

Novell GroupWise®

6.5

February 28, 2005

TROUBLESHOOTING 3: MESSAGE FLOW
AND DIRECTORY STRUCTURE

www.novell.com

N

Novell®

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

You may not use, export, or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 1993-2005 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.novell.com/company/legal/patents/> and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.

www.novell.com

GroupWise 6.5 Troubleshooting 3: Message Flow and Directory Structure
[February 28, 2005](#)

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

GroupWise is a registered trademark of Novell, Inc. in the United States and other countries.

NDS is a registered trademark of Novell, Inc. in the United States and other countries.

NLM is a trademark of Novell, Inc.

QuickFinder is a trademark of Novell, Inc.

Third-Party Trademarks

All third-party trademarks are the property of their respective owners.

Contents

- About This Guide** **9**
- Part I Message Flow Diagrams**
- 1 Message Delivery in the Local Post Office** **13**
 - Access Mode: Client/Server 13
- 2 Message Delivery to a Different Post Office** **15**
 - TCP/IP Link Open: Transfer between Post Offices Successful 15
 - TCP/IP Link Closed: Transfer between Post Offices Delayed 18
 - Mapped/UNC Link Open: Transfer between Post Offices Successful 21
 - Mapped/UNC Link Closed: Transfer between Post Offices Delayed 23
- 3 Message Delivery to a Different Domain** **27**
 - TCP/IP Link Open: Transfer between Domains Successful 27
 - TCP/IP Link Closed: Transfer between Domains Delayed 30
 - Mapped/UNC Link Open: Transfer between Domains Successful 33
 - Mapped/UNC Link Closed: Transfer between Domains Delayed 36
 - Alternate Link Configuration: Transfer Pull 38
- 4 Message Delivery to and from the Internet** **41**
 - Message Transfer from GroupWise to the Internet 41
 - Message Transfer from the Internet to GroupWise 43
- 5 Message Delivery to and from GroupWise Remote** **45**
 - Mapped/UNC Link for "Hit the Road" 45
 - Modem Link to the Remote User through the Async Gateway 47
- 6 Administrative Database Update** **51**
 - TCP/IP Link between the Domain and the Post Office 51
 - Mapped/UNC Link between the Domain and the Post Office 52
- Part II Directory Structure Diagrams**
- 7 Message Transfer/Storage Directories** **57**
 - Domain Directory 57
 - domain* directory 58
 - wpcsin directory 58
 - wptools directory 61
 - wpgate directory 61
 - wpcsout directory 61
 - mtaname file 63
 - wpdomain.db file 63
 - wpdomain.dc file 63
 - wphost.dc file 63
 - gwdom.dc file 64
 - gwpo.dc file 64

agentinstall.txt	64
Post Office Directory.	64
<i>post_office</i> directory	66
wpcsin directory	66
gwdms directory	68
ofmsg directory	69
ofuser directory	70
offiles directory.	71
ofviews directory	71
ofwork directory	71
ofdirect directory	72
wpcnout directory	72
wphost.db file	75
gwpo.dc file	75
ngwguard.db file	75
ngwguard.dc file	75
ngwguard.fbk file.	75
ngwguard.rfi file	76
ngwcheck.db.	76
agentinstall.txt	76
MTA Local Queue Directory.	76
mslocal directory.	77
msglog directory	77
gwinprog directory	78
mshold directory	78
domainms directory	78
<i>postx</i> directories	78
<i>gatewayx</i> directories	78
<i>domainx</i> directories	79
0-7 directories	79
mtaname files	79
gwvscan directory	79
mtaconv directory	79
Internet Agent Queue Directory	80
<i>domain</i> \wpgate\gwia directory	81
gwia directory	84
WebAccess Agent Queue Directory.	86
<i>domain</i> \wpgate\webac65a directory	86
000.prc directory	86
wpcsin directory	87
wpcnout directory	87
gwhold directory	87
gwprob directory	87
files directory.	87
template directory	87
commgr.cfg file.	87
comint.cfg file	88
mimetype.cfg file.	88
gwcache.db file	88
gwac.db file	88
gwac.dc file	88
Caching Mailbox Directory.	89
\novel\groupwise\gwxxxxx directory	89
rofddata directory	90
wpcsin directory	92

wpcoutofs directory	92
wpgwsend directory	92
wpgwrecv directory	93
Remote Mailbox Directory	93
<i>remote_mailbox</i> directory	93
wpcsin directory	94
rofdata directory	94
wpgwsend directory	95
wpgwrecv directory	96
wpcoutofs directory	96
8 Agent Installation Directories	97
GroupWise Agent Installation (POA and MTA)	97
NetWare Installation Directory	97
Linux Installation Directory	100
Windows Installation Directory	105
Internet Agent Installation	107
NetWare Installation Directory	107
Linux Installation Directory	109
Windows Installation Directory	112
WebAccess Agent Installation	113
NetWare Installation Directory	113
Linux Installation Directory	115
Windows Installation Directory	120
Monitor Agent Installation	123
Linux Installation Directory	123
Windows Installation Directory	125
Apache/Tomcat Installation	127
NetWare Installation Directory	127
Linux Installation Directory	133
9 Software Distribution Directory	137
NetWare/Windows Software Distribution Directory	137
\grpwise\software directory	138
agents directory	138
domain directory	139
po directory	140
client directory	141
ofviews directory	141
zen directory	142
admin directory	142
internet directory	142
license directory	143
common directory	143
docs directory	143
Linux Software Distribution Directory	143
/opt/novell/groupwise/software directory	145
agents directory	146
domain directory	146
po directory	147
client directory	148
ofviews directory	148
admin directory	148
internet directory	149
license directory	151
docs directory	151

gwinst directory	151
10 GroupWise Client Installation Directories	153
Windows Client	153
c:\novell\groupwise	154
grpwise.exe file	154
gwtip.exe file	154
notify.exe file	154
addrbook.exe file	154
gwcheck.exe file	154
htrsetup.exe file	154
gwimpexe.exe file	155
gwmalto.exe file	155
gwreload.exe file	155
gwsync.exe file	155
ngwguard.dc file	155
wprof.dc file	155
*.dll files	155
*.ocx files	155
*.flt files	156
*.chm files	156
ofviews directory	156
ppforms directory	156
Cross-Platform Client on Linux	156
/opt/novell/groupwise/client directory	157
bin directory	157
lib directory	157
jre directory	158
logs directory	158
Cross-Platform Client on Macintosh	158
/Applications/GroupWise.app directory	159
Contents directory	159
MacOS directory	159
Resources directory	159
lib directory	159
Java directory	159
 Part III Documentation Updates	
February 28, 2005 (GroupWise 6.5 SP4)	161
September 30, 2004	161
September 30, 2003	162
July 16, 2003 (GroupWise 6.5 SP1)	162

About This Guide

This Novell® *GroupWise*® *Troubleshooting 3 Guide* provides diagrams to help you understand the structure and functioning of your GroupWise system. The guide is divided into the following sections:

- ◆ “Message Flow Diagrams” on page 11
- ◆ “Directory Structure Diagrams” on page 55

Other sources of troubleshooting assistance include:

- ◆ Novell Support (<http://support.novell.com>)
- ◆ Novell Support Knowledgebase (http://support.novell.com/search/kb_index.jsp)
- ◆ GroupWise 6.x Support Forums (<http://support.novell.com/forums/2gw.html>)
- ◆ GroupWise Cool Solutions (<http://www.novell.com/coolsolutions/gwmag/index.html>)

Additional Documentation

For additional GroupWise documentation, see the following guides at the [Novell GroupWise 6.5 documentation Web site](http://www.novell.com/documentation/lg/gw65) (<http://www.novell.com/documentation/lg/gw65>):

- ◆ *Installation Guide*
- ◆ *Administration Guide*
- ◆ *Multi-System Administration Guide*
- ◆ *Interoperability Guide*
- ◆ *GroupWise Client User Guides*

Documentation Updates

For the most recent version of the *GroupWise 6.5 Troubleshooting 3 Guide*, visit the [Novell GroupWise 6.5 documentation Web site](http://www.novell.com/documentation/lg/gw65) (<http://www.novell.com/documentation/lg/gw65>).

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (™, ®, etc.) denotes a Novell trademark. An asterisk denotes a third-party trademark.

User Comments

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comment feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.



Message Flow Diagrams

This part of *Troubleshooting 3: Message Flow and Directory Structure* helps you understand how messages travel between GroupWise® users and how administrative updates to GroupWise databases occur.

- ◆ “Message Delivery in the Local Post Office” on page 13
- ◆ “Message Delivery to a Different Post Office” on page 15
- ◆ “Message Delivery to a Different Domain” on page 27
- ◆ “Message Delivery to and from the Internet” on page 41
- ◆ “Message Delivery to and from GroupWise Remote” on page 45
- ◆ “Administrative Database Update” on page 51

1

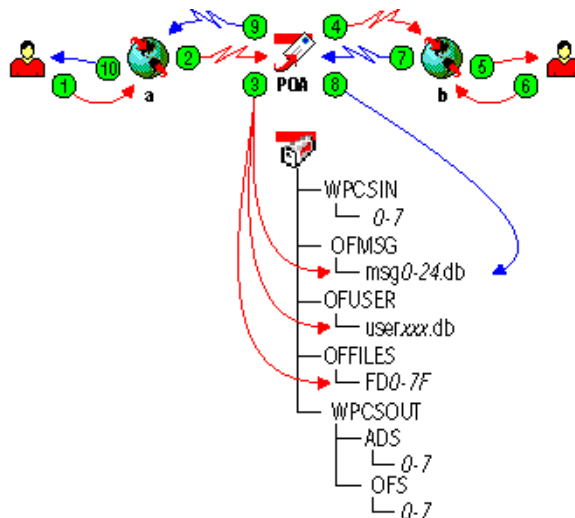
Message Delivery in the Local Post Office



The GroupWise 6 client functions optimally using client/server access to the post office. For an explanation of client/server access mode, see “[Post Office Access Mode](#)” in “[Post Office Agent](#)” in the *GroupWise 6.5 Administration Guide*.









- ♦ “[Access Mode: Client/Server](#)” on page 13

Access Mode: Client/Server

This message flow diagram illustrates how a GroupWise message travels from one user to another in the local post office when the client and POA communicate by way of TCP/IP.



Stage	Icon	Description
1 Sender		The user sends a message to recipients in the same post office. The access mode setting for the post office is Client/Server Only.
2 Sender's GroupWise Client		The GroupWise client communicates with the POA by way of TCP/IP.

Stage	Icon	Description
<p>2 POA for Local Post Office</p>		<p>The POA receives the message from the GroupWise client and performs the following actions for the sender:</p> <ul style="list-style-type: none"> ♦ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ♦ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ♦ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) <p>The POA also performs the following actions for the recipient:</p> <ul style="list-style-type: none"> ♦ Creates a pointer in each recipient's user database (<i>userxxx.db</i>) to the message in the message database (<i>msgnn.db</i>) so the new message appears in the recipient's mailbox. ♦ Updates the message in the message database (<i>msgnn.db</i>) with a Delivered status for each recipient.
<p>3 POA for Local Post Office</p>		<p>The POA communicates to the GroupWise client by way of TCP/IP that a new message has arrived.</p>
<p>4 Recipient's GroupWise Client</p>		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
<p>5 Recipient</p>		<p>Each recipient opens the message in the GroupWise client.</p>
<p>6 Recipient's GroupWise Client</p>		<p>Each recipient's GroupWise client communicates the Opened status to the POA by way of TCP/IP.</p>
<p>8 POA for Local Post Office</p>		<p>The POA receives the Opened status from the GroupWise client and updates the message in the message database with the Opened status for each recipient who opens the message.</p>
<p>9 POA for Local Post Office</p>		<p>The POA communicate the Opened status to the sender's GroupWise client by way of TCP/IP.</p>
<p>10 Sender</p>		<p>When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).</p>

2

Message Delivery to a Different Post Office

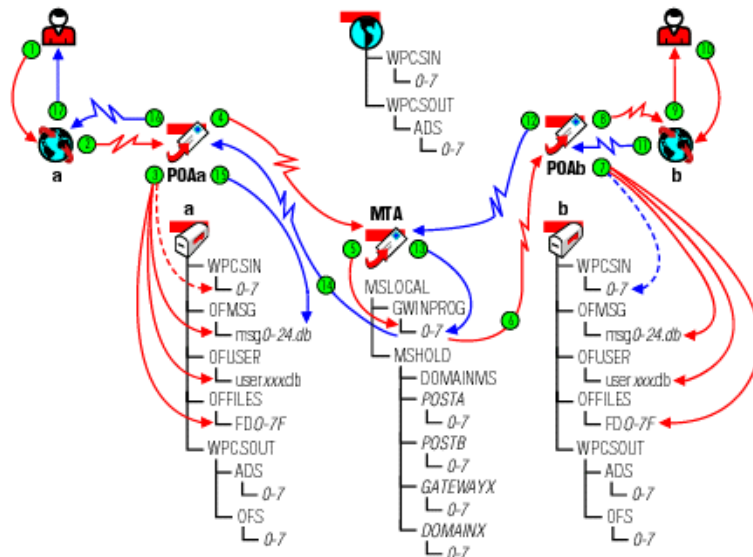
The MTA handles message transfer between post offices.


- ♦ “TCP/IP Link Open: Transfer between Post Offices Successful” on page 15
- ♦ “TCP/IP Link Closed: Transfer between Post Offices Delayed” on page 18
- ♦ “Mapped/UNC Link Open: Transfer between Post Offices Successful” on page 21
- ♦ “Mapped/UNC Link Closed: Transfer between Post Offices Delayed” on page 23






For an explanation of link types and link protocols, see “Understanding Link Configuration” in “Domains” in the *GroupWise 6.5 Administration Guide*.









TCP/IP Link Open: Transfer between Post Offices Successful




This message flow diagram illustrates how a GroupWise® message travels from one user to another between post offices in the same domain when the TCP/IP link between the post office and the domain is open.



Stage	Icon	Description
1 Sender		The user sends a message to recipients in a different post office in the same domain. In this diagram, the access mode setting in the local post office is Client/Server Only.

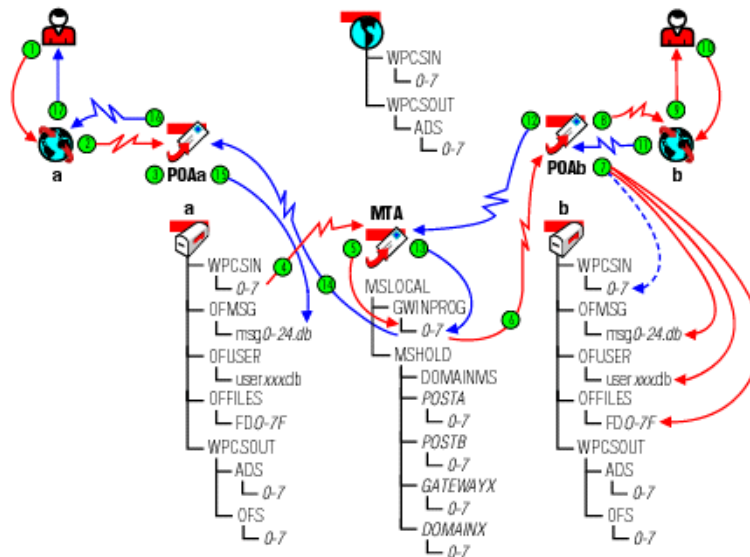
Stage	Icon	Description
2 Sender's GroupWise Client		The GroupWise client communicates the message to the POA by way of TCP/IP.
3 POA for Sender's Post Office		<p>The POA receives the message from the GroupWise client and performs the following actions for the sender:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office, in case the TCP/IP link to the MTA is currently closed.
4 POA for Sender's Post Office		<p>The POA then communicates the message to the MTA by way of TCP/IP, and deletes the copy in the MTA input queue because the TCP/IP transfer to the MTA was successful.</p> <p>To see what would happen if the TCP/IP link to the MTA were closed, see "TCP/IP Link Closed: Transfer between Post Offices Delayed" on page 18.</p>
5 MTA for Local Domain		The MTA receives the message and places it into the MTA "in progress" (<i>gwinprog</i>) queue.
6 MTA for Local Domain		<p>The MTA then communicates the message to the POA in the recipient's post office by way of TCP/IP. When the transmission is successful, the MTA deletes the message from the MTA "in progress" queue.</p> <p>If the TCP/IP link to the recipient's post office were closed, the message would be placed in the closed post office's holding queue in the MTA's <i>mslocal</i> directory for later transfer. The resulting message flow would be parallel to what occurs when a domain is closed. See "TCP/IP Link Closed: Transfer between Domains Delayed" on page 30 for a similar message flow that illustrates how messages to closed locations are handled.</p>



Stage	Icon	Description
7 POA for Recipient's Post Office		<p>When it receives the new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ♦ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ♦ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ♦ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ♦ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office. It also communicates the Delivered status message directly to the MTA by way of TCP/IP. When that transmission is successful, the copy in the MTA input queue is deleted.
8 POA for Local Post Office		The POA communicates to the GroupWise client by way of TCP/IP that a new message has arrived.
9 Recipient's GroupWise Client		The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.
10 Recipient		Each recipient opens the message in the GroupWise client.
11 Recipient's GroupWise Client		Each recipient's GroupWise client communicates the Opened status message to the POA by way of TCP/IP.
12 POA for Recipient's Post Office		The POA for the recipient's post office communicates the status message to the MTA by way of TCP/IP.
13 MTA for Local Domain		The MTA places the status message into the MTA "in progress" (<i>gwinprog</i>) queue.
14 MTA for Local Domain		The MTA communicates the status message to the POA for the sender's post office by way of TCP/IP.









Stage	Icon	Description
16 POA in Sender's Post Office		The POA for the sender's post office updates the sender's message database (msgn.db) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).
17 POA for Local Post Office		The POA communicates the status to the sender's GroupWise client by way of TCP/IP.
18 Sender		When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).

















TCP/IP Link Closed: Transfer between Post Offices Delayed




This message flow diagram illustrates how a GroupWise message travels from one user to another between post offices in the same domain when the TCP/IP link between the post office and the domain is closed.



Stage	Icon	Description
18 Sender		The user sends a message to recipients in a different post office in the same domain.
19 Sender's GroupWise Client		In this diagram, the access mode setting in the local post office is Client/Server Only. The GroupWise client communicates the message to the POA by way of TCP/IP.

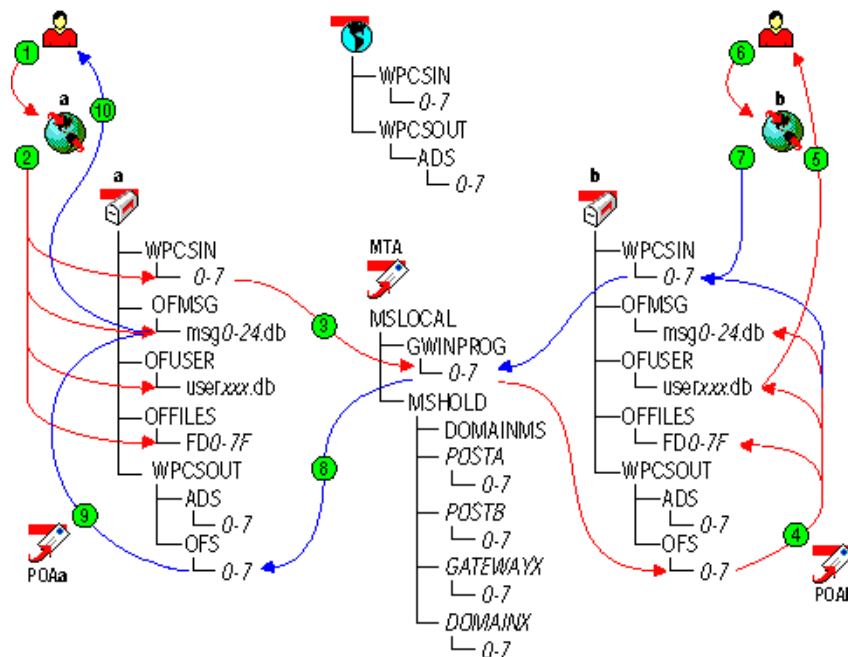
Stage	Icon	Description
 POA for Sender's Post Office	 POA	<p>The POA receives the message from the GroupWise client and performs the following actions for the sender:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office, in case the TCP/IP link to the MTA is currently closed.
 POA for Sender's Post Office	 POA	<p>The POA then attempts to communicate the message to the MTA by way of TCP/IP, but the MTA does not respond. The POA leaves the copy of the message in the MTA input queue and periodically attempts to contact the MTA. When the MTA responds again, the POA communicates the message and deletes the copy in the MTA input queue after the TCP/IP transmission to the MTA is successful.</p>
 MTA for Local Domain	 MTA	<p>The MTA receives the message and places it into the MTA "in progress" (<i>gwinprog</i>) queue.</p>
 MTA for Local Domain	 MTA	<p>The MTA then communicates the message to the POA in the recipient's post office by way of TCP/IP. When the transmission is successful, the MTA deletes the message from the MTA "in progress" (<i>gwinprog</i>) queue.</p> <p>If the TCP/IP link to the recipient's post office were closed, the message would be placed in the closed post office's holding queue in the MTA's <i>mslocal</i> directory for later transfer. The resulting message flow would be parallel to what occurs when a domain is closed. For a similar message flow that illustrates how messages to closed locations are handled, see "TCP/IP Link Closed: Transfer between Domains Delayed" on page 30.</p>


Stage	Icon	Description
 POA for Recipient's Post Office		<p>When it receives the new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office. It also communicates the Delivered status message directly to the MTA by way of TCP/IP and when that transmission is successful, the copy in the MTA input queue is deleted.
 POA for Local Post Office		<p>The POA communicates to the GroupWise client by way of TCP/IP that a new message has arrived.</p>
 Recipient's GroupWise Client		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
 Recipient		<p>Each recipient opens the message in the GroupWise client.</p>
 Recipient's GroupWise Client		<p>Each recipient's GroupWise client communicates the Opened status message to the POA by way of TCP/IP.</p>
 POA for Recipient's Post Office		<p>The POA for the recipient's post office communicates the status message to the MTA by way of TCP/IP.</p>
 MTA for Local Domain		<p>The MTA places the status message into the MTA "in progress" (<i>gwinprog</i>) queue.</p>
 MTA for Local Domain		<p>The MTA communicates the status message to the POA for the sender's post office by way of TCP/IP.</p>











Stage	Icon	Description
15 POA in Sender's Post Office		The POA for the sender's post office updates the sender's message database (msgnn.db) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).
16 POA for Local Post Office		The POA communicates the Opened status to the sender's GroupWise client by way of TCP/IP.
17 Sender		When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).





Mapped/UNC Link Open: Transfer between Post Offices Successful

This message flow diagram illustrates how a GroupWise message travels from one user to another between post offices in the same domain when the mapped or UNC link between the post office and the domain is open.



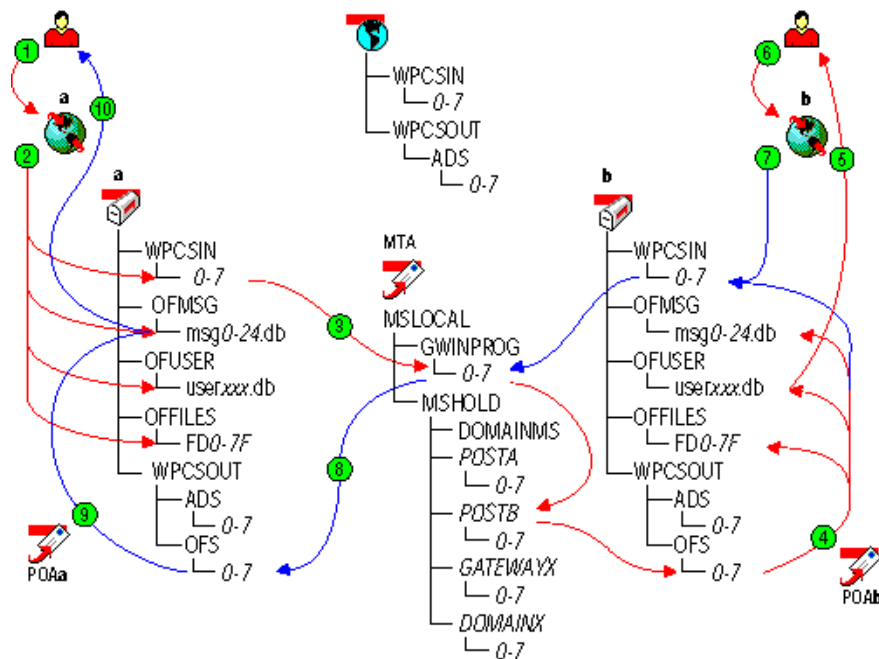
Stage	Icon	Description
1 Sender		The user sends a message to recipients in a different post office in the same domain. In this diagram, the delivery mode setting for the local post office is Use App Thresholds and the threshold is not exceeded.




Stage	Icon	Description
 Sender's GroupWise Client		<p>The GroupWise client performs the following actions for the sender:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office. <p>If client/server mode were in effect for the post office, these actions would be performed by the POA instead of the GroupWise client. See "Access Mode: Client/Server" on page 13.</p>
 MTA for Local Domain		<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See "Adjusting MTA Polling of Input Queues in the Domain and Post Offices" in "Message Transfer Agent" in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects a new message, the MTA picks up the message from the sender's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there the MTA transfers the message to the output queue in the recipient's post office. The MTA output queue is the input queue for the POA in the recipient's post office.</p>
 POA for Recipient's Post Office		<p>The POA polls its input queue regularly. When it detects a new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
 Recipient's GroupWise Client		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
 Recipient		<p>Each recipient opens the message in the GroupWise client.</p>








Stage	Icon	Description
1 Recipient's GroupWise Client		Each recipient's GroupWise client creates an Opened status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
2 MTA for Local Domain		For mapped and UNC links, the MTA scans its input queue for messages to process. You can control the rate at which the MTA scans its input queues. See "Adjusting MTA Polling of Input Queues in the Domain and Post Offices" in "Message Transfer Agent" in the <i>GroupWise 6.5 Administration Guide</i> . When it detects the status message, the MTA picks it up from the recipient's post office and transfers it to the MTA "in progress" (gwinprog) queue. From there the MTA transfers the message to the output queue in the sender's post office. The MTA output queue is the input queue for the POA in the sender's post office.
3 POA for Sender's Post Office		The POA for the sender's post office updates the sender's message database (MSGnn.DB) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).
4 Sender		When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).

Mapped/UNC Link Closed: Transfer between Post Offices Delayed

This message flow diagram illustrates how a GroupWise message travels from one user to another between post offices in the same domain when the mapped or UNC link between the post office and the domain is closed.



Stage	Icon	Description
1 Sender		<p>The user sends a message to recipients in a different post office in the same domain.</p> <p>In this diagram, the delivery mode setting for the local post office is Use App Thresholds and the threshold is not exceeded.</p>
2 Sender's GroupWise Client		<p>The GroupWise client performs the following actions for the sender:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office. <p>If client/server mode were in effect for the post office, these actions would be performed by the POA instead of the GroupWise client. See “Access Mode: Client/Server” on page 13.</p>
3 MTA for Local Domain		<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See “Adjusting MTA Polling of Input Queues in the Domain and Post Offices” in “Message Transfer Agent” in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects a new message, the MTA picks up the message from the sender's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. However, because the recipient's post office is closed, the MTA cannot complete the delivery. Therefore, the MTA stores the message in the holding queue for the recipient's post office in the <i>mslocal</i> directory.</p> <p>When the recipient's post office is open again, the MTA transfers the delayed message from the post office holding queue to the MTA output queue in the recipient's post office. The MTA output queue is the input queue for the POA in the recipient's post office.</p>

Stage	Icon	Description
4 POA for Recipient's Post Office	 POA	<p>The POA polls its input queue regularly. When it detects a new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
5 Recipient's GroupWise Client		The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.
6 Recipient		Each recipient opens the message in the GroupWise client.
7 Recipient's GroupWise Client		Each recipient's GroupWise client creates an Opened status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
8 MTA for Local Domain	 MTA	<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See “Adjusting MTA Polling of Input Queues in the Domain and Post Offices” in “Message Transfer Agent” in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects the status message, the MTA picks it up from the sender's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. Because the sender's post office is open, the MTA can immediately transfer the message to the output queue in the sender's post office. The MTA output queue is the input queue for the POA in the sender's post office.</p>
9 POA for Sender's Post Office	 POA	The POA for the sender's post office updates the sender's message database (<i>msgnn.db</i>) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).
10 Sender		When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).

3

Message Delivery to a Different Domain

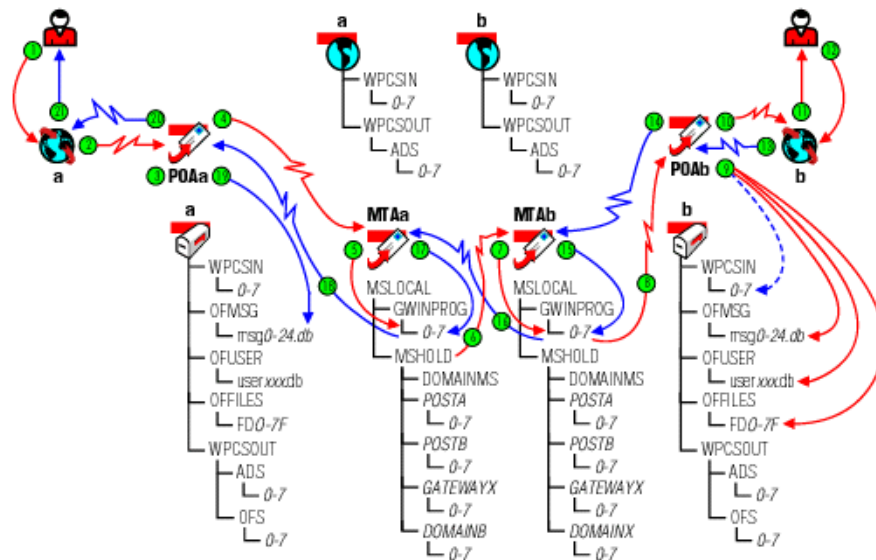
The MTA handles message transfer between domains.


- ◆ “TCP/IP Link Open: Transfer between Domains Successful” on page 27
- ◆ “TCP/IP Link Closed: Transfer between Domains Delayed” on page 30
- ◆ “Mapped/UNC Link Open: Transfer between Domains Successful” on page 33
- ◆ “Mapped/UNC Link Closed: Transfer between Domains Delayed” on page 36
- ◆ “Alternate Link Configuration: Transfer Pull” on page 38








For an explanation of link types and link protocols, see “Understanding Link Configuration” in “Domains” in the *GroupWise 6.5 Administration Guide*.









TCP/IP Link Open: Transfer between Domains Successful






This message flow diagram illustrates how a GroupWise® message travels from one user to another when the domains are connected by a TCP/IP link and the link is open.



Stage	Icon	Description
1 Sender		The user sends a message to recipients in a post office in a different domain. In this diagram, the access mode setting for the local post office is Client/Server Only.

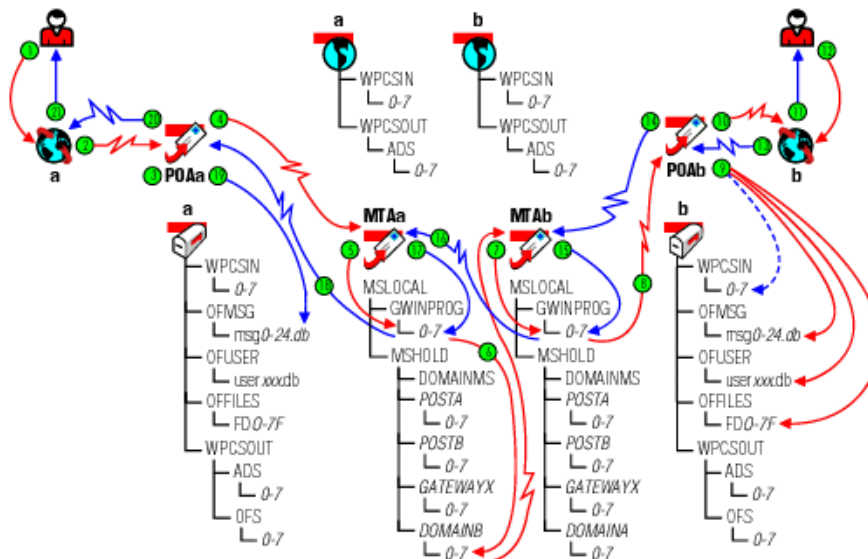
Stage	Icon	Description
2 Sender's GroupWise Client		The GroupWise client communicates the message to the POA by way of TCP/IP.
3 POA for Sender's Post Office		<p>The POA receives the message from the GroupWise client and performs the following actions for the sender:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office, in case the TCP/IP link to the MTA is currently closed.
4 POA for Sender's Post Office		<p>The POA then communicates the message to the MTA for the sender's domain by way of TCP/IP, and deletes the copy in the MTA input queue because the TCP/IP transfer to the MTA was successful.</p> <p>To see what would happen if the TCP/IP link to the MTA were closed, see "TCP/IP Link Closed: Transfer between Post Offices Delayed" on page 18.</p>
5 MTA for Sender's Domain		The MTA for the sender's domain receives the message and places it into the MTA "in progress" (<i>gwinprog</i>) queue.
6 MTA for Sender's Domain		<p>The MTA for the sender's domain then communicates the message to the MTA for the recipient's domain by way of TCP/IP.</p> <p>If the TCP/IP link to the recipient's domain were closed, the message would be placed in the closed domain's holding queue in the MTA's <i>mslocal</i> directory for later transfer. See "TCP/IP Link Closed: Transfer between Domains Delayed" on page 30.</p>
7 MTA for Recipient's Domain		The MTA for the recipient's domain receives the message and places it into the MTA "in progress" (<i>gwinprog</i>) queue.
8 MTA for Recipient's Domain		The MTA for the recipient's domain then communicates the message to the POA in the recipient's post office by way of TCP/IP.









Stage	Icon	Description
<p>9</p> <p>POA for Recipient's Post Office</p>		<p>When it receives the new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office. It also communicates the Delivered status message directly to the MTA by way of TCP/IP and when that transmission is successful, the copy in the MTA input queue is deleted.
<p>10</p> <p>POA for Recipient's Post Office</p>		<p>The POA for the recipient's post office communicates to the GroupWise client by way of TCP/IP that a new message has arrived.</p>
<p>11</p> <p>Recipient's GroupWise Client</p>		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
<p>12</p> <p>Recipient</p>		<p>Each recipient opens the message in the GroupWise client.</p>
<p>13</p> <p>Recipient's GroupWise Client</p>		<p>Each recipient's GroupWise client communicates the Opened status message to the POA by way of TCP/IP.</p>
<p>14</p> <p>POA for Recipient's Post Office</p>		<p>The POA for the recipient's post office communicates the status message to the MTA for the recipient's domain by way of TCP/IP.</p>
<p>15</p> <p>MTA for Recipient's Domain</p>		<p>The MTA for the recipient's domain places the status message into the MTA "in progress" (<i>gwinprog</i>) queue.</p>
<p>16</p> <p>MTA for Recipient's Domain</p>		<p>The MTA for the recipient's domain communicates the status message to the MTA for the sender's domain by way of TCP/IP.</p>









Stage	Icon	Description
11 MTA for Sender's Domain		The MTA for the sender's domain places the status message into the MTA "in progress" (gwinprog) queue.
12 MTA for Sender's Domain		The MTA for the sender's domain communicates the status message to the POA for the sender's post office by way of TCP/IP.
13 POA for Sender's Post Office		The POA for the sender's post office updates the sender's message database (MSGnn.DB) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).
14 POA for Sender's Post Office		The POA for the sender's post office communicates the status to the sender's GroupWise client by way of TCP/IP.
21 Sender		When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).






TCP/IP Link Closed: Transfer between Domains Delayed

This message flow diagram illustrates how a GroupWise message travels from one user to another when the domains are connected by a TCP/IP link and the link is closed.



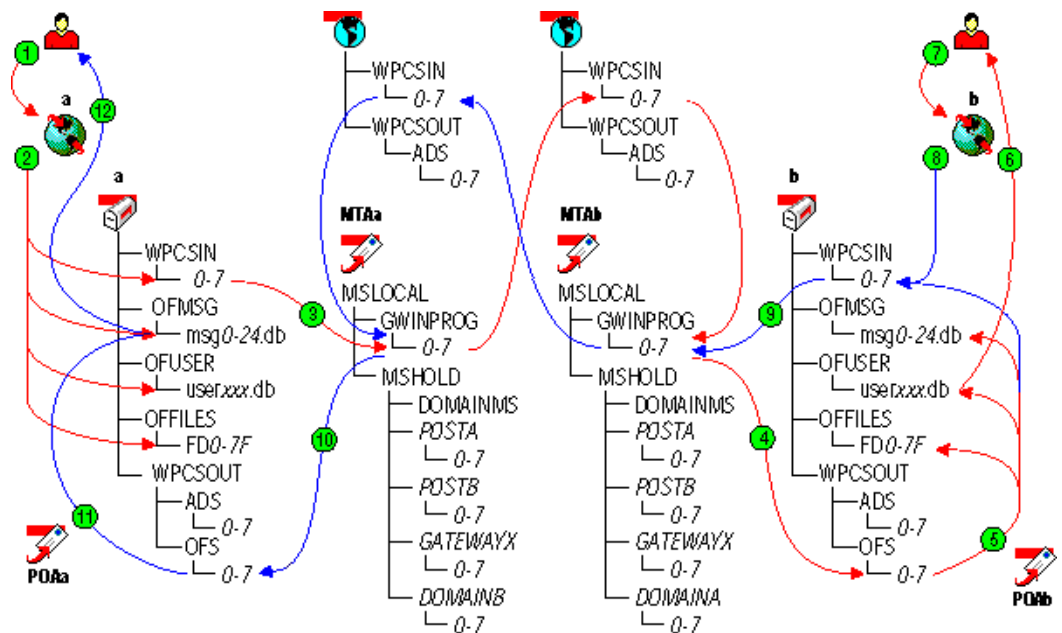
Stage	Icon	Description
1 Sender		The user sends a message to recipients in a post office in a different domain. In this diagram, the access mode setting for the local post office is Client/Server Only.
2 Sender's GroupWise Client		The GroupWise client communicates the message to the POA by way of TCP/IP.
3 POA for Sender's Post Office		The POA receives the message from the GroupWise client and performs the following actions for the sender: <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office, in case the TCP/IP link to the MTA is currently closed.
4 POA for Sender's Post Office		The POA then communicates the message to the MTA for the sender's domain by way of TCP/IP, and deletes the copy in the MTA input queue because the TCP/IP transfer to the MTA was successful. To see what would happen if the TCP/IP link to the MTA were closed, see "TCP/IP Link Closed: Transfer between Post Offices Delayed" on page 18.
5 MTA for Sender's Domain		The MTA for the sender's domain receives the message and places it into the MTA "in progress" (<i>gwinprog</i>) queue.
6 MTA for Sender's Domain		The MTA for the sender's domain then attempts to communicate the message to the MTA for the recipient's domain by way of TCP/IP, but the recipient MTA does not respond. Therefore, the MTA stores the message in its holding queue for the recipient's domain in the <i>mshold</i> directory. When the MTA in the recipient's domain responds again, the MTA for the sender's domain transfers the delayed message from the domain holding queue to the MTA in the recipient's domain by way of TCP/IP.
7 MTA for Recipient's Domain		The MTA for the recipient's domain receives the message and places it into the MTA "in progress" (<i>gwinprog</i>) queue.
8 MTA for Recipient's Domain		The MTA for the recipient's domain then communicates the message to the POA in the recipient's post office by way of TCP/IP.





Stage	Icon	Description
<p>9</p> <p>POA for Recipient's Post Office</p>		<p>When it receives the new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ♦ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ♦ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ♦ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ♦ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office. It also communicates the Delivered status message directly to the MTA by way of TCP/IP and when that transmission is successful, the copy the MTA input queue is deleted.
<p>10</p> <p>POA for Recipient's Post Office</p>		<p>The POA for the recipient's post office communicates to the GroupWise client by way of TCP/IP that a new message has arrived.</p>
<p>11</p> <p>Recipient's GroupWise Client</p>		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
<p>12</p> <p>Recipient</p>		<p>Each recipient opens the message in the GroupWise client.</p>
<p>13</p> <p>Recipient's GroupWise Client</p>		<p>Each recipient's GroupWise client communicates the Opened status message to the POA by way of TCP/IP.</p>
<p>14</p> <p>POA for Recipient's Post Office</p>		<p>The POA for the recipient's post office communicates the status message to the MTA for the recipient's domain by way of TCP/IP.</p>
<p>15</p> <p>MTA for Recipient's Domain</p>		<p>The MTA for the recipient's domain places the status message into the "in progress" (<i>gwinprog</i>) queue.</p>
<p>16</p> <p>MTA for Recipient's Domain</p>		<p>The MTA for the recipient's domain communicates the status message to the MTA for the sender's domain by way of TCP/IP.</p>

















Stage	Icon	Description
17 MTA for Sender's Domain		The MTA for the sender's domain places the status message into the MTA "in progress" (gwinprog) queue.
18 MTA for Sender's Domain		The MTA for the sender's domain communicates the status message to the POA for the sender's post office by way of TCP/IP.
19 POA for Sender's Post Office		The POA for the sender's post office updates the sender's message database (msgnn.db) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).
20 POA for Sender's Post Office		The POA for the sender's post office communicates the status to the sender's GroupWise client by way of TCP/IP.
21 Sender		When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).

Mapped/UNC Link Open: Transfer between Domains Successful

This message flow diagram illustrates how a GroupWise message travels from one user to another when the domains are connected by a mapped or UNC link and the link is open.

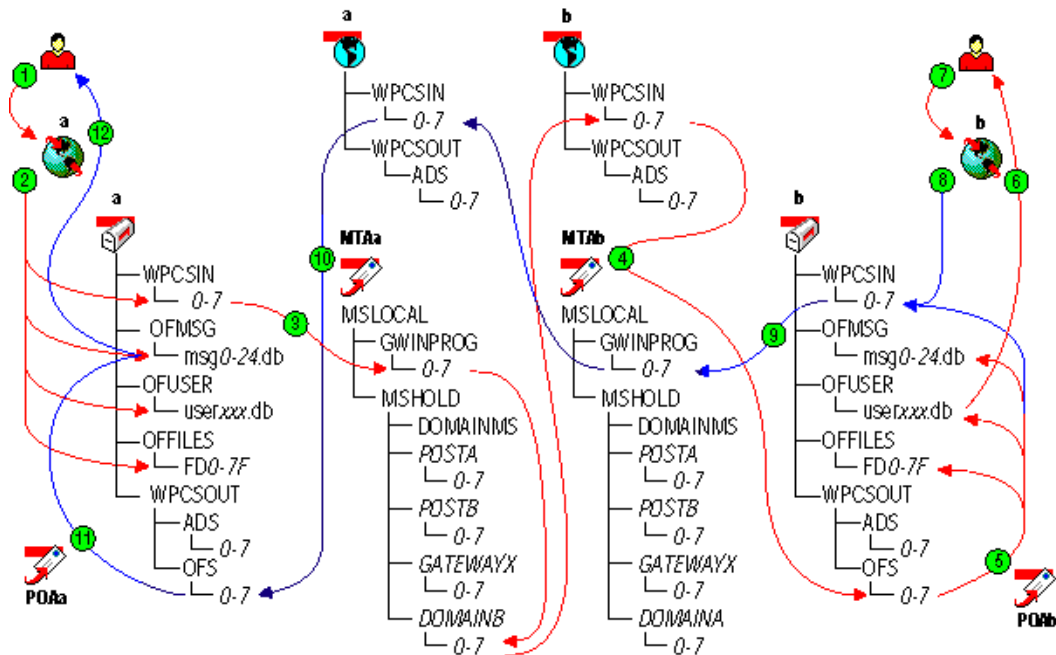




Stage	Icon	Description
1 Sender		<p>The user sends a message to recipients in a post office in a different domain.</p> <p>In this diagram, the delivery mode setting in the local post office is Use App Thresholds and the threshold is not exceeded.</p>
2 Sender's GroupWise Client		<p>The GroupWise client performs the following actions for the sender:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. ◆ Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office. <p>If client/server mode were in effect for the post office, these actions would be performed by the POA instead of the GroupWise client. See “Access Mode: Client/Server” on page 13.</p>
3 MTA for Sender's Domain		<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See “Adjusting MTA Polling of Input Queues in the Domain and Post Offices” in “Message Transfer Agent” in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects a new message, the MTA picks up the message from the sender's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there, the MTA in the sender's domain transfers the message to the output queue in the recipient's domain.</p>
4 MTA for Recipient's Domain		<p>The MTA in the recipient's domain also polls its input queue. When it detects a new message, the MTA in the recipient's domain picks up the message from its domain input queue and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there, the MTA in the recipient's domain transfers it to the output queue in the recipient's post office. The MTA output queue is the input queue for the POA in the recipient's post office.</p>







Stage	Icon	Description
 POA for Recipient's Post Office	 POA	<p>The POA polls its input queue regularly. When it detects a new message, the POA for the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
 Recipient's GroupWise Client		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
 Recipient		<p>Each recipient opens the message in the GroupWise client.</p>
 Recipient's GroupWise Client		<p>Each recipient's GroupWise client creates an Opened status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.</p>
 MTA for Recipient's Domain	 MTA	<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See “Adjusting MTA Polling of Input Queues in the Domain and Post Offices” in “Message Transfer Agent” in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects the status message, the MTA picks it up from the recipient's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there, the MTA in the recipient's domain transfers the status message to the output queue in the sender's domain.</p>
 MTA for Sender's Domain	 MTA	<p>The MTA in the sender's domain also polls its domain input queues. When it detects the status message, the MTA in the sender's domain picks it up and transfers it to the output queue in the sender's post office. The MTA output queue is the input queue for the POA in the sender's post office.</p>
 POA for Sender's Post Office	 POA	<p>The POA in the sender's post office updates the sender's message database (<i>MSGnn.DB</i>) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).</p>
 Sender		<p>When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).</p>





Mapped/UNC Link Closed: Transfer between Domains Delayed

This message flow diagram illustrates how a GroupWise message travels from one user to another when the domains are connected by a mapped or UNC link and the link is closed.



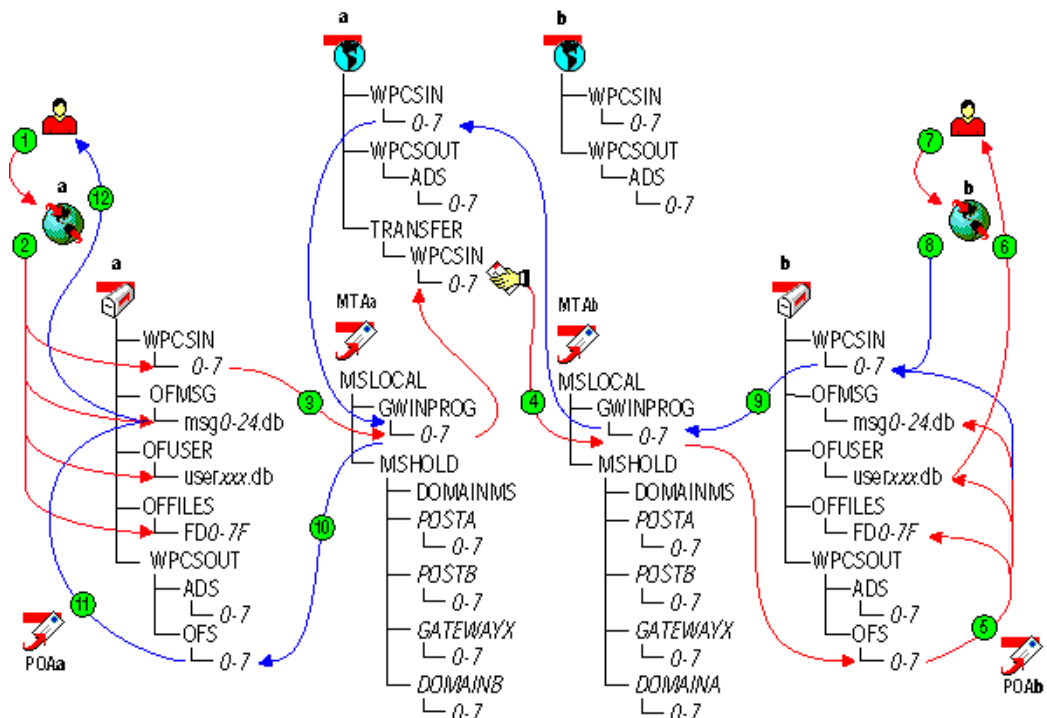
Stage	Icon	Description
1 Sender		The user sends a message to recipients in a post office in a different domain. In this diagram, the delivery mode setting for the local post office is Use App Thresholds and the threshold is not exceeded.
2 Sender's GroupWise Client		The GroupWise client performs the following actions for the sender: <ul style="list-style-type: none"> • Adds the message to the message database (<code>msgnn.db</code>) assigned to the sender. • Creates a pointer in the sender's user database (<code>userxxx.db</code>) so the message appears in the sender's mailbox as a sent item. • Places attachments larger than 2 KB in the <code>post_office\offiles\fd0-f6</code> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) • Creates a copy of the message in the appropriate priority 0-7 subdirectory of the sender's MTA input queue. <p>If client/server mode were in effect for the post office, these actions would be performed by the POA instead of the GroupWise client. See “Access Mode: Client/Server” on page 13.</p>





Stage	Icon	Description
<p>3</p> <p>MTA for Sender's Domain</p>		<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See "Adjusting MTA Polling of Input Queues in the Domain and Post Offices" in "Message Transfer Agent" in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects a new message, the MTA picks up the message from the sender's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. However, because the recipient's domain is closed, the MTA in the sender's domain cannot complete the delivery. Therefore, the MTA in the sender's domain stores the message in its holding queue for the recipient's domain in the <i>mslocal</i> directory. From there, the MTA in the sender's domain transfers the message to the output queue in the recipient's domain.</p> <p>When the recipient's domain is open again, the MTA in the sender's domain transfers the delayed message from the domain holding queue to the MTA output queue in the recipient's domain.</p>
<p>4</p> <p>MTA in Recipient's Domain</p>		<p>The MTA in the recipient's domain also polls its input queue. When it detects a new message, the MTA in the recipient's domain picks up the message from its domain input queue and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there, the MTA transfers the message to the output queue in the recipient's post office. The MTA output queue is the input queue for the POA in the recipient's post office.</p>
<p>5</p> <p>POA in Recipient's Post Office</p>		<p>The POA polls its input queue regularly. When it detects a new message, the POA in the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
<p>6</p> <p>Recipient's GroupWise Client</p>		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
<p>7</p> <p>Recipient</p>		<p>Each recipient opens the message in the GroupWise client.</p>
<p>8</p> <p>Recipient's GroupWise Client</p>		<p>Each recipient's GroupWise client creates an Opened status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.</p>









Stage	Icon	Description
<p>9</p> <p>MTA for Recipient's Domain</p>		<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See "Adjusting MTA Polling of Input Queues in the Domain and Post Offices" in "Message Transfer Agent" in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects the status message, the MTA picks it up from the recipient's post office and transfers it to the MTA "in progress" (gwinprog) queue. From there, the MTA in the recipient's domain transfers the status message to the output queue in the sender's domain.</p>
<p>10</p> <p>MTA for Sender's Domain</p>		<p>The MTA in the sender's domain also polls its domain input queues. When it detects the status message, the MTA in the sender's domain picks it up and transfers it to its output queue in the sender's post office. The MTA output queue is the input queue for the POA in the sender's post office.</p>
<p>11</p> <p>POA for Sender's Post Office</p>		<p>The POA in the sender's post office updates the sender's message database (msgnn.db) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).</p>
<p>12</p> <p>Sender</p>		<p>When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).</p>

Alternate Link Configuration: Transfer Pull

This message flow diagram illustrates how a GroupWise message travels from one user to another when the domains are connected by a transfer pull configuration.



Stage	Icon	Description
1 Sender		The user sends a message to recipients in a post office in a different domain. In this diagram, the delivery mode setting for the local post office is Use App Thresholds and the threshold is not exceeded.
2 Sender's GroupWise Client		The GroupWise client performs the following actions for the sender: <ul style="list-style-type: none"> • Adds the message to the message database (<i>msgnn.db</i>) assigned to the sender. • Creates a pointer in the sender's user database (<i>userxxx.db</i>) so the message appears in the sender's mailbox as a sent item. • Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) • Creates a copy of the message in the appropriate priority 0-7 subdirectory of the MTA input queue in the sender's post office. <p>If client/server mode were in effect for the post office, these actions would be performed by the POA instead of the GroupWise client. See “Access Mode: Client/Server” on page 13.</p>
3 MTA for Sender's Domain		For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See “Adjusting MTA Polling of Input Queues in the Domain and Post Offices” in “Message Transfer Agent” in the <i>GroupWise 6.5 Administration Guide</i>.
		When it detects a new message, the MTA picks up the message from the sender's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there, the MTA for the sender's domain would normally transfer the message to the output queue in the recipient's domain. However, in this transfer pull configuration, the MTA for the sender's domain does not have access to the recipient's domain. Therefore, the MTA for the sender's domain transfers the message into the appropriate priority 0-7 subdirectory in the transfer directory in its own domain. It is now the job of the MTA for the recipient's domain to pick up (or 'pull') the message from the sender's domain to the recipient's domain.
		See “Using a Transfer Pull Configuration” in “Message Transfer Agent” in the <i>GroupWise 6.5 Administration Guide</i>.
4 MTA for Recipient's Domain		The MTA for the recipient's domain polls its input queue in the local domain and any transfer directories in other domains. The transfer pull configuration is the only situation where an MTA scans directories outside its own directory structure.
		When it detects a new message in a transfer directory of another domain, the MTA picks up (or "pulls") the message from the other domain and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there, the MTA for the recipient's domain transfers the message to the output queue in the recipient's post office. The MTA output queue is the input queue for the POA in the recipient's post office.

Stage	Icon	Description
<p>6 POA for Recipient's Post Office</p>		<p>The POA polls its input queue regularly. When it detects a new message, the POA in the recipient's post office performs the following actions:</p> <ul style="list-style-type: none"> ♦ Adds the message to the message database (<i>msgnn.db</i>) corresponding to the one assigned to the sender. ♦ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ♦ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ♦ Creates a Delivered status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.
<p>8 Recipient's GroupWise Client</p>		<p>The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.</p>
<p>7 Recipient</p>		<p>Each recipient opens the message in the GroupWise client.</p>
<p>9 Recipient's GroupWise Client</p>		<p>Each recipient's GroupWise client creates an Opened status message in the appropriate priority 0-7 subdirectory of the MTA input queue in the recipient's post office.</p>
<p>4 MTA for Recipient's Domain</p>		<p>For mapped and UNC links, the MTA scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See "Adjusting MTA Polling of Input Queues in the Domain and Post Offices" in "Message Transfer Agent" in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects the status message, the MTA picks it up from the recipient's post office and transfers it to the MTA "in progress" (<i>gwinprog</i>) queue. From there the MTA in the recipient's domain transfers the status message to the output queue in the sender's domain.</p>
<p>10 MTA for Sender's Domain</p>		<p>The MTA in the sender's domain also polls its domain input queues. When it detects the status message, the MTA in the sender's domain picks it up and transfers it to its output queue in the sender's post office. The MTA output queue is the input queue for the POA in the sender's post office.</p>
<p>11 POA for Sender's Post Office</p>		<p>The POA in the sender's post office updates the sender's message database (<i>MSGnn.DB</i>) with the Delivered status information (and possibly Opened as well if the recipient has opened the message).</p>
<p>12 Sender</p>		<p>When the sender checks the sent items in his or her mailbox in the GroupWise client, the message displays a status of Delivered for each recipient (and possibly Opened as well if the recipient has opened the message).</p>

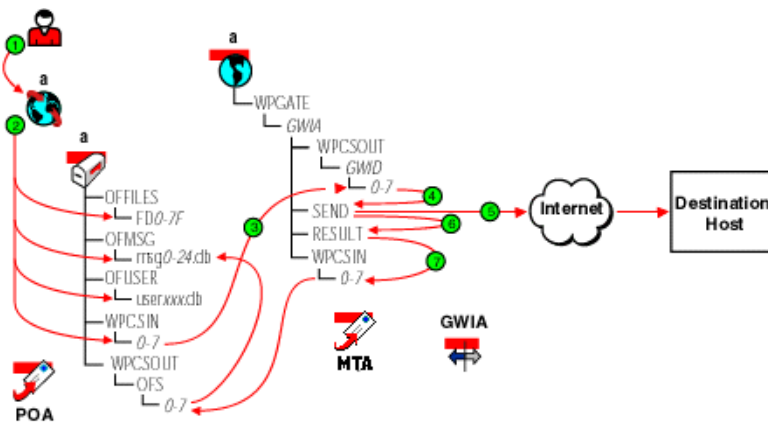
4


Message Delivery to and from the Internet











- ♦ “Message Transfer from GroupWise to the Internet” on page 41
- ♦ “Message Transfer from the Internet to GroupWise” on page 43



Message Transfer from GroupWise to the Internet

The diagram below shows the flow of outbound messages through the GroupWise® directory structure to the Internet.



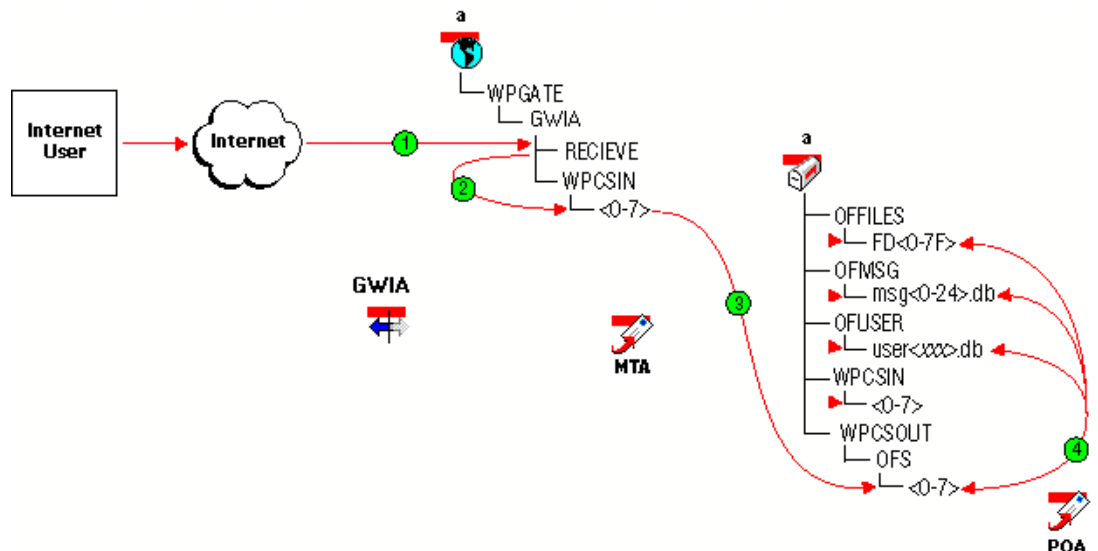
Stage	Actor	Action
1 Sender		The user enters information for the message, including the recipient's Internet address, and sends it. In this diagram, the delivery mode in the local post office is Use App Thresholds and the threshold is not exceeded.







Stage	Actor	Action
 Sender's GroupWise Client		<p>The GroupWise client performs the following actions for the sender:</p> <ul style="list-style-type: none"> • Adds the message to the message database (<i>msgnn.db</i> file) assigned to the sender. • Creates a pointer in the sender's user database (<i>userxxx.db</i> file) so the message appears in the sender's Out Box. • Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory assigned to the sender and creates pointers from the message to its attachments. • Creates a copy of the message in the appropriate priority 0-7 subdirectory of the sender's MTA input queue. <p>If client/server mode were in effect for the post office, these actions would be performed by the POA instead of the GroupWise client. See "Access Mode: Client/Server" on page 13.</p>
 MTA for Sender's Domain		<p>The MTA for the sender's domain polls its input queue as specified by Scan Cycle in Agent Settings in ConsoleOne®. When it detects a new message, the MTA recognizes that it must be sent through a gateway and places the message in its output queue in the program's directory structure: <i>domain\wpgate\gwia\wpcout\id0-7</i>.</p>
 Internet Agent for Sender's Domain		<p>The Internet Agent picks up the file in binary-encrypted format from the <i>wpcout\id0-7</i> directory and converts it. It UUencodes any attachments unless the Internet Agent is configured with the <i>/mime</i> switch to use MIME specifications or unless the sender specified the MIME foreign domain.</p> <p>If MIME is specified, the Internet Agent encodes the message attachments with the appropriate encoding scheme, as specified in the <i>mimetype.cfg</i> file.</p> <p>When the message file is built, the Internet Agent saves it with S as the first character of the filename and places the file in the <i>wpgate\gwia\send</i> directory.</p>
 Internet Agent for Sender's Domain		<p>The Internet Agent processes the message in the <i>domain\wpgate\gwia\send</i> directory and sends the message to the destination host across the Internet.</p>
 Internet Agent for Sender's Domain		<p>The Internet Agent creates a file, R*.*, that records the SMTP reply codes (error messages or transmission confirmation) in the <i>wpgate\gwia\result</i> directory. After the Internet Agent completes the transmission with the destination host, it moves the S file from the send directory to the result directory.</p>

Stage	Actor	Action
 Internet Agent for Sender's Domain		<p>The Internet Agent analyzes the files in the result directory, comparing the SMTP reply codes in the R file.</p> <p>If the Internet Agent was successful at sending the message (meaning the R file had the 250 OK SMTP reply code), then the two files are deleted. The Internet Agent sends a Transferred status message to the sender's Sent Items folder.</p> <p>If the R file has a temporary transmission error (meaning it has a 400-level SMTP reply code such as 450 Host Down), the Internet Agent moves the S file to the defer directory and re-queues the message to the send directory according to the Retry Schedule. At the end of the Retry Schedule, if the Internet Agent is still unsuccessful at transmitting the message, the Internet Agent moves the S file to the gwprob directory and deletes all the schedule files in the defer/work directory. The Internet Agent sends an Undeliverable status message to the sender's Mailbox and Sent Items folder.</p> <p>If the R file has a fatal rather than a temporary error (meaning it has a 500-level SMTP reply code such as 550 Host Unknown), the Internet Agent sends an Undeliverable status message immediately to the sender's Mailbox and Sent Items folder. The Internet Agent deletes the R file and moves the S file to the gwprob directory.</p> <p>Only the msgnn.db (not the userxxx.db) is updated when the Internet Agent sends a status message for the user's Sent Items folder.</p>

Message Transfer from the Internet to GroupWise

The diagram below shows the flow of inbound messages from the Internet through the GroupWise directory structure to the GroupWise recipient.



Stage	Actor	Action
1 Internet Agent for Recipient's Domain	 GWIA	A message is sent from an Internet user to a GroupWise user and the Internet Agent places the message in the <code>wpgate\gwia\receive</code> directory.
2 Internet Agent for Recipient's Domain	 GWIA	The Internet Agent polls the receive directory, converts the message to GroupWise format and places it in the <code>wpgate\gwia\wpcsin\0-7</code> directory, where 0-7 is one of the priority directories from 0-7. The Internet Agent puts messages only in the 4 directory.
3 MTA for Recipient's Domain	 MTA	The Message Transfer Agent polls the <code>domain\wpgate\gwia\wpcsin\fd0-7f</code> directory and moves the file to the <code>post_office\wpcout\ofs\0-7</code> directory.
4 POA for Recipient's Post Office	 POA	The POA polls its input queue regularly. When it detects a new message, the POA in the recipient's post office performs the following actions: <ul style="list-style-type: none"> ♦ Adds the message to the message database (<code>msgnn.db</code> file) corresponding to the one assigned to the sender. ♦ Creates a pointer in the recipient's user database (<code>userxxx.db</code> file), so the message appears in the recipient's Mailbox and updates the notification information in the user database so the recipient can be notified of the message. ♦ Places attachments larger than 2 KB in the <code>post_office\offiles\fd0-f6</code> subdirectory corresponding to the one assigned to the sender and creates pointers from the message to its attachments. (For database efficiency, messages and recipient lists larger than 2 KB are also handled as attachments.)
5 Recipient's GroupWise Client		The Notify component of the recipient's GroupWise client notifies the recipient that a new message has arrived.
6 Recipient		Each recipient opens the message in the GroupWise client.

5

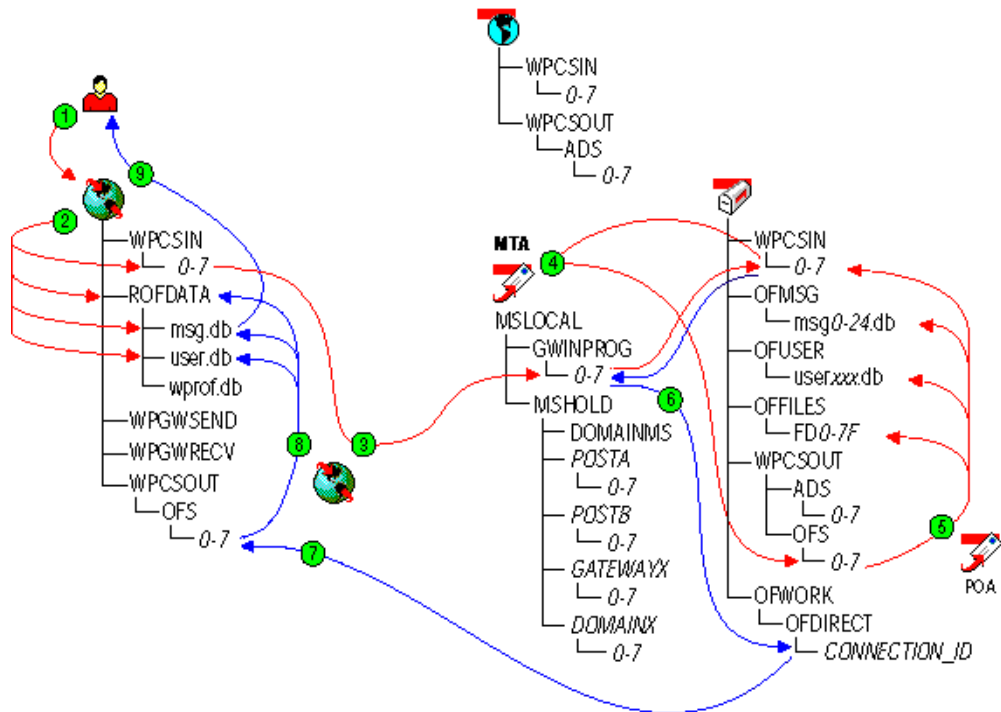
Message Delivery to and from GroupWise Remote







GroupWise® client users can still access information in their mailboxes even when not connected to the network.







- ♦ “Mapped/UNC Link for "Hit the Road"” on page 45
- ♦ “Modem Link to the Remote User through the Async Gateway” on page 47




Mapped/UNC Link for "Hit the Road"







This message flow diagram illustrates how a user who will be away from the network prepares to access GroupWise from a remote location.



Stage	Icon	Description
<p>1 Remote User</p>		The GroupWise user requests items from the master mailbox to be downloaded to the Remote mailbox in preparation for disconnecting from the master GroupWise system. For example, the user could be preparing a laptop computer for use away from the network using the Hit the Road feature of the GroupWise client.
<p>2 GroupWise Remote</p>		When the GroupWise Remote user sends a request for items from the master mailbox, GroupWise Remote places the request in the <code>wpcsin\1</code> subdirectory of the remote input queue on the user's remote computer.
<p>3 GroupWise Remote</p>		When the GroupWise Remote user establishes a direct network connection between the remote computer and the master GroupWise system, the GroupWise client places the request into the priority <code>1</code> subdirectory of the MTA input queue (WPC SIN) in the post office to which the GroupWise Remote user has connected. The MTA then transfers the request to the post office where the GroupWise Remote user's master mailbox is located.
<p>4 MTA for Remote User's Domain</p>		<p>For mapped and UNC links, the MTA for the GroupWise Remote user's domain scans its input queue in each post office for messages to process. You can control the rate at which the MTA scans its input queues. See “Adjusting MTA Polling of Input Queues in the Domain and Post Offices” in <i>“Message Transfer Agent”</i> in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects the new request from the GroupWise Remote user, the MTA picks it up from its domain input queue and transfers it to the post office where the GroupWise Remote user's master mailbox is located. The MTA output queue is the input queue for the POA in the post office where the GroupWise Remote user's master mailbox is located.</p>
<p>5 POA for Remote User's Post Office</p>		<p>The POA polls its input queue regularly. It quickly detects the new request from the GroupWise Remote user.</p> <p>When the GroupWise Remote user sends a request for items from the master mailbox, the POA performs the following actions:</p> <ul style="list-style-type: none"> ♦ Gathers the requested items from the GroupWise Remote user's master mailbox (<code>msgnn.db</code> and <code>userxxx.db</code>). ♦ Gathers any attachments for requested items from the <code>post_office\offiles\fd0-f6</code> subdirectory in the GroupWise Remote user's post office. ♦ Compiles the information into a response file and places it in the priority <code>1</code> subdirectory of the MTA input queue (<code>wpcsin</code>) for return to the GroupWise Remote user.
<p>6 MTA for Remote User's Domain</p>		When the MTA in the GroupWise Remote user's domain detects the response for the GroupWise Remote user, the MTA picks it up from its post office input queue and transfers it to its output queue in the <code>ofdirect</code> directory in the post office where the GroupWise Remote user's master mailbox is located. The MTA output queue in the <code>ofdirect</code> directory is the input queue for the GroupWise client on the remote computer.

Stage	Icon	Description
1 Remote User		<p>The GroupWise user, who is not currently connected to the network, sends a message to another GroupWise user or creates a request for items from the master mailbox.</p> <p>This message flow diagram illustrates only the simplest case where the recipient is in the same post office as the GroupWise Remote user's master mailbox. If the GroupWise Remote user sends a message to a user in any other post office, the MTA would route the message to the appropriate destination.</p>
2 GroupWise Remote		<p>When the GroupWise Remote user sends a message to another GroupWise user, GroupWise Remote performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the GroupWise Remote message database (msg.db) on the user's remote computer. ◆ Creates a pointer in the GroupWise Remote user database (user.db) so the message appears in the mailbox on the user's remote computer as a sent item. ◆ Places attachments larger than 2 KB in the rofddata directory and creates pointers from the message to its attachments on the user's remote computer. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a copy of the message in the wpcsin\1 subdirectory of the remote input queue on the user's remote computer. <p>When the GroupWise Remote user sends a request for items from the master mailbox, GroupWise Remote places the request in the WPC SIN\1 subdirectory of the remote input queue on the user's remote computer.</p>
3 GroupWise Remote		<p>When the user initiates the modem connection, the GroupWise client on the remote computer polls the wpcsin\1 subdirectory and compresses the outgoing messages and/or requests into a file. If the compressed file totals over 50 KB, additional compressed files are created.</p> <p>The GroupWise client next moves the compressed files into the wpgwsend directory, then dials in to the gateway and logs in.</p>
4 GroupWise Remote (Modem Connection)		<p>After the login process is completed, GroupWise Remote transmits the compressed message/request file(s) across the modem connection to the system where the user's master mailbox is located.</p>
5 Async Gateway		<p>The GroupWise Async Gateway picks up the message/request files and decompresses them.</p>
6 Async Gateway		<p>The Async Gateway moves the message/request files to the MTA input queue.</p>

Stage	Icon	Description
<p>7</p> <p>MTA for Remote User's Domain</p>		<p>For mapped and UNC links, the MTA for the GroupWise Remote user's domain scans its input queue for messages to process. You can control the rate at which the MTA scans its input queues. See "Adjusting MTA Polling of Input Queues in the Domain and Post Offices" in "Message Transfer Agent" in the <i>GroupWise 6.5 Administration Guide</i>.</p> <p>When it detects the new message or request from the GroupWise Remote user, the MTA picks it up from its domain input queue and transfers it to its output queue in the post office where the GroupWise Remote user's master mailbox is located. The MTA output queue is the input queue for the POA in the post office where the GroupWise Remote user's master mailbox is located.</p> <p>This message flow diagram illustrates only the simplest case where the recipient is in the same post office as the GroupWise Remote user's master mailbox. If the GroupWise Remote user sends a message to a user in any other post office, the MTA would route the message to the appropriate destination.</p>
<p>8</p> <p>POA for Remote User's Post Office</p>		<p>The POA polls its input queue regularly. It quickly detects the new message or request from the GroupWise Remote user.</p> <p>When the GroupWise Remote user sends a message, the POA performs the following actions:</p> <ul style="list-style-type: none"> ◆ Adds the message to the GroupWise Remote user's message database (<i>msgnn.db</i>). This updates the GroupWise Remote user's master mailbox. ◆ Creates a pointer in the recipient's user database (<i>userxxx.db</i>) so the new message appears in the recipient's mailbox and updates the notification information in the user database so the recipient can be notified of the message. ◆ Places attachments larger than 2 KB in the <i>post_office\offiles\fd0-f6</i> subdirectory in the GroupWise Remote user's post office and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) ◆ Creates a "Delivered" status message in the priority 1 subdirectory of the GroupWise Remote user's MTA input queue (<i>wpcsin</i>). <p>When the GroupWise Remote user sends a request for items from the master mailbox, GroupWise Remote performs the following actions:</p> <ul style="list-style-type: none"> ◆ Gathers the requested items from the GroupWise Remote user's master mailbox (<i>msgnn.db</i>). ◆ Gathers any attachments for requested items from the <i>post_office\offiles\fd0-f6</i> subdirectory in the GroupWise Remote user's post office. ◆ Compiles the information into a response file and places it in the priority 1 subdirectory of the MTA input queue for return to the GroupWise Remote user.
<p>9</p> <p>MTA in Remote User's Domain</p>		<p>When the MTA in the GroupWise Remote user's domain detects the response for the GroupWise Remote user, the MTA picks it up from its post office input queue and transfers it to its output queue in the Async Gateway directory under <i>wpgate</i> in the GroupWise Remote user's domain. The MTA output queue in the Async Gateway directory is the input queue for the Async Gateway.</p>

Stage	Icon	Description
10 Async Gateway		The Async Gateway picks up the response file and moves it to the priority 1 subdirectory of its output queue.
11 Async Gateway		<p>If the connection to the GroupWise Remote user is still active, the Async Gateway compresses the response files and moves them to the CMP directory.</p> <p>If the connection is no longer available, the response files wait in the <i>connection_id\1</i> subdirectory until a new connection is established by the GroupWise Remote user.</p>
12 Async Gateway (Modem Connection)		The Async Gateway transmits the response files through the modem connection to GroupWise Remote on the remote computer.
13 GroupWise Remote		The GroupWise client on the remote computer decompresses the response files and places them in its input queue on the GroupWise Remote user's computer.
14 GroupWise Remote		<p>Taking the items from its input queue, the GroupWise client performs the following actions for the GroupWise Remote user:</p> <ul style="list-style-type: none"> ◆ Updates the GroupWise Remote message database (msg.db) with any items requested from the GroupWise Remote user's master mailbox. ◆ Creates pointers in the GroupWise Remote user database (user.db) so the messages gathered from the master mailbox appear in the GroupWise Remote user's mailbox. ◆ Places any requested attachments larger than 2 KB in the rofdata directory and creates pointers from the message to its attachments. (For database efficiency, messages and distribution lists larger than 2 KB are also handled as attachments.) <p>The user's Remote mailbox now contains current copies of requested items from the user's master mailbox, plus any messages received in the user's master mailbox from other GroupWise users.</p>
15 Remote User		The GroupWise user can now review current GroupWise mail when the connection to the master GroupWise system is no longer available.

6

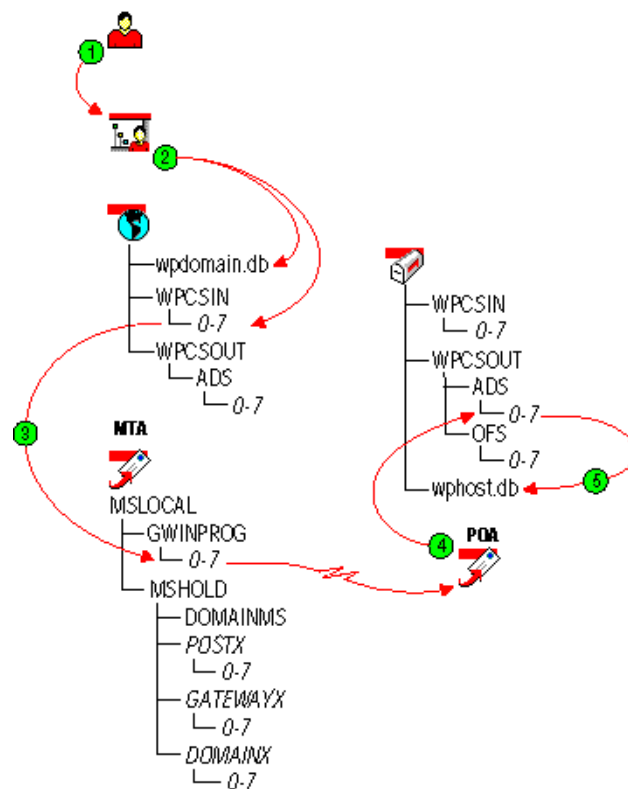
Administrative Database Update






NetWare® Administrator and the agents handle database updates throughout the GroupWise® system.

- ♦ “TCP/IP Link between the Domain and the Post Office” on page 51
- ♦ “Mapped/UNC Link between the Domain and the Post Office” on page 521

TCP/IP Link between the Domain and the Post Office

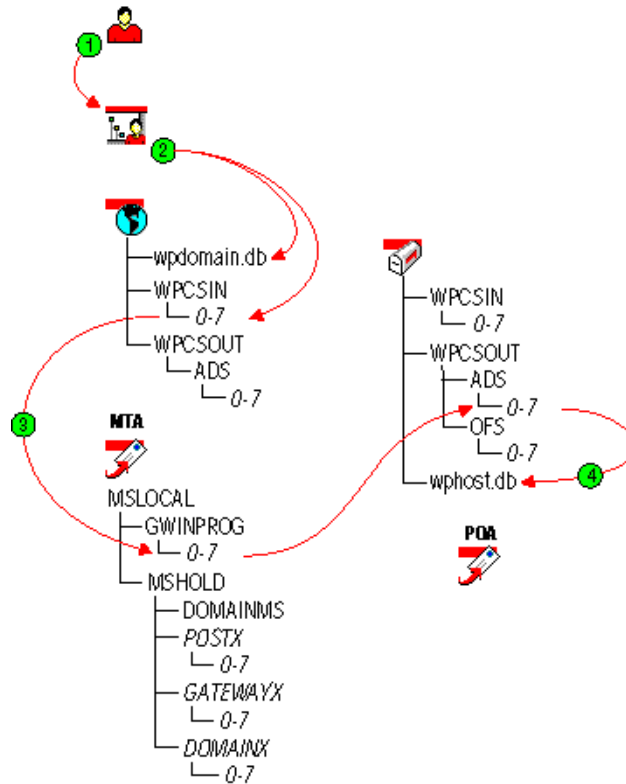
This message flow diagram illustrates how an administrative message, such as a database update request, passes from ConsoleOne® to the agents so that databases are updated throughout the GroupWise system. In this diagram, the MTA transfers the administrative message to the POA by way of TCP/IP.







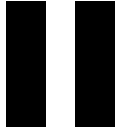
Stage	Actor	Action
1 GroupWise Administrator		The administrator uses GroupWise Administrator in ConsoleOne to add, modify, or delete a GroupWise object in a single-domain, single-post office GroupWise system. An object could be a GroupWise user, resource, distribution list, post office, secondary domain, and so on.
2 ConsoleOne		ConsoleOne performs the following actions: <ul style="list-style-type: none"> • Updates the domain database (wpdomain.db) to reflect the addition, modification, or deletion performed in GroupWise Administrator. • Creates an administrative message in the priority 2 subdirectory of the domain's MTA input queue (wpcsin) to replicate the update.
3 MTA for Domain		The MTA for the domain transfers the administrative message to the MTA "in progress" (gwinprog) queue. From there, the MTA communicates the administrative message to the POA in the post office by way of TCP/IP. The administrative message notifies the POA a GroupWise object has been added, modified, or deleted. Historical Note: In earlier versions of GroupWise, this function of the POA was handled by a separate agent, the Administration Agent (ADA). The ADA no longer exists in GroupWise.
4 POA for Post Office		The POA creates a copy of the administrative message in the priority 2 subdirectory of the administrative input queue (wpcout\ads) in the post office. After the update is made successfully, the copy will be deleted.
5 POA for Post Office		The POA updates the post office database (wphost.db) to reflect the addition, modification, or deletion performed in GroupWise Administrator and deletes the administrative message from its administrative input queue.

Mapped/UNC Link between the Domain and the Post Office

This message flow diagram illustrates how an administrative message, such as a database update request, passes from ConsoleOne to the agents so that databases are updated throughout the GroupWise system. In this diagram, the MTA writes the administrative message into the POA input queue in the post office.



Stage	Actor	Action
1 GroupWise Administrator		The administrator uses GroupWise Administrator in ConsoleOne to add, modify, or delete a GroupWise object in a single-domain, single-post office GroupWise system. An object could be a GroupWise user, resource, distribution list, post office, secondary domain, and so on.
2 ConsoleOne		ConsoleOne performs the following actions: <ul style="list-style-type: none"> • Updates the domain database (wpdomain.db) to reflect the addition, modification, or deletion performed in GroupWise Administrator. • Creates an administrative message in the priority 2 subdirectory of the domain's MTA input queue (wpcsin) to replicate the update.
3 MTA for Domain		The MTA for the domain transfers the administrative message to the MTA "in progress" (gwinprog) queue. From there, the MTA transfers the administrative message to the output queue in the post office. The MTA output queue is the input queue for the POA in the post office. The administrative message notifies the POA a GroupWise object has been added, modified, or deleted. Historical Note: In earlier versions of GroupWise, this function of the POA was handled by a separate agent, the Administration Agent (ADA). The ADA no longer exists in GroupWise.
4 POA for Post Office		The POA picks up the message from its administrative input queue in the post office, then updates the post office database (wphost.db) to reflect the addition, modification, or deletion performed in GroupWise Administrator.



Directory Structure Diagrams

This part of *Troubleshooting 3: Message Flow and Directory Structure* helps you understand the structure of GroupWise® message transfer/storage directories (such as domains and post offices) and software installation directories.

- ◆ “Message Transfer/Storage Directories” on page 57
- ◆ “Agent Installation Directories” on page 97
- ◆ “Software Distribution Directory” on page 137
- ◆ “GroupWise Client Installation Directories” on page 153














7













Message Transfer/Storage Directories

Message transfer and storage directories are the locations through which messages pass as they travel from user to user through your GroupWise® system.

- ◆ “Domain Directory” on page 57
- ◆ “Post Office Directory” on page 64
- ◆ “MTA Local Queue Directory” on page 76
- ◆ “Internet Agent Queue Directory” on page 80
- ◆ “WebAccess Agent Queue Directory” on page 86
- ◆ “Caching Mailbox Directory” on page 89
- ◆ “Remote Mailbox Directory” on page 93

Domain Directory

 <i>domain</i>	Domain directory
 <i>wpcsin</i>	MTA input queue directory
 0	Live interactive requests
 1	Other interactive requests
 2	High priority messages
 3	High priority status responses
 4	Normal priority messages
 5	Normal priority status responses
 6	Low priority messages
 7	Low priority status responses
 <i>wptools</i>	Supporting program directory
 <i>wpgate</i>	GroupWise gateway directory
 <i>wpcout</i>	MTA output queue directory

 ads	MTA admin thread input queue directory
 0	Restart requests
 1	Directory synchronization requests
 2	Database updates
 3	Reserved; not currently used
 4	Reserved; not currently used
 5	Reserved; not currently used
 6	Reserved; not currently used
 7	Reserved; not currently used
 css	MTA input queue directory for administrative messages
 0	MTA restart requests
 1	Statistics requests
 2	Other non-priority administrative requests
 3	Reserved; not currently used
 4	Reserved; not currently used
 5	Reserved; not currently used
 6	Reserved; not currently used
 7	Reserved; not currently used
 problem	Directory for undeliverable messages
 mtaname	Domain name identifier
 wpdomain.db	Domain database
 wpdomain.dc	Data dictionary for 4.x domain databases
 wphost.dc	Data dictionary for 4.x post office databases
 gwdom.dc	Data dictionary for 5.x and 6.x domain databases
 gwpo.dc	Data dictionary for 5.x and 6.x post office databases
 AgentInstall.txt	List of information provided during agent installation

domain directory

Within the GroupWise system, a **domain** is hierarchically the highest level object. It organizes post offices into a logical grouping for addressing and routing purposes. Each user in the domain has an address that consists of the user's GroupWise user ID, the user's post office name, and the domain name (*user.post_office.domain*). The explicit name is not displayed in the Address Book, but is stored in the domain database (**wpdomain.db**).

wpcsin directory

The wpcsin subdirectory in the **domain** is the MTA input queue in each domain. It contains eight priority subdirectories to handle different types of message traffic.

- ◆ Incoming user messages are queued by priority for routing to recipients' post offices in the local domain.

- ◆ Incoming status messages are queued by priority for routing to senders' post offices in the local domain.
- ◆ Outgoing administrative messages are queued for replication to other domains.
- ◆ In a routing domain, messages pass through this directory on their way to the next domain.

When a new message arrives, the MTA routes it to the appropriate destination.

For TCP/IP links, the MTA is notified immediately when a message arrives for processing. For mapped and UNC links, the MTA scans its input queue for messages to process. You can control the rate at which the MTA scans its input queues. See [“Adjusting MTA Polling of Input Queues in the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

Historical Note: WP Office*, the predecessor of GroupWise, was originally designed by WordPerfect Corporation* (WPCorp*). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory name wpcsin for the MTA input queue. Some naming conventions were originally preserved for backward compatibility.

0 directory

The priority 0 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for service requests that demand an immediate response from the MTA. For example:

- ◆ ConsoleOne[®] places restart requests and queue reconfiguration requests here for the MTA and gateways.
- ◆ MTAs for other domains route Busy Search requests through here when users in other domains check schedules of users in the local domain.

You can increase throughput for the priority 0 subdirectory. See [“Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

1 directory

The priority 1 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for service requests of the next highest priority. For example:

- ◆ ConsoleOne places directory synchronization requests here for the MTA admin thread.
- ◆ ConsoleOne places statistics requests here for the MTA to relay to the message logging module for processing.
- ◆ MTAs for other domains route GroupWise Remote requests through here when GroupWise Remote users do not connect to the post office where their master mailboxes are located.

You can increase throughput for the priority 1 subdirectory. See [“Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

2 directory

The priority 2 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for high priority messages. For example:

- ◆ MTAs for other domains place incoming high priority user messages here. The local MTA then routes the messages to recipients' post offices.

- ◆ MTAs for other domains place incoming administrative messages here to replicate database updates in the local domain.
- ◆ The MTA admin thread places outgoing administrative messages here to replicate database updates to other domains.

You can increase throughput for the priority 2 and 3 subdirectories. See [“Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

3 directory

The priority 3 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for high priority status messages routed back to senders in local post offices.

For example, MTAs for other domains place status responses to high priority user messages here. The local MTA then routes the status messages to senders’ post offices, so senders’ mailboxes can be updated with current message status.

You can increase throughput for the priority 2 and 3 subdirectories. See [“Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

4 directory

The priority 4 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for normal priority user messages routed to recipients in local post offices.

For example, MTAs for other domains place normal priority user messages here. The local MTA then routes the messages to recipients’ post offices. Most messages in your GroupWise system pass through the priority 4 subdirectory.

You can increase throughput for the priority 4 subdirectory. See [“Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

5 directory

The priority 5 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for normal priority status messages routed back to senders in local post offices.

For example, MTAs for other domains place status responses to normal priority user messages here. The local MTA then routes the status messages to senders’ post offices, so senders’ mailboxes can be updated with current message status.

6 directory

The priority 6 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for low priority user messages routed to recipients in local post offices.

For example, MTAs for other domains place low priority user messages here. The local MTA then routes the messages to recipients' post offices.

7 directory

The priority 7 subdirectory of the MTA input queue (**wpcsin**) in the **domain** is for low priority status messages routed back to senders in local post offices.

For example, MTAs for other domains place status responses to low priority user messages here. The local MTA then routes the status messages to senders' post offices, so senders' mailboxes can be updated with current message status.

wptools directory

The wptools subdirectory in the **domain** contains programs that support GroupWise administration.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wptools. Some naming conventions were originally preserved for backward compatibility.

wpgate directory

The wpgate subdirectory in the **domain** contains a subdirectory for each GroupWise gateway you have installed in your GroupWise system. For a list of gateways, see [GroupWise 6.x Gateways \(http://www.novell.com/documentation/gw6xgate/index.html\)](http://www.novell.com/documentation/gw6xgate/index.html). GroupWise 5.5 gateways can be used with GroupWise 6.x.

wpcsout directory

The wpcsout subdirectory in the **domain** is the MTA output queue in each domain. It contains subdirectories that function as input queues for the processes to which the MTA delivers messages.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory name wpcsout for the MTA output queue. Some naming conventions were originally preserved for backward compatibility.

ads directory

The ads subdirectory of the MTA output queue (**wpcsout**) in the **domain** is the input queue for the MTA admin thread in each domain. It contains priority subdirectories where incoming administrative messages are queued for processing. When a new administrative message arrives, the MTA admin thread performs the requested action.

Historical Note: The MTA admin thread was previously part of a separate agent, the Administration Agent (ADA), which was originally named the Administration Server (ADS). Hence, the directory name ads. Some naming conventions were originally preserved for backward compatibility.

0 directory

The priority 0 subdirectory of the MTA admin thread input queue (**wpcsout\ads**) in the **domain** is for service requests that demand an immediate response from the MTA admin thread.

For example, when you create or delete a post office in ConsoleOne, a restart request is placed here. The domain MTA admin thread processes the request and then restarts.

1 directory

The priority 1 subdirectory of the MTA admin thread input queue (**wpcsout\ads**) in the **domain** is for service requests of the next highest priority.

2 directory

The priority 2 subdirectory of the MTA admin thread input queue (**wpcsout\ads**) in the **domain** is for high priority administrative messages. For example:

- ◆ The MTA places administrative messages from other domains here. The administrative messages might instruct the MTA admin thread to add, modify, or delete users, post offices, or other objects in the domain. The MTA admin thread then processes the messages and makes the specified updates.
- ◆ When you use the Synchronize utility in ConsoleOne, a synchronization request is placed here. The MTA admin thread then resends the specified administrative messages to produce the required database updates.

css directory

The css subdirectory of the MTA output queue (**wpcsout**) in the **domain** is processed by a specialized MTA thread that responds to requests regarding its own configuration. It contains the eight standard priority subdirectories.

Historical Note: In an earlier version of GroupWise, the Message Transfer Agent (MTA) was called the Connection Server (CS) and this specialized subprocess was called the Connection Server Server (css). Some naming conventions were originally preserved for backward compatibility.

0 directory

The priority 0 subdirectory of the CSS input queue (**wpcsout\css**) in the **domain** is for service requests that demand an immediate response from the MTA.

For example, when you restart the MTA at the MTA agent console or in ConsoleOne, a restart request is placed here. The MTA processes the request and restarts.

1 directory

The priority 1 subdirectory of the CSS input queue (**wpcsout\css**) in the **domain** is for service requests of the next highest priority.

For example, each time the statistics are updated on the MTA agent console, a statistics request is placed here. The MTA then gathers the statistics and displays them on the MTA agent console.

2 directory

The priority 2 subdirectory of the css input queue (**wpcsout\css**) in the **domain** is for non-priority requests.

problem directory

The problem subdirectory of the MTA output queue (**wpcsout**) in the **domain** is where the MTA places message files that cannot be delivered because they are damaged in some way. Message files in the problem directory must be handled by the GroupWise administrator. See [“Message Is Dropped in the problem Directory in the Domain”](#) in *GroupWise 6.5 Troubleshooting 2: Solutions to Common Problems*.

mtaname file

The mtaname file in the **domain** provides the domain name associated with the domain directory structure. This can help you locate the domain information for the directory structure in ConsoleOne. It can also help you check links between MTAs.

wpdomain.db file

The wpdomain.db file in the **domain** is the domain database. It contains all administrative information for the domain.

In the primary domain, the wpdomain.db file contains all administrative information for your entire GroupWise system (all its domains, post offices, users, and so on). Because the wpdomain.db file in the primary domain is so crucial, you should back it up regularly and keep it secure. (You can re-create your entire GroupWise system from the primary domain wpdomain.db file; however, if the primary domain wpdomain.db file becomes unusable, you can no longer make administrative updates to your GroupWise system.)

In a secondary domain, the wpdomain.db file contains administrative information about that secondary domain only.

In GroupWise 6.x and 5.x domains, the data dictionary for the wpdomain.db file is the **gwdom.dc** file. In groupwise 4.x domains, the data dictionary is the **wpdomain.dc** file. As a result, wpdomain.db files have different structures (schemas) depending on whether they were created for 6.x/ 5.x or 4.x domains.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wpdomain.db. Some naming conventions were originally preserved for backward compatibility.

wpdomain.dc file

The wpdomain.dc file in the **domain** is the data dictionary for rebuilding GroupWise 4.x domain databases (**wpdomain.db** files) in secondary domains.

If the wpdomain.dc file is missing from the primary domain, you cannot rebuild GroupWise 4.x secondary domains. The original wpdomain.dc file is located in the **domain** subdirectory of the software distribution directory or on the GroupWise CD.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wpdomain.dc. Some naming conventions were originally preserved for backward compatibility.

wphost.dc file

The wphost.dc file in the **domain** is the data dictionary for rebuilding GroupWise 4.x post office databases (**wphost.db** files).

If the wphost.dc file is missing from a domain, you cannot rebuild GroupWise 4.x post offices in that domain. The original wphost.dc file is located in the **domain** directory of the software distribution directory or on the GroupWise CD.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Post offices were originally called hosts. Hence, the name wphost.dc. Some naming conventions were originally preserved for backward compatibility.

gwdom.dc file

The gwdom.dc file in the **domain** is the data dictionary for creating and rebuilding GroupWise 6.x and 5.x domain databases (**wpdomain.db** files) in secondary domains.

If the gwdom.dc file is missing from the primary domain, you cannot create or rebuild GroupWise 6.x/5.x secondary domains. The original gwdom.dc file is located in the **domain** directory of the software distribution directory or on the GroupWise distribution media.

gwpo.dc file



















The gwpo.dc file in the **domain** is the data dictionary for creating and rebuilding GroupWise 6.x and 5.x post office databases (**wphost.db** files).

If the gwpo.dc file is missing from a domain, you cannot create or rebuild GroupWise 6.x/5.x post offices in that domain. The original gwpo.dc file is located in the **domain** directory of the software distribution directory or on the GroupWise distribution media.











agentinstall.txt

The agentinstall.txt file in the **domain** records all the information that you provide when you install the MTA for the domain.

Post Office Directory

 post_office	Post office directory
 wpcsin	MTA input queue directory
 0	Live interactive requests
 1	Other interactive requests
 2	High priority messages
 3	High priority status responses
 4	Normal priority messages
 5	Normal priority status responses
 6	Low priority messages
 7	Low priority status responses
 gwdms	Document Management Services directory
 dmsh.db	Shared Document Management Services database
 lib0001-ff	Library directories
 dmxxnn01-ff.db	Document databases
 index	QuickFinder™ index for library
 archive	Archive directory for library
 docs	Large document directory for library
 fd00-ff	Subdirectories for documents

 ofmsg	Message database directory
 msgnn.db	As many as 25 message databases
 ngwdfn.db	Deferred message database
 guardbak	Backup guardian database
 ofuser	User database directory
 userxxx.db	User databases (one per user)
 puxxxx.db	Databases for shared folders
 index	QuickFinder index for messages
 offiles	Attachment store directory
 fd0-f6	Subdirectories for attachments
 ofviews	GroupWise client view files
 ofwork	GroupWise working directory
 ofdirect	Remote direct connection directory
 wpcout	MTA output queue directory
 ofs	POA input queue directory
 0	Live interactive requests
 1	Other interactive requests
 2	High priority messages
 3	High priority status responses
 4	Normal priority messages
 5	Normal priority status responses
 6	Low priority messages
 7	Low priority status responses
 mmdpqa.nnn	POA log files
 wprof50.db	Downloadable system Address Book
 ads	POA admin thread input queue directory
 0	Restart requests
 1	Directory synchronization requests
 2	Database updates
 3	Reserved; not currently used
 4	Reserved; not currently used
 5	Reserved; not currently used
 6	Reserved; not currently used
 7	Reserved; not currently used
 chk	GWCheck working directory
 0-3	GWCheck priority subdirectories

 defer	GWCheck subdirectory for deferred database maintenance requests
 problem	Directory for undeliverable messages
 wphost.db	Post office database
 gwpo.dc	Data dictionary for GroupWise 6.x/5.x post office databases
 ngwguard.db	Guardian database
 ngwguard.dc	Data dictionary for databases
 ngwguard.fbk	Guardian database backup
 ngwguard.rfl	Guardian database roll forward log
 ngwcheck.db	GWCheck control database
 agentInstall.txt	List of information provided during agent installation

post_office directory

Conceptually, a **post office** contains mailboxes for a set of network users. The users on the post office send and receive messages through their mailboxes.

Physically, a post office is a directory structure on a network file server. The directory structure contains subdirectories and databases that store messages and the information used to distribute the messages.

wpcsin directory

The wpcsin subdirectory in the **post office** is the MTA input queue in each post office. It contains eight priority subdirectories to handle different types of message traffic.

- ◆ Outgoing user messages are queued by priority for routing to recipients in other post offices.
- ◆ Outgoing status messages are queued by priority for routing back to senders' post offices.
- ◆ Outgoing Busy Search requests are queued for routing to other post offices so users' schedules can be checked.
- ◆ GroupWise Remote requests are queued for routing to GroupWise Remote users' master mailboxes.

When a new message arrives, the MTA routes it to the appropriate destination.

For mapped and UNC links, the MTA scans its input queue for messages to process. You can control the rate at which the MTA scans its input queues. See [“Adjusting MTA Polling of Input Queues in the Domain and Post Offices”](#) in [“Optimizing the MTA”](#) in the *GroupWise 6.5 Administration Guide*.

For TCP/IP links, the POA passes messages to the MTA via TCP/IP. A copy is kept in the MTA input queue until the POA has successfully transferred the message.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory name wpcsin for the MTA input queue. Some naming conventions were originally preserved for backward compatibility.

0 directory

The priority 0 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for service requests that demand an immediate response from the MTA.

For example, the GroupWise client places Busy Search requests here. The MTA then routes the requests to the appropriate post offices, so users' schedules can be checked.

You can increase throughput for the priority 0 subdirectory. See “[Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices](#)” in “[Optimizing the MTA](#)” in the *GroupWise 6.5 Administration Guide*.

1 directory

The priority 1 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for service requests of the next highest priority. For example:

- ◆ Remote with a direct connection places requests here for routing to GroupWise Remote users' master mailboxes.
- ◆ The POA places outgoing status messages to GroupWise Remote users here for routing to the async connection.

You can increase throughput for the priority 1 subdirectory. See “[Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices](#)” in “[Optimizing the MTA](#)” in the *GroupWise 6.5 Administration Guide*.

2 directory

The priority 2 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for high priority user messages routed to recipients in other post offices, domains, or systems.

For example, the GroupWise client places high priority user messages here. The MTA then routes the messages to the appropriate destinations.

You can increase throughput for the priority 2 and 3 subdirectories. See “[Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices](#)” in “[Optimizing the MTA](#)” in the *GroupWise 6.5 Administration Guide*.

3 directory

The priority 3 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for high priority status messages routed back to senders in other post offices, domains, or systems.

For example, the GroupWise client and local POA place status responses to high priority user messages here. The MTA then routes the status messages to the appropriate post offices, so senders' mailboxes can be updated with current message status.

You can increase throughput for the priority 2 and 3 subdirectories. See “[Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices](#)” in “[Optimizing the MTA](#)” in the *GroupWise 6.5 Administration Guide*.

4 directory

The priority 4 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for normal priority user messages routed to recipients in other post offices, domains, or systems.

For example, the GroupWise client places normal priority user messages here. The MTA then routes the messages to the appropriate destinations. Most messages in your GroupWise system pass through the priority 4 subdirectory.

You can increase throughput for the priority 4 subdirectory. See “[Adjusting the Number of MTA Scanner Threads for the Domain and Post Offices](#)” in “[Optimizing the MTA](#)” in the *GroupWise 6.5 Administration Guide*.

5 directory

The priority 5 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for normal priority status messages routed back to senders in other post offices, domains, or systems.

For example, the GroupWise client and local POA place status responses to normal priority user messages here. The MTA then routes the status messages to the appropriate post offices, so senders’ mailboxes can be updated with current message status.

6 directory

The priority 6 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for low priority user messages routed to recipients in other post offices, domains, or systems.

For example, the GroupWise client places low priority user messages here. The MTA then routes the messages to the appropriate destinations.

7 directory

The priority 7 subdirectory of the MTA input queue (**wpcsin**) in the **post office** is for low priority status messages routed back to senders in other post offices, domains, or systems.

For example, the GroupWise client and local POA place status responses to low priority user messages here. The MTA then routes the status messages to the appropriate post offices, so senders’ mailboxes can be updated with current message status.

gwdms directory

The gwdms subdirectory in the **post office** is the Document Management Services (DMS) directory in each post office. It contains the document libraries associated with the post office.

dmsh.db file

The dmsh.db file in the document management subdirectory (**gwdms**) in the **post office** is a database shared by all libraries in the post office. It contains a list of all available libraries and lookup tables for each library.

lib0001-ff directories

The lib0001-ff subdirectories in the **gwdms** subdirectory in the **post office** contain the libraries for the post office, with one library per directory. You can create a maximum of 256 libraries in a post office.

dmxxnn01-ff.db files

The dmxxnn01-ff.db files in the library subdirectories (**lib0001-ff**) in the **post office** are databases for library and document information.

The *nn* in the filenames represents the partition number, which is generated by a hashing algorithm to guarantee uniqueness.

The *01-ff* in the filenames represents the library number, matching the number on the library directory in which the database is found.

dmsdnn01-ff.db file

The *dmsdnn01-ff.db* file in each library holds system data for the library, such as library configuration information.

dmddnn01-ff.db file

The *dmddnn01-ff.db* file in each library holds document data for the library. Document data is the document property information for documents in the library.

dmdl01-ff.db file

The *dmdl01-ff.db* file in each library holds document logging data for the library. Document logging data records all activities performed on documents in the library.

index directory

The index subdirectories in the library subdirectories (*lib0001-ff*) in the **post office** contain the QuickFinder index for the documents contained in the library.

archive directory

The archive subdirectories in the library subdirectories (*lib0001-ff*) in the **post office** contain an array of subdirectories for holding archived documents. The subdirectories are numbered sequentially. When the first archive subdirectory reaches its maximum allowable size, archived documents are stored in the next sequential directory, and so on.

docs directory

The docs subdirectories in the library subdirectories (*lib0001-ff*) in the **post office** contain an array of subdirectories for storing documents.

fd0-ff directories

The *fd0-ff* subdirectories in the **docs** subdirectory in the **post office** store documents that are equal to or greater than 2 KB in size. The *0-ff* variable represents hexadecimal number 0 through ff, so the subdirectories are named fd0 through fdff. The document databases (*dmxxnn01-ff.db* files) contain pointers to documents stored in the subdirectories of the docs directory.

ofmsg directory

The ofmsg subdirectory in the **post office** contains as many as 25 databases where messages are stored. It serves as centralized storage for all users in the post office. A message must be stored only once to be delivered to any number of users in the same post office.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in ofmsg. Some naming conventions were originally preserved for backward compatibility.

msgnn.db file

The *msgnn.db* files in the **ofmsg** subdirectory in the **post office** are the message databases where users' messages smaller than 2 KB are stored. To increase database efficiency, messages, attachments, and recipient lists equal to or greater than 2 KB are stored outside the *msgnn.db* files

in an array of subdirectories in the **offiles** directory. After the 2 KB limit is reached, only pointers are stored in the message databases.

The *nn* variable in the database names is a two-digit number from zero to 24. A hashing algorithm takes each user's GroupWise file ID (FID) to derive which database the user's outgoing mail is assigned to. The contents of the messages databases are encrypted so the text of message can only be read through GroupWise.

Multiple users are assigned to the same message database. You can use GWCheck to determine which database a specific user has been assigned to. See “GroupWise Check” in “Standalone Database Maintenance Programs” in the *GroupWise 6.5 Administration Guide*.

ngwdfc.db file

The ngwdfc.db file in the **ofmsg** subdirectory in the **post office** holds deferred messages that users have specified for delivery at a later time. When users delay delivery on messages, the messages are transferred to the receiving post office and held in the ngwdfc.db file until the delay expires.

Historical Note: Earlier versions of GroupWise handled deferred messages through the ofpend directory in the post office.

guardbak directory

The guardbak subdirectory in the **ofmsg** subdirectory in the **post office** holds a backup copy of the **ngwguard.fbk** file.

ofuser directory

The ofuser subdirectory in the **post office** contains a separate database (mailbox) for each GroupWise user.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in ofuser. Some naming conventions were originally preserved for backward compatibility.

userxxx.db file

The userxxx.db files in the **ofuser** subdirectory in the **post office** are user databases where the contents of users' mailboxes are stored, as displayed in the GroupWise client. In addition, each user database contains:

- ◆ Some personal GroupWise client program settings
- ◆ Personal appointments
- ◆ Personal groups
- ◆ Personal notes
- ◆ Rules

Personal client settings that remain the same regardless of what workstation a user logs in to are stored in the user database. Personal client settings that are customized for a particular workstation are stored in the Windows* Registry.

The *xxx* variable in the database names is each user's GroupWise file ID (FID).

puxxxxx.db file

The puxxxxx.db files in the **ofuser** subdirectory in the **post office** are databases for replicated items such as shared folders. These databases prevent conflicts between user names of shared items from users in other post offices and user names in the local post office.

index directory

The index subdirectory in the **ofuser** subdirectory in the **post office** contains the QuickFinder index for users' messages stored in the post office.

offiles directory

The offiles subdirectory in the **post office** contains subdirectories for messages, attachments, and recipient lists that are equal to or greater than 2 KB in size. These larger messages, attachments, and recipient lists are stored outside the actual message databases in the **ofmsg** directory to increase database efficiency.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in offiles. Some naming conventions were originally preserved for backward compatibility.

fd0-f6 directories

The fd0-f6 subdirectories in the **offiles** subdirectory in the **post office** store messages, attachments, and recipient lists that are equal to or greater than 2 KB in size. The *nn* variable represents hexadecimal number 0 through f6, so the subdirectories are named fd0 through fdf6. The message databases (**msgnn.db** files) contain pointers to messages, attachments, and recipient lists stored in the subdirectories of offiles.

ofviews directory

The ofviews subdirectory in the **post office** contains subdirectories for GroupWise client platforms. Within the platform-specific subdirectories (for example, win) are view (*.view) files that create the various views displayed in the GroupWise client.

The gwviewxx.ini and ofviewxx.ini files configure the standard views on the menus where users select views. The gwviewxx.ini file configures GroupWise 6 and GroupWise 5.5 standard views. The ofviewxx.inI file configures standard views from earlier versions of GroupWise.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in ofviews. Some naming conventions were originally preserved for backward compatibility.

ofwork directory

The ofwork subdirectory in the **post office** is a working directory for GroupWise Remote.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in ofwork. Some naming conventions were originally preserved for backward compatibility.

ofdirect directory

The ofdirect subdirectory in the working directory (**ofwork**) in the **post office** is used by GroupWise Remote for direct connections.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in ofdirect. Some naming conventions were originally preserved for backward compatibility.

wpcsout directory

The wpcsout subdirectory in the **post office** is the MTA output queue in each post office. It contains subdirectories which function as input queues for the other agents to which the MTA delivers messages.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory name wpcsout for the MTA output queue. Some naming conventions were originally preserved for backward compatibility.

ofs directory

The ofs subdirectory of the mta output queue (**wpcsout**) in the **post office** is the POA input queue in each post office. It contains eight priority subdirectories to handle different types of message traffic.

- ◆ Incoming user messages are queued by priority for delivery to recipients' mailboxes in the local post office.
- ◆ Incoming status messages are queued by priority for delivery to senders' mailboxes in the local post office.
- ◆ Incoming Busy Search requests are queued for the POA to check users' schedules in the local post office.

The POA scans these priority subdirectories regularly. When a new message arrives, the POA processes the messages and performs the required actions.

0 directory

The priority 0 subdirectory of the POA input queue (**wpcsout\ofs**) in the **post office** is for service requests that demand an immediate response from the POA.

For example, the MTA places Busy Search requests here so the POA can check recipients' schedules and quickly return the schedule information to the sender.

1 directory

The priority 1 subdirectory of the POA input queue (**wpcsout\ofs**) in the **post office** is for service requests of the next highest priority.

For example, the MTA places requests from GroupWise Remote users for items in their master mailboxes here. The POA then processes the messages and returns the requested items.

2 directory

The priority 2 subdirectory of the POA input queue (**wpcsout\ofs**) in the **post office** is for high priority user messages being delivered to recipients in the local post office.

For example, the MTA places high priority user messages here. The POA then updates the message databases and recipients' mailboxes.

3 directory

The priority 3 subdirectory of the POA input queue (`wpcout\ofs`) in the **post office** is for high priority status messages coming back to senders in the local post office.

For example, the MTA places status responses to high priority user messages here. The POA then updates the message databases and senders' mailboxes with current message status.

4 directory

The priority 4 subdirectory of the POA input queue (`wpcout\ofs`) in the **post office** is for normal priority user messages being delivered to recipients in the local post office.

For example, the MTA places normal priority user messages here. The POA then updates the message databases and recipients' mailboxes. Most messages in your GroupWise system pass through the priority 4 subdirectory.

5 directory

The priority 5 subdirectory of the POA input queue (`wpcout\ofs`) in the **post office** is for normal priority status messages coming back to senders in the local post office.

For example, the MTA places status responses to normal priority user messages here. The POA then updates the message databases and senders' mailboxes with current message status.

6 directory

The priority 6 subdirectory of the POA input queue (`wpcout\ofs`) in the **post office** is for low priority user messages being delivered to recipients in the local post office.

For example, the MTA places low priority messages here. The POA then updates the message databases and recipients' mailboxes.

7 directory

The priority 7 subdirectory of the POA input queue (`wpcout\ofs`) in the **post office** is for low priority status messages coming back to senders in the local post office.

For example, the MTA places status responses to low priority user messages here. The POA then updates the message databases and senders' mailboxes with current message status.

***mmdp*poa.*nnn* files**

The *mmdp*poa.*nnn* files are POA log files. The POA creates log files to inform you of its processing and any problems it encounters. By default, these log files are created in the `wpcout\ofs` directory. You can change the location if needed. See "Using POA Log Files" in "Post Office Agent" in the *GroupWise 6.5 Administration Guide* guide.

The first two digits of the filename represent the month, the next two digits represent the day of the month, and the next three characters indicate what program created the log. The three-digit extension is a sequence number for multiple log files created on the same day. For example, 0518poa.002 is the second POA log file created on May 18.

wprof50.db file

The wprov50.db file in the `wpcsout\ofs` directory is the downloadable system Address Book for Remote client users. By default, it is automatically re-created once a day to keep it up to date. See “Performing Nightly User Upkeep” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide* guide.

ads directory

The ads subdirectory of the MTA output queue (`wpcsout`) in the `post office` is the input queue for the POA admin thread in each post office. It contains priority subdirectories where administrative messages are queued for processing.

Historical Note: The POA admin thread was previously part of a separate agent, the Administration Agent (ADA), which was originally named the Administration Server (ADS). Hence, the directory name ads. Some naming conventions were originally preserved for backward compatibility.

0 directory

The priority 0 subdirectory of the POA admin thread input queue (`wpcsout\ads`) in the `post office` is for service requests that demand an immediate response from the POA admin thread.

1 directory

The priority 1 subdirectory of the POA admin thread input queue (`wpcsout\ads`) in the `post office` is for service requests of the next highest priority.

For example, a directory synchronization request that could not be performed when the POA admin thread received it in its domain input queue would be placed here in the post office for later processing.

2 directory

The priority 2 subdirectory of the POA admin thread input queue (`wpcsout\ads`) in the `post office` is for high priority administrative messages.

For example, a database update request that could not be performed when the POA admin thread received it in its domain input queue would be placed here in the post office for later processing.

chk directory

The chk subdirectory of the MTA output queue (`wpcsout`) in the `post office` is the working directory where the multithreaded GWCheck process keeps temporary files during database maintenance and where it tracks the activities of its various threads.

problem directory

The problem subdirectory of the mta output queue (`wpcsout`) in the `post office` is a holding area for damaged message files. Problem files are marked with an extension indicating which GroupWise agent placed each file in the problem directory.

You should check this directory periodically for problem files, resolve the problem, then place the files back into the appropriate queue for continued processing. For assistance, see “Message Is Dropped in the problem Directory in the Post Office” in “Strategies for Message Delivery Problems” in the *GroupWise 6.5 Troubleshooting 2: Solutions to Common Problems*.

wphost.db file

The wphost.db file in the **post office** is the post office database. It contains all administrative information for the post office. It also contains the Address Book for the post office.

In GroupWise 6.x and 5.x post offices, the data dictionary for the wphost.db file is the **gwpo.dc** file. In GroupWise 4.x post offices, the data dictionary is the **wphost.dc** file. As a result, wphost.db files have different structures (schemas) depending on whether they were created for GroupWise 6.x/5.x or 4.x post offices.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Post offices were originally called hosts. Hence, the name wphost.db. Some naming conventions were originally preserved for backward compatibility.

gwpo.dc file

The gwpo.dc file in the **post office** is the data dictionary for creating and rebuilding GroupWise 6.x and 5.x post office databases (**wphost.db** files).

If the gwpo.dc file is missing from a post office and its domain, you cannot create or rebuild GroupWise 6.x/5.x post offices in that domain. The original gwpo.dc file is located in the **domain** directory of the software distribution directory or on the GroupWise CD.

ngwguard.db file

The ngwguard.db file in the **post office** is the guardian database. See “**Information Stored in the Post Office**” in “**Post Office Agent**” in the *GroupWise 6.5 Administration Guide*.

ngwguard.dc file

The ngwguard.dc file in the **post office** is the data dictionary for building the following databases in the post office:

- ◆ **ngwguard.db** (guardian database)
- ◆ **dmxxnn01-ff.db** (document management databases)
- ◆ **msgnn.db** (message databases)
- ◆ **userxxx.db** (user databases)
- ◆ **puxxxxx.db** (databases for replicated items like shared folders)

ngwguard.fbk file

The ngwguard.fbk file in the **post office** is a “fall back” copy of the **ngwguard.db** file. If the ngwguard.db file becomes damaged, the ngwguard.fbk file, along with the **ngsguard.rfl** file, can be used to rebuild a valid, current ngwguard.db file. The ngwguard.fbk file is so important that an additional copy of it is kept in the ofmsg/guardbak subdirectory in case the copy in the post office directory is inadvertently deleted. See “**Guardian Databases**” in “**Databases**” in the *GroupWise 6.5 Administration Guide*.

ngwguard.rfl file

The ngwguard.rfl file in the **post office** is a roll-forward transaction log of every database transaction that has taken place since the last copy of the **ngwguard.fbk** file was created. See “Guardian Databases” in “Databases” in the *GroupWise 6.5 Administration Guide*.





















ngwcheck.db

The ngwcheck.dbfile in the **post office** is the database that controls GWCheck’s multithreaded processing. It contains job and task records that are used to synchronize and summarize GWCheck requests as they progress.

agentinstall.txt

The agentinstall.txt file in the **post office** records all the information that you provide when you install the POA for the post office.

MTA Local Queue Directory

 mslocal	MTA local working directory
 mmdxxx.nn	MTA log files
 msglog	Message logging directory
 mmdmsg.nn	Message logging files
 gwinprog	MTA "in progress" queue directory
 0-7	Priority subdirectories
 mshold	MTA holding directory
 domainms	Processing directory for MTA
 0-7	Priority subdirectories
 mtaname	Location identifier
 postx	Holding directories for post offices
 0-7	Priority subdirectories
 mtaname	Location identifier
 gatewayx	Holding directories for gateways
 0-7	Priority subdirectories
 mtaname	Location identifier
 domainx	Holding directories for other domains
 0-7	Priority subdirectories
 mtaname	Location identifier
 gwvsscan	Working directory for third-party virus scanning programs

mslocal directory

The **mslocal** directory is the MTA local working directory. The `/work` startup switch of the MTA specifies the location of the mslocal directory. It must be located on the hard disk of the server where the MTA runs so it is always accessible. Adequate disk space must be available to hold messages going to destinations that are temporarily closed.

Typical locations for the mslocal directory include:

- ◆ `sys:\system` on a NetWare® server
- ◆ `c:\` on a Windows server

To move the mslocal directory, stop the MTA, the copy the mslocal directory, along with all its subdirectories, to the new location. Then restart the MTA and specify the new location using the `/work` startup switch.

Historical Note: In earlier versions of GroupWise, the Message Transfer Agent (MTA) was called the Message Server (MS). Hence, the `ms` in mslocal. Some naming conventions were originally preserved for backward compatibility.

mmddxxx.nnn files

The *mmddxxx.nnn* file are MTA log files. The MTA creates log files to inform you of its processing and any problems it encounters. By default, these log files are created in the **mslocal** directory. You can change the location if needed. See “Using MTA Log Files” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

The first two digits of the filename represent the month; the next two digits represent the day of the month; the next three characters indicate what program created the log. The three-digit extension is a sequence number for multiple log files created on the same day. For example, `0518mta.002` is the second MTA log file created on May 18.

Historical Note: In earlier versions of GroupWise, the Message Transfer Agent (MTA) was called the Message Server (MS). Hence, the `ms` indicator representing the MTA. Some naming conventions were originally preserved for backward compatibility.

msglog directory

The msglog subdirectory contains message logging files. It is created when you turn on message logging. The MTA receiver threads log messages as they arrive so the MTA worker threads can process messages without having to scan the MTA input queues to look for work.

The resources used for message logging are configurable. See “Optimizing the Routing Queue” in “Optimizing the MTA” in the *GroupWise 6.5 Administration Guide*.

More detailed message logging by the MTA is also available, but is turned off by default. See “Enabling MTA Message Logging” in “Configuring the MTA” in the *GroupWise 6.5 Administration Guide*.

mmddmsg.nnn files

The *mmddmsg.nnn* files in the message logging subdirectory (**msglog**) in the **MTA local directory** are used by the MTA to track messages in its “in progress” queue.

The first two digits of the filename represent the month; the next two digits represent the day of the month. The three-digit extension is a sequence number for multiple files created on the same day. For example, 0518msg.002 is the second message logging file created on May 18.

gwinprog directory

The gwinprog subdirectory is the MTA “in progress” queue. It contains eight priority subdirectories parallel to those found in wpcsin. All messages for recipients in the domain pass through gwinprog, no matter whether they arrived by way of TCP/IP or by way of message files deposited into the MTA input queue by a POA or another MTA.

The resources used to process the “in progress” queue are configurable. See “[Optimizing the Routing Queue](#)” in “[Optimizing the MTA](#)” in the *GroupWise 6.5 Administration Guide*.

mshold directory

The mshold subdirectory is a holding queue for messages addressed to domains, post offices, or gateways that are currently closed.

A location might be closed because its server is down or because the MTA is unable to communicate with it for any other reason. When a closed location is again open, the MTA moves messages from the holding queue back into the normal message flow.

Historical Note: In earlier versions of GroupWise, the Message Transfer Agent (MTA) was called the Message Server (MS). Hence, the ms in mshold. Some naming conventions were originally preserved for backward compatibility.

domainms directory

The domainms subdirectory in the holding directory (**mshold**) is used for internal processing by the MTA. It does not contain any files a GroupWise administrator needs to access.

Historical Note: In earlier versions of GroupWise, the Message Transfer Agent (MTA) was called the Message Server (MS). Hence, the ms in domainms. Some naming conventions were originally preserved for backward compatibility.

postx directories

The *postx* subdirectories in the holding directory (**mshold**) represent post offices in the domain. If a post office is closed, the MTA routes messages for that post office into its holding queue in mshold. When the post office is open, the MTA moves the messages from the holding queue back into the regular message flow. For more information, see “[Message Delivery to a Different Post Office](#)” on page 15.

The name of the holding queue for each post office consists of the first three characters of the post office name, followed by four hashed characters to ensure uniqueness.

gatewayx directories

The *gatewayx* subdirectories in the holding directory (**mshold**) represent gateways in the domain. If a gateway is closed, the MTA routes messages for that gateway into its holding queue in mshold. When the gateway is open, the MTA moves the messages from the holding queue back into the regular message flow through the gateway.

The name of the holding queue for each gateway consists of the first three characters of the gateway name, followed by four hashed characters to ensure uniqueness.

domainx directories

The *domainx* subdirectories in the holding directory (**mshold**) represent domains to which the current domain has a direct link. If a domain is closed, the MTA routes messages for that domain into its holding queue in MSHOLD. When the domain is open, the MTA moves the messages from the holding queue back into the regular message flow. For more information, see [“Message Delivery to a Different Domain” on page 27](#).

The name of the holding queue for each domain consists of the first three characters of the domain name, followed by four hashed characters to ensure uniqueness.

0-7 directories

The priority 0-7 subdirectories in each holding queue in the **mshold** subdirectory correspond to the priority 0-7 subdirectories located in each domain, post office, or gateway. See the following directory structures for more information about its priority 0-7 subdirectories:

- ◆ [“Domain Directory” on page 57](#)
- ◆ [“Post Office Directory” on page 64](#)

mtaname files

The mtaname files in the closed location holding queues provide the name associated with the domain, post office, or gateway holding queue. They can help you check links between MTAs in ConsoleOne without going to the MTA agent console to determine the location name. To associate a location name with its holding queue directory from the MTA agent console, click Configuration Status > select the location > click Details.

gwvsscan directory































The gsvsscan subdirectory is the working directory where third-party virus scanning programs that snap in to the MTA can perform their processing.












mtaconv directory

The mtaconv subdirectory is the working directory where the MTA converts GroupWise 6.x and GroupWise 5.x messages to 4.x format for transfer to a GroupWise 4.x system. After the conversion is finished, this directory should be empty.

Internet Agent Queue Directory

The following directories and files are found under the `\domain\wpgate\` structure for the Internet Agent after the software has been installed and the Internet Agent has processed messages.

 <code>domain\wpgate\gwia</code>	GroupWise Internet Agent home directory
 <code>000.prc</code>	Internet Agent message processing directory
 <code>cmd</code>	Not currently used
 <code>gwwork</code>	Hold directory for temporary files using during processing
 <code>mmdlog.nnn</code>	Log files
 <code>acct</code>	Accounting file
 <code>set</code>	Settings file for screen colors, log levels, and so on
 <code>stat</code>	Statistics file for Internet Agent operation
 <code>proc</code>	Process lock file indicating that the Internet Agent is running
 <code>pulse.tmp</code>	Temporary file to verify Internet Agent operation
 <code>wpcsin</code>	MTA input queue directory
 <code>0-7</code>	Message priority subdirectories
 <code>wpcsout</code>	MTA output queue
 <code>gwixxx</code>	System-defined directory
 <code>0-7</code>	Message priority subdirectories
 <code>problem</code>	Hold directory for damaged outbound messages
 <code>gwhold</code>	Message hold directory
 <code>qfiles</code>	Delayed delivery hold directory
 <code>gwprob</code>	Hold directory for damaged inbound messages
 <code>gwchars</code>	Directory for character conversion tables
 <code>save</code>	Directory for old configuration files from reinstalls or upgrades
 <code>gwia.cfg</code>	Internet Agent configuration file for startup switches
 <code>route.cfg</code>	Route configuration file to customize routing
 <code>gwauth.cfg</code>	Host authentication configuration file
 <code>mimetype.cfg</code>	MIME encoding configuration file for various file types
 <code>exepath.cfg</code>	Configuration file pointing ConsoleOne to the <code>gwia.cfg</code> file
 <code>gwac.db</code>	Access control database
 <code>gwac.dc</code>	Database dictionary file used to create the access control database
 <code>preamble.txt</code>	Message for recipients who lack a MIME-compliant mail reader
 <code>preamble.all</code>	Preamble message in various languages

 blocked.txt	List of blocked Internet sites
 gwia.mib	Internet Agent Management Information Base (MIB) (NetWare only)
 statusxx.xml	File for customizing status messages
 gwia	SMTP service (daemon) home directory
 send	Outbound hold directory for converting messages into Internet format
 receive	Incoming hold directory for converting messages into GroupWise format
 result	Send and result files to confirm transmission
 defer	Hold directory for re-queued and deferred messages
 work	Schedule files for SMTP service operations on deferred messages
 dsnhold	Delivery Status Notification (DSN) hold directory
 help	Internet Agent Web console help file directory

domain\wpgate\gwia directory

The *domain\wpgate\gwia* directory is the GroupWise Internet Agent home directory where Internet Agent configuration files and queue directories are located. The name is established when you install the Internet Agent. The default is *wpgate\gwia* in the domain directory. You can change the location using the **/home** startup switch in the Internet Agent configuration file (*gwia.cfg*).

000.prc directory

The Internet Agent uses the *000.prc* directory to process messages.

gwwork directory

The *gwwork* directory stores temporary files created by the Internet Agent as it converts and builds messages for transfer across the Internet.

mmddlog.nnn file

The *mmddlog.nnn* files hold error and status messages about the functioning of the Internet Agent. The Internet Agent creates a log file each day with a unique name, where *mm* is the month, *dd* is the day, and *nnn* is a sequential number indicating the sequence of log files in a single day. For more information log files, see “Using Internet Agent Log Files” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

acct file

The *acct* file contains information about the messages the Internet Agent sends each day. It is e-mailed to the accounts each day at midnight. For more information about the accounting files, see “Setting Up Accounting” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

set file

The *set* file stores Internet Agent console settings such as color, log settings, and so on. For more information, see “Monitoring the Internet Agent through the Server Console”.

stat file

The stat file stores statistics about the Internet Agent's functioning. For information about the statistics provided by the Internet Agent, see "Statistics" in "Internet Agent" in the *GroupWise 6.5 Administration Guide*.

proc file

The proc file is the lock file for the Internet Agent process. The proc file is opened and locked when the Internet Agent starts. This prevents multiple Internet Agents from being started for the same domain.

pulse.tmp file

The pulse.tmp file is re-created by the Internet Agent every time it completes a cycle (after an idle loop). If you are not at the Internet Agent console but need to know if the Internet Agent is running, you can delete the pulse.tmp file. If the Internet Agent is running, it re-creates the file.

wpcsin directory

The Internet Agent places inbound messages in one of the wpcsin priority subdirectories (0-7). Most messages go in the 4 directory, although some administrative and status messages might go in other directories. The Message Transfer Agent retrieves the messages and delivers them to the proper destinations.

wpcout directory

The wpcout directory is the MTA output queue as well as being the Internet Agent input queue.

gwixxxx directory

The gwixxxx directory is a system-defined directory, where *gwi* represents the first three letters of the Internet Agent object name as defined during installation and displayed in ConsoleOne, and *xxxx* is a randomly-generated string. Here, the Message Transfer Agent places outbound messages in the appropriate 0-7 priority subdirectory for the Internet Agent to retrieve and process.

problem directory

The problem directory holds messages that the MTA cannot process.

gwhold directory

The gwhold directory holds messages that are scheduled for delayed delivery.

qfiles directory

The qfiles directory holds messages that cannot be sent during the current Send/Receive cycle. The messages are queued to this directory until the next cycle.

The delayed delivery messages waiting in the qfiles directory remain in encrypted format until the Internet Agent transfers them to the send directory for processing by the SMTP service.

gwprob directory

The Internet Agent uses the gwprob directory for messages it cannot process. These are usually messages that have been damaged during transmission or that have the wrong Internet address.

gwchars directory

This directory contains conversion tables that the Internet Agent uses to convert message attachments between character sets.

save directory

If you reinstall or upgrade the Internet Agent, your old configuration files are copied to the save directory as a backup. If you reinstall or upgrade repeatedly, the files are overwritten each time.

gwia.cfg file

The *gwia.cfg* file is the Internet Agent configuration file that contains startup switches. Some switches are set during installation. You can set others as needed. For more information, see “Using Internet Agent Startup Switches” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

Only the Windows Internet Agent actually uses the *gwia.cfg* file under the domain. The NetWare Internet Agent uses the *gwia.cfg* file created in `sys:\system` during installation. The Linux Internet Agent uses the *gwia.cfg* file created in `/opt/novell/groupwise/agents/share` during installation. On NetWare and Linux, the *gwia.cfg* file under the domain is just a boilerplate file with no switches set during installation.

route.cfg file

The *route.cfg* file enables you to customize routing for specific hosts. For more information, see “Using a Route Configuration File” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

gwauth.cfg file

The *gwauth.cfg* file enables the Internet Agent to log in to SMTP hosts that require authentication. For more information, see “Configuring SMTP Host Authentication” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

mimetype.cfg file

The *mimetype.cfg* file enables you to customize MIME content-type mappings for various attachment types. For more information, see “Customizing MIME Content-Type Mappings” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

exepath.cfg file

The *exepath.cfg* file is used by ConsoleOne to locate the *gwia.cfg* file. This enables ConsoleOne to write any configuration setting changes to the *gwia.cfg* file or update Novell® eDirectory™ with any changes from the file. The file must contain the path to the *gwia.cfg* file in the `sys:\system` directory on NetWare, the `/opt/novell/groupwise/agents/share` directory on Linux, or the `domain\wpgate\gwia` directory on Windows.

gwac.db file

The *gwac.db* file is the access control database that stores information about the classes of service you have created. For more information, see “Maintaining the Access Control Database” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

gwac.dc file

The gwac.dc file is the data dictionary file from which the gwac.db is created.

preamble.txt file

The preamble.txt file is an ASCII text file that is automatically included with any MIME multipart message and is displayed when the message recipient lacks a MIME-compliant mail reader. For more information, see “[Customizing MIME Preamble Text](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

preamble.all file

The preamble.all file contains the preamble text in multiple languages. For more information, see “[Customizing MIME Preamble Text](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

blocked.txt file

The blocked.txt file contains a list of Internet sites that you have added to the Prevent Messages From list for your default class of service in ConsoleOne. For more information, see “[Controlling User Access](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

gwia.mib file (NetWare only)

The gwia.mib file contains all the Trap, Set, and Get variables used for communication between the Internet Agent and SNMP management consoles. For setup instructions, see “[Monitoring the Internet Agent through an SNMP Management Console](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

statusxx.xml file

The statusxx.xml file enables you to customize the messages that users receive regarding message delivery status. For more information, see “[Customizing Delivery Status Notifications](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

gwia directory

The gwia directory is the SMTP service (daemon) home directory where messages are converted between GroupWise format and Internet format. On NetWare and Linux, the default location is wpgate/gwia, the same as the Internet Agent home directory. On Windows, the default location is the Internet Agent installation directory. You can change the location using the `/dhome` startup switch in the Internet Agent configuration file (*gwia.cfg*).

send directory

The Internet Agent SMTP service places outbound messages in the send directory after they have been converted out of GroupWise format into SMTP format. The SMTP service polls the send directory and sends any messages to the destination SMTP host.

receive directory

The Internet Agent SMTP service places inbound messages in the receive directory, converts them into GroupWise format, and then passes them to the Message Transfer Agent by placing them in the **wpcsin** directory.

result directory

When the Internet Agent SMTP service processes the message, it builds a file, **r*.***, in the result directory that contains several lines of comments and SMTP reply codes, which might indicate possible errors or confirm correct transmission. After the Internet Agent SMTP service has completed the transmission with the destination host, it moves another file, **s*.*** from the send directory to the result directory. The filenames for both files are identical, except for the first letter, which is either “s” or “r”. The **s*.*** file is the converted message file. The SMTP service looks at the “s” and “r” files in the result directory and compares the conversation. If the **r*.*** file contains the correct (250 OK) SMTP reply codes, the SMTP service deletes the file and sends a transferred status message to the user’s Sent Items folder in the GroupWise client.

defer directory

The defer directory holds messages that are deferred and re-queued according to the Retry Schedule. If the Internet Agent SMTP service receives a temporary error, such as Host Down, it places the message in the defer directory for a specified time, then transfers the file to the send directory for another attempt at sending to the Internet. For more information, see **“Configuring Basic SMTP/MIME Settings”** in **“Internet Agent”** in the *GroupWise 6.5 Administration Guide*.

dsnhold directory




















The DSN hold directory stores header information for inbound messages that request delivery status notifications. For more information, see **“Using Extended SMTP (ESMTP) Options”** in **“Internet Agent”** in the *GroupWise 6.5 Administration Guide*.

help directory

The help directory holds the help files that you can view from the Internet Agent Web console. See **“Monitoring the Internet Agent through the Web Console”** in **“Internet Agent”** in the *GroupWise 6.5 Administration Guide*.

WebAccess Agent Queue Directory

The following directories and files are found under the `\domain\wpgate\` structure for the WebAccess Agent after the software has been installed and the WebAccess Agent has processed messages.

 <code>domain\wpgate\webac65a</code>	GroupWise WebAccess Agent home directory
 <code>000.prc</code>	WebAccess Agent log file processing directory
 <code>mmddweb.nnn</code>	WebAccess Agent log files
 <code>wpcsin</code>	MTA input queue directory
 <code>0-7</code>	Message priority subdirectories
 <code>wpcout</code>	MTA output queue
 <code>webxxx</code>	System-defined directory for the WebAccess Agent
 <code>0-7</code>	Message priority subdirectories
 <code>problem</code>	Hold directory for damaged outbound messages
 <code>gwhold</code>	Hold directory for delayed delivery messages
 <code>gwprob</code>	Hold directory for damaged inbound messages
 <code>files</code>	Directory for documents attached to messages
 <code>template</code>	Directory for templates for viewing documents
 <code>commgr.cfg</code>	Communications Manager configuration file
 <code>comint.cfg</code>	Communications initialization configuration file
 <code>mimetype.cfg</code>	MIME encoding configuration file for various file types
 <code>gwcachе.db</code>	Database catalog of documents in the files directory
 <code>gwac.db</code>	Access control database
 <code>gwac.dc</code>	Database dictionary file used to create the gwac.db file

`domain\wpgate\webac65a` directory

The `domain\wpgate\webac65a` directory is the WebAccess Agent home directory where WebAccess Agent configuration files and queue directories are located. The name is established when you install the WebAccess Agent. The default is `wpgate\webac65a` in the domain directory. You can change the location using the `/home` startup switch in the WebAccess Agent configuration file (`webac65a.waa` in the WebAccess Agent installation directory) if you want to move the WebAccess Agent home directory.

000.prc directory

The NetWare and Windows WebAccess Agents use the `000.prc` directory to store log files.

On Linux, the `000.prc` directory is located under `/var/log`.

***mmddlog.nnn* file**

The *mmddlog.nnn* files hold error and status messages about the functioning of the WebAccess Agent. The WebAccess Agent creates a log file each day with a unique name, where *mm* is the month, *dd* is the day, and *nnn* is a sequential number indicating the sequence of log files in a single day. For more information about log files, see “[Controlling WebAccess Agent Logging](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

wpcsin directory

No longer used. The WebAccess Agent and the MTA communicate by way of TCP/IP and do not need queue directories.

wpcout directory

No longer used.

webxxxx directory

No longer used.

problem directory

No longer used.

gwhold directory

No longer used.

gwprob directory

No longer used.

files directory

The files directory holds documents that are cached after they have been viewed. The *gwcache.db* file keeps track of what documents are available in the files directory. You can configure the maximum amount of disk space you want the cached documents to consume. See “[Configuring the WebPublisher Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

template directory

The template directory holds the HTML templates used for viewing documents in HTML format.

commgr.cfg file

The *commgr.cfg* file in the WebAccess Agent queue directory contains information for communication between the WebAccess Agent and the WebAccess Application, including the IP address and port where the WebAccess Agent is running, the number of threads that are running, and the encryption key for the WebAccess Agent. This communications information is gathered during installation. For more information, see “[Configuring the GroupWise Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

As part of the installation process, the commgr.cfg file is automatically copied to the Web server installation ([sys:\novell\webaccess](#) on NetWare, [/opt/novell/groupwise/webaccess](#) on Linux, and [c:\novell\webaccess](#) on Windows). The copies are synchronized automatically by the WebAccess Application. The commgr.cfg file is also copied to the webpublisher subdirectory on the Web server.

comint.cfg file

The comint.cfg file in the WebAccess Agent queue directory is read by the WebAccess Agent on startup. It contains the same communications information as the commgr.cfg file and is synchronized with it automatically.

mimetype.cfg file

The mimetype.cfg file enables you to customize MIME content-type mappings for various attachment types. The WebAccess Agent handles this just as the Internet Agent does. For more information, see [“Customizing MIME Content-Type Mappings”](#) in [“Internet Agent”](#) in the *GroupWise 6.5 Administration Guide*

gwcache.db file

The gwcache.db file is like a catalog of documents in the files directory. When a user requests a document, WebAccess checks gwcache.db and if the document is already in the files directory, it can be quickly returned to the user. If the document is not yet in gwcache.db, it is retrieved, recorded in gwcache.db, converted for display in HTML format, and then returned to the user.



























gwac.db file

The gwac.db file is the access control database that stores information about the classes of service you have created. For more information, see [“Maintaining the Access Database”](#) in [“WebAccess”](#) in the *GroupWise 6.5 Administration Guide*.

gwac.dc file

The gwac.dc file is the data dictionary file from which the gwac.db is created.

Caching Mailbox Directory

 <code>\novell\groupwise\gwxxxxxx</code>	GroupWise Caching mailbox
 <code>rofdata</code>	Caching mailbox database directory
 <code>msg.db</code>	Cached message database
 <code>user.db</code>	Cached user database
 <code>wprof.db</code>	Cached Address Book
 <code>wprof.dc</code>	Data dictionary for cached Address Book
 <code>ngwguard.db</code>	Guardian database
 <code>ngwguard.dc</code>	Data dictionary for guardian database
 <code>ngwguard.rfl</code>	Guardian database roll forward log
 <code>ngwguard.fbk</code>	Guardian database "fall back" file
 <code>puxxxx.db</code>	Database for shared folders
 <code>ngwcheck.db</code>	GroupWise Check database
 <code>gwcheckn.log</code>	Log file created by the Repair Mailbox feature
 <code>remoten.log</code>	Connection log
 <code>gwdms</code>	Document Management Services directory
 <code>dmsb.db</code>	Shared DMS database
 <code>dmxxnn01-ff.db</code>	Document databases
 <code>docs</code>	Subdirectory for documents in the Caching mailbox
 <code>index</code>	QuickFinder index for documents in the Caching mailbox
 <code>index</code>	QuickFinder index for messages in the Caching mailbox
 <code>wpcsin</code>	Input queue for the Caching mailbox
 <code>0-7</code>	Priority subdirectories
 <code>wpcsoutofs</code>	Output queue for the Caching mailbox
 <code>0-7</code>	Priority subdirectories
 <code>wpgwsend</code>	Output queue to the Online mailbox
 <code>wpgwrecv</code>	Input queue from the Online mailbox

`\novell\groupwise\gwxxxxxx` directory

Your GroupWise Caching mailbox is a directory structure that functions similarly to a post office. Like a post office, it contains databases and input/output queues. It is created in the directory where the GroupWise client is installed, which is typically `\novell\groupwise`.

The same directory structure is used for a Caching mailbox as for a Remote mailbox. However, a Caching mailbox is a complete copy of your Online mailbox, while you can restrict what gets downloaded into your Remote mailbox.

rofdata directory

The rofdata directory contains the databases accessed by the GroupWise Windows client when running in Caching mode. The databases in rofdata are similar to the databases found in post offices. For comparison, see “[Post Office Directory](#)” on page 64.

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the rof in rofdata. Some naming conventions were originally preserved for backward compatibility.

msg.db file

The msg.db file is the cached equivalent of the [msgm.db](#) files in the [ofmsg](#) directory in your post office. The msg.db file contains copies of messages from your Online mailbox.

user.db file

The user.db file is the cached equivalent of the [userxxx.db](#) files in the [ofuser](#) directory in your post office.

wprof.db file

The wprof.db file contains the cached version of the GroupWise Address Book.

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the wprof in wprof.db. Some naming conventions have been preserved for backward compatibility.

wprof.dc file

The wprof.dc file is the data dictionary for the cached Address Book ([wprof.db](#)).

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the wprof in wprof.dc. Some naming conventions have been preserved for backward compatibility.

ngwguard.db file

The ngwguard.db file is the guardian database for your Caching mailbox. It is parallel in function to the [ngwguard.db](#) file in the post office.

ngwguard.dc file

The ngwguard.dc file is the data dictionary for building the databases in the GroupWise Caching mailbox. It is parallel in function to the [ngwguard.dc](#) file in the post office.

ngwguard.rfl file

The ngwguard.rfl file is a roll-forward transaction log of every database transaction that has taken place since the last copy of the [ngwguard.fbk](#) file was created. It is parallel in function to the [ngwguard.rfl](#) file in the post office.

ngwguard.fbk

The ngwguard.fbk file “fall back” copy of the **ngwguard.db** file. It is parallel in function to the **ngwguard.fbk** file in the post office.

puxxxxx.db files

The puxxxxx.db files are databases for replicated items such as shared folders. These databases prevent conflicts between user names of shared items from users in other post offices and user names in your own post office. They are parallel to the **puxxxxx.db** files in the post office.

ngwcheck.db file

The ngwcheck.db file tracks GroupWise Check threads and the databases being checked. In the GroupWise client, GroupWise Check is run using Tools > Repair Mailbox.

gwcheckn.log

The gwcheckn.log file records any errors that occurred during mailbox repair. For assistance with GroupWise Check errors, see “**GroupWise Check Error Codes**” in “**Administration Error Messages**” in the *GroupWise 6.5 Troubleshooting 1: Error Messages*.

remoten.log

The remoten.log files in the connection output queue (**wpcsout\ofs**) are saved versions of the connection logs you can view in the GroupWise client by clicking Accounts > Connection Log. These log files can be useful for troubleshooting problems with your connection to your Online mailbox.

gwdms directory

The gwdms directory is the Document Management Services directory. It contains information about the libraries in your GroupWise system. It has the same structure as the **gwdms** subdirectory in the post office.

dmsh.db file

The dmsh.db file is a database shared by all libraries that contains a list of all available libraries and lookup tables for each library.

dmxxnn01-ff.db files

The dmxxnn01-ff.db files are databases for library and document information. They are parallel to the **dmxxnn01-ff.db** files in the post office.

docs directory

The docs directory holds cached copies of the documents in your Online mailbox.

index directory

The index directory under the gwdms directory contains the QuickFinder index for the documents in your Caching mailbox.

index directory

The index directory under the rofdata directory contains the QuickFinder index for the messages in your Caching mailbox.

wpcsin directory

The wpcsin subdirectory is the input queue for the connection that transfers messages to your GroupWise system for delivery. Messages from the GroupWise client in Caching mode are processed through the priority 1 subdirectory of wpcsin.

When you send a message in Caching mode, the GroupWise client connects to your GroupWise system. It polls the wpcsin\1 directory and compresses any outgoing messages, requests, or both into a file. If the compressed file totals over 50 KB, additional compressed files are created. The GroupWise client then moves the compressed files into the **wpgwsend** directory.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory name wpcsin for the input queue, although the MTA is not involved in processing messages in your Caching mailbox. Some naming conventions were originally preserved for backward compatibility.

0-7 directories

The priority 0-7 subdirectories in the connection input queue (**wpcsin**) parallel those found in the **wpcsin** directory in your post office.

wpcout\ofs directory

The wpcout\ofs directory is the output queue for the connection that transfers messages from your Online mailbox. Messages from your GroupWise system are processed through the priority 1 subdirectory of wpcout\ofs.

The GroupWise client scans the wpcout\ofs\1 subdirectory and updates the **user.db** and **msg.db** files with the information received from your Online mailbox.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory names wpcsin and ofs for the input queue, though the MTA is not involved in processing messages in your Remote mailbox. Some naming conventions were originally preserved for backward compatibility.

0-7 directories

The priority 0-7 subdirectories in the connection output queue (**wpcout\ofs**) parallel those found in the **ofs** directory in your post office.

wpgwsend directory

The wpgwsend directory holds compressed files that contain outgoing messages, requests, or both. When a connection to your GroupWise system is established, the GroupWise client uploads the files to your Online mailbox.




















Historical Note: WP Office Remote, the predecessor of GroupWise Remote, was originally designed by WordPerfect Corporation (WPCorp). Hence, the name wpgwsend. Some naming conventions were originally preserved for backward compatibility.

wpgwrecv directory

The wpgwrecv directory holds compressed files that contain messages or other information that have been received from your Online mailbox. The GroupWise client decompresses the files and places the message files into the wpcout\ofs\1 directory.

Historical Note: WP Office Remote, the predecessor of GroupWise Remote, was originally designed by WordPerfect Corporation (WPCorp). Hence, the name wpgwrecv. Some naming conventions were originally preserved for backward compatibility.

Remote Mailbox Directory

 remote_mailbox	GroupWise Remote mailbox
 wpcsin	Input queue for GroupWise Remote
 0-7	Priority subdirectories
 rofdata	Remote database directory
 msg.db	Remote message database
 user.db	Remote user database
 wprof.db	Remote Address Book
 wprof.dc	Data dictionary for Remote Address Book
 ngwguard.db	Remote guardian database
 ngwguard.dc	Data dictionary for Remote guardian database
 puxxxx.db	Database for shared folders
 gwdms	Document Management Services directory
 dmsb.db	Shared DMS database
 index	QuickFinder index for Remote mailbox
 wpgwsend	Output queue to master mailbox
 wpgwrecv	Input queue from master mailbox
 wpcoutofs	Output queue for GroupWise Remote
 0-7	Priority subdirectories
 remoten.log	Remote connection log

remote_mailbox directory

The GroupWise **remote mailbox** is a directory structure that functions similarly to a post office. Like a post office, it contains databases and input/output queues.

wpcsin directory

The wpcsin subdirectory in the **remote mailbox directory** is the input queue for the connection that will transfer messages to your master GroupWise system. Messages from GroupWise Remote are processed through the priority 1 subdirectory of wpcsin.

When the user initiates the connection to the master GroupWise system, GroupWise Remote polls the wpcsin\1 directory and compresses the outgoing messages, requests, or both into a file. If the compressed file totals over 50 KB, additional compressed files are created. GroupWise Remote then moves the compressed files into the **wpgwsend** directory.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory name wpcsin for the input queue, although the MTA is not involved in processing messages in your Remote mailbox. Some naming conventions were originally preserved for backward compatibility.

0-7 directories

The priority 0-7 subdirectories in the connection input queue (**wpcsin**) in the **remote mailbox directory** parallel those found in the **wpcsin** directory in the remote user's post office on the master GroupWise system.

rofdata directory

The rofdata subdirectory in the **remote mailbox directory** contains the databases accessed by GroupWise Remote. The databases in rofdata are similar to the databases found in post offices.

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the rof in rofdata. Some naming conventions were originally preserved for backward compatibility.

msg.db file

The msg.db file in the remote data directory (**rofdata**) in the **remote mailbox directory** is the remote equivalent of the **msgm.db** files in the **ofmsg** directory in the post office where your master mailbox is located. The msg.db file contains messages you have downloaded from your master mailbox.

user.db file

The user.db file in the remote data directory (**rofdata**) in the **remote mailbox directory** is the remote equivalent of the **userxxx.db** files in the **ofuser** directory in the post office where your master mailbox is located. The user.db file contains user information you have downloaded from your master mailbox.

wprof.db file

The wprof.db file in the remote data directory (**rofdata**) in the **Remote mailbox directory** contains the remote version of the GroupWise Address Book if you have downloaded it.

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the wprof in wprof.db. Some naming conventions have been preserved for backward compatibility.

wprof.dc file

The wprof.dc file in the remote data directory (**rofdata**) in the **Remote mailbox directory** is the data dictionary for the remote Address Book (**wprof.db**).

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the wprof in wprof.dc. Some naming conventions have been preserved for backward compatibility.

ngwguard.db file

The ngwguard.db file in the remote data directory (**rofdata**) in the **Remote mailbox directory** is the guardian database for the GroupWise Remote mailbox. It is parallel in function to the **ngwguard.db** file in the post office.

ngwguard.dc file

The ngwguard.dc file in the remote data directory (**rofdata**) in the **Remote mailbox directory** is the data dictionary for building the databases in the GroupWise Remote mailbox. It is parallel in function to the **ngwguard.dc** file in the post office.

puxxxxx.db files

The puxxxxx.db files in the remote data directory (**rofdata**) in the **Remote mailbox directory** are databases for replicated items such as shared folders. These databases prevent conflicts between user names of shared items from users in other post offices and user names in the Remote user's post office. They are parallel to the **puxxxxx.db** files in the post office.

gwdms directory

The gwdms subdirectory in the remote data directory (**rofdata**) in the **remote mailbox directory** is the Document Management Services directory. It contains the document libraries you have downloaded from your master GroupWise system. It has the same structure as the **gwdms** subdirectory in the post office.

dmsh.db file

The dmsh.db file in the remote document management subdirectory (**gwdms**) in the **remote mailbox directory** is a database shared by all libraries that have been downloaded to your GroupWise Remote mailbox. It contains a list of all available libraries and lookup tables for each library.

index directory

The index directory in the remote document management subdirectory (**gwdms**) in the **remote mailbox directory** contains the QuickFinder index for the documents contained in the library.

wpgwsend directory

The wpgwsend subdirectory in the **remote mailbox directory** holds compressed files that contain outgoing messages, requests, or both. When a connection to your master GroupWise system is established, GroupWise Remote uploads the files across the link.

Historical Note: WP Office Remote, the predecessor of GroupWise Remote, was originally designed by WordPerfect Corporation (WPCorp). Hence, the name wpgwsend. Some naming conventions were originally preserved for backward compatibility.

wpgwrecv directory

The wpgwrecv subdirectory in the **Remote mailbox directory** holds compressed files that contain messages or other information that have been downloaded from your master mailbox. GroupWise Remote decompresses the files and places the message files into the wpcout\ofs\1 directory.

Historical Note: WP Office Remote, the predecessor of GroupWise Remote, was originally designed by WordPerfect Corporation (WPCorp). Hence, the name wpgwrecv. Some naming conventions were originally preserved for backward compatibility.

wpcout\ofs directory

The wpcout\ofs subdirectory in the **remote mailbox directory** is the output queue for the connection that transfers messages from your master GroupWise system. Messages from GroupWise Remote are processed through the priority 1 subdirectory of wpcout\ofs.

GroupWise Remote scans the wpcout\ofs\1 subdirectory and updates the **user.db** and **msg.db** files with the information received from the master mailbox.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). The Message Transfer Agent (MTA) was originally named the Connection Server (CS). Hence, the directory names wpcsin and ofs for the input queue, though the MTA is not involved in processing messages in your Remote mailbox. Some naming conventions were originally preserved for backward compatibility.

0-7 directories

The priority 0-7 subdirectories in the connection output queue (**wpcout\ofs**) in the **remote mailbox directory** parallel those found in the **ofs** subdirectory in the remote user's post office on the master GroupWise system.

remoten.log

The *remoten.log* files in the connection output queue (**wpcout\ofs**) in the **remote mailbox directory** are saved versions of the connection logs you can view in GroupWise Remote. These log files can be useful for troubleshooting problems with your modem or with your connection to your master GroupWise system.

8

Agent Installation Directories














- ◆ “GroupWise Agent Installation (POA and MTA)” on page 97
- ◆ “Internet Agent Installation” on page 107
- ◆ “WebAccess Agent Installation” on page 113
- ◆ “Monitor Agent Installation” on page 123
- ◆ “Apache/Tomcat Installation” on page 127








GroupWise Agent Installation (POA and MTA)

The Post Office Agent (POA) and the Message Transfer Agent (MTA) are always installed together. The agent installation directory differs depending on the platform where the agents are installed.

- ◆ “NetWare Installation Directory” on page 97
- ◆ “Linux Installation Directory” on page 100
- ◆ “Windows Installation Directory” on page 105

NetWare Installation Directory

 sys:\system	NetWare® agent installation directory
 gwenn4.nlm	GroupWise® 6.5 Agent Engine
 gwpoa.nlm	Post Office Agent (POA) program
 gwmta.nlm	Message Transfer Agent (MTA) program
 strtupxx.poa	Boilerplate POA startup file
 strtupxx.mta	Boilerplate MTA startup file
 post_office.poa	POA startup file for a specific post office
 domain.mta	MTA startup file for a specific domain
 gwpoanxx.hlp	POA online help files
 gwmtanxx.hlp	MTA online help files
 gwpoanxx.fil	POA language information files
 gwmtanxx.fil	MTA language information files
 grpwise.ncf	Agent load file

 x*10.nlm	XIS NLM™ programs
 agent.xml	XIS agent configuration files
 gwtsa.nlm	GroupWise Target Service Agent (GWTSa) program
 gwtsaxx.fil	GWTSa language information files
 gwtsa.ncf	GWTSa load file
 gwtmstmp.nlm	GroupWise Time Stamp program
 help	Subdirectory for GroupWise agent Web console help files

NetWare agent installation directory

When you first install GroupWise Administration, the GroupWise agents are initially installed in platform-specific subdirectories in the **agent** directory in the software distribution directory. However, the agents cannot be run from this initial location. They must be installed on the servers where they will run.

On a NetWare® server, the GroupWise agents are typically installed in the sys:\system directory. You can choose a different location during installation.

gwenn4.nlm file

The gwenn4.nlm file is the GroupWise 6 Agent Engine, a program that is shared by all GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations
- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent loaded on a server automatically loads the GroupWise Agent Engine.

gwpoa.nlm file

The gwpoa.nlm file is the Post Office Agent program. You load this NLM program to start the Post Office Agent. See “Starting the POA” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide*.

gwmta.nlm file

The gwmta.nlm is the Message Transfer Agent program. You load this NLM program to start the Message Transfer Agent. See “Starting the MTA” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

strtpxx.poa file

The POA startup file contains startup switches for the POA. Switch settings placed in the POA startup file override comparable options set for the POA in ConsoleOne®. The *xx* in the startup filename represents a two-letter language code.

During installation, a customized version of the POA startup file, named *post_office.POA*, is created. This customized version has the */home* startup switch automatically set to the post office directory the POA will service. See “Using POA Startup Switches” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide*.

strtpxx.mta file

The MTA startup file contains startup switches for the MTA. Switch settings placed in the MTA startup file override comparable options set for the MTA in ConsoleOne. The *xx* in the startup filename represents a two-letter language code.

During installation, a customized version of the MTA startup file, named *domain.MTA*, is created. This customized version has the */home* startup switch automatically set to the domain directory the MTA will service. See “Using MTA Startup Switches” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

nlm_agentnxx.hlp files

Online help is available by pressing F1 at the agent console on the server where it runs. See “Using the POA Agent Console” in “Post Office Agent” and “Using the MTA Agent Console” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*

The first five characters of the filename are the agent name. The digit *n* is a version number. The last two characters *xx* are a language code.

nlm_agentnxx.fil files

These files contain all language-specific information for the POA and the MTA. The first five characters of the filename are the agent name. The digit *n* is a version number. The last two characters *xx* are a language code.

grpwise.ncf file

The *grpwise.ncf* file is the NetWare configuration file that automatically loads the agents whenever the server is restarted. A typical *grpwise.ncf* file would look similar to the following:

```
load sys:\system\gwmta.nlm @corpdom.mta
load sys:\system\gwpoa.nlm @acctpo.poa
```

For more information, see “Starting the NetWare POA” in “Post Office Agent” and “Starting the NetWare MTA” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

x*10.nlm files

The *x*10.nlm* programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.

nlm_agent.xml files

The *nlm_agent.xml* files are specialized agent configuration files for use in the XIS environment.

gwtsa.nlm file

The gwtsa.nlm is the GroupWise Target Service Agent (GWTSA), which provides reliable backups of a running GroupWise system by successfully backing up open files and locked files, rather than skipping them as some backup software does. For more information, see “GroupWise Target Service Agent” in “Databases” in the *GroupWise 6.5 Administration Guide*.

gwtmstmp.nlm file

If you use the GroupWise Target Service Agent (GWTSA) to back up your GroupWise system, the time stamping is an automatic part of the backup process. However, if you choose not to use the GWTSA, you must still make sure that user databases are time-stamped so that items will not be prematurely purged. The gwtmstmp.nlm accomplishes this task. For more information, see “GroupWise Time Stamp Utility” in “Databases” in the *GroupWise 6.5 Administration Guide*.

gwtsaxx.fil file

These files contain all language-specific information for the GWTSA. The last two characters, *xx*, are a language code.

gwtsa.ncf file





The gwtsa.ncf file is the NetWare configuration file that automatically loads the GWTSA whenever the server is restarted. A typical gwtsa.ncf file would look similar to the following:































```
load sys:\system\smdr
load sys:\system\gwtsa /home-sys:\gwsystem\corpdm
    /home-sys:\gwsystem\acctpo
```

help directory

The help directory contains language-specific subdirectories for the help files available from the POA and MTA Web consoles. See “Using the POA Web Console” in “Post Office Agent” and “Using the MTA Web Console” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

Linux Installation Directory

 /opt/novell/groupwise/agents	Linux agent installation directory
 bin	Subdirectory for GroupWise agent executables
 gwpoa	Post Office Agent (POA) executable
 gwmta	Message Transfer Agent (MTA) executable
 gwpoa.xml	POA XIS configuration file
 gwmta.xml	MTA XIS configuration file
 strtupxx.poa	Boilerplate POA startup file
 strtupxx.mta	Boilerplate MTA startup file
 gwcsrgen	GroupWise Generate CSR utility
 gwtmstmp	GroupWise Time Stamp utility

 lib	Subdirectory for GroupWise agent library files
 gwpoaxxx.fil	POA language information file
 gwmtaxxx.fil	MTA language information file
 libpoa.so*	POA shared library files
 libpoa_ui.so*	POA shared library files for the optional user interface
 libmta.so*	MTA-specific library files
 libmta_ui.so*	MTA-specific library files for the optional user interface
 *.so*	Supporting library files
 image	Subdirectory for icon images
 share	Subdirectory for agent shared files
 post_office.poa	POA startup file for a specific post office
 domain.mta	MTA startup file for a specific domain
 agtcon	Subdirectory for agent console files
 help	Subdirectory agent console help files
 webcon	Subdirectory for agent Web console files
 help	Subdirectory for agent Web console help files
 /etc/init.d	Standard Linux location for application startup scripts
 grpwise	Startup script for the POA and MTA
 rc3.d	Standard Linux location for run-level-3 symbolic links
 S99grpwise	Symbolic link to the startup script for the POA and MTA
 rc5.d	Standard Linux location for run-level-5 symbolic links
 S99grpwise	Symbolic link to the startup script for the POA and MTA
 /var/log/	Standard Linux location for application log files
 novell/groupwise	Subdirectory for GroupWise agent log files
 post_office.poa	Post office-specific subdirectory for POA log files
 mmddpoa.nnn	POA log files
 domain.mta	Domain-specific subdirectory for MTA log files
 mmddmta.nnn	MTA log files
 /var/opt/novell/groupwise/log	Location for script log files
 mmddlog.nnn	Log file for the agent startup script (grpwise)

agents directory

On a Linux server, the agents are always installed in subdirectories of `/opt/novell/groupwise/agents`.

bin directory

The bin directory holds GroupWise executable files.

gwpoa file

The gwpoa file is the Post Office Agent executable. You run this executable file to start the Post Office Agent. See “Starting the POA” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide*.

gwmta file

The gwmta file is the Message Transfer Agent executable. You run this executable file to start the Message Transfer Agent. See “Starting the MTA” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

*.xml files

The *agent.xml* files are specialized agent configuration files for use in the XIS environment.

strtpxx.poa file

The strtpxx.poa file is the boilerplate file from which a post office-specific *post_office.poa* file is created in the *share* directory. The *xx* in the startup filename represents a two-letter language code.

strtpxx.mta file

The strtpxx.mta file is the boilerplate file from which a domain-specific *domain.poa* file is created in the *share* directory. The *xx* in the startup filename represents a two-letter language code.

gwcsrgen file

The gwcsrgen file is the GroupWise Generate CSR utility. If you enable SSL for the agents, they need access to a server certificate and private key. You can use the GroupWise Generate CSR utility (GWCSRGEN) to generate a Certificate Signing Request (CSR) file and a Private Key file. For more information, see “Server Certificates and SSL Encryption” in “Security” in the *GroupWise 6.5 Administration Guide*.

gwtmstmp file

The gwtmstmp file is the GroupWise Time Stamp utility. If you deselect Allow Purge of Items Not Backed Up in ConsoleOne, user databases (*userxxx.db*) must be time-stamped every time a backup is performed so that items can be purged only after being backed up. You can use the GroupWise Time Stamp (GWTMSTMP) utility to ensure that GroupWise user databases include the dates when they were last backed up, restored, and retained. For more information, see “GroupWise Time Stamp Utility” in “Databases” in the *GroupWise 6.5 Administration Guide*.

lib directory

The lib directory holds GroupWise library files.

***.fil files**

These files contain all language-specific information for the POA and the MTA. The first five characters of the filename are the agent name. The last two characters *xx* are a language code.

***.so* files**

These files are Linux library files that provide information to the GroupWise agent executables.

image directory

The image directory has subdirectories for agent icons for both color and monochrome displays.

share directory

The share directory holds agent startup files and files that are used by the agent consoles and Web consoles.

post_office.poa

The POA startup file contains startup switches for the POA. Switch settings placed in the POA startup file override comparable options set for the POA in ConsoleOne.

During installation, a customized version of the *strdupxx.poa* file, named *post_office.poa*, is created in the share directory. This customized version has the **--home** startup switch automatically set to the post office directory the POA will service. See “Using POA Startup Switches” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide*.

domain.mta

The MTA startup file contains startup switches for the MTA. Switch settings placed in the MTA startup file override comparable options set for the MTA in ConsoleOne.

During installation, a customized version of the *strdupxx.mta* file, named *domain.mta*, is created in the share directory. This customized version has the **--home** startup switch automatically set to the domain directory the MTA will service. See “Using MTA Startup Switches” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

agtcon directory

The agtcon directory holds subdirectories and files used by the agent consoles, such as help files. See “Using the POA Agent Console” in “Post Office Agent” and “Using the MTA Agent Console” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

webcon directory

The webcon directory holds subdirectories and files used by the agent Web consoles, such as help files. See “Using the POA Web Console” in “Post Office Agent” and “Using the MTA Web Console” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

/etc/init.d directory

The /etc/init.d directory is the standard location for Linux startup scripts.

grpwise file

The grpwise script is created automatically during installation. You can use the script to start, restart, stop, and display status information about the POA and MTA. For more information about

starting the agents, see “[Installing and Starting the GroupWise Linux Agents](#)” in “[Installing a Basic GroupWise System](#)” in the *GroupWise 6.5 Installation Guide*.

rc3.d directory

The rc3.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 3 (multi-user; boots to a text mode login prompt without The X Window System). The symbolic link to the grpwise script is S99grpwise. It is created if you choose during installation to have the agents start automatically when the server boots. See “[Starting the Linux POA](#)” in “[Post Office Agent](#)” and “[Starting the Linux MTA](#)” in “[Message Transfer Agent](#)” in the *GroupWise 6.5 Administration Guide*.

rc5.d directory

The rc5.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 5 (multi-user; boots to The X Window System login dialog box). The symbolic link to the grpwise script is S99grpwise. It is created if you choose during installation to have the agents start automatically when the server boots. See “[Starting the Linux POA](#)” in “[Post Office Agent](#)” and “[Starting the Linux MTA](#)” in “[Message Transfer Agent](#)” in the *GroupWise 6.5 Administration Guide*.

/var/log/novell/groupwise directory

The /var/log directory is the standard location for log files on Linux. All GroupWise agent log files are created in the novell/groupwise subdirectory.

post_office.poa directory

The *post_office.poa* directory is a post office-specific location for POA log files.

Within the *post_office.poa* directory, the POA creates log files (*mmddpoa.nnn*) to inform you of its processing and any problems it encounters. For more information about log files, see “[Using POA Log Files](#)” in “[Post Office Agent](#)” in the *GroupWise 6.5 Administration Guide* guide.

The first two digits of the filename represent the month, the next two digits represent the day of the month, and the next three characters indicate what program created the log. The three-digit extension is a sequence number for multiple log files created on the same day. For example, 0518poa.002 is the second POA log file created on May 18.

domain.mta directory

The *domain.mta* directory is a domain-specific location for MTA log files.

Within the *domain.mta* directory, MTA creates log files (*mmddxxx.nnn*) to inform you of its processing and any problems it encounters. For more information about log files, see “[Using MTA Log Files](#)” in “[Message Transfer Agent](#)” in the *GroupWise 6.5 Administration Guide*.

The first two digits of the filename represent the month; the next two digits represent the day of the month; the next three characters indicate what program created the log. The three-digit extension is a sequence number for multiple log files created on the same day. For example, 0518mta.002 is the second MTA log file created on May 18.


















/var/opt/novell/groupwise/log directory

The /var directory is used for various types of files. The log file for the agent startup script is created in the opt/novell/groupwise/log subdirectory.

mmddlog.nnn

The *mmddlog.nnn* file stores messages from the agent startup script (**grpwise**) as it tries to start the GroupWise agents.

Windows Installation Directory

 c:\grpwise	Windows agent installation directory
 gwenv1a.dll	GroupWise Agent Engine
 gwpoa.exe	Post Office Agent (POA) program
 gwmta.exe	Message Transfer Agent (MTA) program
 strtpxx.poa	Boilerplate POA startup file [need post.poa]
 strtpxx.mta	Boilerplate MTA startup file [need domain.mta]
 post_office.poa	POA startup file for a specific post office
 domain.mta	MTA startup file for a specific domain
 gwpoanxx.chm	POA online help file
 gwmtanxx.chm	MTA online help file
 gwpoanxx.dll	POA language information files
 gwmtanxx.dll	MTA language information files
 gwsnmp.dll	Customized DLL program for SNMP
 gwww1.dll	Customized DLL program for MIME
 x*10.dll	XIS DLL programs
 agent.xml	XIS agent configuration files
 help	Subdirectory for GroupWise agent Web console help files

c:\grpwise

When you first install GroupWise Administration, the GroupWise agents are initially installed in platform-specific subdirectories in the **agent** directory. However, the agents cannot be run from this initial location. They must be installed on the servers where they will run.

On a Windows server, the GroupWise agents can be installed in any directory you choose. The default is c:\grpwise. The agent icons are set up to include the full path to whatever directory you choose.

gwenv1a.dll file

The gwenv1a.dll file is the GroupWise Agent Engine, a program that is shared by both GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations

- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent started on a server automatically starts the GroupWise Agent Engine DLL.

gwpoa.exe file

The gwpoa.exe file is the Post Office Agent program. You run this executable file to start the Post Office Agent. See “Starting the POA” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide*.

gwmta.exe file

The gwmta.exe file is the Message Transfer Agent program. You run this executable file to start the Message Transfer Agent. See “Starting the MTA” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

gwsnmp.dll file

The gwsnmp.dll file provides interaction with the Windows SNMP Service, so that you can monitor the GroupWise agents using an SNMP monitoring program.

gwww1.dll file

The gwww1.dll file provides parsing of MIME messages received from the Internet.

strtupxx.poa file

The POA startup file contains startup switches for the POA. Switch settings placed in the POA startup file override comparable options set for the POA in ConsoleOne. The xx in the startup filename represents a two-letter language code.

During installation, a customized version of the POA startup file, named *post_office.poa*, is created. This customized version has the */home* startup switch automatically set to the post office directory the POA will service. See “Using POA Startup Switches” in “Post Office Agent” in the *GroupWise 6.5 Administration Guide*.

strtupxx.mta file

The MTA startup file contains startup switches for the MTA. Switch settings placed in the MTA startup file override comparable options set for the MTA in ConsoleOne. The xx in the startup filename represents a two-letter language code.

During installation, a customized version of the MTA startup file, named *domain.MTA*, is created. This customized version has the */home* startup switch automatically set to the domain directory the MTA will service. See “Using MTA Startup Switches” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

agentnxx.chm files

Online Help is available by clicking Help in the agent consoles on the server where the agents are running. In addition, dialog boxes have a Help button for context-sensitive Help.

The first five characters of the filename are the agent name. The digit *n* is a version number. The last two characters *xx* are a language code.

agentnxx.dll files

The *agentnxx.dll* files contain all language-specific information for the agents. The digit *n* is a version number. The last two characters *xx* are a language code.

x*10.dll files

The *x*10.dll* programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.

agent.xml files

The *agent.xml* files are specialized agent configuration files for use in the XIS environment.










help directory

The help directory contains language-specific subdirectories for the help files available from the POA and MTA Web consoles. See “Using the POA Web Console” in “Post Office Agent” and “Using the MTA Web Console” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

Internet Agent Installation

- ◆ “NetWare Installation Directory” on page 107
- ◆ “Linux Installation Directory” on page 109
- ◆ “Windows Installation Directory” on page 112

NetWare Installation Directory

 sys:\system	NetWare Internet Agent installation directory
 gwenn4.nlm	GroupWise 6.5 Agent Engine
 gwia.nlm	Internet Agent NLM program
 gwia.cfg	Internet Agent configuration file
 gwianxx.fil	Internet Agent language information file
 gwia.ncf	Internet Agent load file
 x*10.nlm	XIS NLM programs
 gwia.xml	XIS Internet Agent configuration file
 help	Subdirectory for GroupWise agent console help files

See also “Internet Agent Queue Directory” on page 80.

NetWare Internet Agent installation directory

On a NetWare server, the GroupWise Internet Agent is typically installed in the `sys:\system` directory. You can choose a different location during installation.

gwenn4.nlm file

The `gwenn4.nlm` file is the GroupWise 6.x Agent Engine, a program that is shared by all GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations
- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent loaded on a server automatically loads the GroupWise Agent Engine.

gwia.nlm file

The `gwia.nlm` file is the Internet Agent program. You load this NLM program to start the Internet Agent. See “Starting the Internet Agent” in “Installing the GroupWise Internet Agent” in the *GroupWise 6.5 Installation Guide*.

gwia.cfg file

The Internet Agent configuration file contains startup switches for the Internet Agent. Switch settings placed in the Internet Agent startup file override comparable options set for the Internet Agent in ConsoleOne. The startup file is named the same as the Internet Agent object in ConsoleOne. The default is `gwia`.

During installation, a customized version of the Internet Agent startup file is created in `sys:\system`. This customized version has the `/home` startup switch automatically set to the domain directory where the Internet Agent’s queue directories are located. See “Using Internet Agent Startup Switches” in “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

gwianxx.fil files

This file contains all language-specific information for the Internet Agent. The digit *n* is a version number. The last two characters *xx* are a language code.

gwia.ncf file

The `gwia.ncf` file is the NetWare configuration file that automatically loads the Internet Agent whenever the server is restarted. A typical `gwia.ncf` file would look similar to the following:

```
load sys:\system\gwia @gwia.cfg
```



























x*10.nlm files

The `x*10.nlm` programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.

gwia.xml file

The gwia.xml file is a specialized Internet Agent configuration file for use in the XIS environment.

Linux Installation Directory

 /opt/novell/groupwise/agents	Linux agent installation directory
 bin	Subdirectory for GroupWise agent executables
 gwia	Internet Agent executable
 gwia.xml	Internet Agent XIS configuration file
 lib	Subdirectory for GroupWise agent library files
 gwiaxxx.fil	Internet Agent language information file
 libgwia.so	Internet Agent shared library file
 libgwia_ui.so	Internet Agent user interface shared library file
 *.so	Supporting shared library files
 image	Subdirectory for icon images
 share	Subdirectory for agent shared files
 gwia.cfg	Internet Agent configuration file
 agtcon	Subdirectory for agent console files
 help	Subdirectory agent console help files
 webcon	Subdirectory for agent Web console files
 help	Subdirectory for agent Web console help files
 /etc/init.d	Standard Linux location for application startup scripts
 grpwise-ia	Startup script for the Internet Agent
 rc3.d	Standard Linux location for run-level-3 symbolic links
 S99grpwise-ia	Symbolic link to the startup script for the Internet Agent
 rc5.d	Standard Linux location for run-level-5 symbolic links
 S99grpwise-ia	Symbolic link to the startup script for the Internet Agent
 /var/log/	Standard Linux location for application log files
 novell/groupwise	Subdirectory for GroupWise agent log files
 domain.gwia	Domain-specific subdirectory for Internet Agent log files
 mmdgwia.nnn	Internet Agent log files

See also “[Internet Agent Queue Directory](#)” on page 80.

agents directory

On a Linux server, the Internet Agent is always installed in subdirectories of /opt/novell/groupwise/agents.

bin directory

The bin directory holds GroupWise executable files.

gwia file

The gwia file is the Internet Agent executable. You run this executable file to start the Internet Agent. See “[Starting the Internet Agent](#)” in “[Installing the GroupWise Internet Agent](#)” in the *GroupWise 6.5 Installation Guide*.

gwia.xml file

The gwia.xml file is a specialized configuration file for use in the XIS environment.

lib directory

The lib directory holds GroupWise shared library files.

gwiaxxx.fil files

This file contains all language-specific information for the Internet Agent. The last two characters *xx* are a language code.

***.so files**

These files are Linux shared library files that provide information to the Internet Agent executable.

image directory

The image directory has subdirectories for agent icons for both color and monochrome displays.

share directory

The share directory holds agent startup files and files that are used by the agent consoles and Web consoles.

gwia.cfg

The Internet Agent configuration file contains startup switches for the Internet Agent. Switch settings placed in the Internet Agent configuration file override comparable configuration options set for the Internet Agent in ConsoleOne. The startup file is named the same as the Internet Agent object in ConsoleOne. The default is gwia.

During installation, the Internet Agent startup file is created in the share directory with the **--home** startup switch automatically set to the domain directory where the Internet Agent queue directory is located. See “[Using Internet Agent Startup Switches](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

agtcon directory

The agtcon directory holds subdirectories and files used by the agent consoles, such as help files.

webcon directory

The webcon directory holds subdirectories and files used by the agent Web consoles, such as help files.

/etc/init.d directory

The /etc/init.d directory is the standard location for Linux startup scripts.

grpwise-ia file

The grpwise script is created automatically during installation. You can use the script to start, restart, stop, and display status information about the Internet Agent. For more information about starting the Internet Agent, see “[Starting the Internet Agent](#)” in “[Installing the GroupWise Internet Agent](#)” in the *GroupWise 6.5 Installation Guide*.

rc3.d directory

The rc3.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 3 (multi-user; boots to a text mode login prompt without The X Window System). The symbolic link to the grpwise-ia script is S99grpwise-ia. It is created if you choose during installation to have the Internet Agent start automatically when the server boots.

rc5.d directory

The rc5.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 5 (multi-user; boots to The X Window System login dialog box). The symbolic link to the grpwise-ia script is S99grpwise-ia. It is created if you choose during installation to have the Internet Agent start automatically when the server boots.

/var/log/novell/groupwise directory

The /var/log directory is the standard location for log files on Linux. All GroupWise agent log files are created in the novell/groupwise subdirectory.










domain.gwia directory

The *domain.gwia* directory is a domain-specific location for Internet Agent log files.

Within the *domain.gwia* directory, the Internet Agent creates log files (*mmdgwia.nnn*) to inform you of its processing and any problems it encounters. For more information about log files, see “[Using Internet Agent Log Files](#)” in “[Internet Agent](#)” in the *GroupWise 6.5 Administration Guide*.

The first two digits of the filename represent the month; the next two digits represent the day of the month. The three-digit extension is a sequence number for multiple log files created on the same day. For example, 0518gwia.002 is the second Internet Agent log file created on May 18.

Windows Installation Directory

 c:\grpwise\gwia	Windows Internet Agent installation directory
 gwenv1a.dll	GroupWise Agent Engine
 gwia.exe	Internet Agent program
 gwia.cfg	Internet Agent configuration file
 gwianxx.chm	Internet Agent online help file
 gwianxx.dll	Internet Agent language information file
 gwww1.dll	Customized DLL program for MIME
 x*10.dll	XIS DLL programs
 gwia.xml	XIS agent configuration file

See also “[Internet Agent Queue Directory](#)” on page 80.

c:\grpwise\gwia

On a Windows server, the Internet Agent can be installed in any directory you choose. The default location is c:\grpwise\gwia. The Internet Agent icon is set up to include the full path to whatever directory you choose.

gwenv1a.dll file

The gwenv1a.dll file is the GroupWise Agent Engine, a program that is shared by both GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations
- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent started on a server automatically starts the GroupWise Agent Engine DLL.

gwia.exe file

The gwia.exe file is the Internet Agent program. You run this executable file to start the Internet Agent. See “[Starting the Internet Agent](#)” in “[Installing the GroupWise Internet Agent](#)” in the *GroupWise 6.5 Installation Guide*.

gwia.cfg file

On Windows, the gwia.cfg file in the installation directory is simply a pointer to the gwia.cfg file under the domain directory where the actual Internet Agent configuration file is located.

gwianxx.chm files

Online help is available by clicking Help in the Internet Agent console on the server where the agent is running. In addition, dialog boxes have a Help button for context-sensitive Help.

The digit *n* is a version number. The last two characters *xx* are a language code.

gwianxx.dll files

The gwianxx.dll files contain all language-specific information for the Internet Agent. The digit *n* is a version number. The last two characters *xx* are a language code.

gwww1.dll file

The gwww1.dll file provides parsing of MIME messages received from the Internet.

x*10.dll files

The x*10.dll programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.


gwia.xml file

The gwia.xml file is a specialized agent configuration file for use in the XIS environment.

WebAccess Agent Installation

- ◆ [“NetWare Installation Directory” on page 113](#)
- ◆ [“Linux Installation Directory” on page 115](#)
- ◆ [“Windows Installation Directory” on page 120](#)

NetWare Installation Directory

 sys:\system	NetWare WebAccess Agent installation directory
 gwenn4.nlm	GroupWise 6.5 Agent Engine
 gwinter.nlm	WebAccess Agent program
 strtweb.ncf	WebAccess Agent load file
 webac65a.waa	WebAccess Agent startup file
 x*10.nlm	XIS NLM programs
 gwinter.xml	XIS Internet Agent configuration file
 cache\template\xx	Subdirectory for HTML template files
 webacc.tpt	WebAccess template files
 publish.tpt	WebPublisher template files
 help	Subdirectory for GroupWise agent console help files

Additional WebAccess files are integrated into your Web server to support the WebAccess client and the WebAccess Agent Web console. See [“Apache/Tomcat Installation” on page 127](#). See also [“WebAccess Agent Queue Directory” on page 86](#).

NetWare Internet Agent installation directory

On a NetWare server, the GroupWise WebAccess Agent is typically installed in the sys:\system directory. You can choose a different location during installation.

gwenn4.nlm file

The gwenn4.nlm file is the GroupWise 6.x Agent Engine, a program that is shared by all GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations
- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent loaded on a server automatically loads the GroupWise Agent Engine.

gwinter.nlm file

The gwinter.nlm file is the WebAccess Agent program.

strtweb.ncf

The strtweb.ncf file is the NetWare configuration file that automatically loads the WebAccess Agent whenever the server is restarted. A typical strtweb.ncf file would look similar to the following:

```
load sys:\system\gwinter @webac65a.waa
```

It includes the WebAccess Agent startup file in the load command. For more information, see [“Starting GroupWise WebAccess”](#) in [“Installing GroupWise WebAccess”](#) in the *GroupWise 6.5 Installation Guide*.

webac65a.waa file

The WebAccess Agent startup file contains startup switches for the WebAccess Agent. Switch settings placed in the WebAccess Agent startup file override comparable options set for the WebAccess Agent in ConsoleOne. The startup file is named the same as the WebAccess Agent object in ConsoleOne. The default is webac65a.

During installation, a customized version of the WebAccess Agent startup file is created. This customized version has the `/home` startup switch automatically set to the domain directory where the WebAccess Agent’s queue directories are located. See [“Using WebAccess Agent Startup Switches”](#) in [“WebAccess”](#) in the *GroupWise 6.5 Administration Guide*.

x*10.nlm files

The x*10.nlm programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.

gwinter.xml file

The gwinter.xml file is a specialized WebAccess Agent configuration file for use in the XIS environment.

cache\template\xx directory

The cache\template\xx directory holds language-specific HTML template files for viewing documents in WebAccess and WebPublisher. The xx is a two-letter language code.



























You can change the location of the cache directory, as described in “[Modifying WebPublisher Settings](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

help directory

The help directory holds help files and images used by the WebAccess Agent Web console. See “[Monitoring the WebAccess Agent through the Web Console](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

Linux Installation Directory

 /opt/novell/groupwise	Novell GroupWise installation directory
 agents	Linux agent installation directory
 bin	Subdirectory for GroupWise agent executables
 gwinter	WebAccess Agent executable
 gwinter.xml	WebAccess Agent XIS configuration file
 webac65axx.waa	Boilerplate WebAccess startup file
 lib	Subdirectory for GroupWise agent library files
 *.so	Supporting shared library files
 image	Subdirectory for icon images
 share	Subdirectory for agent shared files
 webac65a.waa	Domain-specific WebAccess Agent startup file
 webcon	Subdirectory for agent Web console files
 help	Subdirectory for agent Web console help files
 webaccess	Subdirectory for WebAccess Application files

 <code>webacc.cfg</code>	WebAccess Application configuration file
 <code>commgr.cfg</code>	Communications Manager configuration file
 <code>spellchk.cfg</code>	Novell Speller Application configuration file
 <code>ldap.cfg</code>	LDAP service provider configuration file
 <code>default</code>	Subdirectory for boilerplate configuration files
 <code>logs</code>	Subdirectory for WebAccess Application log files
 <code>users</code>	Subdirectory for users' WebAccess session information
 <code>webpublisher</code>	Subdirectory for WebAccess Application files
 <code>webpub.cfg</code>	WebPublisher Application configuration file
 <code>commgr.cfg</code>	Communications Manager configuration file
 <code>default</code>	Subdirectory for boilerplate configuration files
 <code>logs</code>	Subdirectory for WebPublisher Application log files
 <code>cache/template\xx</code>	Subdirectory for HTML template files
 <code>webacc.tpt</code>	WebAccess template files
 <code>publish.tpt</code>	WebPublisher template files
 <code>/etc/init.d</code>	Standard Linux location for application startup scripts
 <code>grpwise-wa</code>	Startup script for the WebAccess Agent
 <code>rc3.d</code>	Standard Linux location for run-level-3 symbolic links
 <code>S99grpwise-wa</code>	Symbolic link to the startup script for the WebAccess Agent
 <code>rc5.d</code>	Standard Linux location for run-level-5 symbolic links
 <code>S99grpwise-wa</code>	Symbolic link to the startup script for the WebAccess Agent
 <code>/var/log/</code>	Standard Linux location for application log files
 <code>novell/groupwise</code>	Subdirectory for GroupWise agent log files
 <code>domain.webac65a</code>	Domain-specific subdirectory for WebAccess Agent log files
 <code>000.prc</code>	Processing directory for log files
 <code>mmdgwia.nnn</code>	WebAccess Agent log files

Additional WebAccess files are integrated into your Web server to support the WebAccess client and the WebAccess Agent Web console. See [“Apache/Tomcat Installation” on page 127](#). See also [“WebAccess Agent Queue Directory” on page 86](#).

agents directory

On a Linux server, the WebAccess Agent is always installed in subdirectories of `/opt/novell/groupwise/agents`.

bin directory

The bin directory holds GroupWise executable files.

gwinter file

The gwinter file is the WebAccess Agent executable. You run this executable file to start the WebAccess Agent. See “[Setting Up GroupWise WebAccess on Linux](#)” in “[Installing GroupWise WebAccess](#)” in the *GroupWise 6.5 Installation Guide*.

gwinter.xml file

The gwinter.xml file is a specialized configuration file for use in the XIS environment.

webac65axx.waa file

The *webac6axx.waa* file is the boilerplate file from which a domain-specific *webac65a.waa* file is created in the share directory. The *webac65a* represents the name of the WebAccess Agent object in eDirectory. The *xx* represents a two-letter language code.

lib directory

The lib directory holds GroupWise shared library files.

*.so files

The *.so files are Linux shared library files that provide information to the WebAccess Agent executable.

image directory

The image directory has subdirectories for agent icons for both color and monochrome displays.

share directory

The share directory holds agent startup files and files that are used by the agent consoles and Web consoles.

webac65a.waa

The WebAccess Agent startup file contains startup switches for the WebAccess Agent. Switch settings placed in the WebAccess Agent startup file override comparable options set for the WebAccess Agent in ConsoleOne. The startup file is named the same as the WebAccess Agent object in ConsoleOne. The default is webac65a.

During installation, a customized version of the WebAccess Agent startup file is created. This customized version has the */home* startup switch automatically set to the domain directory where the WebAccess Agent’s queue directories are located. See “[Using WebAccess Agent Startup Switches](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

webcon directory

The webcon directory holds subdirectories and files used by the agent Web consoles, such as help files.

webaccess directory

The webaccess directory holds files used by the WebAccess Application.

webacc.cfg file

The webacc.cfg file holds the same configuration information that is stored on the WebAccess Application object (named GroupWiseWebAccess) in eDirectory. For more information, see “[Configuring the WebAccess Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

commgr.cfg file

The commgr.cfg file holds the same configuration information that is stored on the GroupWise Provider object (named GroupWiseProvider) in eDirectory, including the IP address and encryption key for the WebAccess Agent. This information enables the WebAccess Application to communicate with the WebAccess Agent. For more information, see “[Configuring the GroupWise Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

A copy of the commgr.cfg file is automatically maintained in the WebAccess Agent queue directory (*domain/wpgate/webac65a*).

spellchk.cfg file

The spellchk.cfg file holds the same configuration information that is stored on the Novell Speller Application object (named NovellSpeller) in eDirectory. For more information, see “[Configuring the Novell Speller Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

ldap.cfg file

The ldap.cfg file holds the same configuration information that is stored on the LDAP Provider object (named LDAPProvider) in eDirectory. For more information, see “[Configuring the LDAP Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

default directory

The default directory holds boilerplate versions of the various .cfg files.

logs directory

The logs subdirectory holds WebAccess Application log files. For more information, see “[Controlling WebAccess Application Logging](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

users directory

The users subdirectory holds session files for WebAccess users. If WebAccess times out after a period of user inactivity, the user’s session information is saved. When the user logs back in, the session information is retrieved so that the user can continue working without loss of data. Also, users’ message text is saved during each session, so that if the WebAccess Application is restarted or goes down, users do not lose the message text they were composing at the time. Session files are deleted when users log back in.

webpublisher directory

The webpublisher directory holds files used by the WebPublisher Application.

webpub.cfg

The webpub.cfg file holds the same configuration information that is stored on the WebPublisher Application object (named GroupWiseWebPublisher) in eDirectory. For more information, see “[Configuring the WebPublisher Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

commgr.cfg file

The commgr.cfg file holds the same configuration information that is stored on the GroupWise Provider object (named GroupWiseProvider) in eDirectory, including the IP address, port number, number of threads, and encryption key for the WebAccess Agent. This information enables the WebAccess Application to communicate with the WebAccess Agent. For more information, see “[Configuring the GroupWise Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

A copy of the commgr.cfg file is automatically maintained in the WebAccess Agent queue directory (*domain/wpgate/webac65a*).

default directory

The default directory holds boilerplate versions of the webpub.cfg and commgr.cfg files.

logs directory

The logs subdirectory holds WebPublisher Application log files. For more information, see “[Controlling WebPublisher Application Logging](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

cache\template\xx directory

The cache\template\xx directory holds language-specific HTML template files for viewing documents in WebAccess and WebPublisher. The xx is a two-letter language code.

You can change the location of the cache directory, as described in “[Modifying WebPublisher Settings](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

/etc/init.d directory

The /etc/init.d directory is the standard location for Linux startup scripts.

grpwise-wa file

The grpwise-wa script is created automatically during installation. You can use the script to start, restart, stop, and display status information about the WebAccess Agent. For more information about starting the WebAccess Agent, see “[Setting Up GroupWise WebAccess on Linux](#)” in “[Installing GroupWise WebAccess](#)” in the *GroupWise 6.5 Installation Guide*.

rc3.d directory

The rc3.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 3 (multi-user; boots to a text mode login prompt without The X Window System). The symbolic link to the grpwise-wa script is S99grpwise-wa. It is created if you choose during installation to have the WebAccess Agent start automatically when the server boots.

rc5.d directory

The rc5.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 5 (multi-user; boots to The X Window System login dialog box). The symbolic link to the grpwise-wa script is S99grpwise-wa. It is created if you choose during installation to have the WebAccess Agent start automatically when the server boots.

/var/log/novell/groupwise directory

The /var/log directory is the standard location for log files on Linux. All GroupWise agent log files are created in the novell/groupwise subdirectory.

domain.webac65a directory















The *domain.webac65a* directory is a domain-specific location for WebAccess Agent log files.

000.prc directory

Within the 000.prc directory, the WebAccess Agent creates log files (*mmddweb.nnn*) to inform you of its processing and any problems it encounters. For more information about log files, see “Controlling WebAccess Agent Logging” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

The first two digits of the filename represent the month; the next two digits represent the day of the month. The three-digit extension is a sequence number for multiple log files created on the same day. For example, 0518web.002 is the second WebAccess Agent log file created on May 18.

Windows Installation Directory

 c:\webacc	Windows WebAccess Agent installation directory
 gwenv1a.dll	GroupWise Agent Engine
 gwinter.exe	WebAccess Agent program
 strtweb.bat	WebAccess Agent batch file
 webac65a.waa	WebAccess Agent startup file
 gwwanxx.dll	WebAccess Agent language information file
 gwwasnmp.dll	Customized DLL program for SNMP
 gwww1.dll	Customized DLL program for MIME
 x*10.dll	XIS DLL programs
 gwinter.xml	XIS agent configuration file
 cache\template\xx	Subdirectory for HTML template files
 webacc.tpt	WebAccess template files
 publish.tpt	WebPublisher template files
 help	Subdirectory for GroupWise agent console help files

Additional WebAccess files are integrated into your Web server to support the WebAccess client and the WebAccess Agent Web console. See [“Apache/Tomcat Installation” on page 127](#). See also [“WebAccess Agent Queue Directory” on page 86](#).

c:\webacc

On a Windows server, the WebAccess Agent can be installed in any directory you choose. The default location is c:\webacc. The WebAccess Agent desktop icon is set up to include the full path to whatever directory you choose.

gwenv1a.dll file

The gwenv1a.dll file is the GroupWise Agent Engine, a program that is shared by all GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations
- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent started on a server automatically starts the GroupWise Agent Engine DLL.

gwinter.exe file

The gwinter.exe file is the WebAccess Agent program.

strtweb.bat file

The strtweb.bat file is used to start the WebAccess Agent. The batch file specifies the WebAccess startup file. A typical strtweb.bat file includes the following:

```
title Novell GroupWise WebAccess
@echo off
cls
gwinter.exe @webac65a.waa
```

You run this batch file to start the WebAccess Agent. See [“Starting GroupWise WebAccess”](#) in [“Installing GroupWise WebAccess”](#) in the *GroupWise 6.5 Installation Guide*.

webac65a.waa file

The WebAccess Agent startup file contains startup switches for the WebAccess Agent. Switch settings placed in the WebAccess Agent startup file override comparable configuration options set for the WebAccess Agent in ConsoleOne. The startup file is named the same as the WebAccess Agent object in eDirectory. The default is webac65a.

During installation, the WebAccess Agent startup file is created with the `/home` startup switch automatically set to the domain directory where the WebAccess Agent queue directory (`domain\wpgate\webac65a`) is located. See [“Using WebAccess Agent Startup Switches”](#) in [“WebAccess”](#) in the *GroupWise 6.5 Administration Guide*.

gwwanxx.dll files

The gwwanxx.dll files contain all language-specific information for the WebAccess Agent. The digit *n* is a version number. The last two characters *xx* are a language code.

gwwasmp.dll file

The gwwasmp.dll file provides interaction with the Windows SNMP Service, so that you can monitor the WebAccess Agent using an SNMP monitoring program. This file is not installed unless you enable SNMP during installation. It cannot be installed if the SNMP Service is not installed on the Windows server.

gwww1.dll file

The gwww1.dll file provides parsing of MIME messages received from the Internet.

x*10.dll files

The x*10.dll programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.

gwinter.xml file

The gwia.xml file is a specialized agent configuration file for use in the XIS environment.

cache\template\xx directory

The cache\template\xx directory holds language-specific HTML template files for viewing documents in WebAccess and WebPublisher. The *xx* is a two-letter language code.

You can change the location of the cache directory, as described in “[Modifying WebPublisher Settings](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.



















help directory

The help directory holds help files and images used by the WebAccess Agent Web console. See “[Monitoring the WebAccess Agent through the Web Console](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

Monitor Agent Installation

- ◆ “Linux Installation Directory” on page 123
- ◆ “Windows Installation Directory” on page 125

Linux Installation Directory

 /opt/novell/groupwise	GroupWise installation directory
 agents	Linux agent installation directory
 bin	Subdirectory for GroupWise agent executables
 gwmon	Monitor Agent executable
 lib	Subdirectory for GroupWise agent library files
 *.so	Supporting shared library files
 share	Subdirectory for agent shared files
 monitor.xml	Monitor Agent configuration file
 gwmonitor	Subdirectory for Monitor Application files
 gwmonitor.cfg	Monitor Application configuration file
 default	Subdirectory for boilerplate configuration files
 logs	Subdirectory for Monitor Application log files
 users	Directory for Monitor users' session files
 /etc/init.d	Standard Linux location for application startup scripts
 grpwise-ma	Startup script for the Monitor Agent
 rc3.d	Standard Linux location for run-level-3 symbolic links
 S99grpwise-ma	Symbolic link to the startup script for the Monitor Agent
 rc5.d	Standard Linux location for run-level-5 symbolic links
 S99grpwise-ma	Symbolic link to the startup script for the Monitor Agent
 /var/log/	Standard Linux location for application log files
 novell/groupwise	Subdirectory for GroupWise agent log files
 gwmon	Subdirectory for Monitor Agent log files
 mmdmon.nnn	Monitor Agent event log files
 mmdhist.nnn	Monitor Agent history log files

agents directory

On a Linux server, the Monitor Agent is always installed in subdirectories of `/opt/novell/groupwise/agents`.

bin directory

The bin directory holds GroupWise executable files.

gwmon file

The gwmon file is the Monitor Agent executable. You run this executable file to start the Monitor Agent. See “Starting the Monitor Agent” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

lib directory

The lib directory holds GroupWise shared library files.

*.so files

These files are Linux shared library files that provide information to the Monitor Agent executable.

share directory

The share directory holds agent startup files and files that are used by the agent consoles and Web consoles.

monitor.xml file

The monitor.xml file is a specialized configuration file for use in the XIS environment.

gwmonitor directory

The gwmonitor directory holds files used by the Monitor Application.

gwmonitor.cfg file

The gwmonitor.cfg file holds the same configuration information that is stored on the Monitor Application object (named GroupWiseMonitor) in eDirectory. For more information, see “Configuring the Monitor Application” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

default directory

The default directory holds boilerplate versions of the gwmonitor.cfg file and the monitor.xml file.

logs directory

The logs subdirectory holds Monitor Application log files. For more information, see “Modifying Monitor Application Log Settings” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

users directory

The users subdirectory holds session files for Monitor users.

/etc/init.d directory

The /etc/init.d directory is the standard location for Linux startup scripts.

grpwise-ma file

The grpwise-ma script is created automatically during installation. You can use the script to start, restart, stop, and display status information about the Monitor Agent. For more information about starting the Monitor Agent, see “Starting the Monitor Agent” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

rc3.d directory

The rc3.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 3 (multi-user; boots to a text mode login prompt without The X Window System). The symbolic link to the grpwise-ma script is S99grpwise-ma. It is created if you choose during installation to have the Monitor Agent start automatically when the server boots.

rc5.d directory

The rc5.d directory holds symbolic links to scripts that you want your Linux server to run when it is booted to run level 5 (multi-user; boots to The X Window System login dialog box). The symbolic link to the grpwise-ma script is S99grpwise-ma. It is created if you choose during installation to have the Monitor Agent start automatically when the server boots.

/var/log/novell/groupwise directory







The /var/log directory is the standard location for log files on Linux. All GroupWise agent log files are created in the novell/groupwise subdirectory.




gwmon directory

Within the gwmon directory, the Monitor Agent creates two different types of log files. The *mddmon.nnn* log files inform you of its processing and any problems it encounters. The *mddhist.nnn* log files record dumps of all MIB values gathered during each poll cycle. For more information about log files, see “Configuring Monitor Agent Log Settings” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

The first two digits of the filename represent the month; the next two digits represent the day of the month. The three-digit extension is a sequence number for multiple log files created on the same day. For example, 0518mon.002 is the second Monitor Agent log file created on May 18.

Windows Installation Directory

 c:\gwmon	Windows Monitor Agent installation directory
 gwenv1a.dll	GroupWise Agent Engine
 gwmon.exe	Monitor Agent program
 monitor.xml	Monitor Agent configuration file
 gwmonxx.dll	Monitor Agent language information file
 gwsnmp.dll	Customized DLL program for SNMP

 x*10.dll	XIS DLL programs
 gwmon.xml	XIS agent configuration file
 monwork	Monitor Agent working directory

Additional Monitor files are integrated into your Web server to support the Monitor Web console. For an example, see [“Apache/Tomcat Installation” on page 127](#).

c:\gwmon

On a Windows server, the Monitor Agent can be installed in any directory you choose. The default location is c:\gwmon. The Monitor Agent desktop icon is set up to include the full path to whatever directory you choose.

gwenv1a.dll file

The gwenv1a.dll file is the GroupWise Agent Engine, a program that is shared by all GroupWise agents. It provides the following services to the agents:

- ◆ Database management
- ◆ File operations
- ◆ Message handling
- ◆ Thread management
- ◆ Semaphores (file/record locking)
- ◆ Date/time services

The first agent started on a server automatically starts the GroupWise Agent Engine DLL.

gwmon.exe file

The gwmon.exe file is the Monitor Agent program. You run this executable file to start the Monitor Agent. See [“Starting the Monitor Agent”](#) in [“Monitor”](#) in the *GroupWise 6.5 Administration Guide*.

monitor.xml

The monitor.xml file stores the Monitor Agent configuration information that you establish in the Monitor Agent console, as described in [“Configuring the Monitor Agent”](#) in [“Monitor”](#) in the *GroupWise 6.5 Administration Guide*.

gwmonxx.dll files

The gwmonxx.dll files contain all language-specific information for the WebAccess Agent. The last two characters xx are a language code.

gwsnmp.dll file

The gwsnmp.dll file provides interaction with the Windows SNMP Service, so that the Monitor Agent can throw SNMP traps. See [“Configuring SNMP Trap Notification of Agent Problems”](#) in [“Monitor”](#) in the *GroupWise 6.5 Administration Guide*.

x*10.dll files

The x*10.dll programs provide XIS capability for the GroupWise agents, so that you can create XML documents to monitor and change the agents as needed.

gwmon.xml file

The gwmon.xml file is a specialized agent configuration file for use in the XIS environment.

monwork directory
















The monwork directory is used for temporary files used in calculating system performance, as described in “[Measuring Agent Performance](#)” in “[Monitor](#)” in the *GroupWise 6.5 Administration Guide*




























Apache/Tomcat Installation

























WebAccess, WebPublisher, and Monitor rely on the presence of a Web server in order to fulfil their functions. A number of Web servers are supported. See “[GroupWise System Requirements](#)” in “[Installing a Basic GroupWise System](#)” in the *GroupWise 6.5 Installation Guide*. In this section, the Apache Web server and the Tomcat Servlet Engine used to illustrate how GroupWise integrates with your Web server.

- ◆ “[NetWare Installation Directory](#)” on page 127
- ◆ “[Linux Installation Directory](#)” on page 133

NetWare Installation Directory

 sys:\apache	Apache installation directory
 bin	Standard Apache directory
 cgi-bin	Standard Apache directory
 conf	Standard Apache directory
 httpd.conf	Main Apache configuration file
 GWApache2.conf	Specialized Apache configuration file for use with GroupWise
 error	Standard Apache directory
 htdocs	Apache document root directory
 index.html	Web server default Web page
 com/novell	Directory for Novell product files
 collexion	Directory for Novell class files and properties files
 GWMonitor	Directory for GroupWise Monitor files
 help	Subdirectory for Monitor Web console help files
 images	Subdirectory for Monitor Web console image files
 palm	Subdirectory for Monitor Palm* Query Application (PQA) files

 images	Directory for Novell product image files
 inso	Directory for Inso* viewer files for viewing attachments
 webaccess	Directory for GroupWise WebAccess files
 applets	Subdirectory for WebAccess client properties files
 help	Subdirectory for WebAccess client help files
 images	Subdirectory for WebAccess client image files
 palm	Subdirectory for WebAccess PQA files
 webconsole	Subdirectory for WebAccess Application Web console help files
 webpublisher	Directory for GroupWise WebPublisher files
 help	Subdirectory for WebPublisher help files
 images	Subdirectory for WebPublisher image files
 icons	Standard Apache directory
 logs	Standard Apache directory
 man	Standard Apache directory
 manual	Standard Apache directory
 modules	Standard Apache directory
 sys:\tomcat\4	Tomcat installation directory
 bin	Standard Tomcat directory
 common	Standard Tomcat directory
 conf	Standard Tomcat directory
 examples	Standard Tomcat directory
 logs	Standard Tomcat directory
 server	Standard Tomcat directory
 shared	Standard Tomcat directory
 temp	Standard Tomcat directory
 webapps/ROOT	Standard Tomcat directory
 WEB-INF	Web server information directory
 web.xml	Web server configuration file
 classes/com/novell	Directory for Novell product files
 collexion	Directory for language-specific morphology files
 GWMonitor	Directory for GroupWise Monitor files
 providers	Subdirectory for service provider files
 templates	Subdirectory for template files

 webaccess	Directory for GroupWise WebAccess files
 providers	Subdirectory for service provider files
 templates	Subdirectory for template files
 webpublisher	Directory for GroupWise WebPublisher files
 providers	Subdirectory for service provider files
 templates	Subdirectory for template files
 work	Standard Tomcat directory
 sys:\novell	Directory for Novell product configuration files
 gwmonitor	Subdirectory for GroupWise Monitor
 gwmonitor.cfg	Monitor Application configuration file
 logs	Directory for Monitor Application log files
 webaccess	Subdirectory for WebAccess
 webacc.cfg	WebAccess Application configuration file
 commgr.cfg	Communications Manager configuration file
 spellchk.cfg	Novell Speller Application configuration file
 ldap.cfg	LDAP service provider configuration file
 logs	Directory for WebAccess Application log files
 temp	Temporary working directory
 users	Directory for WebAccess users' session files
 webpublisher	Subdirectory for WebAccess Application files
 webpub.cfg	WebPublisher Application configuration file
 commgr.cfg	Communications Manager configuration file
 default	Subdirectory for boilerplate configuration files
 logs	Subdirectory for WebPublisher Application log files

conf/httpd.conf file

The httpd.conf file is the main Apache configuration file. On NetWare 6.5, the GroupWise Installation program modifies it to include a reference to the GWApache2.conf file.

conf/GWApache2.conf file

The GWApache2.conf file includes the jkmount command that enables Apache and Tomcat to communicate with each other.

htdocs/index.html file

The index.html file is the default HTML file that displays in your Web browser when you hit the IP address or hostname of the server. When you install WebAccess or Monitor, you have the option of replacing the default Web page with a Novell-specific Web page. See [“Web Server Default](#)

Page” in “Installing GroupWise WebAccess” and “Web Server Default Page” in “Installing GroupWise Monitor” in the *GroupWise 6.5 Installation Guide*.

GWMonitor/palm directory

The Monitor Installation program gives you the opportunity to create a Palm Query Application (PQA) file so that you can monitor the GroupWise agents from any Palm OS device. If you want to create a PQA file after initial installation, you can run setup.exe in the \admin\monitor directory of your software distribution directory or on the *GroupWise 6.5 Administrator* CD and use the /pqa startup option to create the PQA file without installing GroupWise Monitor. For more information, see “Web Clipping Application (PQA) URL” in “Installing GroupWise Monitor” in the *GroupWise 6.5 Installation Guide* and “Creating a PQA File for the Monitor Web Console” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

webaccess/palm directory

The WebAccess Installation program gives you the opportunity to create a Palm Query Application (PQA) file so that you can use the WebAccess client from any Palm OS device. If you want to create a PQA file after initial installation, you can run setup.exe in the \internet\webaccess directory of your software distribution directory or on the *GroupWise 6.5 Administrator* CD and use the /pqa startup option to create the PQA file without installing GroupWise WebAccess. For more information, see “Web Clipping Application (PQA) URL (NetWare and Windows WebAccess)” in “Installing GroupWise WebAccess” in the *GroupWise 6.5 Installation Guide* and “Creating a PQA File for the WebAccess Client” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

WEB-INF/web.xml file

The web.xml file tells the Web server where the various GroupWise service providers and their configuration files are located. By default, they are located in subdirectories under sys:\novell.

GWMonitor/providers directory

The providers directory holds the Java* class and properties files for the Monitor provider. For more information, see “Adding or Removing Service Providers” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

GWMonitor/templates directory

The templates directory holds subdirectories for the various file formats used in various browser environments. For more information, see “Modifying Monitor Application Template Settings” in “Monitor” in the *GroupWise 6.5 Administration Guide*.

webaccess/providers directory

The providers directory holds the Java class and properties files for the WebAccess providers. For more information, see “Adding or Removing Service Providers” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

webaccess/templates directory

The templates directory holds subdirectories for the various file formats used in various browser environments. For more information, see “Modifying WebAccess Application Template Settings” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

webpublisher/providers directory

The providers directory holds the Java class and properties files for the WebPublisher providers. For more information, see “[Adding or Removing Service Providers](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

webpublisher/templates directory

The templates directory holds subdirectories for the various file formats used in various browser environments. For more information, see “[Modifying WebPublisher Application Template Settings](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

sys:\novell directory

The sys:\novell directory holds subdirectories for each Novell product that uses application configuration files.

gwmonitor directory

The gwmonitor directory is created when you install GroupWise Monitor.

gwmonitor.cfg file

The gwmonitor.cfg file holds the same configuration information that is stored on the Monitor Application object (named GroupWiseMonitor) in eDirectory. For more information, see “[Configuring the Monitor Application](#)” in “[Monitor](#)” in the *GroupWise 6.5 Administration Guide*.

logs directory

The logs subdirectory holds Monitor Application log files. For more information, see “[Modifying Monitor Application Log Settings](#)” in “[Monitor](#)” in the *GroupWise 6.5 Administration Guide*.

webaccess directory

The webaccess directory holds files used by the WebAccess Application.

webacc.cfg file

The webacc.cfg file holds the same configuration information that is stored on the WebAccess Application object (named GroupWiseWebAccess) in eDirectory. For more information, see “[Configuring the WebAccess Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

commgr.cfg file

The commgr.cfg file holds the same configuration information that is stored on the GroupWise Provider object (named GroupWiseProvider) in eDirectory, including the IP address, port number, number of threads, and encryption key for the WebAccess Agent. This information enables the WebAccess Application to communicate with the WebAccess Agent. For more information, see “[Configuring the GroupWise Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

A copy of the commgr.cfg file is automatically maintained in the WebAccess Agent queue directory (*domain\wpgate\webac65a*).

spellchk.cfg file

The spellchk.cfg file holds the same configuration information that is stored on the Novell Speller Application object (named NovellSpeller) in eDirectory. For more information, see “[Configuring the Novell Speller Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

ldap.cfg file

The ldap.cfg file holds the same configuration information that is stored on the LDAP Provider object (named LDAPProvider) in eDirectory. For more information, see “[Configuring the LDAP Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

logs directory

The logs subdirectory holds WebAccess Application log files. For more information, see “[Controlling WebAccess Application Logging](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

users directory

The users subdirectory holds session files for WebAccess users. If WebAccess times out after a period of user inactivity, the user’s session information is saved. When the user logs back in, the session information is retrieved so that the user can continue working without loss of data. Also, users’ message text is saved during each session, so that if the WebAccess Application is restarted or goes down, users do not lose the message text they were composing at the time.

webpublisher directory

The webpublisher directory holds files used by the WebPublisher Application.

webpub.cfg

The webpub.cfg file holds the same configuration information that is stored on the WebPublisher Application object (named GroupWiseWebPublisher) in eDirectory. For more information, see “[Configuring the WebPublisher Application](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

commgr.cfg file

The commgr.cfg file holds the same configuration information that is stored on the GroupWise Provider object (named GroupWiseProvider) in eDirectory, including the IP address, port number, number of threads, and encryption key for the WebAccess Agent. This information enables the WebAccess Application to communicate with the WebAccess Agent. For more information, see “[Configuring the GroupWise Service Provider](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

A copy of the commgr.cfg file is automatically maintained in the WebAccess Agent queue directory (*domain\wpgate\webac65a*).


























default directory































The default directory holds boilerplate versions of the webpub.cfg and commgr.cfg files.





















logs directory

The logs subdirectory holds WebAccess Application log files. For more information, see “Controlling WebPublisher Application Logging” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

Linux Installation Directory

 /opt/novell/http	Novell Apache installation directory
 bin	Standard Apache directory
 build	Standard Apache directory
 doc	Standard Apache directory
 logs	Standard Apache directory
 modules	Standard Apache directory
 /etc/opt/novell/http	Novell Apache configuration directory
 conf	Standard Apache directory
 conf.d	Standard Apache directory
 gconf.d	Standard Apache directory
 init.d	Standard Apache directory
 logrotate.d	Standard Apache directory
 sslconf.d	Standard Apache directory
 /etc/opt/novell/tomcat4	Novell Tomcat configuration directory
 catalina.properties	Standard Tomcat file
 jk2.properties	Standard Tomcat file
 server.xml	Standard Tomcat file
 tomcat-users.xml	Standard Tomcat file
 tomcat4.conf	Standard Tomcat file
 web.xml	Web server configuration file
 init.d	Standard Tomcat directory
 logrotate.d	Standard Tomcat directory
 /var/opt/novell	Novell subdirectory under /var
 gw	Apache document root directory for WebAccess and WebPublisher
 index.html	Web server default Web page

 com/novell	Directory for Novell product files
 collexion	Directory for Novell class files and properties files
 images	Directory for Novell product image files
 inso	Directory for Inso viewer files for viewing attachments
 webaccess	Directory for GroupWise WebAccess files
 applets	Subdirectory for WebAccess client properties files
 help	Subdirectory for WebAccess client help files
 images	Subdirectory for WebAccess client image files
 webconsole	Subdirectory for WebAccess Application Web console help files
 webpublisher	Directory for GroupWise WebPublisher files
 help	Subdirectory for WebPublisher help files
 images	Subdirectory for WebPublisher image files
 WEB-INF	Web server information directory
 web.xml	Web server configuration file
 classes/com/novell	Web server configuration file
 collexion	Directory for language-specific morphology files
 webaccess	Directory for GroupWise WebAccess files
 providers	Subdirectory for service provider files
 templates	Subdirectory for template files
 webpublisher	Directory for GroupWise WebPublisher files
 providers	Subdirectory for service provider files
 templates	Subdirectory for template files
 gwmon	Apache document root directory for Monitor
 index.html	Web server default page
 com/novell	Directory for Novell product files
 gwmonitor	Directory for GroupWise WebPublisher files
 help	Subdirectory for WebPublisher help files
 images	Subdirectory for WebPublisher image files
 WEB-INF	Web server information directory
 web.xml	Web server configuration file
 gwmonitor	Directory for GroupWise Monitor files
 providers	Subdirectory for service provider files
 templates	Subdirectory for template files

 httpd	Standard Apache directory
 log	Standard Apache directory
 novlwww	Home directory for the novlwww user created at installation
 run	Standard Apache directory
 tomcat4	Standard Tomcat directory
 bin	Standard Tomcat directory
 common	Standard Tomcat directory
 conf	--> /etc/opt/novell/tomcat4
 logs	Standard Tomcat directory
 server	Standard Tomcat directory
 shared	Standard Tomcat directory
 temp	Standard Tomcat directory
 webapps	Standard Tomcat directory
 admin.xml	Standard Tomcat file
 manager.xml	Standard Tomcat file
 examples	Standard Tomcat directory
 gw	--> /var/opt/novell/gw
 gwmon	--> /var/opt/novell/gwmon
 tomcat-docs	Standard Tomcat directory
 webdav	Standard Tomcat directory

gw/index.html file

The index.html file in the gw directory is the initial page for accessing the WebAccess client. See “Starting GroupWise WebAccess” in the *GroupWise 6.5 WebAccess Client User Guide*.

gw/WEB-INF/web.xml file

The web.xml file in the gw/WEB-INF directory tells the Web server where the various WebAccess service providers and their configuration files are located. By default, they are located in [/opt/novell/groupwise/webaccess](#).

webaccess/providers directory

The providers directory holds the Java class and properties files for the WebAccess providers. For more information, see “Adding or Removing Service Providers” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

webaccess/templates directory

The templates directory holds subdirectories for the various file formats used in various browser environments. For more information, see “Modifying WebAccess Application Template Settings” in “WebAccess” in the *GroupWise 6.5 Administration Guide*.

webpublisher/providers directory

The providers directory holds the Java class and properties files for the WebPublisher providers. For more information, see “[Adding or Removing Service Providers](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

webpublisher/templates directory

The templates directory holds subdirectories for the various file formats used in various browser environments. For more information, see “[Modifying WebPublisher Application Template Settings](#)” in “[WebAccess](#)” in the *GroupWise 6.5 Administration Guide*.

gwmon/index.html file

The index.html file in the gwmon directory is the login Web page for GroupWise Monitor if you have provided a password as described in “[Configuring Authentication and Intruder Lockout for the Monitor Web Console](#)” in “[Monitor](#)” in the *GroupWise 6.5 Administration Guide*. See also “[Using the Monitor Agent Console](#)”.

gwmon/WEB-INF/web.xml file

The web.xml file in the gwmon/WEB-INF directory tells the Web server where the Monitor service provider and its configuration file are located. By default, they are located in [/opt/novell/groupwise/gwmonitor](#).

gwmonitor/providers directory

The providers directory holds the Java class and properties files for the Monitor provider. For more information, see “[Adding or Removing Service Providers](#)” in “[Monitor](#)” in the *GroupWise 6.5 Administration Guide*.

gwmonitor/templates directory












The templates directory holds subdirectories for the various file formats used in various browser environments. For more information, see “[Modifying Monitor Application Template Settings](#)” in “[Monitor](#)” in the *GroupWise 6.5 Administration Guide*.
















9

Software Distribution Directory

- ♦ “NetWare/Windows Software Distribution Directory” on page 137
- ♦ “Linux Software Distribution Directory” on page 143

NetWare/Windows Software Distribution Directory

 software_distribution directory	Master copy of GroupWise® software
 agents	GroupWise agent software
 install.exe	GroupWise Agent Installation program
 nlm	NetWare® agent software
 nt	Windows agent software
 startups	Agent startup files
 snmp	GroupWise MIB files
 help	Agent Web console help files
 domain	Domain data dictionary files
 wdomain.dc	Data dictionary for GroupWise 4.x domain databases
 gwdom.dc	Data dictionary for GroupWise 6.x and 5.x domain databases
 wphost.dc	Data dictionary for GroupWise 4.x post office databases
 gwpo.dc	Data dictionary for GroupWise 6.x and 5.x post office databases
 po	Post office data dictionary files
 ngwguard.dc	Data dictionary for message store databases
 wphost.dc	Data dictionary for GroupWise 4.x post office databases
 gwpo.dc	Data dictionary for GroupWise 6.x and 5.x post office databases

 client	GroupWise client software
 setup.cfg	GroupWise client installation configuration file
 win32	GroupWise client for Windows
 setup.exe	GroupWise client installation program
 addons	GroupWise client add-on programs
 gwmailto	Program to make GroupWise your browser e-mail program
 gwtip	Program to display tips each time GroupWise starts
 ofviews	GroupWise client view files
 zen	ZENworks supporting files
 admin	GroupWise administrator software
 c1admin	GroupWise snap-ins to ConsoleOne®
 monitor	GroupWise Monitor software
 utility	GroupWise administrative utilities
 gwcheck	GroupWise Check utility
 gwcsrgen	GroupWise Generate CSR utility
 setupip	GroupWise TCP/IP Setup utility
 internet	Internet connectivity software
 gwia	GroupWise Internet Agent software
 webaccess	GroupWise WebAccess software
 license	GroupWise Software License Agreement in multiple languages
 common	Java files used by GroupWise programs
 docs	GroupWise <i>Installation Guide</i>

\grpwise\software directory

The GroupWise software distribution directory resides initially on the GroupWise distribution CD. GroupWise Administration is installed directly from the CD. In addition, during installation, you create a software distribution directory on your network from which you subsequently install the GroupWise agents and GroupWise client software.

The default software distribution directory is `\grpwise\software`.

agents directory

The agents subdirectory contains all files associated with GroupWise agents:

- ◆ Post Office Agent (POA)
- ◆ Message Transfer Agent (MTA)
- ◆ GroupWise Target Service Agent (GWTSA)

install.exe file

The install.exe file in the agent subdirectory is the installation program you run to install the GroupWise agents on the servers where you will run the POA and/or MTA. For instructions, see “Installing and Starting the POA” in “Post Office Agent” and “Installing and Starting the MTA” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

nlm directory

The nlm subdirectory in the agent subdirectory contains the GroupWise agent NLM™ files installed on NetWare servers. See “NetWare Installation Directory” on page 97.

nt directory

The nt subdirectory contains the GroupWise agent files installed on Windows servers. See “Windows Installation Directory” on page 105.

startups directory

The startups subdirectory contains the default startup files for the GroupWise agents. During installation, a customized startup file is created for each agent that includes the location of the domain or post office serviced by that agent. The customized startup files are named after the domain or post office for which they are created. See “NetWare Installation Directory” on page 97 or “Windows Installation Directory” on page 105.

snmp directory

The snmp subdirectory contains GroupWise MIB files. For more information, see “Using SNMP Monitoring Programs” in “Post Office Agent” and “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

help directory

The help directory holds the help files that you can view from the agent Web consoles. See “Using the POA Web Console” in “Post Office Agent” and “Using the MTA Web Console” in “Message Transfer Agent” in the *GroupWise 6.5 Administration Guide*.

domain directory

The domain subdirectory contains the files from which domains are created.

wpdomain.dc file

The wpdomain.dc file is the distribution copy of the data dictionary for rebuilding GroupWise 4.x domain databases (**wpdomain.db** files) in secondary domains.

If the wpdomain.dc file is missing from the primary domain, you cannot rebuild GroupWise 4.x secondary domains. The original wpdomain.dc file is located in the domain directory of the GroupWise distribution media.

Historical Note: WordPerfect Office (WP Office), the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wpdomain.dc. Some naming conventions were originally preserved for backward compatibility.

gwdom.dc file

The gwdom.dc file is the distribution copy of the data dictionary for creating and rebuilding GroupWise 6.x and GroupWise 5.x domain databases (**wpdomain.db** files) in secondary domains.

If the gwdom.dc file is missing from the primary domain, you cannot create or rebuild GroupWise 6.x/5.x secondary domains. The original gwdom.dc file is located in the domain subdirectory of the GroupWise distribution media.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wpdomain.db. Some naming conventions were originally preserved for backward compatibility.

wphost.dc file

The wphost.dc file is the distribution copy of the data dictionary for rebuilding GroupWise 4.x post office databases (**wphost.db** files).

If the wphost.dc file is missing from a domain, you cannot rebuild GroupWise 4.x post offices in that domain. The original wphost.dc file is located in the domain subdirectory of the GroupWise distribution media. There is also a copy in the **po** subdirectory.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Post offices were originally called hosts. Hence, the name wphost.dc. Some naming conventions were originally preserved for backward compatibility.

gwpo.dc file

The gwpo.dc file is the distribution copy of the data dictionary for creating and rebuilding GroupWise 6.x and GroupWise 5.x post office databases (**wphost.db** files).

If the gwpo.dc file is missing from a domain, you cannot create or rebuild GroupWise 6.x/5.x post offices in that domain. The original gwpo.dc file is located in the domain directory of the GroupWise CD. There is also a copy in the **po** directory.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Post offices were originally called hosts. Hence, the name wphost.db. Some naming conventions were originally preserved for backward compatibility.

po directory

The po subdirectory contains the files from which post offices are created.

ngwguard.dc file

The ngwguard.dc file is the distribution copy of the data dictionary for building the following databases in the post office:

- ◆ **ngwguard.db** (guardian database)
- ◆ **dmxxxnn01-FF** (document management databases)
- ◆ **msgnn.db** (message databases)
- ◆ **userxxx.db** (user databases)
- ◆ **puxxxx.db** (databases for replicated items like shared folders)

If the `ngwguard.dc` file is missing from a post office, new databases cannot be created in the post office, so the post office cannot grow. The original `ngwguard.dc` file is located in the `po` directory of the GroupWise CD.

GroupWise Remote also uses the `ngwguard.dc` file as the data dictionary for its local databases.

client directory

The client subdirectory contains all files associated with GroupWise Windows client. See “Client” in the *GroupWise 6.5 Administration Guide*.

setup.cfg file

The `setup.cfg` file enables you to automate installation of the GroupWise Windows client so that your users do not need to respond to the Setup program’s prompts. For more information, see “Using a Configuration File to Enable AutoUpdate and to Select Installation Responses” in “Client” in the *GroupWise 6.5 Administration Guide*.

win32 directory

The `client\win32` subdirectory contains all GroupWise client files installed for use with Windows. See “Windows Client” on page 153.

setup.exe file

The `client\win32\setup.exe` file is the program GroupWise client users run to install and set up the GroupWise client on their Windows workstations. See “Client” in the *GroupWise 6.5 Administration Guide*.

addons directory

The addons directory contains subdirectories for programs that can be used along with the GroupWise client to enhance aspects of its functionality.

gwmalto directory

The `gwmalto` directory contains a program that makes the GroupWise Windows client the default e-mail application in your Web browser. For more information, see “[GWMALTO]” in “Modifying the Configuration File” in “Client” in the *GroupWise 6.5 Administration Guide*.

gwtip directory

The `gwtip` directory contains a program that display a helpful tip about GroupWise each time the client starts. For more information, see “[GWTIP]” in “Modifying the Configuration File” in “Client” in the *GroupWise 6.5 Administration Guide*.

ofviews directory

The `client/ofviews` subdirectory contains platform-specific subdirectories of view files for use by the GroupWise client. In addition, the `gwviewxx.ini` and `ofviewxx.ini` files configure custom views on the menus where users select views. The `gwviewxx.ini` file configures GroupWise 6.x and GroupWise 5.5 views. The `ofviewxx.ini` file configures views from earlier versions of GroupWise.

zen directory

The zen directory contains files that let you use ZENworks Desktop Management to distribute the GroupWise Windows client to users' workstations. See [“Using ZENworks Desktop Management to Distribute the GroupWise Windows Client”](#) in [“Client”](#) in the *GroupWise 6.5 Administration Guide*.

admin directory

The admin subdirectory contains subdirectories for administrative tools that can be used with GroupWise.

c1admin directory

The admin\c1admin subdirectory contains the GroupWise snap-ins to ConsoleOne.

monitor directory

The admin\monitor subdirectory contains the GroupWise 6 Monitor program, an SNMP monitoring program for use with the GroupWise agents. See [“Monitor”](#) in the *GroupWise 6.5 Administration Guide*.

utility directory

The admin\utility subdirectory contains helpful GroupWise utilities.

gwcheck directory

The admin\utility\gwcheck subdirectory contains the GroupWise Check utility. See [“GroupWise Check”](#) in [“Standalone Database Maintenance Programs”](#) in the *GroupWise 6.5 Administration Guide*.

gwcsrgen directory

The admin\utility\gwcheck subdirectory contains the GroupWise Generate CSR utility. See [“Server Certificates and SSL Encryption”](#) in [“Security”](#) in the *GroupWise 6.5 Administration Guide*.

setupip directory

The admin\utility\setupip subdirectory contains supporting programs for the client AutoUpdate feature. See [“Enabling AutoUpdate”](#) in [“Client”](#) in the *GroupWise 6.5 Administration Guide*.

internet directory

The internet subdirectory contains subdirectories for GroupWise components that provide Internet connectivity.

gwia directory

The internet\gwia subdirectory contains the GroupWise Internet Agent software, used to connect GroupWise systems across the Internet and to allow GroupWise users to exchange e-mail with users of various Internet e-mail programs. For information about Internet Agent files and directories, see [“Internet Agent Queue Directory”](#) on page 80 and [“Internet Agent Installation”](#) on page 107.

For information about the Internet Agent, see “**Internet Agent**” in the *GroupWise 6.5 Administration Guide*.

webaccess directory

The internet\webaccess subdirectory contains the GroupWise WebAccess software, which allows users to access their GroupWise mailboxes from a Web browser. For information about WebAccess files and directories, see “**WebAccess Agent Queue Directory**” on page 86, “**WebAccess Agent Installation**” on page 113, and “**Apache/Tomcat Installation**” on page 127

For information about WebAccess, see “**WebAccess**” in the *GroupWise 6.5 Administration Guide*.

license directory

The license subdirectory contains the GroupWise Software License Agreement in multiple languages. Run license.exe > select a language.











common directory

The common subdirectory contains Java files used by various components of GroupWise.


























docs directory

The docs subdirectory contains language-specific subdirectories for the *Installation Guide* in PDF format. You can find the most up-to-date version of the *GroupWise 6.5 Installation Guide* on the [GroupWise 6.5 Documentation Web site \(http://www.novell.com/documentation/gw65/index.html\)](http://www.novell.com/documentation/gw65/index.html).

Linux Software Distribution Directory

 /opt/novell/groupwise/software	Master copy of GroupWise software
 install	GroupWise Installation script
 agents	GroupWise agent software
 linux	Subdirectory for Linux* software
 novell-groupwise-agents-ver.date.i386.rpm	GroupWise agent package
 startups	Subdirectory for startup files
 language	Language-specific files
 strtupxx.poa	POA startup file
 strtupxx.mta	MTA startup file
 S99grpwise	GroupWise agent startup script (POA and MTA)

 domain	Domain data dictionary files
 wpdomain.dc	Data dictionary for GroupWise 4.x domain databases
 gwdom.dc	Data dictionary for GroupWise 6.x and 5.x domain databases
 wphost.dc	Data dictionary for GroupWise 4.x post office databases
 gwpo.dc	Data dictionary for GroupWise 6.x and 5.x post office databases
 po	Post office data dictionary files
 ngwguard.dc	Data dictionary for message store databases
 wphost.dc	Data dictionary for GroupWise 4.x post office databases
 gwpo.dc	Data dictionary for GroupWise 6.x and 5.x post office databases
 client	GroupWise client software
 linux	Subdirectory for Linux software
 install	Cross-Platform client installation script
 novell-groupwise-gwclient-ver.date.i386.rpm	Cross-Platform client package
 ofviews	GroupWise client view files
 admin	GroupWise administrator software
 novell-groupwise-admin-ver.date.i386.rpm	GroupWise snap-ins to ConsoleOne package
 novell-groupwise-dbcopy-ver.date.i386.rpm	GroupWise Database Copy Utility package
 novell-groupwise-gwcheck-ver.date.i386.rpm	GroupWise Check Utility package
 NOVLc1Linuxjre-ver.i386.rpm	ConsoleOne JRE update package
 monitor	GroupWise Monitor software
 linux	Subdirectory for Linux software
 novell-groupwise-gwmon-ver.date.i386.rpm	Monitor Agent package
 novell-groupwise-monitor-ver.date.i386.rpm	Monitor Application package
 startup	Subdirectory for startup files
 S99grpwise-ma	Monitor startup script
 internet	Internet connectivity software

 gwia	GroupWise Internet Agent software
 linux	Subdirectory for Linux software
 novell-groupwise-gwia-ver.date.i386.rpm	GroupWise Internet Agent package
 root	Subdirectory for boilerplate supporting files
 gwia.cfg	Internet Agent startup file
 gwauth.cfg	Host authentication configuration file
 mimetype.cfg	MIME encoding configuration file
 gwac.dc	Database dictionary for the access control database
 preamble.txt	Message for non-MIME recipients
 preamble.all	Preamble message in various languages
 statusxx.xml	File for customizing status messages
 startup	Subdirectory for startup files
 S99grpwise-ia	Internet Agent startup script
 webaccess	GroupWise WebAccess software
 linux	Subdirectory for Linux software
 novell-groupwise-gwinter-ver.date.i386.rpm	GroupWise WebAccess Agent package
 novell-groupwise-webaccess-ver.date.i386.rpm	GroupWise WebAccess Application package
 root	Subdirectory for boilerplate supporting files
 mimetype.cfg	MIME encoding configuration file
 gwac.dc	Database dictionary for the access control database
 gwgs.ini	Gateway identifier file
 startup	Subdirectory for startup files
 S99grpwise-wa	WebAccess Agent startup script
 webserver	Web server to support WebAccess and Monitor
 linux	Subdirectory for Linux software
 *.rpm	Apache, Tomcat, and related packages
 license	GroupWise Software License Agreement in multiple languages
 docs	GroupWise Readme, <i>Quick Start</i> , and <i>Installation Guide</i>
 gwinst	Directories and files used by the Installation program

/opt/novell/groupwise/software directory

The GroupWise software distribution directory resides initially on the GroupWise distribution CD. GroupWise Administration is installed directly from the CD. In addition, during installation, you can create a software distribution directory on your network from which you subsequently install the GroupWise agents and GroupWise client software.

The default software distribution directory is `/opt/novell/groupwise/software`.

install file

The install file is the script you use to install all components of GroupWise on Linux. See “[Setting Up a Basic GroupWise System on Linux](#)” in the *GroupWise 6.5 Installation Guide*.

agents directory

The agents subdirectory contains all files associated with GroupWise agents:

- ◆ Post Office Agent (POA)
- ◆ Message Transfer Agent (MTA)
- ◆ GroupWise Target Service Agent (GWTSA)

novell-groupwise-agents.version.date.i386.rpm

The novell-groupwise-agents.version.date.i386.rpm file is the GroupWise agent package that is installed by the Installation script. You can install the agent package manually if necessary. To see what files are installed by the package, see “[Linux Installation Directory](#)” on page 100.

startups directory

The startups subdirectory contains the default startup files for the GroupWise agents. During installation, a customized startup file is created for each agent that includes the location of the domain or post office serviced by that agent. The customized startup files are named after the domain or post office for which they are created. See “[Linux Installation Directory](#)” on page 100.

S99grpwise file

The S99grpwise file is the GroupWise agent startup script. During installation, it is named grpwise and placed in `/etc/init.d`. If you choose to have the agents start automatically when the server reboots, the Installation script places symbolic links to it named S99grpwise are placed in the rc3.d and rc5.d subdirectories of `/etc/init.d`. For more information, see “[Installing and Starting the GroupWise Linux Agents](#)” in “[Installing a Basic GroupWise System](#)” in the *GroupWise 6.5 Installation Guide*.

domain directory

The domain subdirectory contains the files from which domains are created.

wpdomain.dc file

The wpdomain.dc file is the distribution copy of the data dictionary for rebuilding GroupWise 4.x domain databases (`wpdomain.db` files) in secondary domains.

If the wpdomain.dc file is missing from the primary domain, you cannot rebuild GroupWise 4.x secondary domains. The original wpdomain.dc file is located in the domain directory of the GroupWise distribution media.

Historical Note: WordPerfect Office (WP Office), the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wpdomain.dc. Some naming conventions were originally preserved for backward compatibility.

gwdom.dc file

The gwdom.dc file is the distribution copy of the data dictionary for creating and rebuilding GroupWise 6.x and GroupWise 5.x domain databases (**wpdomain.db** files) in secondary domains.

If the gwdom.dc file is missing from the primary domain, you cannot create or rebuild GroupWise 6.x/5.x secondary domains. The original gwdom.dc file is located in the domain subdirectory of the GroupWise distribution media.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Hence, the wp in wpdomain.db. Some naming conventions were originally preserved for backward compatibility.

wphost.dc file

The wphost.dc file is the distribution copy of the data dictionary for rebuilding GroupWise 4.x post office databases (**wphost.db** files).

If the wphost.dc file is missing from a domain, you cannot rebuild GroupWise 4.x post offices in that domain. The original wphost.dc file is located in the domain subdirectory of the GroupWise distribution media. There is also a copy in the **po** subdirectory.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Post offices were originally called hosts. Hence, the name wphost.dc. Some naming conventions were originally preserved for backward compatibility.

gwpo.dc file

The gwpo.dc file is the distribution copy of the data dictionary for creating and rebuilding GroupWise 6.x and GroupWise 5.x post office databases (**wphost.db** files).

If the gwpo.dc file is missing from a domain, you cannot create or rebuild GroupWise 6.x/5.x post offices in that domain. The original gwpo.dc file is located in the domain directory of the GroupWise CD. There is also a copy in the **po** directory.

Historical Note: WP Office, the predecessor of GroupWise, was originally designed by WordPerfect Corporation (WPCorp). Post offices were originally called hosts. Hence, the name wphost.db. Some naming conventions were originally preserved for backward compatibility.

po directory

The po subdirectory contains the files from which post offices are created.

ngwguard.dc file

The ngwguard.dc file is the distribution copy of the data dictionary for building the following databases in the post office:

- ◆ **ngwguard.db** (guardian database)
- ◆ **dmxxxnn01-FF** (document management databases)
- ◆ **msgnn.db** (message databases)
- ◆ **userxxx.db** (user databases)
- ◆ **puxxxx.db** (databases for replicated items like shared folders)

If the `ngwguard.dc` file is missing from a post office, new databases cannot be created in the post office, so the post office cannot grow. The original `ngwguard.dc` file is located in the `po` directory of the GroupWise CD.

GroupWise Remote also uses the `ngwguard.dc` file as the data dictionary for its local databases.

client directory

The `client` subdirectory contains all files associated with GroupWise Windows client. See “[Client](#)” in the *GroupWise 6.5 Administration Guide*.

install file

The `install` file is the script that GroupWise Cross-Platform client users run to install and set up the GroupWise client on their Linux or Macintosh* workstations. See “[Setting Up the GroupWise Cross-Platform Client](#)” in the *GroupWise 6.5 Installation Guide*.

novell-groupwise-gwclient-version.date.i386.rpm

The `novell-groupwise-client.version.date.i386.rpm` file is the GroupWise client package that is installed by the Installation script. You can install the client package manually if necessary. To see what files are installed by the package, see “[Cross-Platform Client on Linux](#)” on page 156.

ofviews directory

The `client\ofviews` subdirectory contains platform-specific subdirectories of view files for use by the GroupWise client. On Linux, the `win` subdirectory contains the view files for the Windows client, so that users on a Linux post office can use the Windows client if desired. The Cross-Platform client does not use view files like the Windows client does.

In addition, the `gwviewxx.ini` and `ofviewxx.ini` files configure custom views on the menus where users select views in the Windows client. The `gwviewxx.ini` file configures GroupWise 6.x and GroupWise 5.5 views. The `ofviewxx.ini` file configures views from earlier versions of GroupWise.

admin directory

The `admin` subdirectory contains subdirectories for administrative tools that can be used with GroupWise.

novell-groupwise-admin-version.date.i386.rpm file

The `novell-groupwise-admin.version.date.i386.rpm` file is the GroupWise administrator package that is installed by the Installation script. It contains the GroupWise Administrator snap-ins to ConsoleOne. You can install the admin package manually if necessary.

novell-groupwise-dbcopy-version.date.i386.rpm file

The `novell-groupwise-dbcopy.version.date.i386.rpm` file is the GroupWise Database Copy utility package. It is not installed by the Installation script but must be installed manually, as described in “[Using DBCopy on Linux Servers](#)” in “[Databases](#)” in the *GroupWise 6.5 Administration Guide*.

novell-groupwise-gwcheck-version.date.i386.rpm file

The novell-groupwise-gwcheck.version.date.i386.rpm file is the GroupWise Check utility package. It is not installed by the Installation script but must be installed manually, as described in “Starting GWCheck on a Linux Workstation” in “Databases” in the *GroupWise 6.5 Administration Guide*.

NOVLc1Linuxjre-version.i386.rpm file

The NOVLc1Linuxjre-version.i386.rpm file is the package that updates ConsoleOne so that it is compatible with the GroupWise Administrator snap-ins. It is installed for you automatically when you install GroupWise administration.

monitor directory

The monitor subdirectory contains the GroupWise Monitor software. See “Monitor” in the *GroupWise 6.5 Administration Guide*.

novell-groupwise-gwmon-version.date.i386.rpm file

The novell-groupwise-gwmon.version.date.i386.rpm file is the GroupWise Monitor Agent package that is installed by the Installation script. You can install the package manually if necessary. To see what files are installed by the package, see “Monitor Agent Installation” on page 123 and “Apache/Tomcat Installation” on page 127.

novell-groupwise-monitor-version.date.i386.rpm file

The novell-groupwise-monitor.version.date.i386.rpm file is the GroupWise Monitor Application package that is installed by the Installation script. You can install the package manually if necessary. To see what files are installed by the package, see “Monitor Agent Installation” on page 123 and “Apache/Tomcat Installation” on page 127.

S99grpwise-ma file

The S99grpwise-ma file is the Monitor Agent startup script. During installation, it is named grpwise-ma and placed in `/etc/init.d`. If you choose to have the Monitor Agent start automatically when the server reboots, the Installation script places symbolic links to it named S99grpwise-ma in the rc3.d and rc5.d subdirectories of `/etc/init.d`. For more information, see “Installing the Linux Monitor Agent” in “Installing GroupWise Monitor” in the *GroupWise 6.5 Installation Guide*.

internet directory

The internet subdirectory contains subdirectories for GroupWise components that provide and support Internet connectivity.

gwia directory

The gwia subdirectory contains the Internet Agent software, used to connect GroupWise systems across the Internet and to allow GroupWise users to exchange e-mail with users of various Internet e-mail programs. See “Internet Agent” in the *GroupWise 6.5 Administration Guide*.

novell-groupwise-gwia-version.date.i386.rpm file

The novell-groupwise-gwia.version.date.i386.rpm file is the Internet Agent package that is installed by the Installation script. You can install the package manually if necessary. To see what files are installed by the package, see [“Internet Agent Installation” on page 107](#).

root directory

The root directory contains boilerplate versions of the configuration files that are installed to the Internet Agent root directory under the domain directory. For more information, see [“Internet Agent Queue Directory” on page 80](#).

S99grpwise-ia file

The S99grpwise-ia file is the Internet Agent startup script. During installation, it is named grpwise-ma and placed in `/etc/init.d`. If you choose to have the Internet Agent start automatically when the server reboots, the Installation script places symbolic links to it named S99grpwise-ma in the rc3.d and rc5.d subdirectories of `/etc/init.d`. For more information, see [“Starting the Internet Agent on Linux”](#) in [“Installing the GroupWise Internet Agent”](#) in the *GroupWise 6.5 Installation Guide*.

webaccess directory

The internet\webaccess subdirectory contains the GroupWise WebAccess software, which allows users to access their GroupWise mailboxes from a Web browser. See [“WebAccess”](#) in the *GroupWise 6.5 Administration Guide*.

novell-groupwise-gwinter-version.date.i386.rpm file

The novell-groupwise-gwinter.version.date.i386.rpm file is the WebAccess Agent package that is installed by the Installation script. You can install the package manually if necessary. To see what files are installed by the package, see [“WebAccess Agent Installation” on page 113](#).

novell-groupwise-webaccess-version.date.i386.rpm file

The novell-groupwise-webaccess.version.date.i386.rpm file is the WebAccess Application package that is installed by the Installation script. You can install the package manually if necessary. To see what files are installed by the package, see [“WebAccess Agent Installation” on page 113](#) and [“Apache/Tomcat Installation” on page 127](#).

root directory

The root directory contains boilerplate versions of the configuration files that are installed to the WebAccess Agent root directory under the domain directory. For more information, see [“WebAccess Agent Queue Directory” on page 86](#).

S99grpwise-wa file

The S99grpwise-ia file is the WebAccess Agent startup script. During installation, it is named grpwise-ma and placed in `/etc/init.d`. If you choose to have the Internet Agent start automatically when the server reboots, the Installation script places symbolic links to it named S99grpwise-ma in the rc3.d and rc5.d subdirectories of `/etc/init.d`. For more information, see [“Setting Up GroupWise WebAccess on Linux”](#) in [“Installing GroupWise WebAccess”](#) in the *GroupWise 6.5 Installation Guide*.

webservice directory

On Linux, you can install Apache and Tomcat along with WebAccess and Monitor. This automatically integrates WebAccess and Monitor into Apache and Tomcat for you. If you are already running Apache and Tomcat, you can choose to integrate WebAccess and Monitor into that installation, or you can have two Apache/Tomcat installations on your server. WebAccess and Monitor cannot be used without a supporting Web server. For more information, see “[Determining the WebAccess and WebPublisher Applications' Configuration](#)” in “[Installing GroupWise WebAccess](#)” and “[Determining the Monitor Application's Configuration](#)” in “[Installing GroupWise Monitor](#)” *GroupWise 6.5 Installation Guide*.

license directory

The license subdirectory contains the GroupWise Software License Agreement in multiple languages. Use the more command to view the License Agreement.

docs directory

The docs subdirectory contains language-specific subdirectories for the *GroupWise 6.5 Installation Guide* and *Quick Start* in PDF format, along with the Readme in HTML format. You can find the most up-to-date version of the GroupWise documentation on the [GroupWise 6.5 Documentation page \(http://www.novell.com/documentation/gw65/index.html\)](http://www.novell.com/documentation/gw65/index.html).






















gwinst directory




The gwinst subdirectory contains supporting files and subdirectories used by the Installation script.

10 GroupWise Client Installation Directories

- ◆ “Windows Client” on page 153
- ◆ “Cross-Platform Client on Linux” on page 156
- ◆ “Cross-Platform Client on Macintosh” on page 158

Windows Client

 c:\novell\groupwise	Installation directory for GroupWise® 6.x Windows client software
 grpwise.exe	GroupWise client program
 gwtip.exe	GroupWise Tip of the Day program
 notify.exe	GroupWise Notify program
 addrbook.exe	GroupWise Address Book program
 gwcheck.exe	GroupWise GWCheck program
 htrsetup.exe	Hit the Road setup program
 gwimpexe.exe	GroupWise E-Mail Importer Utility
 gwmailto.exe	Web browser support program
 gwreload.exe	Mailbox mode switching program
 gwsync.exe	Mailbox synchronization program
 ngwguard.dc	Data dictionary for databases
 wprof.dc	Data dictionary for Remote Address Book
 *.dll	DLL programs to support GroupWise Windows client
 *.ocx	OLE custom controls
 *.flt	Graphics filters for Inso viewers
 *.chm	Help files to support the GroupWise client
 ofviews	GroupWise view files
 win	GroupWise view files for Windows
 *.vew	View files
 *.ini	View initialization files

 ppforms	Day planner forms
 *.bfp	Form description files
 *.prs	Print resource files

c:\novell\groupwise

The GroupWise Windows client installation directory contains all files necessary to run the GroupWise client. The default location is c:\novell\groupwise.

- ◆ For a standard installation, the GroupWise client directory resides on the user's workstation.
- ◆ For a workstation installation, the GroupWise client directory resides on the network and is run by users on many workstations.
- ◆ For a GroupWise Remote installation, the GroupWise Remote directory contains the same files as would be installed on a workstation connected to the network. Whenever GroupWise cannot make a network connection, it checks for a GroupWise Remote mailbox to access. See [“Remote Mailbox Directory” on page 93](#).

grpwise.exe file

The grpwise.exe file is the GroupWise 6.x client program for use with Windows.

gwtip.exe file

The gwtip.exe file is the GroupWise Tip of the Day program, which displays helpful information about a GroupWise feature each time you start the GroupWise client.

notify.exe file

The notify.exe file is the program that alerts you whenever you have incoming items, when your outgoing items are opened, or when you have an upcoming appointment. Notify can alert you in four ways: a sound, a dialog box, a small icon, or by launching an application.

addrbook.exe file

The addrbook.exe file is the program that accesses the GroupWise Address Book. The Address Book contains information for all Novell® eDirectory™ users, GroupWise external users, GroupWise external entities, GroupWise e-mail distribution lists, and so on.

gwcheck.exe file

The gwchek.exe file is the program that checks and repairs GroupWise databases located on users' workstations or other locations not accessible to the Mailbox/Library Maintenance feature of Console. See [“GroupWise Check”](#) in [“Databases”](#) in the *GroupWise 6.5 Administration Guide*.

htrsetup.exe file

The htrsetup.exe file is the setup program for Hit the Road, which creates your Remote mailbox. See [“Remote Mode”](#) in [“Client”](#) in the *GroupWise 6.5 Administration Guide*.

gwimpexe.exe file

The gwimpexe.exe file is the GroupWise E-Mail Importer Utility, which enables you to import supported POP3/IMPA4 accounts into GroupWise. See “[Importing POP3 and IMAP4 Accounts](#)” in “[Working with Items in Your Mailbox](#)” in the *GroupWise 6.5 Windows Client User Guide*.

gwmailto.exe file

The gwmailto.exe file is the program that integrates the GroupWise client with your Web browser. If you select Internet Browser Mail Integration during installation, the GroupWise client becomes the default e-mail program on your workstation. See “[Setting Up the GroupWise Client](#)” in “[Installing the GroupWise Windows and Cross-Platform Clients](#)” in the *GroupWise 6.5 Installation Guide*.

gwreload.exe file

The gwreload.exe file is the program that restarts the GroupWise client when you switch between Online, Caching, and Remote modes. See “[Using Caching Mode](#)” and “[Using Remote Mode](#)” in the *GroupWise 6.5 Windows Client User Guide*.

gwsync.exe file

The gwsync.exe file is the program that synchronizes your Online and Caching mailboxes. See “[Using Caching Mode](#)” in the *GroupWise 6.5 Windows Client User Guide*.

ngwguard.dc file

The ngwguard.dc file is the data dictionary for building databases in the post office. GroupWise Remote also uses the ngwguard.dc file as the data dictionary for its local databases.

wprof.dc file

The wprof.dc file is the data dictionary for the GroupWise Remote Address Book ([wprof.db](#)).

Historical Note: An earlier version of GroupWise Remote, designed by WordPerfect Corporation (WPCorp), was named WP Office Remote. Hence, the wprof in wprof.dc. Some naming conventions were originally preserved for backward compatibility.

*.dll files

*.dll files are dynamically linked libraries of program code used by executable programs. Language-independent *.dll files are typically located in the same directory with the executable programs they support. Language-specific *.dll files are grouped into subdirectories by language.

*.ocx files

*.ocx files are OLE custom controls. For example, *.ocx files control the Address Book and name completion.

*.flt files

*.flt files are graphics filters used by the Inso viewers incorporated into the GroupWise client. The viewers enable you to view a wide variety of file types from within GroupWise.

*.chm files

*.chm files contain the online documentation for all components of GroupWise. *.chm files are typically located in the same directory with the executable programs they support.

ofviews directory

The ofviews directory contains platform-specific subdirectories of view files for use by the GroupWise client.

Historical Note: An earlier version of GroupWise, designed by WordPerfect Corporation (WPCorp), was named WP Office. Hence, the of in ofviews. Some naming conventions were originally preserved for backward compatibility.

win directory

The win subdirectory contains view (*.vew) files for use by the GroupWise client. It also contains initialization (*.ini) files to control display of views.

ppforms directory

The ppforms directory contains day planner forms for printing GroupWise calendars and tasks.






*.bfp files









*.bfp files are form description files that contain binder, filler, and page attributes for forms.

*.prs files

*.prs files are print resource files that indicate which forms are available for which languages. For example, day planners used by English-speaking people in the United States have different forms from day planners used for various languages in Europe.

Cross-Platform Client on Linux

 /opt/novell/groupwise/client	GroupWise Cross-Platform client installation directory
 gwclient.desktop	Desktop shortcut for the Cross-Platform client
 gwclient.png	Desktop icon for the Cross-Platform client shortcut
 bin	Subdirectory for the Cross-Platform client executable
 groupwise	Cross-Platform client executable

 lib	Subdirectory for Cross-Platform client library files
 gwenlxxx.fil	Cross-Platform client language resource files
 ngwguard.dc	Data dictionary for local databases on the user's workstation
 wprof.dc	Data dictionary for the local copy of the Address Book
 *.jar	Java archive (.jar) files used by the Cross-Platform client
 *.so	Shared library files used by the Cross-Platform client
 jre	Java Runtime Environment (JRE) used by the Cross-Platform client
 logs	Log files generated by the Cross-Platform client

/opt/novell/groupwise/client directory

The /opt/novell/groupwise/client directory is the default location for the GroupWise Cross-Platform software. The Installation program does not provide the opportunity to specify a different location and typically there is no need to move the software.

gwclient.desktop file

The gwclient.desktop file create the GroupWise Cross-Platform client shortcut on your Linux desktop.

gwclient.png file

The gwclient.png file is the GroupWise icon used for Cross-Platform client shortcut.

bin directory

The bin directory is the standard Linux location for executable files.

groupwise file

The groupwise file is the GroupWise Cross-Platform client executable. See [“Starting the GroupWise Cross-Platform Client”](#) in the *GroupWise 6.5 Cross-Platform Client User Guide*.

lib directory

The lib directory is the standard Linux location for library files.

gwenlxxx.fil file

The gwenlxxx.file provides the language-specific strings for the localized versions of the Cross-Platform client. The *xx* is a two-letter language code.

ngwguard.dc file

The ngwguard.dc file is the data dictionary for building the databases used by the Cross-Platform client. It is parallel in function to the **ngwguard.dc** file in the post office.

wprof.dc file

The wprof.dc file is the data dictionary used to create the local copy of the Address Book that resides on the user's workstation (wprof.db).

*.jar files

The *.jar files efficiently store information referenced by the Cross-Platform client. The gwclient.jar file contains the Cross-Platform online help information.

*.so files

The *.so files are Linux shared library files that provide information to the Cross-Platform client executable.















jre directory

The jre directory holds the version of the Java Runtime Environment (JRE) that is required by the Cross-Platform client.

logs directory

The logs directory stores log files generated by the Cross-Platform client when an error occurs.

Cross-Platform Client on Macintosh

 /Applications/GroupWise.app	GroupWise Cross-Platform client installation directory
 Contents	Standard Macintosh directory for application files
 PkgInfo	Standard Macintosh package info file
 Info.plist	Standard Macintosh package list file
 MacOS	Standard Macintosh directory for operating system files
 Resources	Standard Macintosh directory for resource files
 GroupWise.icns	Icons used by the Cross-Platform client
 lib	Subdirectory for Cross-Platform client library files
 gwenlxxx.fil	Data dictionary for local databases on the user's workstation
 ngwguard.dc	Data dictionary for the local copy of the Address Book
 wprof.dc	Cross-Platform client language information files
 Java	JRE used by the Cross-Platform client
 *.jar	Java archive (.jar) files used by the Cross-Platform client
 *.dylib	Dynamic libraries used by the Cross-Platform client

/Applications/GroupWise.app directory

The /Applications/GroupWise.app directory contains the GroupWise Cross-Platform client on Macintosh.

Contents directory

The Contents directory is the standard location on Macintosh for all files that make up an application.

MacOS directory

The MacOS directory contains the JavaApplicationStub file that enables the Cross-Platform client to invoke the Java Virtual Machine (JVM) on Macintosh.

Resources directory

The Resources directory is the standard location on Macintosh for files that support an application.

GroupWise.icns file

The GroupWise.icns file contains all the images that display in the Cross-Platform client.

lib directory

The lib directory holds library files that provide information required by the Cross-Platform client.

gwenlxxx.fil file

The gwenlxxx.file provides the language-specific strings for the localized versions of the Cross-Platform client. The xx is a two-letter language code.

ngwguard.dc file

The ngwguard.dc file is the data dictionary for building the databases used by the Cross-Platform client. It is parallel in function to the [ngwguard.dc](#) file in the post office.

wprof.dc file

The wprof.dc file is the data dictionary used to create the local copy of the Address Book that resides on the user's workstation (wprof.db).

Java directory

The Java directory holds the version of the Java Runtime Environment (JRE) that is required by the Cross-Platform client.

***.jar files**

The *.jar files efficiently store information referenced by the Cross-Platform client. The gwclient.jar file contains the Cross-Platform online help information.

***.dylib files**

The *.dylib files are dynamic library files that provide information to the Cross-Platform client program.



Documentation Updates

This section lists updates to *Troubleshooting 3: Message Flow and Directory Structure* that have been made since the initial release of GroupWise® 6.5. The information will help you to keep current on documentation updates and, in some cases, software updates (such as a Support Pack release).

The information is grouped according to the date when *Troubleshooting 3: Message Flow and Directory Structure* was republished. Within each dated section, the updates are listed by the names of the main table of contents sections.

Troubleshooting 3: Message Flow and Directory Structure has been updated on the following dates:

- ♦ “February 28, 2005 (GroupWise 6.5 SP4)” on page 161
- ♦ “September 30, 2004” on page 161
- ♦ “September 30, 2003” on page 162
- ♦ “July 16, 2003 (GroupWise 6.5 SP1)” on page 162

February 28, 2005 (GroupWise 6.5 SP4)

Location	Change
Message Flow Diagrams	
“Message Delivery to and from the Internet” on page 41	Updated some outdated icons.

September 30, 2004

Location	Change
Directory Structure Diagrams	
“WebAccess Agent Queue Directory” on page 86	Added the WebAccess Agent queue directory structure.
“Caching Mailbox Directory” on page 89	Added the Caching mailbox directory structure.

Location	Change
“GroupWise Agent Installation (POA and MTA)” on page 97	Added the Linux agent directory structure.
“Internet Agent Installation” on page 107	Added the Internet Agent directory structure for all platforms.
“WebAccess Agent Installation” on page 113	Added the WebAccess Agent directory structure for all platforms.
“Monitor Agent Installation” on page 123	Added the Monitor Agent directory structure for Linux and Windows.
“Apache/Tomcat Installation” on page 127	Added the Apache and Tomcat directory structures for NetWare® and Linux.
“Linux Software Distribution Directory” on page 143	Added the Linux software distribution directory structure.
“Cross-Platform Client on Linux” on page 156	Added the Cross-Platform client for Linux installation directory.
“Cross-Platform Client on Macintosh” on page 158	Added the Cross-Platform client for Macintosh installation directory.

September 30, 2003

Location	Change
Message Flow Diagrams	
“Message Flow Diagrams” on page 11	Added a missing item to the bulleted list.

July 16, 2003 (GroupWise 6.5 SP1)

The following sections list the updates that were made to GroupWise 6.5 *Troubleshooting 3: Message Flow and Directory Structure*:

Location	Change
Directory Structure Diagrams	
“Post Office Directory” on page 64, <i>mmddpoa.nnn</i>, and <i>wprof50.db</i>	Added POA log files and the downloadable system Address Book for Remote client users to the post office directory structure.