



TEMENOS™

SYSTEM ADMINISTRATION MANUAL

ACCOUNT SETUP



Copyright

Copyright (c) 2008 TEMENOS HOLDINGS NV

All rights reserved.

This document contains proprietary information that is protected by copyright. No part of this document may be reproduced, transmitted, or made available directly or indirectly to a third party without the express written agreement of TEMENOS UK Limited. Receipt of this material directly from TEMENOS UK Limited constitutes its express permission to copy. Permission to use or copy this document expressly excludes modifying it for any purpose, or using it to create a derivative therefrom.

Acknowledgements

Information regarding Unicode has been provided in part courtesy of the Unicode Consortium. The Unicode Consortium is a non-profit organization founded to develop, extend and promote use of the Unicode Standard, which specifies the representation of text in modern software products and standards. The membership of the consortium represents a broad spectrum of corporations and organizations in the computer and information processing industry. The consortium is supported financially solely through membership dues. Membership in the Unicode Consortium is open to organizations and individuals anywhere in the world who support the Unicode Standard and wish to assist in its extension and implementation.

Portions of the information included herein regarding IBM's ICU has been reprinted by permission from International Business Machines Corporation copyright 2001

jBASE, jBASIC, jED, jSHELL, jLP, jEDI, jCL, jQL, j1, j2 j3 j4 and jPLUS files are trademarks of TEMENOS Holdings NV.

REALITY is a trademark of Northgate Solutions Limited.

PICK is a trademark of Raining Data Inc.

All other trademarks are acknowledged.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Windows, Windows NT, and Excel are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names used in this publication may be trademarks or service marks of others.

Errata and Comments

If you have any comments regarding this manual or wish to report any errors in the documentation, please document them and send them to the address below:

Technical Publications Department

TEMENOS UK Limited

2 PeopleBuilding Estate

Hemel Hempstead

Herts

England

Tel SB: +44 (0) 1442 431000

Fax: +44 (0) 1442 431001

Please include your name, company, address, and telephone and fax numbers, and email address if applicable. documentation@temenos.com

Contents

SYSTEM ADMINISTRATION MANUAL	I
ACCOUNT SETUP	I
Documentation Conventions	1
ACCOUNTS	3
SYSTEM FILE.....	3
Encrypted Password	4
Optional Parameters	4
Extended Values.....	5
MD	5
LOGTO	6
REMOTE FILES	8
Q Pointers.....	8
Remote File Service	9
jCONNECT Library	10
Client Connections	11
Configuration Files.....	12

DOCUMENTATION CONVENTIONS

This manual uses the following conventions:

Convention	Usage
BOLD	In syntax, bold indicates commands, function names, and options. In text, bold indicates keys to press, function names, menu selections, and MS-DOS commands.
UPPERCASE	In syntax, uppercase indicates JBASE commands, keywords, and options; BASIC statements and functions; and SQL statements and keywords. In text, uppercase also indicates JBASE identifiers such as filenames, account names, schema names, and Windows NT filenames and pathnames.
UPPERCASE <i>Italic</i>	In syntax, italic indicates information that you supply. In text, italic also indicates UNIX commands and options, filenames, and pathnames.
Courier	Courier indicates examples of source code and system output.
Courier Bold	Courier Bold In examples, courier bold indicates characters that the user types or keys (for example, <Return>).
[]	Brackets enclose optional items. Do not type the brackets unless indicated.
{ }	Braces enclose nonoptional items from which you must select at least one. Do not type the braces.
ItemA itemB	A vertical bar separating items indicates that you can choose only one item. Do not type the vertical bar.
. . .	Three periods indicate that more of the same type of item can optionally follow.
⇒	A right arrow between menu options indicates you should choose each option in sequence. For example, “Choose File ⇒ Exit ” means you should choose File from the menu bar, and then choose Exit from the File pull-down menu.

Syntax definitions and examples are indented for ease in reading.

All punctuation marks included in the syntax—for example, commas, parentheses, or quotation marks—are required unless otherwise indicated.

Syntax lines that do not fit on one line in this manual are continued on subsequent lines. The continuation lines are indented. When entering syntax, type the entire syntax entry, including the continuation lines, on the same input line.

Accounts

System File

The location of the jBASE SYSTEM file is determined by the JEDIFILENAME_SYSTEM environment or registry variable. The environment or registry variable is usually setup when the user logs on. By default jBASE uses a hash file, named SYSTEM, in a subdirectory, named src, in the jBASE release directory as defined by the environment or registry variable JBCRELEASEDIR.

Use of the SYSTEM file also depends upon the setting of the environment or registry variable JEDIFILENAME_MD, which describes the users MD/VOC file. This environment or registry variable should be set to the users required MD file. E.g. to use an MD file in the users home directory set the environment or registry variable, JEDIFILENAME_MD to \$HOME/MD or %HOME%\MD.

In its minimum form, a record in the SYSTEM file must contain two fields. Field 1 contains the character "D" to specify a local account, and field 2 contains the absolute path of the account directory. For example:

```
ACCOUNTX
001 D
002 C:\users\accx or /accounts/accx
```

Fields 7 and 11 and fields 20 – 37 (if specified) of the SYSTEM record are used by the jBASE LOGTO and PASSWORD commands. All other fields are reserved. The full format of a SYSTEM record is as follows:

```
AccountId
001 Type
002 Absolute Pathname
003 Reserved
004 Reserved
005 Reserved
006 Reserved
007 {Optional encrypted password}
008 Reserved
009 Reserved
010 Reserved
011 {Optional parameters}
...
```

020 ESYSTEM_START
021 value for JBCEMULATE
022 value for HOME
023 value for JBCDEVBIN
024 value for JBCDEVLIB
025 value for PATH
026 value for JBCOBJECTLIST
027 value for JEDIFILEPATH
028 value for JEDIFILENAME_MD
029 value for JBC_TCLRESTART
030 value for JBC_ENDRESTART
031 value for JBCRESTARTPROG
032 value for JBCLOGINPROG
033 value for JBCLOGNAME
034 value for JBASE_I18N
035 value for JBCPORTNO
036 values for any other environment variables (multi-valued pairs)
037 ESYSTEM_END

Encrypted Password

This optional field should only be maintained through the jBASE PASSWORD command. The PASSWORD command prompts for an Account Name, which must be a valid entry in the SYSTEM file (i.e. the file defined by the JEDIFILENAME_SYSTEM environment variable). Enter the password twice for verification. NOTE: this password is case sensitive (e.g. "MyPassword" is different than "mypassword").

Optional Parameters

The jBASE LOGTO command uses the optional parameters field. It allows the environment of each account to be set up during the LOGTO event. When using the -X option, enclose multiple parameters or parameters with spaces in single (') or double quotes ("). See EXAMPLES below.

The optional parameters are:

- A Accountname Parsed Into Jbclogname
- H Directory Parsed Into Home
- M Filename Parsed Into Jedifilename_Md
- V Termtyp Parsed Into Term
- X Environment Variable Overrides Any Previous Specifications

variable determines the location of the jBASE MD file. This environment variable is usually set-up in the “. Profile” script when the user logs on. If the environment variable is not set-up then the default path \$HOME/MD is used when attempting to open the jBASE MD file. Most accounts/user directories have an MD file specific to that account/user directory. If users wish use an MD and the MD file does not already exist then the user should use the jBASE CREATE-FILE command to create a single level dictionary file.

```
CREATE-FILE DICT MD 3
```

Increase the modulo of the MD file depending on the number of jCL programs and Qpointers it is expected to contain. Copy the jQL modifiers SORT and SELECT into the new MD file to use the jQL modifiers. This can be achieved using a jBASE script called UpdateMD or by copying from the jBASE release directory file:

UNIX	Windows
/opt/jbase4/src/MD	\JJBASE30\src\MD

UpdateMD

To action, other jCL programs in a different file and/or action catalogued jBASIC programs place short jCL programs in the MD file.

```
E.g. APPL Account startup jCL program
001 PQ jCL type identifier
002 [PROCS APPL.CHK] jCL external routine, APPL.CHK
003 HAPPL.INIT jBC catalogued program, APPL.INIT
004 P
```

Keep the MD file as a single level dictionary file to avoid confusion. The real filename is “./MDJD” (or \MDJD on Windows). All single level dictionary files contain a reflexive Q-pointer entry of the same name as the filename without the appended “JD”, for DICT. e.g. the reflexive Qpointer entry in the “./MDJD” file is MD, therefore the following jBASE copy command will copy the same record from the same file to the terminal.

E.g. CT MD - MD does the same as CT MD] D MD

LOGTO

Switches to the specified user account

LOGTO account-name

Where: account-name is the pseudo account (defined in the SYSTEM file) or, under UNIX, the user account to switch to

NOTES

This command is provided for legacy use and applications. Ideally, new development should use “su - account-name” (for UNIX) or the Windows Logout/Login functions.

Execution of this command changes environmental variables differently depending upon the type of account (pseudo or real) and the OS being used, as shown in the table below:

Variable	Effect of LOGTO
HOME	Changed to the directory of the specified account
JBCDEV_BIN	Under Windows, only reset if set prior to LOGTO. Under UNIX pseudo accounts, this is always set. For UNIX accounts, this is not set.
JBCDEV_LIB	Under Windows, only reset if set prior to LOGTO. Under UNIX pseudo accounts, this is always set. For UNIX accounts, this is not set.
JBCLOGNAME	Under Windows (and UNIX pseudo accounts), set to the name of the account you log to. With UNIX accounts, LOGNAME is set.
JBCOBJECTLIST	Reset, but only if set prior to LOGTO
LOGNAME	With UNIX accounts, set to the name of the account you LOGTO
PATH	jBASE automatically appends %HOME%\bin (or \$HOME/bin for UNIX systems), so by definition it is changed after LOGTO
PWD	same as HOME

For use under Windows, the user must have a valid SYSTEM file entry defined for the command to work. In addition, this command does not work when ran from “cmd.exe” (instead, use jLoginNT.exe).

This command does not directly support any options. However, it is possible to specify options specific to the new account’s environment via the SYSTEM file entry as follows:

- A Accountname Parsed Into JBCLOGNAME
- H Directory Parsed Into HOME
- M Filename Parsed Into JEDIFILENAME_MD
- V Termtype Parsed Into TERM
- X Environmental Variable Overrides Any Previous Specification

UNIX Only

- E Programname Parsed Into JBC_ENDRESTART
- L Language Parsed Into LANG
- R Programname Parsed To Restart After Logoff (Jloginmsg)
- S Programname Parsed To Execute At Logon. Usually Jsh -S Jsh.

-T Programname Parsed Into JBC_TCLRESTART

This command cannot be used in the command stream for PH-START.

This command cannot be used in the Primary input buffer of a jCL command.

Examples

LOGTO homer

Logs off the current account and into the account "homer"

Remote Files

Q POINTERS

Remote file pointers are one mechanism by which a client application can access files located on a remote system.

A Q-pointer is created in the MD file and set to point to a SYSTEM file entry as normal for Q pointers. E.g

QPTR

001 Q

002 AccountName

003 FileName

The Qpointer is resolved in the normal way to the SYSTEM file entry. However, the SYSTEM file entry is an Rpointer, usually of the following form.

AccountName

001 R

002

003

004 HostName

The full format of and R pointer SYSTEM entry is as follows:

AccountName

001 R

002 {RemoteAccountName} {ServerName}

003

004 {HostName}

Where

RemoteAccountName - Optional Remote Host SYSTEM file reference.

ServerName - Optional network service name.

HostName - Remote HostName or IP Address

If the optional RemoteAccountName is specified then searching for remote relative, filenames will start in the directory path specified by the SYSTEM entry on the remote host. If the specified RemoteAccountName is not a valid accountname entry in the remote SYSTEM file then the open request will fail with file not found. If left blank then searching for remote relative filenames will begin in the home directory of the network userid.

The optional ServerName is used to specify an alternative network service required by locating a matching entry in the services file. If left blank the server name will default to jRFS, which is then used to reference the services file and find the required network service. NOTE: that a dash, “-“, should be used as a placeholder to signify a blank RemoteAccountName when only an alternate ServerName is specified

The HostName is optional and if specified is used to determine the network address of the target system by locating a matching entry in the hosts file. If left blank the target system name will default to the local host node name or on some systems localhost, which is then used to reference the hosts file and find the network address of the local system. The HostName may also be entered as a dotted IP address.

In order to gain access to the remote system via the jRFS, the client must provide a remoteusername, which will either be the same as the client user name or a mapped alternative, see later. The supplied remoteusername must be a valid user name on the remote system or the connection will be denied. The remoteusername is also used to determine file access permissions. The combination of the second attribute from the remote Dpointer, or the remote home directory of the remoteusername, and the third attribute from the Qpointer produces the filepath for the file referenced by the Qpointer. However, if the third attribute from the Qpointer is an absolute path then this reference will be used on the remote system.

Remote File Service

The jRFS process must be running on the target system before access via Rpointers can be achieved. The jRFS process, once initiated on the target system, waits passively for a client request from another system. When a client request is detected, the server then spawns another process to handle the request. The spawned process performs a security check of the connecting client and if successful then becomes a remote user process as determined by the remoteusername. The remote user process then attempts to open the specified file via the standard jEDI interface. The result of the file open is returned to the client process. The remote user process will continue to handle all further remote file operations for the client process. The remote user process continues to be active until all opened remote files are closed and the client issues a disconnect or an error/shutdown signal is received, in which case all files still open are closed using the jEDI interface.

Before the jRFS process can accept client requests the services file must be updated to include an entry for the server. This entry describes the connection type and an arbitrary connecting port. A port should be chosen which does not clash with any existing services on both the server and client systems.

jRFS 5001/tcp

UNIX	NT
<p>The jRFS can be initiated at boot time or manually as root by installing the script, jRFS.init.d, located in the “src” subdirectory of the jBASE release directory.</p>	<p>The jRFS process can be installed as a service by the jRFSInstall.exe and removed from the services by the jRFSRemove.exe. Once installed and configured the jRFS can be started by the services applet in the control panel. To install and start the jRFS the user must have administrator privileges.</p>

Once started the jRFS process attempts to read the jRFS configuration file, jrfs_config, This configuration file contains flags which can be asserted to “flag” the jRFS and client request modules to produce trace information for remote file operations. The trace facility can be useful for determining successful Rpointer resolution or confirmation that a remote file operation has been successfully constructed, sent and received. The jRFS process passes control to the jConnect library, which reads the jConnect configuration file, jnet_config. This configuration file contains flags, which can be utilized to specify the security mechanism for client requests, and “flag” the jConnect library to provide trace information. The jConnect trace information can be used to ascertain why a client connect request has been denied.

WARNING: The trace features can severely degrade performance and generate large trace files. The trace features should only be enabled when required and care taken to allow sufficient space in the log file partition.

JCONNECT LIBRARY

UNIX	NT
<p>The jConnect library provides for two mutually exclusive security mechanisms. The default mechanism is the standard UNIX ruserok mechanism usually utilized by ftp, rcp, etc. This mechanism first checks the /etc/hosts.equiv file and then the target users ‘ rhosts’ file to determine whether</p>	<p>The jConnect library provides user security on NT by the jnet_access file. This file should be placed in a NTFS directory set with administrator read only privileges. The directory location should then be configured into a system environment variable JBCNETACCESS.</p>

further processing is allowed. An alternative method, which can be configured in the `jnet_config` configuration file, is to use the `jnetok` mechanism. The `jnetok` mechanism requires that certain components included in the client connect request match an entry in the `/etc/jnet_access` file. For security purposes, this file should only be readable by root.

NOTE: Both security mechanisms utilize the connecting hostname as listed in the hosts file, the `remoteusername` and the client username as supplied by the client connect request. The `jnetok` mechanism also requires the correct servername before permission is granted to continue the client connection. The `jConnect` library will deny access to clients whose `remoteusername` resolves to a user id of less than one hundred.

Once the security check has completed successfully the `jConnect` library then proceeds to set the `groupid`, `userid` and home directory along with certain other required environment variables. The `jConnect` library then attempts to match the client connects request components with an entry in the `jnet_env` file. If successful then any related environment entries are setup before returning control back to the `jBASE Remote File Service` modules.

CLIENT CONNECTIONS

Client connections can be configured to map client connect request components to alternative values by using the client mapping feature of the `jConnect` library. The client-mapping feature requires that certain components from the client connect request match an entry in the `jnet_map` file. If a valid entry is found then some of the components can be mapped to alternatives before executing the connect request proper. The components that can be mapped are the `hostname`, the `servername` and the `remoteusername`. This feature enables administrators to configure an alternative `remoteusername` for specific clients or rename a remote system without having to change all the references in the local `SYSTEM` file.

NOTE: Client mapping is invoked before the resolution of the network address and service, therefore the mapped names will be used in place of original to obtain the connection parameters.

CONFIGURATION FILES

Templates for all jRFS configuration files can be found in the “config” subdirectory of the jBASE release directory. It is recommended that the template file be copied to a read only directory and accessed using the JBCNETDIR environment variable. The following files are used for configuration:

File	Description
jnet_config	jConnect client and server configuration file
jnet_map	jConnect client and server configuration file
jnet_env	jConnect server environment map file
jvfs_config	jRFS client and server configuration file

QPOINTERS

Due to the additional overhead of resolving Q-to-Q pointers, which is not normally necessary in a well-maintained system, an environment variable JEDIENABLEQ2Q must be set. The use of Q-to-Q pointers should be discouraged.

NOTE: If an ACCOUNT name does not exist in the JEDIFILENAME_SYSTEM when resolving system entries then on UNIX, the ACCOUNT is looked up in the /etc/password file. If successful then the home directory from the /etc/password file is used for SystemEntryAttr2.

Q and F pointer Resolution

Structure	Resolution
1 <1> Q . <2> <3>	Reflexive Qpointer to MD JEDIFILENAME_MD else HOME/MD
2 <1> Q . <2> <3> RealFILE	Append File to HOME directory HOME/RealFILE
3 <1> F . <2> RealFILE <3> RealDictFILE	Append File to HOME directory HOME/RealFILE or HOME/RealDictFile
4 <1> F . <2> RealFILE <3> D_RealDictFILE	Append File to HOME directory HOME/RealFILE or HOME/RealDictFileJD
5 <1> Q	Recursively look up until failure AnotherQpointer or

	Structure	Resolution
.	<2> <3>AnotherQ/Fpointer	AnotherFpointer in HOME/MD or JEDIFILENAME_MD Then resolve as 2 or 3 HOME/RealFILE
6	<1> Q	Qpointer to SYSTEM
.	<2> SYSTEM <3>	JEDIFILENAME_SYSTEM
7	<1> Q	Qpointer to SYSTEM
.	<2> SYSTEM <3> SYSTEM	JEDIFILENAME_SYSTEM
8	<1> Q	Qpointer to ACCOUNT MD
.	<2> SYSTEM <3> RealACCOUNT	SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD
9	<1> Q	Qpointer to ACCOUNT MD
.	<2> SYSTEM <3> QpointerACCOUNT	Recursively look up until failure QpointerACCOUNT in JEDIFILENAME_SYSTEM Then resolve as 8. SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD
1	<1> Q	Qpointer to ACCOUNT MD
0	<2> RealACCOUNT <3>	SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD
1	<1> Q	Qpointer to ACCOUNT MD
1	<2> QpointerACCOUNT <3>	Recursively look up until failure QpointerACCOUNT in JEDIFILENAME_SYSTEM Then resolve as 10 SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD
1	<1> Q	Qpointer to ACCOUNT FILE
2	<2> RealACCOUNT <3> RealFile	SystemEntryAttr2/RealFILE
1	<1> Q	Qpointer to ACCOUNT FILE
3	<2> QpointerACCOUNT <3> RealFile	Recursively look up until failure QpointerACCOUNT in JEDIFILENAME_SYSTEM Then resolve as 12 SystemEntryAttr2/RealFILE
1	<1> Q	Qpointer to ACCOUNT FILE

Structure		Resolution
4	<2> RealACCOUNT . <3> AnotherQ/Fpointer	Recursively look up until failure AnotherQpointer or AnotherFpointer in SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD Then resolve as 12
1	<1> Q	Qpointer to ACCOUNT FILE
5	<2> QpointerACCOUNT . <3> AnotherQ/Fpointer	Recursively look up until failure QpointerACCOUNT in JEDIFILENAME_SYSTEM Recursively look up until failure AnotherQpointer or AnotherFPointer in SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD Then resolve as 12
1	<1> Q	Qpointer to ACCOUNT FILE,SECTION
6	<2> ACCOUNT . <3> RealFILE,SECTION	SystemEntryAttr2/FILE]MSECTION
1	<1> Q	Qpointer to ACCOUNT FILE,SECTION
7	<2> ACCOUNT . <3> AnotherQ/Fpointer,SECTION	Recursively look up until failure AnotherQpointer or AnotherFpointer in SystemEntryAttr2/SuffixofJEDIFILENAME_MD or SystemEntryAttr2/MD Then resolve as 16 SystemEntryAttr2/FILE]MSECTION
1	<1> Q	Remote Q pointer for use with the jBASE Remote file service. See jRFS
8	<2> Remote Account . <3> Remote File	
1	<1> QL	Remote Q pointer for use with the jBASE Remote file service. See jRFS. The L indicates that updates to remote file are not logged.
9	<2> Remote Account . <3> Remote File	
2	<1> F	Fpointer to absolute file
0	<2> /absolute/path/for/DATA . <3> /absolute/path/for/DICT	/absolute/path/for/DATA

Please ask with regard to creating an index

COMMENT SHEET

Please give page number and description for any errors found:

Page	Error

Please use the box below to describe any material you think is missing; describe any material which is not easily understood; enter any suggestions for improvement; provide any specific examples of how you use your system which you think would be useful to readers of this manual. Continue on a separate sheet if necessary.

Copy and paste this page to a word document and include your name address and telephone number. Email to documentation@temenos.com

