close all folders in the event of an uncontrolled process termination e.g. kill -9 on Unix systems.

Folder Commit

Flushes all pending status updates.

```
extern int EFMCommitFolder(int id,  
    EFM_FOLDER_HANDLE folderhandle);
```

Folder Purge

Expunged messages are removed from the folder using the following API. This will apply on POP QUIT or IMAP EXPUNGE. All messages marked as EXPUNGED should be removed. Please note that this API works on EXPUNGED messages and not DELETED messages.

```
extern int EFMRemoveFolder(int id,  
    EFM_FOLDER_HANDLE folderhandle);
```

Folder Store

An API is provided to store a message in a folder. Note that this function appends to a folder by name rather than the using a folder handle. This is because, in general, the folder is not open and hence we avoid the overhead of reading an index file on append. The function may return NOT_HANDLED to indicate that the

message has not been stored and should be stored in the internal database. Both the UID and UDIL of the stored message are returned.

```
extern int EFMStoreFolder(int id,
                          EFM_SESSION_HANDLE sessionHandle,
                          char *foldername,
                          char *filename,
                          char *flags,
                          int *uid,
                          char *udil,
                          int size);
```

Message Flags Changes

Different protocols have different mechanisms as to when status changes are committed. For instance in POP this is only done on QUIT whereas in IMAP it is done immediately. All message status changes are cached by both the GMS and EFM layers, a commit is required to flush them. The flags are the complete flags NOT partial updates. All changes to standard flags must result in notification callback.

```
extern int EFMSetMessageFlags(int id,
                              EFM_FOLDER_HANDLE folderhandle,
                              int uid,
                              char *flags);
```

Folder Update Notifications

In IMAP there is a requirement to send untagged responses whenever the following conditions exist:

1. New message appended to a selected folder (all selected mode commands)
2. Flags changed for a message in a selected folder (all selected mode commands)
3. Messages purged from a selected folder (all selected mode commands)
4. IDLE command

Currently IMAP and SMTP use an IPC\Pipe based mechanism to obtain this information. This mechanism causes the folder index to be re-read and flag updates to be sent.

In an external folder there may be other mechanisms whereby these changes occurs (e.g. in a multiple machine environment where the same folder is accessed on different machines).

The current mechanism will be preserved for local notifications. Non local notifications will be handled using a callback. The callback should not be used for local notifications.

An API is provided to allow a user of the EFM library to receive callbacks whenever any of these events take place. A registration API allows custom code to be registered and called asynchronous by the DLL

```
extern int EFMRegisterUpdateCallback(int id,  
    EFM_UPDATE_CALLBACK_FN *updateCallback);
```

The updateCallback function will be a function with the following prototype. The callback causes IMAP to search the session list and identify any sessions with this folder selected. This folder is then updated.

```
extern int updateCallback(char *user,  
    char *domain,  
    char *foldername,  
    EFM_MESSAGE_LIST *msgList);
```

The event string is one of:

1. NEW
2. FLAGS
3. EXPUNGED - sent after a PurgeFolder has removed the message.

Message parsing and log DLLs

A message sent to the SMTP server passes through the following stages within a Gordano server:

1. Each time an RCPT clause is encountered, the redirect file is parsed and (if necessary) the protocol RCPT clause is modified.
2. Once the DATA stage has been reached, the message is loaded into a temporary file in the /Temp directory.
3. When it has completely arrived from the source mail server, the following stages are followed for each RCPT clause:
 - Must it be logged? If full message logging is switched on the message is logged in the domain specified by the RCPT clause. If this domain is not on this machine, the message is logged in the domain represented by the IP address the message was delivered to.
 - Is extra processing required? If the Message Parsing DLL has been installed, it will be called. It will be passed the current MAIL, RCPT and data file for any additional processing that may be required. The DLL can change the RCPT clause or message, but cannot change the MAIL or the file's name.
 - Message delivery. The message is passed to the delivery rule code which arranges its arrival at the correct location.

The Message Parsing DLL is only loaded **once** when the Gordano services start up. The following Registry variable is checked and an attempt to load the named DLL is made:

Software\InternetShopper\Mail\Parameters\MailLogDLLLocation:REG_SZ:

Once the DLL has been loaded, the server searches it for this routine:

```
DllExport int LogMessage(
    int major,    // Major number of NTMail (e.g. 3)
    int minor,    // Minor number of NTMail (e.g. 02)

    char *From,   // "MAIL" clause parameter received
                // from the server sending e-mail

    char *To,      // Destination of this message
                // (May be local or remote).

    char *FileName, // Name of temporary file where
                // this message has been saved.

    void *)       // For future expansion
{
}
}
```

If the routine is found, it is called for every message that passes through the Gordano mail server.

Note the following

- If there is more than one RCPT clause in a message delivery, the DLL will be called for each clause. Therefore, if you only wish to analyse the message once, you should mark it in some way (for example, add a clause to the header of the message).
- The DLL may start an extra thread for background processing if required
- To change the DLL, the SMTP server must be stopped, the new DLL installed and the SMTP server restarted. Although the SMTP server will drop the DLL and reload on each UPDATE event, this is likely to be too fast for a file to be replaced manually.

22 Using ColdFusion Tags

You may decide to write your own Web scripts, for example to allow administrators to enter details of new users, or to change user attributes. To help with this, Gordano has made ColdFusion (™) tags available for your use at no charge.

ColdFusion is a scripting tool for Web servers. In ColdFusion terminology, the Web page is a *cfx template*.

For full information on how to write ColdFusion scripts you must refer to the ColdFusion documentation, but in a simple script there are four types of statement:

- Comment — contained within `<!--` and `-->` symbols, like this:
`<!-- comment -->`
- Header — contained within `<h2>` and `</h2>` symbols, like this:
`<h2> Header </h2>`
- Tag — writes a value to the Registry. Tags take this form:
`<cfx_ntmail Action="name"
parameter1="x"
parameter2="y">`
- Output — writes one or more values to the Web page, using this format:
`<cfoutput> x = #var_name# </cfoutput>`

Where "var_name" is a variable which has been input by filling in a Web page form.

The example routines shown below use all of these. These examples add users to the database, change user characteristics and list users, etc. They use and write to several different Gordano Registry parameters. You can reference and modify any Gordano Registry parameter in your script.

To interpret and modify these examples, you need to know the values of the user Type parameter; see page 111.



These examples are based on the file cfexamples.txt, which is available in the Gordano directory.

22.1 Example ColdFusion Routines

The code segment given below could be used to obtain the domain name and user name, which would be entered into a form by the administrator. These variables are then used in the other routines which follow.

```
<!--- Lets set up our variables for use elsewhere --->
<cfset OurDomain = "CompanyA.dom">
<cfset User = "Matthew">
```

Logging on as an administrator

To carry out actions such as adding and removing users you have to log on as an administrator as a first step:

```
<cfx_ntmail
  action="Logon"
  User="postmaster"
  domain="#OurDomain#"
  password="postmasterpassword">
```

Adding a user

The routine shown below adds a user to the database. The type would be obtained from the administrator filling in a form. They would enter the account type they wanted, for example "POP account", and the form would translate this to a Type value like 262146. For details of the user Type parameter; see page 111.

The cfx_ntmail tag named UserAdd is what adds the user to the database. If the user is added successfully, the cfoutput statement returns 0, otherwise it prints one of the error status codes listed at the end of the routine.

```
<P>
<h2>ADD A USER </h2>

<cfx_ntmail
  action="UserAdd"
  User="#User#"
  domain="#OurDomain#"
  type="262146">

<cfoutput>
Status = #status#<br>
StatusNum = #StatusNumber#
</cfoutput>

<!---
  This routine can return the following StatusNumbers:
    4009 - The domain does not exist
    4010 - Failed addition of user (see logs)
    4011 - Exceeded number of licensed users
    4012 - Account already exists
    4013 - Missing variable
--->
```


Changing the user's attributes

The routine shown below changes the attributes of a user in the database. Any attribute of an account can be changed. Even values that are not used by Gordano products can be stored there, if required.

The example sets the type to 128, which signifies an autoresponder. For details of the user Type parameter; see page 111.

The cfx_ntmail tag named UserSet makes the change to the database. If the attribute is changed successfully, the cfoutput statement returns 0, otherwise it prints one of the error status codes listed at the end of the routine.

```
<P>
<h2> CHANGE ATTRIBUTES OF USER</h2>

<!-- Lets change this user by giving them an autoresponder --->
<P>
<cfx_ntmail
  action="UserSet"
  User="#User#"
  domain="#OurDomain#"
  Variable="Type"
  Value="128"
>

<cfoutput>
Status = #status#<br>
StatusNum = #StatusNumber#
</cfoutput>

<!--
  This routine can return the following StatusNumbers:
    4009 - The domain does not exist
    4013 - Missing variable
    4014 - No such user
-->
```

Displaying the user's attributes

The routine shown below prints a user's attributes to the screen. Any attribute of an account can be displayed.

The `cfx_ntmail` tag named `UserGet` gets the information from the database. The `cfoutput` statement returns 0, otherwise it prints one of the error status codes listed at the end of the routine.

This example shows use of two conditional statements, like this:

```
<cfif #value# is x>action </cfif>
```

The action after the first ">" is taken if the condition is true. In this example, if the type is 262146 the text "POP account" is written to the screen. If the type is 128, "Autoresponder" is written out.

```
<P>
<h2>DISPLAY ATTRIBUTES OF USER</h2>
```

```
<cfx_ntmail
  action="UserGet"
  domain="#ourdomain#"
  User="#User#"
  variable="Type"
>
```

```
<cfoutput>
User name: matthew<br>
Domain: #ourdomain#<br>
Account type: <cfif #value# is 262146>POP Account</cfif><cfif #value# is
128>Autoresponder</cfif><br>
</cfoutput>
```

```
<!---
  This routine can return the following StatusNumbers:
    4009 - The domain does not exist
    4014 - No such user
--->
```

Enumerating users in a domain

The routine shown below prints the attributes of users in a domain to the screen. Any attribute of an account can be displayed.

The `cfx_ntmail` tag named `UserEnum` gets the information on users in the specified domain from the database. (The domain was obtained using the code at the start of the section.) The `cfoutput` statement returns 0, otherwise it prints one of the error status codes listed at the end of the routine.

This example again shows use of conditional statements; see above for details. If the type is 262146, the text "POP account" is written to the screen. If the type is 128, "Autoresponder" is written out.

```
<P>
<h2>ENUMERATE USERS IN A DOMAIN</h2>

<cfx_ntmail
  action="UserEnum"
  domain="#ourdomain#"
>

<cfoutput>
Status = #status#<br>
StatusNumber = #StatusNumber#<br>
<cfif StatusNumber is 0>
  Value = #Variable#
</cfif>
</cfoutput>

<!--
  This can return the following StatusNumbers:
    4009 - The domain does not exist
-->
```

Deleting a user

This routine removes a user from the database. The domain name and user would be obtained from the administrator filling in a form.

The `cfx_ntmail` tag named `UserDel` removes the specified user from the database. If the user is deleted successfully, the `cfoutput` statement returns 0, otherwise it prints one of the error status codes listed in the comment at the end of the routine.

```
<P>
<h2>DELETE A USER</h2>

<cfx_ntmail
  action="UserDel"
  domain="#ourdomain#"
  User="#User#"
>

<cfoutput>
Status = #status#<br>
StatusNum = #StatusNumber#
</cfoutput>

<!---
  This routine can return the following StatusNumbers:
    4009 - The domain does not exist
    4014 - No such user
    4015 - Could not delete user (see logs)
--->
```

Add a domain

This routine will add a domain to the database. The domain name would be obtained from the administrator filling in a form.

The cfx_ntmail tag named DomainAdd adds the specified domain to the database. If the domain is added successfully, the cfoutput statement returns 0, otherwise it prints one of the error status codes listed in the comment at the end of the routine.

The BaseDomain and PostFix options are only required if you are setting up a virtual domain.

```
<P>
<h2>ADD A DOMAIN</h2>

<cfx_ntmail
  action="DomainAdd"
  TypeStr=" 1 "
  BaseDomain="#OurDomain#"
  Postfix="postfix"
>

<cfoutput>
Status = #status#<br>
StatusNum = #StatusNumber#
</cfoutput>

<!--
  This routine can return the following StatusNumbers:
    400 - Not enough memory
    4009 - The domain does not exist
    4010 - Failed addition of user (see logs)
    4012 - Account already exists
-->
```


23 Gordano Accessory Pack

The Gordano Accessory Pack provides a number of utilities designed to enhance your use of Gordano Products. The utilities and their uses are listed below. If the files are unzipped to the Gordano base directory using the option "Use Folder Names" in WinZip they should be automatically copied to the appropriate directories.

- Gizmos
- Mailbox Inspector
- NTMetrn Utility
- SQL Authentication DLL
- LDAP Authentication DLL
- MML Examples and Documentation
- Robots
- Full to Virtual Domain Conversion
- Upgrade V3 Mailboxes
- Example SMTP DLL
- Example External Folder Manager (EFM) DLL
- Virtual to Full Domain Conversion
- COM DLL
- Monitoring Software

23.1 Gizmos

There are a number of Gizmos included in the main installation of GMS. All of these, along with a number of additional Gizmos are also included in this Accessory Pack. These Gizmos are provided as examples of what can be achieved using Gizmo technology, please feel free to copy and amend them to your own requirements.

Each of the Gizmos and all of the available Gizmo functions are fully documented in the GMS Reference Guide. If you design any Gizmos you particularly like, and would like to share those Gizmos with others please feel free to do so. Alternatively please contact us and we will make them available to other GMS users via our web site.

The example Gizmos can be found in the following location within the GMS Accessory Pack: `\mml\webmail\plugins`

23.2 Mailbox Inspector

Inspector is a utility, designed to help Network Administrators maintain the email on their server more easily. It will allow you to view the contents of domains and mailboxes and perform

"housekeeping" duties such as renaming a mailbox, or deleting a message from an individual mailbox.

23.3 NTMetrn Utility

This utility is designed to be used with a Gordano mail server to initiate the transfer of files via ETRN. This would be used by one Gordano server collecting mail from another or can be used by completely independent servers such as MS Exchange SBS server to collect mail from a Gordano server. See your mail server documentation for details of how to initiate collection of mail via ETRN.

23.4 MML Examples and Documentation

There are a number of MML examples included, along with a copy of the MML Scripting Guide. Examples include an example mail reading GUI, delivery scripts and timed events. There are three specific types of examples. A syntax file for TextPad is also included to provide colour syntax highlighting for MML files, see <http://www.textpad.com>

Delivery Scripts

These examples should be used by pasting them into the appropriate area of the Gordano GUI.

Timed Events

These examples should be used by pasting them into the appropriate area of the Gordano GUI.

GUI

The example GUI should be placed in a directory under the base MML directory called "usr", i.e. `c:\gordano\mml\usr`. You can then modify the files at will to achieve the look and feel that you would like. The example GUI can be accessed by pointing your browser to port 8888 rather than port 8000, i.e. <http://localhost:8888>

23.5 Robots

A number of robots are included in this option pack, each of them are described below (see "Robots" on page 237). A number of these require a license before they will operate as indicated. All of the robots are User robots with the exception of domain.exe which applies to Robot Domains only. The robots should be placed in the "bin" directory and they will then be automatically recognised by the appropriate configuration pages in the Gordano GUI.

Demo keys for any of the robots that require them can be obtained by contacting sales@gordano.com

Archive

The Archive robot allows you to keep an archive of all messages arriving at a particular account. The robot will accept an email message, thread it and place it into an ODBC compliant database. The command line should look something like:

```
c:\gordano\archive.exe -path c:\gordano\mail.mdb
```

The archive program will create the archive on the first email message that arrives and then add subsequent messages. Once the messages are in the database, you can access them in any way you like. By default, all MIME and UUENCODED binary parts are removed from the message.

Typically, to archive a list, you would add a list member called "list-archive" (for example) and create a robot with this name:

```
c:\gordano\archive.exe -path c:\gordano\list-archive.mdb
```

The advantage of this is that all mail that is posted to members is placed into the archive while email that is rejected (e.g. an unsubscribe message) that is rejected by the list server does not appear in the archive.

Command line options

- **-path full-file-name** - Allows the full name of an Access database to be specified for the data to be placed into. If the database does not exist, Archive will create the database with default parameters. For example:

```
-path c:\gordano\archive\discuss.mdb
```
- **-table name** - Defines the table into which the message will be placed. If the table does not exist, creates a blank table and starts putting data into it.
- **-sql** - The database is SQL Server.
- **-db name** - Name of the dataset to write the data to (mutually exclusive with -path option). For example:

```
-db abd -password pass
```

- **-username archive** - could be used to write data to tables in the dataset called "abd".
- **-password password** - Data set password (if required).
- **-username username** - Data set username (if required).
- **-nodetach** - Causes the archiver NOT to detach MIME and UUENCODED parts of messages which are binary images.
- **-verb** - Allows logging, should only be used if problems are being encountered.

Table fields

The table that archive writes data to must have the following fields.

Field Name	DataType	Comments
ID	AutoNumber	Used as the message id
EMail	Text	Email address of person who sent message
Author	Text	Name of author. If cannot be found, then the email address.
DateSent	Date/Time	Date and time that the message was sent according to the "Date:" clause in the email header
DateArchived	Date/Time	Date and time the message was archived
Subject	Text	Subject of the message with "Re:" and "- Reply" clauses removed
Response	Text	
SentTo	Text	
Message	Memo	The message itself
MessageHTML	Memo	Message converted to HTML
ReplyTold	Number	-1 if this is the first message with this subject Otherwise is the number of the message with the same subject.
OriginalMessageId	Number	
NextMessageId	Number	Possible next message in this thread
NextId	Number	Number of next message in the archive
PreviousId	Number	Number of previous message in the archive

Detach

The Detach robot will remove any attachments arriving in emails to the account and store them in the specified directory.

Domain (free)

This program will take a message and change the domain name section and repost it. This may be used as a domain robot to redirect all mail for a given domain to another domain.

eSarah

"GMS Archive" automates the storage and searching of message archives. eSarah automatically stores all email communications passing through your server to a secure remote server. The archive can be indexed and queried to extract specific messages. See the full section on eSarah below.

Filter

The Filter robot will read a list of words or phrases and redirect any incoming messages containing one of these words or phrases to a given account.

Rotate (free)

The Rotate robot takes an incoming mail message and sends it in turn to each of the email addresses listed for the robot. This is a particularly useful tool for, say, a sales team, where each of the members of the team will receive incoming messages in turn.

23.6 Full to Virtual Domain Conversion

This utility is used to convert full domains to virtual domains and is an ideal solution for those that are running short on IP numbers. Place the executable fulltovirtual.exe in your Gordano\bin directory then run from a command prompt within that directory. Remember though that if you have lists running under the full domain these will not function under the virtual domain. All Gordano Services should be stopped before running the utility

Command Line Parameters

Parameter	Description
-bbaseDomain	Specifies the base domain to be used.
-vvirtualDomain	Specifies the virtual domain to be converted
-ppostfix	Specifies the postfix of the virtual domain to be converted
-h	Display help

23.7 Upgrade V3 Mailboxes

If you wish to install a fresh copy of a Gordano mail server rather than upgrading an existing V3 installation, this utility can be used to copy over all the users mbx files to the new server, upgrading them to the required format while doing so. Run upgradembx.exe on the machine with NTmail V3 installed on it. The files contained in the zip should be copied to NTMail v3 executable directory, then from a command prompt in that directory run:

upgradembx path_to_NTMail

This will copy existing v3 mailboxes etc to the new machine. It assumes you have a network connection to and a shared drive on the new machine.

23.8 Example SMTP DLL

SMTP DLLs act on all e-mail entering the system during the course of a normal mail transaction (see "DLLs" on page 241). The DLLs can act on any stage of the SMTP protocol. This example DLL should set you on the way to developing your own DLL's.

23.9 Example EFM DLL

The External Folder Manager (EFM) DLL allows complete control over GMS message stores and allows message stores to be populated on-the-fly from external applications (see "DLLs" on page 241). This has a multitude of uses but will mostly appeal to those who wish to integrate GMS with other systems such as Telco's, VoIP Switches, Archiving Systems or Document Management Systems. This example DLL should set you on the way to developing your own DLL's.

23.10 Virtual to Full Domain Conversion

This utility is used to convert virtual domains to full domains, you may want to do this for instance, if you have a requirement to run lists under the domain. Place the executable virtualtofull.exe in your Gordano\bin directory then run from a command prompt within that directory. All Gordano Services should be stopped before running the utility.

Command Line Parameters

Parameter	Description
-bbaseDomain	Specifies the base domain to be used.
-wvirtualDomain	Specifies the virtual domain to be converted.
-ppostfix	Specifies the postfix of the virtual domain to be converted
-flog	Specifies that progress should be logged to file <log>
-h	Display help

23.11 COM DLL

Provides a COM interface to any of the Gordano administrative functions.

Installation:

Copy ntmcom.dll to NTMail bin directory.

Run 'regsvr32 ntmcom.dll' from NTMail bin directory to register NTMail COM Interface

Function Spec:

Unless specified otherwise, domain aliases cannot be used where a domain is needed.



Examples of Com object usage are included in the option pack

General Functions

GLogLevel

Sets logging level to 'level', returning previous level in 'prevLevel'. Default logging level is 0 (no logging).

```
HRESULT GLogLevel( [in] int level, [out, retval] int *prevLevel )
```

Authentication Functions

GGetLogonToken

Get a logon type and token for domain 'domain_name'.

```
HRESULT GGetLogonToken( [in] BSTR domain_name, [out, retval] int *type,
[out, retval] BSTR *token )
```

type	logon_info
0 POP Logon	not used (a blank string is returned)
1 APOP Logon	key for an MD5 digest of the password for 'user_name@domain_name'

Status: Not implemented.

GLogon

Attempt to logon as 'user_name@domain'. The meaning of 'logon_info' depends on the logon type 'type' returned by a preceding call to GetLogonToken.

HRESULT GLogon([in] BSTR user_name, [in] BSTR domain_name, [in] BSTR logon_info).

type	logon_info
0 POP Logon	the plain text password for 'user_name@domain_name'
1 APOP Logon	the MD5 digest of the password for 'user_name@domain_name' and the token 'token'

Status: Not implemented.

GLogoff

Logoff as 'user_name@domain'.

HRESULT GLogoff([in] BSTR user_name, [in] BSTR domain_name)

Status: Not implemented.

User Functions

GUserAdd

Add user 'user_name@domain_name'. The initial password is set to 'password'.

HRESULT GUserAdd([in] BSTR user_name, [in] BSTR domain_name, [in] BSTR password, [in] int type)

Status: Implemented.

GUserDel

Delete 'user_name@domain_name'.

HRESULT GUserDel([in] BSTR user_name, [in] BSTR domain_name, [in] BSTR transfer_user_name, [in] int delete_request_flag)

Status: Implemented.

GUserGetVar

Get the current value of the variable 'variable_name' for 'user_name@domain_name'

HRESULT GUserGetVar([in] BSTR user_name, [in] BSTR domain_name, [in] BSTR variable_name, [out, retval] BSTR *value)

Status: Implemented.

GUserGetDir

Get the user directory for 'user_name@domain_name'.

```
HRESULT GUserGetDir( [in] BSTR user_name, [in] BSTR domain_name, [out,
retval] BSTR *directory )
```

Status: Not implemented.

GUserSetVar

Set the value of the variable 'variable_name' for 'user_name@domain_name' to 'new_value'

```
HRESULT GUserSetVar( [in] BSTR user_name, [in] BSTR domain_name, [in]
BSTR variable_name, [in] BSTR new_value )
```

Status: Implemented.

GUserList

Return a list of users in the domain 'domain_name'. The entries are separated by ':'

```
HRESULT GUserList( [in] BSTR domain_name, [out, retval] BSTR *user_list )
```

Status: Implemented.

Alias Functions

GAliasAdd

Add 'alias' as an alias for account 'user_name@domain_name'.

```
HRESULT GAliasAdd( [in] BSTR user_name, [in] BSTR domain_name, [in]
BSTR alias )
```

Status: Implemented.

GAliasDel

Delete 'alias' from the list of aliases for account 'user_name@domain_name'.

```
HRESULT GAliasDel( [in] BSTR user_name, [in] BSTR domain_name, [in] BSTR
alias )
```

Status: Implemented.

Domain Functions

GDomainAdd

Add a domain 'domain_name'. The domain aliases entry is set to 'domain_name *.domain_name'

HRESULT GDomainAdd([in]BSTR domain_name, [in] int type, [in] BSTR param1, [in] BSTR param2)

If the domain is created successfully the parameters 'param1' and 'param2' have the following values:

type	param1	param2	comments
1 (FULL Domain)	IP address (i.e. 1.2.3.4)	not needed	If OK, Postmaster account added, with hostmaster, root as aliases
2 (POP Domain)	not needed	not needed	If OK, 'domain_name' account added
4 (Robot Domain)	not needed	not needed	Robot account needs to be added later
8 (Virtual Domain)	Base domain name	account Postfix	If OK, Postmaster account added, with hostmaster, root as aliases

Status: Implemented.

GDomainDel

Delete the domain 'domain_name'.

HRESULT GDomainDel([in] BSTR domain_name, [in] BSTR transfer_user_name, [in] BSTR transfer_domain_name, [in] int delete_request_flag)

Status: Implemented.

GDomainGetVar

Get the current value of the variable 'variable_name' for the domain 'domain_name'

HRESULT GDomainGetVar([in] BSTR domain_name, [in] BSTR variable_name, [out,retval] BSTR *value)

Status: Implemented.

GDomainSetVar

Set the value of the variable 'variable_name' for the domain 'domain_name' to 'new_value'

HRESULT GDomainSetVar([in] BSTR domain_name, [in] BSTR variable_name, [in] BSTR value)

Status: Implemented.

GDomainGetDir

Get the base directory for the domain 'domain_name'.


```
HRESULT GDomainGetDir( [in] BSTR domain_name, [out, retval] BSTR *value )
```

Status: Not implemented.

GDomainList

Return a list of domains. The entries are separated by ' '

```
HRESULT GDomainList( [out,retval] BSTR * value )
```

Status: implemented.

Global Variable Functions

GGlobalGetVar

```
HRESULT GGlobalGetVar( [in] BSTR variable_name, [out,retval] BSTR *value )
```

Status: Implemented.

GGlobalSetVar

```
HRESULT GGlobalSetVar( [in] BSTR variable_name, [in] BSTR value )
```

Status: Implemented.

Message Queue Functions

GQueueGet

Return a list of queues. The entries are separated by ' '

```
HRESULT GQueueGet( [out,retval] BSTR *value )
```

Status: Not implemented.

GQueueStart

Start post queue 'queue'. The mail will be sent using the appropriate postservers entry

```
HRESULT GQueueStart( [in] BSTR queue )
```

Status: Implemented.

Message Functions

GMessageOpen

This creates a basic message object with the following details

- From: source
- To: destination
- Subject: subject

```
HRESULT GMessageOpen( [in] BSTR destination, [in] BSTR source, [in] BSTR  
subject, [out,retval] GMessageHandle *handle)
```

The email address in the 'From:' field is used as the 'MAIL From' address for the SMTP protocol.

The email address in the 'To:' field is not used. Recipient addressing is handled by explicit calls to GMessageAddRecipient described below.

The returned message handle 'handle' should be used when performing further operations on the message and should be released by a call to MessageClose on completion.

Status: Implemented.

GMessageAppend

This appends the message text in 'text' to the end of the message identified by 'handle'.

```
HRESULT GMessageAppend( [in] GMessageHandle handle, [in] BSTR text )
```

The data in text should be in lines separated by carriage-return / line-feed (CRLF). The text will be wrapped automatically.

Status: Implemented.

GMessageSetHeader

This allows a message header line to be created, eg if clause is 'X-MyHeader' and text is 'my header text', the result will be a line

X-MyHeader: my header text

in the message header. Note the code automatically inserts the ':' at the end of the header field name.

If you intend to CC the message to others, then 'clause' would be 'Cc' and 'text' would be a semicolon separated list of email addresses.

```
HRESULT GMessageSetHeader( [in] GMessageHandle handle, [in] BSTR  
clause, [in] BSTR text )
```

Status: Implemented.

GMessageAttach

This attaches a file specified by 'filename' to the message. The file will be attached as a MIME base64 encoded attachment. The filename passed must be fully specified.

```
HRESULT GMessageAttach( [in] GMessageHandle handle, [in] BSTR filename  
)
```

Status: Implemented.

GMessageAddRecipient

The routine adds a RCPT clause for use in the SMTP protocol.

Note if the address specified is not given in the 'destination' field of the call to MessageOpen or in the list of addresses specified for a 'Cc' line, the address will effectively be a 'Bcc' address.

```
HRESULT GMessageAddRecipient( [in] GMessageHandle handle, [in] BSTR
address )
```

Status: Implemented.

GMessageClose

This is called when the message construction is complete. It calls code to compose the message and place it in the outbox for delivery by NTMail's SMTP server. The 'closure'flag' controls the action

- 0 - delete the message without sending it
- 1 - send immediately.
- 2 - place the message in the out directory.

Message resources such as memory and temporary files are then released.

```
HRESULT GMessageClose( [in] GMessageHandle handle, [in] GClosureFlag
closure_flag )
```

Status: Implemented.

Example for Message Functions

Send a message from "me@my.company.com" to "you@your.comany.com" with carbon copies to "him@your.company.com" and "her@your.company.com" and a blind carbon copy to "bcc@your.company.com". The message contains a MIME text section and a MIME attachment.

```
GMessageOpen( "you@your.company.com", "me@my.comany.com", "test
message", &handle );

GMessageAppend( handle, "one line\n\na second line\n\nfinal line" );

GMessageSetHeader( handle, "Cc", "him@your.company.com;
her@your.company.com" );

GMessageAttach( handle, "d:\path\to\file.doc" );

GMessageAddRecipient( handle, "you@your.comany.com" );
GMessageAddRecipient( handle, "him@your.company.com" );
GMessageAddRecipient( handle, "her@your.company.com" );

// add a BCC recipient
GMessageAddRecipient( handle, "bcc@your.company.com" );

GMessageClose( handle, 1 );
```

23.12 Monitoring Software

Both Watchmail and NTMail Watch can be used to monitor the performance of a Gordano server. Watchmail can be run on the

server allowing you to connect via telnet and monitor the post activity, while MailMon can be run from any workstation on the network to watch the activity of each of the Gordano services. Watchmail requires to be installed as a service on the Gordano server and can thereafter be accessed by telnetting to port 10010 on the server.

23.13 WatchMail

This is a statistics program for Gordano servers - It has been called WatchMail because that is what it does. It will give you information about the messages being posted out onto the Internet.

Install

To install it, put it into your chosen directory and type:

```
Watchmail -i
```

To start it type:

```
net start watchmail
```

Uninstall

To uninstall the service type:

```
net stop watchmail
```

```
watchmail -u
```

The Output

To find the stats, log into port 10010 on the machine running the WatchMail service using telnet. In greater details:

1. Start the telnet program by typing:

```
start telnet
```

2. Select the menu item Connect -> Remote System.

3. Enter the name of the machine running WatchMail as "host Name", enter 10010 for port and leave the TermType as "vt100". Press [CONNECT].

After a short pause you should see information run down the display with the following format:

Time	Wait- ing	Sent	Prog- ress	Bytes	Data	Av B/s
18:02:08	1	3342	0	8363193	0	0.000
18:02:18	349	3342	16	8363193	0	0.000
18:02:28	349	3342	16	8376010	12817	42.723
18:02:38	335	3356	16	8385397	9387	74.013
18:02:48	325	3366	16	8397882	12485	115.630

Every 10 seconds, these stats will be displayed.

- **Time:** Time the measurement of activity was take. Nominally every ten seconds (this depends upon how busy the Windows NT machine is).
- **Waiting:** Number of messages currently queued internally by the POST service.
- **Sent:** Number of messages sent to date (since POST was restarted).
- **Progress:** Number of threads currently actively sending out email.
- **Byte:** Number of bytes of mail data sent.
- **delta:** Number of bytes sent over the last 10 seconds.
- **Av B/s:** A rolling average of the number of bytes sent per second by the post server. This an average over five minutes.

SECURITY WARNING!

This service will allow up to ten people to collect stats at the same time. If this service is running on your machine, and someone knows, they, too, can get this information. There is no logging.

23.14 GMS Watch

The GMS Watch (MailMon) utility can be used to monitor each of the services in a Gordano installation (see "Mailbox Monitor" on page 36).

24 Porting/Migration Tools

There are a number of utilities available to make it easier for you to migrate user information from other applications into a format usable by Gordano products. These utilities can be downloaded from www.Gordano.com or [ftp.Gordano.com](ftp://Gordano.com). This section will cover the porting tools for:

- Exchange
- Linux/Solaris (Sendmail/Unix)

In addition to these tools, Gordano's patented Autoport technology can be used to automatically port users and messages from any incumbent mail server over to GMS. The use of Autoport is fully explained in the GMS Administration Guide.

24.1 Exchange

Download and install the Exchange utility. It can then be used to retrieve all user information from an Exchange Server and save it to a format usable by Gordano products.

Usage:

`exchange [-b baseDir] [-d domain] [-f filename] [-v] where`

- b 'baseDir' specifies the location of the converted files
- d Exchange handles mail for domain 'domain'
- f use specified file as source of Exchange info
- v enables verbose mode

24.2 Linux/Solaris

There are two utilities available one for Sendmail and the other for Unix.

SendMail

The conversion utility will convert information from `sendmail.cf`, `/etc/aliases` (or `/etc/mail/aliases`) and `/etc/passwd` (along with `shadow` if supported) into a standard Gordano configuration file. This can be loaded into an existing Gordano setup. This may be a default demo installation on a new machine or a full version with no accounts configured on it yet. The utility will also convert the sendmail mailboxes to Gordano format. The migration process will produce a set of text files:

- Some intermediate files if the utility is run in verbose mode
- `Setup.txt` - containing the converted sendmail configuration.

- A hierarchy of converted mailboxes named in the format 'domain/Users/name/InBox.mbx' which contain the converted mailbox for user@domain (if the source mailbox existed on the UNIX box).

eg for 'sendmail_to_ntmail -b /convert -d company.com -v -l you' you would have the following directory structure

```

/convert
/convert/aliases.txt      <----- only if -v
                           specified
/convert/accounts.txt     <----- only if -v
                           specified
/convert/setup.txt        <----- passwords
                           masked if -m specified
/convert/company.com
/convert/company.com/Users
/convert/company.com/Users/fred
....
/convert/company.com/Users/joe
/convert/company.com/Users/joe/InBox.mbx <--- in NTMail format
....
/convert/company.com/Users/list
/convert/company.com/Users/list/members.txt <----- exists only if -l
specified
....
/convert/company.com/Users/root

```

If your sendmail was configured to handle mail for 'company.com' with users 'fred', 'joe', 'list' and 'root' where only 'joe' has a mailbox in the sendmail spool dir and 'list' is a list account with members now listed in members.txt

If you have any difficulties or would like further assistance with this process contact support@gordano.com.

Usage:

```
sendmail_to_ntmail -b baseDir -d domain [-l] [-m] [-v]
```

where:

'baseDir' specifies the location of the converted files

SendMail handles mail for domain 'domain'

'-l' convert ':include:' entries to NTMail lists

'-m' masks password text with 'x' (for testing only)

'-v' enables verbose mode

Unix

This conversion utility will convert info in /etc/passwd (along with shadow if supported) into a standard Gordano configuration file.

This can be loaded into an existing Gordano setup. This may be a default demo installation on a new machine or a full version with no accounts configured on it yet. The migration process will produce a set of text files:

- Some intermediate files if the utility is run in verbose mode
- Setup.txt - containing the converted sendmail configuration.

eg for 'users_to_ntmail -b /convert -d company.com -v' you would have the following directory structure:

```
/convert
/convert/accounts.txt <----- only if -v specified
/convert/setup.txt   <----- passwords masked if -m specified
```

Usage:

```
users_to_ntmail -b baseDir -d domain [-m] [-v]
```

where:

'baseDir' specifies the location of the converted files

UNIX system handles mail for domain 'domain'

'-m' masks password text with 'x' (for testing only)

'-v' enables verbose mode

25 Log Contents

This section gives background information on log files, which are saved to the directory GORDANO\LOGS.

At a general level, Gordano products have two types of log:

- Message content — hold the contents of messages as a text file.
- Delivery/acceptance — can be used to prove the e-mail has been received or delivered.

25.1 Log Levels

The following logging parameters can be set using Logs>Transaction level (see “Managing Logs” in the *GMS Administrator’s Guide*):

- Progress — logs start, send etc. of a send/receive.
- Statistics — for every message that passes through the server, logs the date and time, the sender’s address, the receiver’s address and the message size in KB.
- Returns — records all messages that are returned to the sender.
- Protocol — records all messages sent/received.
- GMS Anti-Spam — statistics on e-mail rejected by GMS Anti-Spam.
- DNS — DNS checks.
- Failures — failed delivery of messages.

By default only statistics and failure information is collected, but if you have problems with a service, turn on its other log parameters too. Once the problem is resolved, disable these again or log files will grow very large.

The excerpts from logs in the following section show the format of entries.

25.2 Log file contents

Each line of a log file has the following parameters:

- Service.
- Date & time.
- Type of entry — S = statistics, F = a failure, I = information, A = action, P = protocol, Z = parser (all details).
- Four digit code for use by Gordano Ltd. The following log excerpts show examples.
- Thread id (usually present) is the number of the thread that made the log entry

The remainder of the line describes the entry in more detail.

An example section of SMTP log file is shown below:

```
SMTP 23 Nov 99 10:39:24 I 0016 0 GMS Anti-Spam Key: enabled
SMTP 23 Nov 99 10:42:57 A 3159 1 Connection from[194.5.1.2] [194.5.1.2]
SMTP 23 Nov 99 10:42:57 A 0000 1 220 dean.test.dom NTMail(v4.01.0010/
  NU7173.00.acd975a5) ready for ESMTP transfer
SMTP 23 Nov 99 10:42:57 A 3084 1 EHLO dean.test.dom
SMTP 23 Nov 99 10:42:57 A 0000 1 250-dean.test.dom Hello dean.test.dom
  [194.205.1.254], pleased to meet you
SMTP 23 Nov 99 10:42:57 A 0000 1 250-VRFY
SMTP 23 Nov 99 10:42:57 A 0000 1 250-SIZE 10485760
SMTP 23 Nov 99 10:42:57 A 0000 1 250-ENHANCEDSTATUS CODES
SMTP 23 Nov 99 10:42:57 A 0000 1 250-8BITMIME
SMTP 23 Nov 99 10:42:57 A 0000 1 250-ETRN
SMTP 23 Nov 99 10:42:57 A 0000 1 250-DSN
SMTP 23 Nov 99 10:42:57 A 0000 1 250 HELP
SMTP 23 Nov 99 10:42:57 A 3084 1 MAIL From:<joe@test.dom> SIZE=835
SMTP 23 Nov 99 10:42:57 A 0000 1 250 2.5.0 OK.
SMTP 23 Nov 99 10:42:57 A 3084 1 RCPT To:<john@test.dom>
```

Here is a short section of an IMAP log:

```
IMAP 8 Dec 99 09:57:58 S 6070 Version 4.01.0010
IMAP 8 Dec 99 10:03:31 S 6070 Version 4.01.0010
```

Here is a short section of a LIST log:

```
LIST 9 Dec 99 15:56:59 F 0016 0 GMS Anti-Spam Key: disabled
LIST 9 Dec 99 15:56:59 F 0292 0 LocalIP - 194.205.1.*
```

Here is a short section of a POP log:

```
POP 6 Jan 99 09:18:44 S 1018 Version 4.01.0010
POP 6 Jan 99 09:47:22 F 0016 0 GMS Anti-Spam Key: disabled
```

Here is a short section of a POST log:

```
POST 11 Dec 99 15:03:48 S 2063 Version 4.01.0010
POST 11 Dec 99 15:03:55 F 2113 1 not deliverable or returnable 72 hours
  C:\NTMAIL\Out\ntmail.dom\bill deleted
POST 11 Dec 99 15:34:59 F 0016 0 GMS Anti-Spam Key: disabled
```

Here is a short section of a WWW log:

```
WWW 11 Dec 99 15:03:50 F 0016 0 GMS Anti-Spam Key: disabled
WWW 11 Dec 99 15:03:50 S 7057 Version 4.01.0010
WWW 11 Dec 99 15:34:46 S 7030 0, MML, 127.0.0.1,127.0.0.1, 5224, 229,
  761,200, GET, \, \index.mml, "Mozilla/4.01 [en] WinNT; I)", " "
```

25.3 Log File names

Each filename starts with two letters defining its type:

- CF for CFX tag
- DU for Dial Up
- FX for Fax DLL.
- IM for IMAP.
- LI for LIST.
- PO for POP.
- PT for POST.
- SL for Spam.
- SM for SMTP.
- WM for WebMail
- WW for WWW.
- VS for Virus Scanner
- MG for eSarah

There are also files based on DLL names. The first two characters in the name of such a file will depend on the DLL.

The next six digits of each filename contain the date of the log file in yymmdd format (the log files roll over at midnight). The extension is always '.log'. For example, a POP log produced on January 4th 1999, would be named PO990104.log.



If you have specified the parameter 'LogDir' the log files will be written to that directory in preference to the default directory.

25.4 Logging Bit Map

Use this table to work out logging levels by summing any of the following components:

Bit	Value	Logging	Description
0	1	LOG_START	Logs start and stop events
1	2	LOG_FAILURE	Logs all failures to send
2	4	LOG_PROGRESS	Logs start, send etc of a send/receive
3	8	LOG_RETURN	Logs all messages that are returned
4	16	LOG_STATS	Logs details of message size for each sent
5	32	LOG_HEADER	Logs the header of each message sent
6	64	LOG_CONTENT	Logs entire contents of each message
7	128	LOG_DEBUG	Too much stuff printed!
8	256	LOG_PROTOCOL	All messages sent/received
9	512	LOG_DNS	DNS checks
10	1024	LOG_ALERT	For redirection of incoming messages
11	2048	LOG_TIME	Logs duration of events
12	4096	LOG_SPAM	Logs all spam related events
13	8192	LOG_ERRORS	Logs errors
14	16384	LOG_DATA	Logs all data
15	32768	LOG_QUEUE	Logs post queues
16	65536	LOG_LIB	For internal use
17	131072	LOG_JUCE	Logs GMS Anti-Spam events
18	262144	LOG_IPC	Logs inter process communication
19	524288	LOG_TEST	For internal use
20	1048576	LOG_PARSER	Logs parser information.

25.5 Stats Log Contents

When enabled, Gordano products log the following messages in response to certain events:

Event	Logged Message
Auto Responder	<id> autor@<domain> <from> <to>
Incoming mail	<id> mail <POP-name> <to> <from> <size> bytes
Outgoing Fax	<id> fax user <from> <telephone> <num pages>
Outgoing Fax	<id> fax access <from> <telephone> <num pages> <access>
Outgoing mail	<id> mail <ref> <host> "<from>" "<to>" <size> bytes
Receipt Stat	<id> receipt <from> <to> "<subject>"
Robot	<id> robot <from> <to> <lines> "<subject>"
WWW	<id> www <from> <address>

The fields in the above table have the following meanings:

Field	Description
access	The costing-centre code for the FAX number.
domain	The SMTP domain name.
from	Where the mail claims it is from.
hostname	The IP address or hostname of the machine that sent this message here.
id	The number of the thread.
lines	The number of lines in the message.
num pages	How many FAX pages were created.
size, length	The message size in bytes.
subject	The subject of the message.
telephone	The telephone number the mail was sent to.
to	The mail's destination.

26 Error Messages

This section supplies information on SMTP error codes. There are several sets of codes:

- DNS Resolver error codes.
- POST log errors.
- SMTP error codes, the standard set.
- Three-digit SMTP error codes.
- Enhanced error codes as defined in RFC 1893.

The codes are listed in the tables which follow.

26.1 DNS Resolver Errors

These errors may be produced by DNS trying to resolve domain names:

Code	Meaning
0	Routine successful.
-1	Generic routine failure.
-2	No error available (initial value).
-3	Invalid thread block.
-4	Memory allocation failed.
-5	Corrupt input message (illegal access).
-6	Insufficient space in buffer for output data.
-7	Item not found.
-8	Bad parameters given to routine.
-9	Recoverable error.
-10	Non-existent domain.
-11	Reply timed out
-12	Winsock error (WSAGetLastError should be applicable).
-13	Socket operation with no socket.
-14	Reply truncated.
-15	System error (GetLastError should be applicable).

These codes are produced by ParseReply:

Code	Meaning
-16	Reply corrupt or invalid.
-17	Caller's buffer too small for all mail exchange records.
-18	Too many nested aliases.
-19	Class not supported.
-20	Last error.

26.2 Error Codes in Logs

Error codes in logs appear as shown in this short section of a POST log:

```
POST 11 Dec 98 15:03:48 S 2063 Version 4.01.0010
POST 11 Dec 98 15:03:55 F 2113 1 not deliverable or returnable 72 hours
C:\NTMAIL\Out\CompanyA.dom\bill deleted
```

The number 2113 in the second line is an error code.

PT Log codes

The following errors may appear in POST log files:

Code	Meaning
2001	<p>The remote machine has an illegal Gordano licence. Please contact Gordano Ltd. with details of the domain name so that action can be taken.</p> <p>This message is only displayed as a result of the remote server's logon message.</p>
2002	<p>Remote server cannot accept connections for delivery of e-mail at the moment. Please try sending e-mail at a later time. The full response from the remote server is given in the remainder of the log message.</p> <p>This message is only displayed as a result of the remote server's logon message.</p>
2004	<p>Remote server cannot accept connections for delivery of e-mail at any time. Please return the e-mail to sender. The full response from the remote server is given in the remainder of the log message.</p> <p>This message is only displayed as a result of the remote server's logon message.</p>
2008	<p>The remote server cannot accept connections for delivery of e-mail at any time. Please return the e-mail to sender. The full response from the remote server is given in the remainder of the log message.</p> <p>This message is only displayed as a result of a Gordano server saying HELO to the remote server. Rejection at this stage suggests that some form of anti-spam filter has been applied in order to prevent e-mail being delivered from your mail server.</p>
2015	<p>The remote mail server has indicated that it is not possible to deliver e-mail to the destination account at this time. If we try later, the account may be available for delivery. The reason the account is not available is given in the remainder of the log message.</p> <p>This message only occurs in response to an RCPT clause from a Gordano server to the remote server.</p>

- 2016 The remote mail server has indicated that it will never be possible to deliver e-mail to this account (or, in the case of multiple RCPT clauses, that some other limit has been reached). If this e-mail account does not exist, the e-mail message should be returned to sender. The reason the account is not available is given in the remainder of the log message.
- This message only occurs in response to an RCPT clause from a Gordano server to the remote server.
- 2030 The remote mail server has indicated that this message could not be accepted at this time. Delivery should be attempted later. The reason the message could not be delivered is given in the remainder of the log message.
- This message only occurs in response to the completion of sending the mail message from a Gordano server to the remote server.
- 2031 The remote mail server has indicated that this message can never be accepted. The message should be returned to the sender. The reason the message could not be delivered is given in the remainder of the log message.
- This message only occurs in response to the completion of transmission of the mail message from a Gordano server to the remote server.
- 2040 For some reason, the Gordano server could not open a message on the local disk to send it to a remote mail server. This may be due to a lock being held by another application, or someone deleting the file.
- 2099 An attempt to get the IP address of a host using its name failed. This may occur if there is no DNS assigned to this domain (that is, the domain name does not exist) or if neither the primary or secondary name servers could be reached.
- 2100 Could not resolve the domain name that the e-mail message should be sent to. The e-mail message will be returned with a message indicating that the domain name should be checked.
- 2106 Attempted to connect to the remote mail server, but no connection could be made. This may occur because there is no route to the remote mail server (e.g. the network is down, network card failed, modem not connected, etc.) or the remote mail server is unavailable (e.g. it has been taken down for maintenance). Try to resend the e-mail later.
- 2111 The Gordano server has tried to deliver this e-mail message for a given number of hours and has still not succeeded. The message will be returned to the sender.
- 2112 The Gordano server has tried to deliver this e-mail message for a given number of hours and has still not succeeded. The message will be returned to the sender.

- 2121 Part way through the transaction sequence sending e-mail to the remote host, the connection was lost. This might be due to a network failure, the remote machine failing, etc. The Gordano server will try to send the e-mail message later.
- Note: If this failure occurs after the "." line of the message, the Gordano server must accept that the message was NOT delivered, though the remote server may consider that the message was delivered. In this case, the Gordano server will (correctly) send the message again since it never received a confirmation from the remote server that the message was delivered successfully.
- 2124 Could not send an e-mail message so it has been rescheduled for attempted delivery later. At this time, The Gordano server will try to deliver the e-mail a second time.

Three-digit SMTP error codes in logs

These error codes are three digits long. There are four types of error code, defined by their first digit:

- 2 — success reported.
- 3 — for information only.
- 4 — temporary failure. The message is valid and its reason for failure is transient, so a future attempt to send may succeed.
- 5 — permanent failure, unlikely to be resolved by sending the message again in its current form. Some change is needed for successful delivery.

This table lists the SMTP codes.

Code	Meaning
211	System status or system help reply.
214	Help message: explains how to use the receiver, or the meaning of a particular non-standard command. This reply is useful only to human users.
220	<domain> Service ready.
221	<domain> Service closing transmission channel.
250	Requested mail action okay, completed.
251	User not local; will forward to <forward-path>.
283	The Gordano server is configured so that messages aren't being saved. The From and To address are taken from the e-mail message header.
354	Start mail input; end with <CRLF>.<CRLF>.
421	<domain> Service not available, closing transmission channel (This may be a reply to any command if the service knows it must shut down.)
450	Requested action not taken: mailbox unavailable (for example, mailbox busy).
451	Requested action aborted: local error in processing.
452	Requested action not taken: insufficient system storage.
500	Syntax error, command unrecognized. This includes errors such as command line too long.
501	Syntax error in parameters or arguments.
502	Command not implemented.
503	Bad sequence of commands.

Code	Meaning
504	Command parameter not implemented.
550	Requested action not taken: mailbox unavailable (for example, mailbox not found, no access).
551	User not local; please try <forward-path>.
552	Requested mail action aborted: exceeded storage allocation.
553	Requested action not taken: mailbox name not allowed. For example, the mailbox syntax may be incorrect.
554	Transaction failed.

Enhanced SMTP error codes

These codes are appended to the above set, supplying additional information. The first digit, marked "X" in the table, can be any of the numbers given above — 2, 3, 4 or 5. The other two are as shown below:

Code	Meaning
X.1.0	Other address status.
X.1.1	Bad destination mailbox address.
X.1.2	Bad destination system address.
X.1.3	Bad destination mailbox address syntax.
X.1.4	Destination mailbox address ambiguous.
X.1.5	Destination address valid.
X.1.6	Destination mailbox has moved, No forwarding address.
X.1.7	Bad sender's mailbox address syntax.
X.1.8	Bad sender's system address.
X.2.0	Other or undefined mailbox status.
X.2.1	Mailbox disabled, not accepting messages.
X.2.2	Mailbox full.
X.2.3	Message length exceeds administrative limit.
X.2.4	Mailing list expansion problem.
X.3.0	Other or undefined mail system status.
X.3.1	Mail system full.
X.3.2	System not accepting network messages.
X.3.3	System not capable of selected features.
X.3.4	Message too big for system.
X.4.0	Other or undefined network or routing status.

Code	Meaning
X.1.0	Other address status.
X.4.1	No answer from host.
X.4.2	Bad connection.
X.4.3	Directory server failure.
X.4.4	Unable to route.
X.4.5	Mail system congestion.
X.4.6	Routing loop detected.
X.4.7	Delivery time expired.
X.5.0	Other or undefined protocol status.
X.5.1	Invalid command.
X.5.2	Syntax error.
X.5.3	Too many recipients.
X.5.4	Invalid command arguments.
X.5.5	Wrong protocol version.
X.6.0	Other or undefined media error.
X.6.1	Media not supported.
X.6.2	Conversion required and prohibited.
X.6.3	Conversion required but not supported.
X.6.4	Conversion with loss performed.
X.6.5	Conversion Failed.
X.7.0	Other or undefined security status.
X.7.1	Delivery not authorized, message refused.
X.7.2	Mailing list expansion prohibited.
X.7.3	Security conversion required but not possible.
X.7.4	Security features not supported.
X.7.5	Cryptographic failure.
X.7.6	Cryptographic algorithm not supported.
X.7.7	Message integrity failure.

26.3 Four-digit SMTP error codes

This table shows other SMTP errors:

Code	Meaning
3035	If the unknown user action for a domain is set to "N" (i.e. deliver mail to no-one), this message is written to the log when the e-mail is deleted by the SMTP server.
3053	The Gordano server was waiting for a message from a robot and the robot either did not send the message or did not send all of the message.
3055	The Gordano server tried to deliver a message to a robot. Unfortunately, the executable for the robot did not exist. The name of the executable the Gordano server tried to use is in the log message. Check that the executable is available (for example, use the fully qualified path) and that the file's privileges allow it to be executed.
3060	SMTP could not close the stream writing a message to a robot. This may be because the robot did not accept the message. The standard Windows NT error code is returned in brackets.
3089	Part way through the transaction sequence receiving e-mail from the remote host, the connection was lost. This might be due to a network or remote host failure, etc.
3130	The thread which is accepting messages from the given domain has stopped receiving any TCP/IP communications (either to or from the remote machine). No Windows NT socket error was generated so the SMTP server terminated the thread and restarted it so that it is available for another connection. Note: some resources (memory, handles, etc.) may have been lost.
3182	The remote mail server tried to deliver a message to the same address twice. The second attempt will be ignored.
3200	The service is being closed down or the number of threads the service should use for accepting connections must be reduced (the user has changed the configuration). The message logs the fact that one of the thread's activities has been stopped in mid flow.
3209	The remote server is using ESTMP to deliver an e-mail message to the Gordano server and specified the size of the message. However, the message delivered to the Gordano server was not the quoted size. The Gordano server will continue to treat the e-mail as normal. Please contact the owner of the remote mail server to inform them that their mail server is issuing the incorrect SIZE parameter when sending mail to remote servers.

Code	Meaning
3301	<p>The Gordano server has rejected an e-mail message for some reason at the "." phase of e-mail delivery from a remote host. The last parameter (in brackets) gives the reason that the message failed. It may be one of the following values:</p> <ul style="list-style-type: none">1 = A word in the message is in the bad word file.2 = An incoming message size is larger than the maximum message size for the system, domain or user.3 = An outgoing message size is larger than the maximum message size for the system, domain or user.4 = The End of Message MML script indicated that this message should fail.5 = The Domain MML script indicated that this message should fail. This message occurs if a filter is activated.6 = The User MML script indicated that this message should fail.7 = The Incoming MML script indicated that this message should fail.8 = The SMTP DLL indicated that this message should fail.9 = The Incoming DLL indicated that this message should fail.10 = The ProcessMessage function said that this message should fail.
4090	<p>There was a problem reading the member file. The end of the member file for a list was reached unexpectedly.</p>

27 Security Check

The following outlines some steps you can take to help ensure that your Gordano server is as secure as possible. The settings might not be appropriate to all situations so please bear in mind your own requirements when setting them.

27.1 Disallowing Relay

Whether to allow or disallow relay is a choice that you must make yourself, however if you choose to allow relay through your server then you need to be prepared to put up with a lot of flack from other users of the Internet.

Gordano products come with relay disabled by default and there should be no need to change this, unless you have a particular situation which requires it.

If you do wish to change it you can do so from the Security > Relay tab. By default you will find that the option "Disallow relay, all RCPT clauses must be local" option is checked, this setting depends on the LocalIP entry to determine who is able to relay through your server.

If you have GMS Anti-Spam installed you will find a third option on this page, "Disallow relay, but allow mail where MAIL clause or more than one RCPT clause is local" this is not as secure as the default option as it will allow anyone claiming to be from your domain to relay mail through the server. This can be restricted by setting the IP address ranges that are allowed to send as if from your domain under the Local Clients option.

Another option for those with GMS Anti-Spam installed is "Allow relay for authenticated POP/IMAP users". With this selected users who log on with their user password to check their mailbox will then be able to relay. There is also a timeout that can be set for this which if reached will require the user to logon again. This is particularly useful if you have roaming users.

27.2 LocalIP

The LocalIP entry works in conjunction with the default anti-relay option above, it is set from the Security > LocalIP page. Only senders that come from one of the IP addresses contained in the ranges specified here will be allowed to relay mail through your server, providing that the default anti-relay setting is used. Note you can use the conventional wildcards to specify IP addresses e.g 123.123.123.*

27.3 Local Clients

Only available with GMS Anti-Spam this setting determines which IP address ranges are allowed to send mail purporting to come from your domain. If you set an IP address range here and someone from an IP address outside the specified range tries to send an email with a From address of your domain it will be rejected.

27.4 UnknownUserAction

This setting determines what happens to mail that arrives at your server for a user that does not exist. Some relay testing organisations will test for relay by sending mail to an account on your server that looks like

user%theirdomain.com@yourdomain.com

This is an old and unsupported (by Gordano products) method of relaying mail through a server. If your server is set to fail unknown users to a local account this mail will appear to have been accepted to these organisations and you will appear to fail one of their tests. Although this should not result in you being blacklisted by them as they will never receive the mail back. The simplest method of overcoming this is to set the UnknownUserAction for your domain to reject mail to users that are not recognised as being local.

Another option is to set up a rule in your redirect file that rejects all mail that contains the % symbol in the user portion of the name with a rule similar to

```
* * *%* @* * T F "500 Unknown User"
```

27.5 Disable VRFY

The SMTP VRFY command is sometimes used to attempt to harvest addresses from your mail server. It has three options as described below:

1. Disable VRFY - the VRFY command will not return any information at all.
2. Allow VRFY - allows the verification of a single address. For instance a remote server could connect to your SMTP server and

issue a command such as "VRFY smith", if smith is a valid local account the SMTP service will respond positively as in

```
250 smith@yourdomain.com
```

If the account is not valid it will respond with:

```
557 string does not match anything
```

Remote servers may have a valid reason for issuing this command, for example to ensure a user does exist prior to attempting to send mail to them.

3. Allow WildCard VRFY - this is similar to 2 above but in this instance if a remote server connects to you they may issue the command "VRFY *" the smtp server will return a list of all valid accounts on your machine.

27.6 Restrict Access to Config GUI

As an added security measure you can determine ranges of IP addresses that are allowed to access the Gordano Configuration Server. If all configuration access is from a single workstation then you need only enter the IP address of this workstation. This can be set from the Security > Config Access page. This is set on a per domain basis, users from IP addresses not covered by the range entered here will not be able to access the Configuration Server.

27.7 Password Policies

Password policies are important to ensure that users logging on to your system are who they say they are, it is relatively easy for potential intruders to discover usernames on your system but much more difficult for them to discover passwords. The stricter your password policy is the more difficult it will be for someone to break into the system.

Gordano provides a set of rules to help enforce a suitable password policy. This includes ensuring that passwords contain a suitable mix of letters, digits and symbols, as well as requiring that passwords are a minimum length. It is also possible to state a period that passwords are expired at, users will be sent a mail message prior to

expiry to provide them with the opportunity to change their passwords.



*Passwords cannot have * as their last digit.*

Index

Numerics

8BitMIME ESMTP command
description 198
enabling 68

A

AcceptCommand 147
AcceptVals 147
AcceptWords 128
access 31
Access parameters - values 127
AccessFile 107
AccessIP 60
AccessIPDenied 60
AccessRights 107
Account
maximum number in domain 102
maximum size 73, 109
NULL 76, 87, 103
Addresses
fixing bad 12
AddUser.exe 27
AdminBackgroundBackdrop 60, 99
Administration Gizmos 223
Administrator 31, 60
AdminMenuTitleColour 60, 99
AllPActivate 177
AllPAverageMultiplier 177
AllPCheckActive 177
AllPFailureMes 177
AllPRequiredSamples 178
AllPRunningAverageMin 178
AllMAILActivate 178
AllMAILAverageMultiplier 178
AllMAILCheckActive 178
AllMAILFailureMes 178
AllMAILRequiredSamples 178
AllMAILRunningAverageMin 178
AllRCPTActivate 178
AllRCPTAverageMultiplier 178
AllRCPTCheckActive 178
AllRCPTFailure 178
AllRCPTRequiredSamples 179
AllRCPTRunningAverageMin 179
AlertMaxEntries 60
AlertPort 61
AlertsRequired 61
alias 28, 29
Aliases 108
AliasOf 108
AllowAdminGuiServer 61
AllowAPOPlogin 61
AllowCollaborationServer 61
AllowDailyStats 61
AllowEditScript 108
AllowEditScripts 61

AllowEditVariables 61
AllowedRCPTLimit 62, 179
AllowedRCPTLimitMes 62, 179
AllowedSenderIP 99, 182
AllowedSenderIPMes 99, 182
AllowESMTP 62
AllowEtrn 62
AllowFaxDLL 62
AllowFingerServer 62
AllowHelp 62
AllowIMServer 63
AllowLookup 63, 99, 108
AllowManuals 63
AllowNTInfo 63
AllowNTPassword 63
AllowPasswordServer 63
AllowProxyServer 63
AllowReturnReceipt 63
AllowReverseProxyServer 64
AllowScriptServer 64
AllowSSLProxyServer 64
AllowUserGUIServer 64
AllowVRFY 64
AllowWebMailGUIServer 64, 153
AllowWWWServer 64
AllowXTND 64
APOP login 61
Archive 283
ArchiveBackgroundColor 128
ArchiveBaseURL 128
ArchiveGifURL 128
ArchivePassword 128
ArchiveToHTML 128
ArchiveToMML 129
ArchiveWallpaperURL 129
ArchiveWWWDirectory 129
Auth ESMTP command
bypassing 100
description 198
enabling 68, 87
Authentication
using DLL 256
AutoPortHost 64
AutoResponder 108

B

Backup directory 193
Bad commands
number allowed 73
BadMes directory 73, 87, 195
BadMessageAction 129
BadMessageAddress 129
BadMessageLimit 129
BadMessageNumber 113
BadMessageResetFrequency 129
BadMessageResetStartDate 129

Bandwidth
 for POP users 78, 110
 for POST users 80
BandwidthUsed 130
BannedFromAddressFile 130
BannedMessageAction 130
BannedMessageAddress 130
BannedSubjectFile 130
BaseDir 100, 116
BaseDir directory 191
BaseDomain 100
Bin directory 193
BinStats 116
Bit Map
 Logging 304
Blackberry 70
BounceAttachmentPosts 130
Break 221
BypassAuthIP 100
BypassWordCheckIP 182

C

Cache
 DNS 65
 flushing DNS 66
 maximum files in 76
 proxy 84
 purging DNS 17
 purging proxy cache 84
Cascading Style Sheets 234
CheckMXOnJoin 130
CmdCmdAsSubject 130
CNAME 11
CName
 depth 66
 lookups 11
ColdFusion tags 273
Collaboration
 Parameters 170
Collaboration Server 216
CollaborationLogonDelay 170
CollaborationLogonTimeout 171
CollaborationMaxConnections 171
CollaborationMaxConnectionsPerlIP 170
CollaborationPort 171
CollaborationReceiveThreads 171
CollaborationSessionTimeout 171
CollaborationSSLPort 171
CollaborationTcpBlockSize 65
CollaborationThreads 171
CollLog 171
COM 286
Command
 EHLO 88
 ETRN 199
 HELO 88
 maximum bad allowed 73
 Size 201
 STAT 91
 VRFY 201
CompanyFax 100
CompanyName 100
CompanyTel 100

Compose 220, 232
Compose Quiz 222
Config GUI
 Restrict access to 319
ConfirmExpires 131
conflict 32
Connectivity - testing 46
Container Name 216
CONTENT_LENGTH characters 238
CRLF
 enforcing use 11, 67
 required at end of line 3
 use by robot 238
CSS 236
Custom Themes 235

D

DailyCommands 131
DailyPosts 131
Data Access Account 217
DATAScript.mml 88
DataSourceName 131
DateConfirmSent 147
DateCreated 100, 108
DateJoined 148
DateLeft 148
DateTime Format 261
DBType 131
DefaultAck 131
DefaultDigestType 131
DefaultPersonality 158
DefaultShow 131
Delivery Status Notification
 enabling 68
 enforcing 3, 68
 ESMTP feature 198
DeliveryDLL 65
DeliveryScript 65
Detach 284
DetailsNumber 113
Dial-up
 postservers file for 211
Dial-up connection
 postservers file 82
DialupActions 65
DialupLog 65
DialupRequested 65
DigestSizeLimit 132
DigestTimeHour 132
DirAccess 132
DirAccessFile 132
Directory
 Backup 193
 BadMes 73, 195
 Bin 193
 DNS Cache 65
 for log files 95
 ListIn 193
 ListMan 193
 Log 74, 193, 301
 LogMes 195
 MesLog 193
 MML 193

- SpamMes 194
- Temp 194
- Users 195
- Web Proxy 93
- WWWBaseDir 105
- Directory structure 191
- DirPassword 132
- DirWords 132
- Disable VRFY 318
- DisableMemberLookup 132
- Disconnect Delay 85
- Display 132
- DisplaySupport 65
- DLL
 - available DLLs 241
 - External Folder Manager 260
 - log 270
 - log files 303
 - message parsing 270
 - parameter 108
 - reason for use 241
 - security 92
 - SMTP 248
 - transmission 241
 - user authentication 256
- DLLselected 108
- DNS
 - error codes 307
 - purging cache 11, 17
 - servers list 77
- DNSAliases 65
- DNSBLServerMode 180
- DNSBLServers 180
- DNSBLServersMes 180
- DNSCacheDir 65
- DNSCacheExpiry 66
- DNSCacheFlush 66
- DNSCacheSize 66
- DNSCacheZeroTTL 66
- DNSDomain 66
- DNSHost 66
- DNSMaxCNameDepth 66
- DNSMinNumDots 66
- DNSOptions 67
- DNSPort 67
- DNSQueryOptimise 67
- DNSServers 67
- DNSTimeClient 67
- DNSUseHOSTSFile 67
- DocumentStoreCapacity 158
- Domain 284
- domain 28, 29
- Domain robot 237
- Domain-specific parameters
 - listed 97
 - location 50
- DomainType 100
- DSN. See Delivery Status Notification
- DSNPassword 132
- DSNUser 132
- Dynamic IP addresses and ETRN 62

E

- EFMDLL 260
- EHLOScript.mml 88
- Email1 117
- Email2 117
- Email3 117
- EmailClient 189
- EmailClientCommand 189
- EmailClientWindowName 189
- EmailMode 189
- End of message 238
- EndCmdMsgStr 132
- EnforceCRLF 67
- EnforceDSN 68
- EnforceReturnPath 68
- EnforceRFC822Header 68
- Enhanced SMTP. See ESMTP
- EOMScript.mml 88
- Error Codes 262
- Error codes 307
- eSarah 285
- ESMTP
 - 8BitMIME 198
 - allowing commands 62
 - Auth 198
 - Delivery Status Notification 198
 - Enhanced Status Codes 199
 - ETRN 199
 - overview 197
 - Pipelining 200
 - Restart 200
 - Size 201
 - VRFY 201
 - XTND 201
- ESMTP8BitMIME 68
- ESMTPAuth 68
- ESMTPDeliveryStatus 68
- ESMTPEnhancedStatusCodes 68
- ESMTPPipeLining 68
- ESMTPRestart 69
- ESMTPSize 69
- ETRN
 - and dial-up connections 199
 - configuring 199
 - enabling 62
- Exchange 297
- Executable
 - parameter 108
- Extenal Folder Manager 260

F

- FailedPostAccess 132
- Fax 109
- FaxCompanyName 109
- FaxDLL 69
- FaxFont 109
- FAXHandler 101
- FaxNumber 114
- FAXPageLength 69
- FAXPageToLength 69
- FAXPageWidth 69
- FAXPriority 69

FaxTemplate 109
File
 appending POP entries 10
 DATA Script.mml 88
 EOM Script.mml 88
 error.htm 93
 error.mml 95
 Filter Script.mml 69
 footer 101
 HELO Script.mml 88
 index.mml 95
 log file types 303
 MAIL Script.mml 89
 postservers 211
 postservers.txt 81
 redirect 86, 208
 template 212
Filter 285
Filter Script 69
Filter Script.mml 69
Finger
 enabling server 62
 using command 48
FingerPort 70
Folder Close 267
Folder Commit 267
Folder Create 265
Folder Delete 265
Folder Exists 265
Folder Helper Functions 263
Folder List 265
Folder List Subscribed 266
Folder Locking 264
Folder Open 266
Folder Purge 267
Folder Rename 265
Folder Store 267
Folder Subscribe 266
Folder Update Notifications 268
FolderDisconnectIdle 70
FooterFile 101
ForDays 148
ForDefault 133
Forward 109
ForWords 133
FreeBusyAuth 172
FreeBusyAuthTime 172
FreeBusyLocalIPAllowed 172
FreeBusyPublish 172
FreeBusyPublishPeriod 172
FreeBusyPublishPrivate 172
Full to Virtual Domain 285
FullName 109
Functions 262

G

GetAccess 133
GetAccessFile 133
GetPassword 133
GetWords 133
Gizmos 219, 281
 Access Rights 223
 Ajax Requests 230

Archive 220
Content Helper Methods 230
Current Selection Values Helper
 Methods 227
Description 223
Dictionary 220
Display Name 223
Empty Folder 220
External Scripts 223
Functionality 222
General Helper Methods 226
Google Translate 220
Greasemonkey helpers 231
Helper Methods 226
Hide buttons 220
Installation 224
Mark Mail 220
Menus Helper Methods 228
Mouse Gestures 221, 232
Naming Conventions 222
Object Methods 225
Panels Helper Methods 229
Skype 221
Tree Helper Methods 229
UK Postcodes 221
Whitelist 221
Window Helper Methods 230
Global parameters
 JUCE 176
 listed 53
 location 50
 NTList Version 4 146
GLWebMail 153
GMS 295
Greasemonkey 222, 231
GUI 282

H

HeaderFile 152
HELO Script.mml 88
HelpMsg 133
HelpMsgPeriod 133
HelpSentTime 148
HelpWords 133
Hidden 109
Homedir directory 196
Hops - maximum count 74
HTTP Object 224

I

IDLE 70
IMAP
 parameters 203
IMAPFetchBufferSize 70, 91, 93
IMAPLog 70
IMAPLogonDelay 71
IMAPMaxConnections 71
IMAPMaxConnectionsPerIP 179
IMAPPort 71
IMAPTcpBlockSize 71, 82
IMAPThreads 71
IMAPTimeClient 71

- IncomingDLL 72
- IncomingScript 72
- Index file
 - rebuilding 13
- Initialization 262
- Inspector. See Gordano Inspector.
- Install 114
- InstallDir 190
- interactive 30
- IP addresses
 - entering into NTMail 51
- IPAddress 101
- IPAllowed 72, 179
- IPAllowedMes 72, 179

J

- javascript 222
- Javascript File 223
- JoinAccess 133
- JoinAccessFile 133
- JoinConfirmationMsg 133
- JoinConfirmRequired 134
- JoinMsg 134
- JoinPassword 134
- JoinWords 134
- jQuery 222, 231

L

- Language 158
- LastConfirmed 148
- LastPost 134
- LastPosted 148
- LeaveAccess 134
- LeaveConfirmationMsg 134
- LeaveConfirmRequired 134
- LeaveInSubject 134
- LeaveMsg 134
- LeavePassword 134
- LeaveReason 148
- LeaveWords 134
- Line lengths 74
- LIST
 - parameters 203
- List
 - adding to 13
 - extracting addresses 12
 - File format 121
- List parameters 122
- ListAccess 135
- ListAccessFile 135
- ListAllAccount 151
- ListDescription 135
- ListFileLimit 146
- ListLog 146
- ListMan directory 193
- ListManager 135
- ListMaxAddresses 146
- ListMaxOutFiles 146
- ListMessage 135
- ListMsgSize 152
- ListNameInSubject 135
- ListNumber 114

- ListOwner 135
- ListPassword 135
- ListPOSTThreads 146
- ListReturnAction 152
- ListServerOneFile 135
- ListServerPassword 146
- ListSizeInForm 135
- ListSizeResponse 146
- ListStatus 135
- ListSummary 136
- ListThreads 146
- ListWords 136
- Local Clients 318
- LocalDomainNames 101
- LocalIP 72, 317
- Log
 - directory 74, 193
 - events shown by Monitor 37
 - file contents 301
 - file names 303
 - IIS 95
 - levels 301
 - SMTP, POST and POP logs 36
 - WWW 94
- LogAllMessages 72, 101
- LogAutoDelete 72
- LogDeleteAfter 72, 101
- LogEmailAfter 72, 101
- Logging
 - bit map 304
- Login 217
- LogoFile 190
- LogoLink 190
- LOGON authenticated SMTP 198
- LogonDelay 73, 109
- LogZipAfter 73, 101
- Looking up MX records 48
- Lookups on domain 63, 99, 108
- LurkerStartDate 136

M

- Mail exe
 - Run Script 18
- Mail relay
 - accepting 180
 - localP parameter 72
- Mail.exe
 - Append 10
 - CNAME 11
 - command line switches 8
 - Domain 11
 - Dump DNS Cache 11
 - End of line. 11
 - Fix Addresses 12
 - Help 13
 - Index 13
 - Insert 13
 - Insert List 18
 - introduction 8
 - Kick 14
 - List of Users 14
 - Load 14
 - Load Share 15

- Logon 16
- Max Wait 16
- MX Lookup 16
- Post 17
- Purge DNS Cache 17
- Query Server 18
- Reread Registry 20
- Sleep 19
- Sortlist 19
- Start Gordano services 19
- Stop Gordano services 19
- Transfer 19
- Verbose 20
- Version 20
- Wait 20
- Watch 21
- MailAutoRefresh 159
- Mailbox 281
 - size limit 74, 110
- Mailbox Monitor
 - introduction 36
- MailerMessage 73
- MailLogDLLLocation 73
- MailMon 295
- MailNumber 114
- MAILScript.mml 89
- Manuals button 63
- mashup 219
- MaxAccounts 102
- MaxAccountSize 73, 102, 109
- MaxBadCommand 73
- MaxBadMesDirSize 73
- MaxDailyPosts 136
- MaxHopCount 74
- MaxHTTPDataLength 74
- MaxLineLen 74
- MaxListMembers 136
- MaxListMsgSize 136, 152
- MaxLogDirSize 74
- MaxMailboxSize 74, 102, 110
- MaxMessageExceededMsg 136
- MaxMessageSize 75, 102, 110
- MaxMessageSizeInMes 102
- MaxMessageSizeOut 102
- MaxMessageSizeOutMes 103
- MaxMessagesPerIPAddress 75, 179
- MaxMessagesPerIPAddressMes 75, 179
- MaxMessagesPerUser 75, 180
- MaxMessagesPerUserMes 76, 180
- MaxNumberFilesInCache 76
- MaxPasswordLetterOccur 76
- MaxThreads 117
- MaxToAddresses 136
- MD5 authenticated SMTP 198
- MD5 authentication 198
- MemberAddress 136
- MemberConfirmExpires 137
- MemberConfirmMsg 137
- MemberExpireMsg 137
- MemberExpireWarning 137
- MemberExpireWarningMsg 137
- MemberFile 110, 137
- MemberModeratorAddress 137

- MemberModeratorJoinNotify 137
- MemberModeratorLeaveNotify 137
- MemberModeratorPassword 137
- MesLog directory 73, 101, 193, 195
- Message
 - from Redirect file 208
 - information 210
- Message File Format 261
- Message Flags Changes 268
- Message Functions 263
- Message Index 266
- Message Parsing DLL 270
- Migration Tools 297
 - Exchange 297
 - Linux/Solaris 297
 - Sendmail 297
 - Unix 298
- MIME types 95
- Mimic 219
- MinDiskSpaceRequired 76, 146
- MinPasswordLen 76
- MML
 - Delivery Scripts 282
 - Examples 282
 - GUI 282
 - Scripting Guide 282
 - Timed Events 282
- MML directory 193
- MMLErrorToSupport 76
- ModerateMemberExpires 138
- ModeratePostExpires 138
- ModerateRequestExpires 138
- ModerationDate 148
- ModFile 148
- Moved 110
- MSBugFix 117
- MsgFilename 138
- MSMailAddressFix 76
- MULTI_SZ Registry values 51
- MX records
 - lookup 48
 - lookups failing 67
- MyPassword 149
- MySQL 193

N

- Name 149
- NameServers 77
- NextCookie 77
- NextNtListCookie 77
- Nslookup tool 48
- NT SAM User Database
 - name of host 77, 103
- NT SAM user database
 - effect on Registry 51
- NTComputer 77, 103
- NTDomains 77, 103
- NTMail Inspector
 - in multi-domain system 33
- NTMetrn 282
- NullAccount 103
- NumAdverts 138
- NumAllocUnits 117

NumBadMsgs 149
NumberFilesSent 138
NumberJoined 138
NumberLeft 138
NumberPosted 138
NumberReturned 138
NumBins 117
NumWWWAllocUnits 117
NumWWWBins 117

O

ODBCTableName 138
OnlyAcceptLocal 180
OnlyAcceptLocalMes 180
OnlyJoinOnce 138
Option Pack 281
OptionalFields 139
Organization 149
Out directory
 two file types in 193
OutDir 80
OutgoingDLL 77
OutgoingScript 77

P

Parameters. See Registry parameters
Password 110, 139, 217
 policies 319
Password server
 enabling 63
PasswordChangeDate 110
PasswordPolicy 77
PasswordPort 77
PasswordWords 139
Performance Monitor
 introduction 37
 tracing memory leaks 37
permissions 31
Ping
 options 46
 using 46
Pipelining
 ESMTP command 200
 ESMTP command - enabling 68
Plan 110
POP
 Object Type 43
 parameters 203
 script file 14
POP3 DELE 3
POPAuthSMTP 180
POPAuthSMTPTime 180
POPBandwidth 78, 110
POPLog 78
POPLogonDelay 78
POPMaxConnections 78
POPMaxConnectionsPerIP 180
POPPort 78
POPReetrBufferSize 78
POPSaveOnDisconnect 79
POPTcpBlockSize 79
POPThreads 79

POPTIMEClient 79
port 32
Porting Tools 297
 Exchange 297
 Linux/Solaris 297
 SendMail 297
 Unix 298
Position 111
POST
 Object Type 42
 parameters 203
 starting service 14
PostAccess 139
PostAccessFile 139
PostAllCmdModerators 139
PostAllPostModerators 140
POSTBandwidth 80
PostCmdAsLineOne 140
PostCmdAsSubject 140
PostFix 103
PostFixesFile 103
PostHelpAsLineOne 141
PostHelpAsSubject 140
PostInsertFile 141, 152
POSTLog 80
PostMaster 103
Postmaster
 redirection to 210
PostmasterStats 80
POSTMaxContLine 80
POSTMaxWaiting 114
POSTMinWaiting 114
PostModeratorAddress 141
PostModeratorPassword 141
POSTRescanDelay 115
POSTRescanFilesWaiting 115
PostRetryTime 81
PostReturnTime 81
POSTServers 81
Postservers file 211
PostSSL 82
POSTThreads 83
POSTTimeSend 83
PostWarningTime 83
PostWords 141
PreloadImages 153, 155, 159
PreloadImagesThreads 153, 155, 159
PreloadImagesThreadsTimeout 153, 155, 159
PreloadImagesTimeout 154, 156, 159
Private parameters 113
ProcessMML 141
ProcessMMLAtDigest 141
Profiles 194, 195
Properties 149
Proxy
 disconnect delay 86
Proxy server
 enabling 63
 port 95
ProxyCacheSize 84
ProxyCacheSizeMax 84
ProxyPurgeAge 84

ProxyPurgeAgeLast 84
ProxyPurgeMethod 84
ProxyRASAccName 84
ProxyRASAccName2 85
ProxyRASBackup 85
ProxyRASPassword 85
ProxyRASPassword2 85
ProxyRASPhoneNumber 85
ProxyRASPhoneNumber2 85
ProxyRASSelected 85
Public 141
PurgeFrequency 141
PurgeStartDate 142
PutAccess 142
PutAccessFile 142
PutPassword 142
PutWords 142

Q

QSND. See ETRN
QuarantineAdminPort 85, 103
QuarantineAdminServer 85, 103
QuietWords 142
QuotaMessageFile 103

R

RAS
 primary proxy account 85
RASAccName 85
RASAccName2 85
RASDisconnectDelay 85
RASMaxScheduleTimeout 86
RASMinScheduleTimeout 85
RASPassword 86
RASPassword2 86
RASPhoneNumber 86
RASPhoneNumber2 86
RASProxyDisconnectDelay 86
RCPT
 clauses must be local 180
 failing limit 62
 maximum clauses 86
 redirects 210
 script 89
RcptLimit 86
RCPTScript.mml 89
ReadOnlyAction 142
ReadOnlyAddress 142
Redirect file
 smart routing 208
 syntax 208
RedirectFile 86
Regedt32
 description 45
 keys 49
RegisterSupport 117
Registry
 appending POP entries 10
 keys 45
 mandatory parameters 50
 replacing contents 14
 structure 49

 updating 20
Registry parameter
 RedirectFile 208
Registry parameters
 domain-specific 97
 global 53
 private 113
 three types 50
 user 106
RejectAttachmentExt 181
RejectAttachmentExtOptions 181
RejoinLurkerDays 142
RejoinLurkerExpires 142
RejoinLurkerMsg 143
Relay
 Disallowing Relay 317
RemoveBinaryAttachments 143
RemoveHeaderFields 143
ReportAction 143
ReportAddress 143
Request for Comments. See RFCs
RequestModeratorAddress 143
RequestModeratorPassword 143
RequiredFields 143
ResponseInsertFile 143
ResponseRate 111
Restart ESMTP command 200
RestrictedWordFile 182
RestrictedWordFileEnable 183
RestrictedWordFileMes 183
RestrictedWordFileReturn 183
RestrictedWordFileSave 183
RestrictedWordMode 183
ResumeAccess 143
ResumeAccessFile 144
ResumePassword 144
ReturnReceiptAllow 87
Reverse Proxy Server
 enabling 64
RevisionsHighWater 159
RFC 1893 error codes 307
RFC1123Return 87
RFC822 header 3
RFCs
 defined 3
 override options 3
Robot
 example 238
 example program 238
Robots 283
 Archive 283
 Detach 284
 Domain 284
 eSarah 285
 Filter 285
 Rotate 285
Rotate 285
Rows per Page 216
RSS Reader 219

S

SaveBadMes 87, 115
SaveSetupFiles 87

- SaveSetupSendTo 87
 - Script 181
 - Delivery 65
 - disabling editing 61
 - domain-specific parameter 104
 - EHLO 88
 - enabling 89
 - enabling for user 108
 - End Of Message 88
 - example 274
 - filter 69
 - HELO 88
 - incoming mail 72
 - outgoing mail 77
 - SMTP connection 87
 - SMTP DATA 88
 - SMTP MAIL 89
 - SMTP RCPT 89
 - writing with ColdFusion tags 273
 - Security 317
 - Disable VRFY 318
 - Disallowing Relay 317
 - Local Clients 318
 - LocalIP 317
 - Password 319
 - UnknownUserAction 318
 - SendHelpMsg 144
 - Sending rules 207, 211
 - SendJoinMsg 144
 - SendLeaveMsg 144
 - SendMail 297
 - SendStatistics 144
 - Services
 - parameters 203
 - stopping 19
 - Session Keep Alive 262
 - Sessions 262
 - SetAccess 144
 - SetAccessFile 144
 - SetPassword 144
 - SetWords 144
 - SharePoint 215
 - Calendar Properties 217
 - Configuration 215
 - Installation 215
 - Web Parts 215
 - Sharing Themes 236
 - ShowRealHost 87
 - Size SMTP command 201
 - SMTP
 - enhanced error codes 312
 - four digit error codes 314
 - From & To clauses 3
 - parameters 40, 203
 - real host in Received clause 3
 - resolve hostname 3
 - three digit error codes 311
 - SMTP DLL 286
 - SMTPAuth 87
 - SMTPCheckAccept 87
 - SMTPCheckFrom 87
 - SMTPConnectScript 87
 - SMTPCorrectFrom 88
 - SMTPDATAScript 88
 - SMTPDLL 88
 - SMTPEHLOScript 88
 - SMTPEOMScript 88
 - SMTPHELOScript 88
 - SMTPKickPOST 88
 - SMTPLoadSharingEnabled 104
 - SMTPLoadSharingFile 104
 - SMTPLoadSharingPrimaryServer 104
 - SMTPLog 88
 - SMTPLogonMsg 88
 - SMTPMAILScript 89
 - SMTPMaxConnections 89
 - SMTPMaxConnectionsPerlP 181
 - SMTPPortReceive 89
 - SMTPPortSend 89
 - SMTPRCPTScript 89
 - SMTPResolveHostname 89
 - SMTPScript 89, 181
 - SMTPSSL 90
 - SMTPThreads 90
 - SMTPTimeReceive 90
 - SpamMes directory 194
 - SQLConnectionMode 90
 - Standard input 238
 - StatisticsMessage 91
 - StatPassword 91
 - Status Flags 261
 - StatusAccess 144
 - StatusAccessFile 144
 - StatusPassword 144
 - StatusWords 145
 - StdIn 111
 - Stdin parameter 237
 - StdOut 111
 - Stdout parameter 237
 - StopWWWAction 145
 - SuperAllow 152
 - SuperListEntries 145
 - SuspendAccess 145
 - SuspendAccessFile 145
 - SuspendDays 150
 - SuspendPassword 145
- T**
- Tar pitting 80
 - Telephone 111
 - Temp directory 194
 - Template file
 - format 212
 - Theme
 - CSS 234
 - Full 234
 - Images 234
 - Options 234
 - Partial 234
 - Themes 233
 - Custom 235
 - Sharing 236
 - Threads
 - displayed by Monitor tool 36
 - for POP 79
 - in log file 301

Timeout
 DNS requests 67
 POP server 79
 POST server 83
 setting 16
 Web clients 96, 113
 WWW session 96
Time-To-Live of ICMP packets 47
Timezone Offset 217
TimeZoneName 91
Tools available 7
TotalMessageSize 145
TRAPToSupport 92
TreatAsLocal 92
Twitter 219
Type 111
Types 260

U

Udil 150
Unix 298
UnknownUserAction 104, 318
UpdateSQLConfiguration 115
Upgrade 285
UseNTDatabase 104
user 28, 29
User parameters
 GLCommunicator Version 6 147
 listed 106
User robot 237
UserAccessList 92
UserConfirm 150
UserDLLLocation 92, 256
UserMessage 105
Users subdirectory 195
UseSTDIn 112
UseSTDOOut 112

V

variables 29, 30
VerifyHostname 181
VerifyHostnameMes 181
VerifyIncoming 181
VerifyIncomingMes 182
VerifyOutgoing 182
VerifyOutgoingMes 182
Version 116
Virtual domain 100
Virtual to Full Domain 286
Visual Studio 222
VRFY
 enabling 64
 ESMTP command 201
VScanAction 186, 188
VScanActivate 170, 186, 188
VScanAlert 186, 188
VScanInline 186
VScanLog 187
VScanPostmasterAcc 187
VScanProgram 187
VScanRedirectAcc 187, 188
VScanReturnText 187, 188

VScanTimeout 187
VScanVendor 187

W

WapFolderIndexRange 174
WapMailAckText 174
WapMailAllowed 173, 174
WapMailMainTitle 173, 174
WapMailMOTD 173, 174
WapMailMsgIndexRange 174
WapMailMsgPageSize 174
WapMailSplash 173
WapMailSplashTime 173
WatchHostname 92
WatchMail 294
WatchPort 92
Web Parts 215
WebMailAccessIP 92, 154, 156
WebMailAccessIPDenied 92, 154, 156
WebMailAccessRights 159, 162
WebMailAddressHistory 160
WebMailAddressHistoryCount 160
WebMailAllowCustomisation 160
WebMailAllowUserRunExecutablehtml 1
 54
WebmailAttachDropdown 160
WebMailAutoDisplayNameForma 93
WebMailAutoDisplayNameFormat 105,
 112
WebmailAutoPersonality 160
WebMailBackgroudMenuBar 161
WebmailBackgroundBackdrop 160
WebmailBackgroundForm 161
WebmailBackgroundFrameBorder 161
WebmailBackgroundMail 161
WebmailBackgroundMailHeader 161
WebmailBackgroundMailList 161
WebmailBackgroundMessagebar 161
WebmailBackgroundNavigator 161
WebMailCustomButton1Alt 154
WebMailCustomButton1URL 154
WebmailDateFormat 161
WebMailDateSelectorPosition 161
WebMailDefaultTemplate 161
WebMailDraftsMailboxName 161
WebMailHeaderWindowHeight 162
WebMailKeepTrash 162
WebMailLog 93, 154
WebMailLogoAlt 154, 156
WebMailLogoUrl 154, 156
WebMailMaxForwardText 162
WebMailMeetingGranularity 163
WebMailMessageReadFont 163
WebMailMessageReadFontSize 163
WebMailMessageWriteFont 163
WebMailMessageWriteFontSize 163
WebMailMsgEncoding 163
WebMailNotifyNewMail 163
WebMailPopCheckInterval 163
WebMailPurgePending 163
WebmailReadPreviewWindow 164
WebmailReadWindow 163
WebMailReplyIndentText 164

WebMailRunExecutableHTML 164
WebMailSaveSentMessages 164
WebMailScript 155
WebMailSelectorAppearance 164
WebMailSelectorWindowHeight 164
WebMailSentMailboxName 164
WebMailShowLinks 164
WebMailShowRealNames 164
WebMailSmallMenu 164
WebMailTimeOut 164
WebMailTrashMailboxName 164
WebMailUseDefaultAccessIP 156
WebMailWeekStartday 165
Wildcarding
 LocalIP 317
WildCardMatch 145
WWW
 Parameters 203
WWWBaseDir 105
WWWCacheDir 93
WWWConnectionRetryAttempts 93
WWWConnectionRetryTimeOut 93
WWWCUList 93
WWWDefaultPageName 93
WWWErrorPageName 93
WWWFtpAllowed 93
WWWFtpExpress 93
WWWHostname 94
WWWLog 94
WWWMaxConnections 94
WWWMIMETypes 95
WWWMMLDefaultPageName 95
WWWMMLErrorPage 95
WWWMMLPort 93
WWWMMLRoot 95
WWWMMLScriptPort 95
WWWPassword 95
WWWPort 95
WWWProxyPort 95
WWWRequestLogDir 95
WWWRequestLogTypes 95
WWWResolveHostname 96
WWWSessionTimeOut 96
WWWShareSessionAcrossPorts 96
WWWTcpBlockSize 96
WWWThreads 96
WWWTimeClient 96
WWWTimeOut 113
WWWUserMMLDefaultPageName 97
WWWUserMMLDir 97
WWWUserMMLPort 97
WWWUseSessionCookies 94
WWWUseSessionIPAddress 94
WWWWebMailMMLPort 155
WWWWebMailMMLSSLPort 155
WWWWebMailSessionTimeout 155

X

XLIST 201
X-mailer line in header 73
XMIT 201

Z

ZombiePurgeFrequency 145
ZombiePurgeStartDate 145

Licence Agreements

GORDANO LIMITED SOFTWARE LICENCE AGREEMENT

Copyright © Gordano Ltd, 1995-2015

WARNING: YOU SHOULD CAREFULLY READ THE FOLLOWING TERMS AND CONDITIONS BEFORE USING THIS SOFTWARE PACKAGE. INSTALLING THE SOFTWARE ONTO YOUR COMPUTER INDICATES YOUR ACCEPTANCE OF THESE TERMS AND CONDITIONS. IF YOU DO NOT WISH TO ACCEPT ALL OF THESE TERMS, YOU SHOULD STOP INSTALLING THIS SOFTWARE NOW AND DESTROY ALL COPIES OF THE SOFTWARE AND ALL MANUALS AND OTHER DOCUMENTS SUPPLIED WITH IT.

1 DEFINITIONS

"Agreement" means this Gordano Limited Software Licence Agreement together with all related invoices.

"Company" means the licensee of the Software, being the signatory to this Agreement.

"Gordano" means Gordano Limited.

"Documentation" means any documentation or manuals provided with the Software or provided online or on storage media containing this text.

"Key" means the activation key.

"Software" means the software computer program, Key and Documentation contained in this package.

"Trial Period" means the period of 28 days from installation of the Software.

2 GRANT OF LICENCE

2.1 Subject to the Company's compliance with the terms of this Agreement, Gordano grants to the Company a non-exclusive, non-transferable licence to use the Software strictly for its own internal business operations only under the terms of this Agreement for the Trial Period and thereafter if a key is purchased from Gordano or its authorised representatives. For the avoidance of doubt, operating the Software outside the Trial Period or without a Key from Gordano (or its representatives) constitutes unlicensed use of the Software and will be a material breach of this Agreement, which would allow Gordano to terminate under clause 8.2.

2.2 This Agreement becomes effective upon the Company signing this Agreement or installing the Software.

2.3 On expiry of the Trial Period and on payment of the fee invoiced by Gordano, the Company will be sent the Key which will activate the Software.

2.4 The Company may use the Software on the number of computers that it has purchased a licence for; a separate license is required for any other computers. The number of licenses purchased by the Company under this Agreement will be stated on the invoice issued by Gordano.

2.5 The Company may make one copy of the Software, strictly for backup or archive purposes only.

2.6 The Company shall be responsible for all use of the Software licenced under this Agreement, including but not limited to any use by its agents, contractors, outsourcers, customers and suppliers, and their compliance with this Agreement.

2.7 The Company agrees to maintain accurate and adequate records relating to its use of the Software and compliance with this Agreement. The Company agrees to permit Gordano to audit the Company in relation to its use of the Software and compliance with the terms of this Agreement. The Company shall provide Gordano with reasonable assistance and access to information in the course of any such audit, and the Company

agrees that Gordano may report the audit results to its licensors. Each party shall be responsible for its own costs in relation to any such audit.

2.8 In the event that the Software contains source code from a licensor of Gordano, that source code shall also be governed by the terms of this Agreement.

3 OWNERSHIP OF THE SOFTWARE

3.1 Gordano and its licensors own all title and proprietary rights to the Software and all copies thereof and all rights therein, including without limitation all copyright, patents, know-how, trade secrets, trade marks or names and database rights. All such rights shall remain vested in Gordano and its licensors. The provision of the Software to you does not grant, and you do not receive, any rights under any Microsoft intellectual property with respect to any device or software that you use to access the Software.

3.2 The Company undertakes and agrees as follows:

(a) it may NOT make or permit others to make any copies of the Software except for one backup copy.

(b) it may NOT reverse engineer, disassemble, decompile the Software or attempt to reconstruct, identify or discover any source code except as expressly permissible by law.

(c) it may NOT modify, adapt or translate the Software or incorporate the Software, in whole or in part in any other product or software or permit others to do so without express, written consent of Gordano.

(d) it may NOT disclose, provide or otherwise make available in any form the Software, its functionality or any portion thereof, to any third party other than its employees without the prior written consent of Gordano.

(e) it may NOT remove any copyright, trademark, proprietary rights, disclaimer or warning notice included on or embedded in any part of the Software and the Company agrees to diligently reproduce all copyright notice(s) and other proprietary notices of Gordano on any authorised copy of the Software.

(f) it may NOT assign, sell, transfer (except for temporary transfer in the event of computer malfunction), licence, sub-licence, rent, timeshare, lease or otherwise redistribute the Software or its functionality to any third party without the written permission of Gordano.

(g) it may NOT use the Documentation for any purpose other than to support its use of the Software.

(h) it accepts that from time to time, the Software will send a message containing details of the Key or Keys installed to Gordano and it agrees not to interfere with the delivery of this message.

(i) it accepts, that Gordano may receive error messages from the Software installed on the Company's system in the event that the Software fails for some reason (and that the Company has the option to turn this off).

(j) it agrees to stop using all previous version of the Software immediately following an upgrade.

(k) it may NOT use the Software for any subscription service, hosting or outsourcing.

(l) it may NOT publish any results of benchmark tests run on the programs.

(m) if appropriate, it must comply with all relevant import and export laws to ensure that the Software or anything directly produced using the Software are not exported directly or indirectly contrary to applicable laws.

(n) it agrees that any third party technology that may be appropriate or necessary for use with some or all

of the Software that is notified to the Company (whether via the Documentation or otherwise) shall not be licensed to the Company under this Agreement, but may be

licensed as stated in the Documentation or as otherwise notified to the Company.

(o) The Company shall ensure that its customers and/or employees (and any other persons) that use the Software agree to and are bound by the following condition on their right to access and use the Software: "The provision of the Software to you does not grant, and you do not receive, any rights under any Microsoft intellectual property with respect to any device or software that you use to access the Software."

3.3 No distribution licence or other rights are provided to the Company under this agreement.

3.4 The Software may utilise Microsoft® Exchange ActiveSync, and the use of Microsoft® Exchange ActiveSync is limited to internal use as part of hosting the Software for the sole purpose of providing access by Microsoft® approved devices to email accounts of employees or customers of the Company maintained by the Software.

The provisions of clauses 3, 4, 6, and 7 shall survive termination of this Agreement.

4 CONFIDENTIALITY

4.1 The Company undertakes to treat as confidential and keep secret all information contained or embodied in the Software and Documentation supplied by Gordano.

5 ANTI-VIRUS

5.1 Gordano does not warrant that the Software is free from all known viruses and the Company shall assume responsibility to take appropriate steps to ensure that the Software is virus free and that the running of the Software will not damage or interfere with the computer system on which the Software is used or any data or software which may be used or stored on its computer system.

6 WARRANTY AND DISCLAIMER

6.1 The Company acknowledges that software in general is not error free and agrees that the existence of such errors in the Software shall not constitute a breach of this Agreement.

6.2 The Company further acknowledges that the Software has not been developed to meet its specific individual requirements and that it is the Company's responsibility to ensure that any use of the Software or the information contained on it is suitable for its specific individual requirements.

6.3 THIS SOFTWARE IS PROVIDED 'AS IS'. GORDANO WARRANTS THAT THE SOFTWARE WILL SUBSTANTIALLY COMPLY WITH THE SPECIFICATIONS SET OUT IN THE DOCUMENTATION. EXCEPT AS STATED HEREIN AND TO THE EXTENT PERMITTED BY LAW THE SOFTWARE IS PROVIDED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY AND FITNESS FOR A PARTICULAR PURPOSE. GORDANO DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET THE COMPANY'S REQUIREMENTS OR THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE.

6.4 Gordano does not represent or warrant that the Software furnished hereunder is free of infringement of any third party patents, copyrights, other intellectual property rights or trade secrets. The Company waives any right to indemnification or other relief from Gordano should the Software be found to be defective or to infringe any right of any third party.

6.5 Nothing in this Agreement shall exclude or limit the liability of Gordano for death or personal injury caused by its negligence or for any other liability which cannot by law be excluded.

GORDANO'S SOLE LIABILITY TO THE COMPANY FOR ANY CLAIM, DEMAND OR CAUSE OR ACTION WHATSOEVER, AND REGARDLESS OF FORM OF ACTION, WHETHER IN CONTRACT OR TORT, SHALL BE LIMITED TO REPLACEMENT OF THE PRODUCT OR

REFUND OF THE LICENCE FEE PAID FOR THE SOFTWARE. IN NO EVENT SHALL GORDANO OR ITS LICENSORS BE LIABLE TO THE COMPANY FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL OR PUNITIVE DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOSS OF ANTICIPATED SAVINGS, LOSS OF REVENUES, LOSS OF PROFIT, LOSS OF BUSINESS, LOSS OF DATA OR DATA USE OR ECONOMIC LOSS OF ANY KIND.

7 LIMIT OF LIABILITY

7.1 In the event that any exclusion or limitation in clause 6 above is held to be invalid for any reason and Gordano becomes liable for loss or damage that may lawfully be limited, such liability shall be limited to the sum equivalent to a multiple of 3 (three) times the total annual fee paid by the Company to Gordano for the licence of the Software.

8 TERMINATION OF LICENCE

8.1 Save in the event of any unlicensed use of the Software when the terms of this Agreement shall remain in full force and effect, the Company may terminate this Agreement, at any time, by destroying or returning all copies of the Software.

8.2 Gordano may terminate this Agreement by written notice to the Company if the Company is in default of any terms or conditions of this Agreement or if the Company enters into any form of insolvency including without limitation liquidation, receivership, voluntary arrangement, administration or is unable to pay its debts as they fall due.

8.3 On termination of this Agreement the Company agrees to discontinue all use of the Software and destroy all copies of the Software in any form in its possession or control, and if requested by Gordano certify in writing that such action has been taken. The Company shall not be entitled to any refund of any monies or other consideration paid by it.

9 SUPPORT

9.1 Gordano shall provide support for the first 28 days from your first contact with Gordano or its representatives. First contact means the Company's representative's first telephone call to Gordano, registration on the Gordano website, or installation of the trial software from our website, whichever is the earlier.

9.2 On expiry of this 28 days the Company shall have the option of purchasing support services from Gordano under the terms of the Support Agreement.

10 MAINTENANCE (Software Updates)

10.1 Gordano shall provide maintenance services in the form of updates to the Software for the duration of the Software's licence term, commencing on the expiry of the Trial Period and on the Company's receipt of the Key. Thereafter, the Company shall have the option of renewing annual maintenance services (Software updates) from Gordano.

10.2 Maintenance services shall comprise of the provision of new versions of the Software only as and when they become available, and no other maintenance services or assistance is included.

11 GENERAL

11.1 If any provision of this Agreement is determined to be invalid or unenforceable, by any court of competent jurisdiction it shall be deemed to be omitted and the remaining

provisions shall continue in full force and effect.

11.2 Gordano's waiver of any right shall not constitute a waiver of that right in the future.

11.3 This Agreement shall be governed and construed in accordance with the laws of England and both parties submit to the exclusive jurisdiction of the English courts, save in respect of enforcement where the jurisdiction shall be non-exclusive.

11.4 This Agreement constitutes the entire understanding between the parties with respect to the subject matter hereof. The Company agrees that any of Gordano's licensors that are associated with the Software shall be a third party beneficiary of this Agreement. All prior agreements, representations, statements and undertakings, oral or written, between the Company and Gordano are hereby expressly superseded and canceled.

11.5 All notices under this Agreement shall be in writing and shall be given by registered or certified mail to the following address: Gordano Ltd, 1 Yeo Bank Business Park, Kenn Road, Kenn, Clevedon, North Somerset, BS21 6UW, UK.

© 1995-2015. Gordano Limited. All rights reserved.

GORDANO LIMITED SUPPORT AGREEMENT

WARNING: YOU SHOULD CAREFULLY READ THE FOLLOWING TERMS AND CONDITIONS. BY REGISTERING FOR SUPPORT SERVICES TO BE PROVIDED BY GORDANO YOU ARE ACCEPTING THESE TERMS AND CONDITIONS. IF YOU DO NOT WISH TO ACCEPT ALL OF THESE TERMS YOU SHOULD IMMEDIATELY NOTIFY GORDANO AND ANY SUPPORT FEE YOU MAY HAVE PAID WILL BE REFUNDED FOR THE OUTSTANDING CONTRACT TERM.

1 DEFINITIONS

"Business Days" means weekdays excluding weekends, and UK Bank and Public Holidays and Gordano's training days (which will be notified to the Company in advance and in any case will not be more than 3 (three) days in any one calendar year).

"Company" means the licensee of the Software.

"Gordano" means Gordano Limited.

"Key" means the activation key for the Software or Support Service.

"Software" means the software computer program and documentation licensed to the Company from Gordano.

"Software Licence" means the software licence granting the Company a non-exclusive, nontransferable licence of the Software.

"Support Fee" means the fees payable for the Support Service, which shall be in accordance with Gordano's current price list as amended from time to time.

"Support Agreement" means this Gordano Limited Support Agreement.

"Support Service" means the support services provided by Gordano in relation to the Software and as detailed in clause 3 of this Support Agreement.

2 GRANT

2.1 This Support Agreement is for the provision of Gordano's Support Service in respect of the current version of the Software for the term of your subscription to the Support Service commencing from the date of the commencement of your subscription for the Support Service.

2.2 If further products are licensed from Gordano during the lifetime of this Agreement a "top-up" fee may be added to extend this Support Agreement to cover the additional products at the time of their purchase.

2.3 This Support Agreement becomes effective on the date you pay for the Support Service.

2.4 Customers may register as users on the helpdesk at <https://helpdesk.gordano.com> however this is not required in order to receive support.

3 SUPPORT SERVICES

3.1 Gordano shall provide the Company with the following Support Service:

- (a) telephone support for the Software (currently on +44 (0)1275 340151):
 - (i) between the hours of 0900 to 17:00 or 14:00 to 2200 hours UK Time; or
 - (ii) between the hours of 0900 to 2200 UK Time; on all Business Days or
 - (iii) for 24x7 cover;

telephone support shall be provided at all hours on all days

- (b) email support for the Software at support@gordano.com or helpdesk@gordano.com.

3.2 Messages sent to and Support calls made to Gordano will be processed automatically and assigned a ticket ID. Gordano will send confirmation of these details to the creator of

the ticket.

3.3 All Support Services for the Software will be provided in the English language only.

4 EXCLUDED SERVICES

The Support Service supplied under this Agreement shall not include the provision of Support Service in respect of:

- (a) any version of the Software which is more than 24 months past its release date, except at the discretion of a support engineer or the management of Gordano Ltd;
- (b) any products or services which are not the Software or its components;
- (c) training in the use of the Software;
- (d) any development services;
- (e) defects or errors resulting from any modifications or enhancements of the Software made by any person other than Gordano;
- (f) use of the Software other than in accordance with the documentation or operator error;
- (g) virus protection or bug fixes except in exceptional circumstances as advised by Gordano, for example, when the system has been compromised by some external force and there is no available workaround; or
- (h) any circumstances beyond the reasonable control of Gordano, including (but not limited to) any act of God, fire, flood, war, act of violence or any other similar occurrence or failure or reduced performance of telecommunications networks or the internet.

5 COMPANY OBLIGATIONS

5.1 The Company agrees and undertakes:

- (a) to ensure that the Software is used only in accordance with the documentation or advice from Gordano, by competent trained employees only or by persons under their supervision;
- (b) not to alter or modify the Software in any way whatever nor permit the Software to be combined with any other programs to form a combined work;
- (c) not to request, permit or authorise anyone other than Gordano or its nominated third parties to provide any support services in respect of the Software;
- (d) to co-operate fully with Gordano's personnel in the diagnosis of any error or defect in the Software;
- (e) if necessary, to make available to Gordano free of charge all information facilities and services reasonably required by Gordano to enable Gordano to provide the support services;
- (f) to provide such telecommunication facilities as are reasonably required by Gordano for testing and diagnostic purposes.

6 SUPPORT FEES

In consideration of the Support Services the Company shall pay the Support Fee in advance to Gordano

7 TERMINATION

Gordano may terminate this Support Agreement by written notice to the Company if the Company is in default of any terms or conditions of this Support Agreement by written notice to the Company or if the Company enters into any form of insolvency including without limitation liquidation, receivership, voluntary arrangement, administration or are

unable to pay its debts as they fall due.

8 LIABILITY

Gordano's sole liability to the Company for any claim, demand, cause or action whatsoever, and regardless of form of action, whether in contract or tort, including negligence, shall be limited, at Gordano's sole option, to refund of the purchase price, re-performance of the Support Service or an extension to the length of the Support Service to be provided. In no event shall Gordano be liable for recovery of any special, indirect, incidental, or consequential damages, even if Gordano has been advised of the possibility of such damages, including but not limited to lost profits, lost savings, lost revenues, lost business, lost data or economic loss of any kind, or for any claim by any third party.

9 LIMIT OF LIABILITY

In the event that any exclusion or limitation in clause 8 above is held to be invalid for any reason and Gordano becomes liable for loss or damage that may lawfully be limited, such liability shall be limited to the sum equivalent to a multiple of three times the Support Fees paid by the Company to Gordano.

10 GENERAL

10.1 If any provision of this Support Agreement is determined to be invalid or unenforceable, by any court of competent jurisdiction it shall be deemed to be omitted and the remaining provisions shall continue in full force and effect.

10.2 Gordano's waiver of any right shall not constitute a waiver of that right in the future.

10.3 This Support Agreement shall be governed and construed in accordance with the laws of England and both parties submit to the exclusive jurisdiction of the English courts, save in respect of enforcement where the jurisdiction shall be non-exclusive.

10.4 This Support Agreement constitutes the entire understanding between the parties with respect to the subject matter hereof and all prior agreements, representations, statements and undertakings, oral or written, are hereby expressly superseded and cancelled.

10.5 All notices in connection with this Agreement shall be in writing and shall be given by registered or certified mail to the following address: Gordano Ltd, 1 Yeo Bank Business Park, Kenn, Kenn Road,
Clevedon, North Somerset, BS21 6UW, UK.

© 1995-2015. Gordano Limited. All rights reserved.

LICENCE AGREEMENT MySQL AB

MySQL AB, Bangårdsgatan 8, 753 20 Uppsala, SWEDEN

1. License Grant. Customer is granted a limited, non-exclusive, non-transferable license to run one copy of the object code version of the Licensed Software on one machine or instrument solely as integrated with, and for running and extracting data from, a Licensee Application. Use shall be limited to internal business purposes in accordance with these license terms. If the Integrated Product is licensed for concurrent or network use, Customer may not allow more than the maximum number of authorized users to access and use the Licensed Software concurrently.

2. License Restrictions. Customer may make copies of the Licensed Software only for backup and archival purposes. Customer shall not:

- (a) copy the Licensed Software onto any public or distributed network;
- (b) use the Licensed Software as a general SQL server, as a stand alone application or with applications other than Licensee Applications under this license;
- (c) change any proprietary rights notices which appear in the Licensed Software; or
- (d) modify the Licensed Software.

3. Ownership. MySQL AB and its third party suppliers retain all right, title and interest in the Licensed Software and all copies thereof, including all copyright and other intellectual property rights. MySQL AB may protect its rights in the Licensed Software in the event of any violation of this EULA.

4. Transfer. Customer may transfer the license granted herein provided that it complies with any transfer terms imposed by Licensee and delivers all copies of the Licensed Software to the transferee along with this EULA. The transferee must accept the terms and conditions of this EULA as a condition to any transfer. Customer's license to use the Licensed Software will terminate upon transfer. Customer must comply with all applicable export laws and regulations.

5. Termination. Upon termination of this license, Customer must immediately destroy all copies of the Licensed Software.

The MD5 Message-Digest Algorithm

The MD5 Message-Digest Algorithm used in NTMail is copyright (c) 1992-2, RSA Data Security, Inc. Created 1991. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

jQuery MIT License

Copyright (c) 2008 John Resig, <http://jquery.com/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Installation and Contact Information

For installation you need the following information. Keep a note of the values you used here in case you need to quote them to support.

Your domain name

Your computer's IP address (if static).

Telephone number of ISP's computer.

Your account user name at the ISP and its password.

To contact Gordano Ltd. :

Support

- Email: support@gordano.com

Sales

- Email: sales@gordano.com
- Tel: +44 1275 345100
- Fax: +44 1275 340056
- Unit 1, Yeo Bank Business Park, Kenn Road, Clevedon, North Somerset, BS21 6UW.

Gordano Limited

Unit 1, Yeo Bank Business Park, Kenn Road, Clevedon,
North Somerset, BS21 6UW - UK

<http://www.gordano.com>

