

WitchDoctor-WDX User's Guide

by Dominique Chabord

dominique.chabord@bluedjinn.com

Shaman-X project:

<http://www.shaman-x.org>
contact@shaman-x.org

Reference:

Shaman-X-UG-WDX10-E

Document version:

V0.e draft

Release date:

8/29/03

Software version:

WitchDoctor-WDX Version 0.4 .

Download directory

<http://downloads.shaman-x.org/wdx-userguide.pdf>

Aknowledgements

WitchDoctor-WDX code is based on HP's DECwatchdog/Autopilot GPL code. This document includes information from DECwatchdog/Autopilot documentation set published by HP under special terms and conditions for Shaman-X project.

History

<u>Version</u>	<u>Modification</u>	<u>Author</u>
V0.a draft	Creation	Dominique Chabord
V0.b draft	Corrections	Dominique Chabord
V0.c-draft	Support file documentation	Dominique Chabord
V0.d-draft	Included license text	Dominique Chabord
V0.e-draft	Update for WDX V0.4	Dominique Chabord

Services

Professionnals are on hand to help you to increase your efficiency in deploying and using Shaman-X software tools. Please visit <http://www.shaman-x.com> for more details.

Trademarks

OpenVMS is a trademark of Digital Equipment Corporation
 UNIX is a trademark of Bell Laboratories
 Windows NT is a Trademark of Microsoft Corporation.
 Other trademarks belong to their respective owners.

© Copyright 2003 Dominique Chabord, www.bluedjinn.com

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation;

A copy of the license is available at the end of this document or by Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111, USA

NO WARRANTY

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Contents

Preface

1 WitchDoctor-WDX Presentation

- 1.1 Overview
- 1.2 WitchDoctor-WDX Software Components
- 1.3 WitchDoctor-WDX Software Description

2 WDX Configuration

- 2.1 Configuration Step-by-Step Procedure
- 2.2 Define a Mission Name
- 2.3 UDP Service
- 2.4 Interfaces
- 2.5 Mission Configuration File
- 2.6 Local File
- 2.7 Support File
- 2.8 Script Files

3 WitchDoctor-WDX Management

- 3.1 Prerequisites
- 3.2 Restrictions
- 3.3 Networking Management
- 3.4 WitchDoctor-WDX Version
- 3.5 Configuration Files
- 3.6 System Clock Synchronisation
- 3.7 Mission Maintenance, Start, Stop and Pause

4 WitchDoctor-WDX Operator Interface

- 4.1 Starting the wdx process
- 4.2 wdx Command Format
- 4.3 Mission Management Commands
- 4.4 Process Management Commands
- 4.5 Masterholder/Substitute Management Commands
- 4.6 Resource and Directory Management Commands
- 4.7 Information Commands
- 4.8 Option Restrictions

5 Using WitchDoctor-WDX on Windows NT

- 5.1 Post Installation Configuration
- 5.2 Start the wdx Process
- 5.3 Service Declaration and Configuration of wdx Process
- 5.4 Wdx Initialization service
- 5.5 Startup errors
- 5.6 Wdx Scripts
- 5.7 Time Setting

6 WitchDoctor-WDX behaviour in an OpenVMS cluster environment

- 6.1 Cluster, quorum and processes behaviour
- 6.2 WitchDoctor-WDX resiliency upon cluster transition with loss of quorum
- 6.3 The Cluster WDX resource

7 WitchDoctor-WDX Application Programmable Interface

- 7.1 Architecture
- 7.2 Protocol definition
- 7.3 API definition
- 7.4 Supported commands

Appendix

- A Operating System Specific files
- B Mission, Local and Support Files Templates
- C Mission, Local and Support Files Examples
- D Sample General Management Files
- E Sample Resource Management Files
- F WitchDoctor-WDX messages
- G GNU General Public License
- H GNU Free Documentation License

Preface

This document contains instructions for configuring and using WitchDoctor-WDX on different operating systems and hardware platforms. Please refer to WitchDoctor-WDX Software product description Shaman-X-SPD-WDX10-E for an up to date list of supported environments and on-going projects.

In this document, operating systems which are mentioned refer to:

- GnuLinux on all kinds of platforms,
- True64UNIX on AXP platforms,
- OpenVMS on AXP platforms,
- Windows NT Version 4 on Intel processors.

Intended Audience

This guide is for System Managers or persons responsible for configuring the hardware and designing the shell scripts triggered by the WDX software. This person must have a good understanding of some or all of:

- High availability clustering environments
- GnuLinux, UNIX, OpenVMS and Windows NT environments,
- TCP/IP networks,
- WitchDoctor-WDX functionality.

The WitchDoctor-WDX Documentation Set

The WitchDoctor-WDX documentation set includes the following manuals:

- | | |
|---|---------------------------|
| - WitchDoctor-WDX Software product description | Shaman-X-SPD-WDX10-E |
| - WitchDoctor-WDX User's Guide | Shaman-X-UG-WDX10-E |
| - WitchDoctor-WDX Release Notes | Shaman-X-RL-WDX10-E |
| - WitchDoctor-WDX Installation Guide for GnuLinux | Shaman-X-LINUX-WDX10-E |
| - WitchDoctor-WDX Web GUI for GnuLinux | Shaman-X-LINUX-WDXGUI10-E |
| - WitchDoctor-WDX Installation Guide for Windows NT | Shaman-X-IGNT-WDX10-E |
| - WitchDoctor-WDX Installation Guide for UNIX RISC | Shaman-X-UNIX-WDX10-E |
| - WitchDoctor-WDX Installation Guide for OpenVMS | Shaman-X-IGVMS-WDX10-E |
| - WitchDoctor-WDX specification | Shaman-X-FS-WDX10-E |
| - WitchDoctor-WDX design | Shaman-X-DS-WDX10-E |
| - WitchDoctor-WDX acceptance test suite | Shaman-X-TS-WDX10-E |

Please check <http://www.shaman-x.org> for updates.

Purpose of this Guide

This guide describes WitchDoctor-WDX software and provides:

- an overview of the purpose of WitchDoctor-WDX software,
- WitchDoctor-WDX software capabilities,
- how to configure and use WitchDoctor-WDX.

This document is NOT about:

- High availability concepts, hardware setup, application failover mechanisms, data sharing, networks and file system issues. (please visit <http://www.linux-ha.org>)
- IT infrastructure modelling methodology and Shaman-X crisis management software suite. (please visit <http://www.shaman-x.org>)

Conventions

The following typographical conventions are used throughout this manual:

- **BOLD** is used in examples to indicate input that you type in, and for **commands** in the text.
- <CR> means you press the key named RETURN.
- <CTRL/C> Means you press and hold down the key named CTRL and at the same press the key named C.
- The hash sign # is the default superuser prompt.
- System name followed by > is your default user prompt.
- UPPERCASE, lowercase. Some operating systems differentiate between UPPERCASE and lowercase characters. You must type strings in the same case as they appear in the text.
- `system output`. This typeface indicates system output in interactive examples, code examples and screen displays.
- *variable*. This typeface indicates variable values in syntax descriptions and function descriptions, and to introduce new terms in the text.
- [] means you may omit characters between square brackets in commands.
- { } means that you must enter one of the command options given between braces.
- | separates command options given between braces { }.
- . . . in a syntax description or function definition means that you can repeat an item one or more times.

Operating Systems

This manual covers all the current versions of WitchDoctor-WDX for the different Operating Systems. Note that unless specified, the UNIX syntax is used in the examples of Operator Commands and Displayed Messages. In this manual the term "script" must be understood indifferently as UNIX-like "shell scripts", or command procedure files according to the Operating System used. The term "wdx process" is used to reference the WitchDoctor-WDX process for UNIX and GnuLinux, or its main detached process for OpenVMS and Windows NT.

1 WitchDoctor-WDX Presentation

1.1 Overview

WitchDoctor-WDX software runs on distributed configurations running the following operating systems:

- GnuLinux,
- Windows NT
- UNIX/RISC,
- OpenVMS.

Check www.shaman-x.org for new porting projects.

WitchDoctor-WDX runs on a system with one Ethernet controller. However, two independent Ethernet interfaces are required to benefit from the full functionality of WitchDoctor-WDX. WitchDoctor-WDX is used with groups of computers. A group is composed of 2 to 16 computers communicating across a local area network and collaborating to perform a critical application. A copy of WitchDoctor-WDX software must be installed on each computer in a group. A critical application and the description of the group of computers on which this application runs form a WitchDoctor-WDX mission.

WitchDoctor-WDX provides three main features for a mission (defined below):

- watchdog and failover recovery,
- software and hardware redundancy management (distributed lock management),
- mission maintenance facilities, stop and start

1.1.1.WitchDoctor-WDX Terminology

This section introduces terms and concepts used with WitchDoctor-WDX.

WitchDoctor-WDX is a software product designed to improve Client/Server applications availability by failure detection and automated application recovery. A copy of WitchDoctor-WDX software runs on each computer in a group. A group is composed of 2 to 16 computers communicating across a local area network and collaborating to perform a critical application. A critical application and the description of the group of computers on which this application runs, form a WitchDoctor-WDX mission. WitchDoctor-WDX can manage several missions simultaneously on several jointed or disjointed groups of computers to automate failover of several independent applications. Used on top of redundant or partially redundant hardware, WitchDoctor-WDX provides unattended operation and mission continuity.

WitchDoctor-WDX performs four different roles:

- Mutual watch
- Distributed lock management
- Masterholder/Substitute management
- Automated recovery

Features

Mutual Watch

For each mission, WitchDoctor-WDX periodically watches the different computers of the group. WitchDoctor-WDX detects CPU failures and network failures. Detection delay can be modified from 5 seconds to 2 minutes. When the group uses redundant networks, WitchDoctor-WDX differentiates network failures from computer failures. When the group includes more than 2 computers, WitchDoctor-WDX localises network failures by deduction.

Distributed Lock Management

For each mission, the group of computers can share a set of up to 24 resources.

A resource represents either:

- A physical device identifier such as disk, tape, line or peripheral
- An access path to data or NFS files
- Any virtual resource, flag, or event

Masterholder/Substitute Management

There are three different roles for computers in a mission: Masterholder, Substitute and satellite. WitchDoctor-WDX automatically assigns Masterholder and Substitute roles for each mission, conforming to user's preferences. A computer which is neither Substitute nor Masterholder is called a satellite computer.

Automated Recovery

WitchDoctor-WDX can automatically initiate standard command files on each computer in case of any state transition of

- computers
- networks
- resources

Command files include any application recovery action.

Other Features

- WitchDoctor-WDX is specifically designed to manage heterogeneous groups of computers. While computers do not necessarily run the same operating system, a compatible version of WitchDoctor-WDX must be present on each of them.
- WitchDoctor-WDX supports planned service interruptions for hardware maintenance or operating system upgrades. It allows on-line reconfiguration of applications and groups.
- WitchDoctor-WDX does not prevent erroneous access conflicts on shared peripherals. Additional safe locking mechanisms will be invoked in command files when necessary.

1.1.2 Watchdog and Failover Recovery

Watchdog and failover recovery enable a set of members to support each other, by exchanging wdx messages over the configured interfaces. When one member is detected as having failed, specific shell scripts or command procedure files can be activated on the surviving members in the group.

The triggering mechanism for these scripts provides notification for:

- member state transition,
- network state transition,
- operator initiated transition.

Operator commands are provided to support these features.

1.1.3 Software and Hardware Redundancy Management (Distributed Lock Management)

Software and hardware redundancy management includes the following capabilities:

- the provision of advisory resource locking. Each resource is a virtual representation of either full

- virtual objects or physical components,
- the provision of specific shell script (or command file) activation on resource state transition.

Operator commands are provided to support these features.

1.1.4 Mission Maintenance, Start, Stop and Pause

WitchDoctor-WDX 1.0 provides stop, start and pause commands for the control of WitchDoctor-WDX in a mission which facilitate hardware or software maintenance of this mission.

1.2 WitchDoctor-WDX Software Components

The WitchDoctor-WDX software consists of the following components on every member node of the mission:

- WitchDoctor-WDX executables (see Appendix A) which run either as detached or interactive processes.
- Additional specific components for each mission
 1. the mission file mission.cfg includes a member list which must be identical for all members of the mission.
 2. the local file mission.loc
 3. the optional support file
 4. the log file mission.log
 5. user-written script files called:
 - NEWMASTERSCRIPT file,
 - NOSUBSTITUTESCRIPT file,
 - SATELLITESCRIPT file,
 - EXCEPTIONSCSCRIPT file,
 - EVENTSCSCRIPT file.
 6. the design of a script that will start WitchDoctor-WDX.

1.3 WitchDoctor-WDX Software Description

1.3.1 Masterholder, Substitute and Satellite Concepts

For the mission, the Masterholder is defined as the member which holds the lock on the mastertoken. The mastertoken is a predefined symbolic string used as a virtual resource (see to Section 1.3.2 for information about the Locking Protocol).

Other members are either Substitutes or satellites:

- Substitutes are eligible for mastership.
- *Satellites* are not eligible for mastership.

The roles of Masterholder and Substitute for the mission are mutually exclusive. The Substitutes are watchdogs for the Masterholder's activity. When no Masterholder is available for the mission, all the Substitutes compete to get a lock on the mastertoken. The new Masterholder is the Substitute which gets the lock on mastertoken. Each Substitute asks for the mastership according to its priority_level defined in the mission file (see Chapter 2). When two Masterholders are detected for the mission, the Masterholder which holds the oldest lock on mastertoken releases the lock. The possible role for each member: Masterholder/Substitute or Satellite is defined in the mission file.

1.3.1.1 Masterholder/Substitute Capabilities

Masterholder and Substitutes always trigger scripts when transitions are detected by wdx.

Masterholder can trigger:

- NOSUBSTITUTESCRIPT
- EVENTSCRIPT
- EXCEPTIONSCRIPT

while Substitute can trigger:

- NEWMASTERSCRIPT
- EVENTSCRIPT
- EXCEPTIONSCRIPT

The triggering mechanism of EVENTSCRIPT on the various members is controlled in their local file. The EVENTSCRIPT is triggered on a MasterHolder or a Substitute only if the corresponding event transition is validated in its local file. See Section 1.3.4 for details. When the wdx process is started with Masterholder/Substitute capabilities, the following message is displayed:

"process supports Masterholder/Substitute features"

1.3.1.2 Satellite Capabilities

Satellites:

- are members not eligible for mastership,
- trigger scripts for transitions initiated by a Masterholder, Substitute or a Satellite.

The following scripts can be triggered by Satellite:

- SATELLITESCRIPT
- EVENTSCRIPT
- EXCEPTIONSCRIPT

Triggering conditions of the EVENTSCRIPT are controlled for each satellite in its local file. EVENTSCRIPTs are only triggered when the corresponding transition events are validated in this file. See Section 1.3.4 and Chapter 2.

1.3.1.3 Networks and Interface states

Members perform a watchdog function by exchanging periodic messages through communication interfaces. Periodic messages are called heartbeat messages, while the period is called broadcast time. Each member has a map of the network configuration in the mission file. (See Chapter 4). This map is called the network interface table. It contains the information and state for each network interface of each member. Using the heartbeat messages, members maintain the network interface table.

Network Interfaces States

<u>State of a network interface</u>	<u>Interpretation</u>
N/A	No link is defined in the map.
UNKNOWN	No valid message has been received from the members through this interface, but the failure cannot be localised or confirmed. This is the default state for an interface.
ON	A message has been received. The validity period of this interface has yet not expired.
OFF	No valid message has been received from the member through this interface, which is considered as failed.

1.3.1.4 Member State

<u>State of a member</u>	<u>Interpretation</u>
FAULT IDLE	No heartbeat message was received. The wdx process is idle, but not started (or running). It has not yet performed the "wdx [mission] -join", "wdx [mission] -run" , "wdx [mission] -start [snapshot]" or wdx [mission] -master commands (see Chapter 4). In this state, the member cannot perform either failover recovery mechanism or lock protocol.
SUBSTITUTE	The member is running as a Substitute. It has executed the "wdx[mission] -join", "wdx [mission] -run", "wdx [mission] -start [snapshot]" or "wdx [mission] -master" command but does not hold the mastertoken (see Chapter 4).
MASTERHOLDER SATELLITE	The member holds the mastertoken and thus is running as Masterholder. The member is running as a satellite. It has executed the "wdx [mission] -join", "wdx [mission] -run" or "wdx [mission] -start [snapshot]" command, but is not configured to become a Masterholder (see Chapter 2 and see Chapter 4).

1.3.2 Distributed Lock Management and Resources

1.3.2.1 Distributed Lock Management

The Distributed Lock Management provides the following features:

- a function can be assigned to each member according to its resource availability,
- when a failure is detected it re-distributes the functions.

The Distributed Lock Management includes a set of rules to perform these features on virtual strings called resources. These rules are:

- a lock is decided locally, based on the agreement of all members,
- a lock is unique,
- dynamic configuration is enabled,
- automatic reconfiguration is enabled by triggering the EVENTSCRIPT script file on resource state transition (see Section 1.3.4).

Operator commands are provided to support lock management (see Chapter 4).

1.3.2.2 Resources

Resources are entities used and controlled with the distributed lock management. A resource is a string which is a virtual representation of:

- either an abstract object, or
- a directory.

An abstract object is a virtual representation of:

- either a full virtual object, for example "PAYROLL",
- or a physical component, for example "DISK_RZ0".

A directory is a virtual representation of a mount point directory (UNIX and GnuLinux only), for example "/data". A

resource is defined by:

1. resource_name: an ASCII string from 1 to 20 characters.
2. resource_owner: a member which initiates a transition on the resource.
3. resource_state: the status of the resource.

WitchDoctor-WDX is able to manage up to 24 user resources. This does not include the mastertoken. The management of mastertoken is predefined.

1.3.2.2.1 Resource State

<u>State of a resource</u>	<u>Interpretation</u>
UNKNOWN	The resource has not been declared for the mission
FREE	The resource has been declared but is not locked by anymember
RESERVED/MOUNTED	The resource has been locked successfully by a member. The RESERVED state is used for abstract objects, while the MOUNTED state is used for mount point directories.
FAULT	The resource is not usable.

1.3.2.2.2 Resource Owner

<u>State of a resource</u>	<u>Owner</u>
UNKNOWN	Not applicable
FREE	Not applicable
RESERVED/MOUNTED	The member holding a lock on this resource.
FAULT	The member which has set the resource state to FAULT.

1.3.3 Configuration Files

1.3.3.1 Overview

The configuration files contain descriptive information about the static configuration of a mission. These files are:

- the mission file
- the local file
- the support file

For each mission, a copy of these files is required on each member of the mission. File characteristics are:

- all configuration files consist of ASCII text entries,
- a text editor is required for their update,
- a new line is used to delimit each entry,
- a null entry consists of an empty line,
- comments begin with #.

Note: For Windows NT, file name supports FAT and NTFS file systems. The directory location for the mission file and local file is fixed and imposed for each Operating System version of the WitchDoctor-WDX as follows:

- GnuLinux: /var/shaman-x/wdx
- OpenVMS: [WDXAUT]
- Windows NT:\WDX

1.3.3.2 Mission File

The mission file is a mandatory start up file for the wdx process. The content of this file must be identical on all members of the mission. It lists all the members, their roles in the mission, the priority for mastership, and time parameters. The naming convention for the mission file is mission.cfg where mission is the mission's name. If no mission name is specified in the command line (see Chapter 4) , the default mission file wdx.cfg is used.

1.3.3.3 Local File

The local file is a mandatory start up file for each wdx process and provides local parameters. This file gives the names of the scripts used on that member and contains the list of triggering conditions for the EVENTSCRIPT on that member based on transition events. Users can select and validate all the transition events to trigger their EVENTSCRIPT by modifying this list. The naming convention for the local file is mission.loc, where mission is the mission's name. If no mission name is specified in the command line when mission is started (see Chapter 4), the default local file wdx.loc is used .

1.3.3.4 Support File

The support file is an optional start up file for each wdx process and provides tuning capabilities of local parameters. This file is targetted for advanced users and developers.

1.3.4 Command Files

Applications are managed using command files which are triggered by WitchDoctor-WDX when it detects changes in the mission. These command files are defined by the user. They are in the form of shell scripts for UNIX and GnuLinux Systems, DCL command procedure files for OpenVMS, and executable and batch command files for Windows NT. The table below describes the different WitchDoctor-WDX command files and their triggering conditions:

<u>Script name</u>	<u>Triggering condition</u>	<u>Triggered on</u>
NOSUBSTITUTESCRIPT	Substitutes no longer available for the mission.	Masterholder
NEWMASTERSCRIPT	When it becomes the Masterholder.	Substitute
SATELLITESCRIPT	Masterholder failure or Masterholder has changed.	Satellite
EXCEPTIONSCRIPT	Operator command "wdx[mission] -exception".	All members
EXCEPTIONSCRIPT	Two Masterholders are detected for the mission.	Oldest Masterholder
EVENTSCRIPT	State transition for members, networks and resources.	All members

EVENTSCRIPTs are triggered only when the corresponding transition events are validated in the local file (see Chapter 4).

1.3.5 Mission Maintenance, Start, Stop and Pause

It is possible to stop a mission for maintenance of the computers without invoking automatic WitchDoctor-WDX, by using the stop command. The stop command stop [snapshot_file] stores the current (dynamic) state of the mission at the time it is stopped. When the mission is stopped all the members are in the IDLE state and they do not react to any event. The start [snapshot_file] command restarts the mission using the mission state stored with the stop [snapshot_file] command (see Chapter 3 and Chapter 4). With the pause [number_of_seconds] command, member activity is stopped during specified period of seconds. When this period expires, the wdx software restarts the different nodes taking account of the possible member or configuration modifications.

1.3.6 Log Files

WitchDoctor-WDX uses two log files for startup and stop messages:

- mission log file mission.log for dynamic events which occurred in a mission. This file is created at start-up time by the WitchDoctor-WDX main process, and the previous log file is renamed,
- fatal errors are reported to the operating system log files.

See Chapter 4 for log file management information.

Error Logging

All the events logged in the mission log file have the following format:

```
severity: date, message
```

where:

- severity can have one of the values:
 - INFO: information messages used to log all the state transition events,
 - WARNING: warning messages used to log non fatal error,
 - ERROR: fatal error messages.
- date is the date and time at which the message was logged. Date is expressed in local time.
- message is the text message reported by WitchDoctor-WDX

2 Wdx Configuration

2.1 Configuration Step-by-Step Procedure

Follow these steps to configure WitchDoctor-WDX:

1. install WitchDoctor-WDX software, using the WitchDoctor-WDX Get Ready document for your operating system.
2. define a mission name,
3. create the UDP port name on all computers,
4. configure your communication interfaces,
5. create the mission file and the local file on all computers,
6. create the mission script files on each computer,
7. design a shell script that will start WitchDoctor-WDX.
8. start the WitchDoctor-WDX main process manually or reboot all the computers.

2.2 Define a Mission Name

WitchDoctor-WDX allows you to run several missions on a single computer. Each mission is referenced by its name. Choose the name of your mission. The mission name is an ASCII string of up to 16 characters (up to 8 for a FAT file on a Windows NT system): This mission's name, for example mission, is used:

- to name the mission file: mission.cfg
- to name the local file: mission.loc
- to name the log file: mission.log
- to name the UDP communication port: mission (see Section 2.3.)
- to enter operator commands: wdx [mission] option (see Chapter 4).

2.3 UDP Service

Create a privileged UDP port for use by each WitchDoctor-WDX's mission. The UDP port name must be the mission name. For UNIX and GnuLinux, UDP ports must be created in the services file /etc/services. The following example shows the UDP ports for UNIX missions wdx, WEB-SERV and FIREWALL:

```
wdx                666/UDP                # WitchDoctor-WDX V1.0 : default mission
WEB-SERV           667/UDP                # WitchDoctor-WDX V1.0 : mission WEB-SERV
FIREWALL           668/UDP                # WitchDoctor-WDX V1.0 : mission FIREWALL
```

For OpenVMS, use UCX management command SET SERVICE to create a privileged UDP port : a new service must be defined in the Services Databases for the mission. The following example shows how to create an UDP port for OpenVms mission wdx:

```
UCX>      SET SERVICE wdx  -
          /PORT = 666  -
          /PROTOCOL = UDP  -
          /PROCESS_NAME = wdx  -
          /USER_NAME = SYSTEM  -
          /FILE = wdxaut_wdx_start.com
          where wdxaut_"mission"_start.com is the name of the service's startup command file.
```

This UDP port must be defined on all WitchDoctor-WDX systems.

2.4 Interfaces

When configuring the interfaces, consult the following manuals:

- for GnuLinux: the "TCP/IP How-TO",
- for UNIX: the "UNIX Guide to Networking" or equivalent from your system provider,
- for OpenVMS: the "DEC TCP/IP Services for OpenVMS Installation and Configuration Guide",
- for Windows NT : the "System Guide for Microsoft Windows NT".

Dual Ethernets require the two interfaces to be configured in two distinct internet networks. Although subnets are supported, we recommend using two distinct networks whenever possible. In a configuration with two Ethernets, the interfaces connected to the first segment must have the same subnet address, with different host addresses. The broadcast and netmask value must be identical.

2.5 Mission Configuration File

A template for the configuration file, called `wdx.cfg` is provided with the WitchDoctor-WDX kit. Make a copy of this file using the naming convention (refer to Section 2.2). Edit the file to configure your mission on each computer that is part of the mission. The mission file includes date, software functional level version and member entries. All entries must be identical on all members of the mission.

2.5.1 Version and Date line

The mandatory format keyword value for "VERSION" is:

```
VERSION X.Y
```

where X and Y each represent 1 digit.

The VERSION line must be the same on all the members of the mission. It defines the compatibility level of all WDX processes of the mission and allows different wdx versions to interoperate. It must be 1.0 for WitchDoctor-WDX 1.0.

The format for "DATE" is a zone of up to 20 characters:

```
DATE dd-mmm-yyyy
```

The 20 characters are a free string to identify the version of the mission configuration. It must be the same on all members.

2.5.2 Member Characteristics Entries

The format of "member characteristics" lines is:

```
member name type priority broadcast validity redundancy network_1 network_2
```

where:

- member is a keyword,
- name: the name of the member is either any name or hostname.
- type: wdx process is configured to provide either Masterholder/Substitute or Satellite features only.
- The keyword for Masterholder/Substitute is: MASTER/SUBSTITUTE
- the keyword for Satellite is: SATELLITE
- priority: This parameter defines the priority level of each member in the mission when there is a MASTERTOKEN request. The member with the highest priority is considered as the preferred member to be started as the Masterholder during the MASTERTOKEN request. The parameter is valid only for the Masterholder/Substitute member type. The highest priority value is zero
- broadcast: wdx process periodically broadcasts information. Broadcast time determines the period of the broadcast in seconds. Broadcast must be a positive integer value. There is no default or

maximum value. The minimum value is 3 seconds.

- validity: Each broadcast has a validity period attached. If the validity period has expired, the broadcast is considered as invalid. The validity period is expressed in seconds. The validity period must be greater than broadcast time; the minimum value is (broadcast time + 1) seconds. A useful value for validity is (broadcast time x 3) seconds. There is no default.
- redundancy: Each message is sent (redundancy) times on each configured interface in order to tolerate a single message loss on the interface(s). Redundancy must be a non-zero integer value. There is no default.
- network_1 or network_2: This is the name of a configured Ethernet interface supporting TCP/IP. For example, "In0". If no Ethernet interface is defined for the network, "N/A" is required.

In a line, each element is MANDATORY.

2.6 Local File

A template for the local file, called wdx.loc is provided with the WitchDoctor-WDX kit. Make a copy of this file using the naming convention (see Section 2.2). Edit the file to configure your mission on each computer that is part of the mission. The local file contains the following information:

- configuration version
- date of the last modification
- script entries
- transition line

The local file is not necessarily the same on all the members running the same mission. The only restriction is that the configuration version must be the same on all the members for the mission.

2.6.1 Version and Date line

The obligatory format keyword value for "VERSION" is:

```
VERSION X.Y
where X and Y each represent 1 digit.
```

This version is used to check mission consistency and insure that all members share the same configuration release. Therefore, the VERSION line must be the same on all the members of the mission.

The format for "DATE" is a zone of up to 20 characters:

```
DATE dd-mmm-yyyy
```

The date zone is a free string used to identify the release of the file.

2.6.2 Script Files Path Name Entries

The format of "script file path name" line is:

```
name value
```

where:

- name can be one of:
 NOSUBSTITUTESCRIPT,
 NEWMASTERSCRIPT,
 SATELLITESCRIPT,
 EXCEPTIONSCRIPT,
 EVENTSCRIPT.

- value is the script file path name.

The full path name must be given for the scripts, which can be located anywhere. There is no default directory for the scripts. An erroneous path given for a script is transparent for the wdx, which will not generate any error during triggering attempt. However, error messages will be reported by the operating system.

2.6.3 Transition line

The transition line is used to enable or disable the EVENTSCRIPT activation for a specific event. By default all the possible events triggered in the EVENTSCRIPT are enabled. When you want to disable an event, replace the "YES" value by "NO". For example, to disable the EVENTSCRIPT activation for the NETWORK events, change the values in the local file as follows:

```
NETWORK_UNKNOWN_TO_ON           NO
NETWORK_UNKNOWN_TO_OFF          NO
NETWORK_ON_TO_OFF                NO
NETWORK_ON_TO_UNKNOWN            NO
NETWORK_OFF_TO_UNKNOWN           NO
NETWORK_OFF_TO_ON                NO
```

2.7 Support file

It is an optional start up file for wdx process. This file being optional each line has compiled-in value. The contents of this file is local to a member, and can differ from one member to another. The naming convention for the support_file is mission.sup where mission is the mission's name.

The contents of this file is :

```
<line> ::= <support_parameter> <support_value>
<support_parameter> ::= LOG_FLAG |
                        LOG_FILE
                        PRIORITY
                        PERIOD
                        ACTIVITY
                        THRESHOLD
                        STATISTICS
<support_value> ::= any value defined for the support parameter
```

The meaning of each "support parameter" is :

- LOG_FLAG : indicates logging capabilities which are :
 - DISABLE : specify that logging capabilities are disable. The value is "DISABLE".
 - ENABLE : specify that logging capabilities are enable. The value is "ENABLE".
- LOG_FILE : indicates the log file name. If the logging capabilities are disable, this field has not significant.
- PRIORITY : indicates the wdx process priority in terms of process priority levels. A low priority assigned to wdx allows to detect heavy load periods.
- PERIOD : indicates the internal polling value used by the wdx process. The value is a non null positive integer in seconds.
- ACTIVITY : indicates if the internal activity algorithm is enabled. The value is :
 - DISABLE : algorithm is disabled.
 - ENABLE : algorithm is enabled.

THRESHOLD : indicates that the threshold algorithm is enabled for the member specified as value. Only one threshold per member is allowed. Value indicates also the threshold level in seconds as follows :

THRESHOLD member_name level_in_seconds

STATISTICS : indicates that statistics feature is enable. The value indicates that statistics are provided every "value" seconds.

The default compiled-in values for the "log line" are :

LOG_FLAG :	ENABLE
LOG_FILE :	mission.log
priority	highest
polling (seconds)	1
activity :	DISABLE
threshold :	none
statistics :	DISABLE

2.8 Script Files

Follow these steps to write the script files:

1. Decide which actions will be performed when the Masterholder system becomes unavailable: write the corresponding NEWMASTERSCRIPT (and SATELLITESCRIPT if required)
2. Decide which actions will be performed when the Substitute systems become unavailable: write the corresponding NOSUBSTITUTESCRIPT.
3. Decide which resources your distributed system will use; these resources will have to be managed when starting WitchDoctor-WDX and when a resource state transition occurs.
4. Decide which actions will be performed, and write the corresponding EVENTSCRIPT, when the following events occur:
 - network state transition,
 - member state transition,
 - resource state transition.
5. Design a shell script, or a command file that will start WitchDoctor-WDX. Request for mastership and start your application according to the returned status. Declare and reserve the resources managed by your distributed system. Edit your system startup file to call this script during startup.

Sample scripts are shown in Appendix D.

2.8.1 NOSUBSTITUTESCRIPT

NOSUBSTITUTESCRIPT is the name of the script triggered on the Masterholder when it detects that Substitutes are no longer available for the mission. No argument is passed to this script. Sample scripts are shown in the appendices.

2.8.2 NEWMASTERSCRIPT

NEWMASTERSCRIPT is the name of the script triggered on the Substitute that becomes the new Masterholder after Masterholder failure. No argument is passed to this script. Sample scripts are shown in the appendices.

2.8.3 SATELLITESCRIPT

SATELLITESCRIPT is the name of the script triggered on a Satellite when the wdx process detects one of the following conditions:

1. no valid message received from Masterholder through any configured interface (up to two),
2. The Masterholder has changed.

One argument is passed to the script to indicate the reason for the exception:

- MASTERDISAPPEARENCE: when the Masterholder has failed,
- MASTERCHANGE: when the Masterholder has changed.

2.8.4 EXCEPTIONSCRIPT

EXCEPTIONSCRIPT is the name of the script triggered on all members of the wdx mission with the operator interface command "wdx [mission] -exception". Member states are not affected by this script activation. It is also triggered on the oldest Masterholder (only, not on all members) if two Masterholders are running, and the oldest Masterholder becomes Substitute. One argument is passed to the script to indicate the reason for the exception:

- OPERATOR: "wdx [mission] -exception" has been entered,
- DOUBLEMASTER: double Masterholder has been detected.

2.8.5 EVENTSCRIPT

EVENTSCRIPT is the name of the script triggered when a state transition has been detected. The EVENTSCRIPT is triggered on an event only if the corresponding "TRANSITION LINE" is enabled. The call syntax is as follows:

```
eventscript keyword [arguments...]
```

The first argument is a keyword indicating the kind of transition reported. Depending on the value of the first argument, the script will be able to parse the other arguments (if any). The first argument can take one of the following values:

1. MEMBER: a member state transition has been detected. The other arguments passed to the script are:
 - member_name indicates the member name that is affected by the transition,
 - old_state indicates the old state of the member that is affected by the transition,
 - new_state indicates the new state of the member that is affected by the transition.
 Member states are either FAULT, IDLE, Masterholder, Substitute or Satellite.
2. NETWORK: a network state transition has been detected. The other arguments passed to the script are:
 - member_name indicates the member name that is affected by the transition,
 - network_number indicates the network that is affected by the transition. This value is either "1" or "2". This value is issued from the mission file,
 - interface_name indicates the name of the interface on which the transition has been detected. This name is found in the mission file,
 - old_state indicates the old state of the interface that is affected by the transition,
 - new_state indicates the new state of the interface that is affected by the transition.
 Network states are UNKNOWN, ON or OFF.
3. RESOURCE: a resource state transition has been detected. The other arguments passed to the script are:
 - resource_name indicates the resource's name that is affected by the transition,
 - old_state indicates the old state of the resource that is affected by the transition,
 - new_state indicates the new state of the resource that is affected by the transition,
 - initiator_name indicates the name of the transition's initiator.
 Resource states are either UNKNOWN, FREE, RESERVED, MOUNTED or FAULT.
4. MISMATCH: This indicates that the eventscript is triggered because of persistent owner duplication for a resource.
 - resource_name indicates the name of the resource that is affected by the mismatch.

- owner_name indicates the name of the resource's owner which should release its lock.

Sample scripts are shown in the appendix.

3 WitchDoctor-WDX Management

3.1 Prerequisites

Before starting WitchDoctor-WDX verify that:

1. networking TCP/IP is enabled and correctly configured,
2. the WitchDoctor-WDX license is loaded and valid,
3. all the computers are running the same WitchDoctor-WDX version,
4. the mission file is correctly distributed on each computer of the mission,
5. the local file is correctly defined on each computer of the mission,
6. the system clocks are synchronized.
7. for VMS, the wdx symbol for the user interface is correctly defined:

```
wdx:==$wdxaut_dir:wdxaut_ui.exe
```

3.2 Restrictions

The following restrictions apply to WitchDoctor-WDX:

- all the members in the mission must run the same WitchDoctor-WDX version,
- all the members in the mission must have an identical mission file,
- the maximum number of members in the WitchDoctor-WDX mission is limited to 16,
- the maximum number of user resources that can be managed in a mission is limited to 24,
- the string length of a resource or a "mount point directory" name must not exceed 20 characters,
- Broadcast and validity time parameters do not have default and maximum values.
- the latency time for transition detections of WitchDoctor-WDX depends on the validity of each member.

3.3 Networking Management

3.3.1 TCP/IP

Check that your networking TCP/IP is enabled, using the commands:

- UNIX GnuLinux: `nestart -i`
- OpenVMS: `ucx> show service`
- WINDOWS NT: Use the Network Dialog Box in the Control Panel

3.3.2 WitchDoctor-WDX Interface Configuration

If two communication interfaces are used for your WitchDoctor-WDX mission, check these two interfaces are correctly declared in the mission configuration file. Verify that interfaces connected on the same segment have the same broadcast and netmask value in the same network or subnet (see Section 2.4).

3.4 WitchDoctor-WDX Version

Check that all your computers are running the same WitchDoctor-WDX version. Use the command `wdx -version` to display the software version.

3.5 Configuration Files

To perform a mission, the wdx process must have a mission file and a local file. The mission file must be identical

on all members of the mission.

Note:

If one wdx process is started with a different mission file, the mission stops on all the computers. The following message is logged in the mission log file:

```
ERROR: Message received with invalid configuration file version
```

A snapshot file mission.hot is created.

3.6 System Clock Synchronisation

To perform the mission, the computers have to be synchronized, so that in the event of two Masterholders, the wdx process can decide which is the oldest. For WINDOWS NT configurations refer to Chapter 5.

3.7 Mission Maintenance, Start, Stop and Pause

The Stop and Start commands, with the IDLE state, provide "cold and warm start" of the mission and thus allow maintenance of the mission. Cold Startup allows all the IDLE members of a mission to start from scratch (as if they have never run before). Cold startup is performed by entering the wdx -run operator command on a member of the mission. Warm startup allows all the members of a mission to start from a previously recorded state. A typical sequence for a warm startup situation is:

- the operator issues the stop command,
- the WitchDoctor-WDX records the state of the mission in a snapshot file (one snapshot file per member),
- all wdx processes are stopped in the mission; all the members are placed in the IDLE state.

The configuration of members or of the mission can now be modified. Member systems can be added or removed from the mission, stopped and restarted, etc, without the WitchDoctor-WDX actions that would modify the state of the mission.

- the operator issues the start snapshot command,
- the wdx process restarts on each member,
- all members return to the state they had when the mission was stopped, and which was saved into the snapshot file.

During the startup:

- configuration files are scanned so a different role can be attributed to a member. In this case the member releases all the resources it had locked.
- If a member is no longer defined in the mission file the wdx process exits, after releasing its locks.

With the Pause command, member activity is stopped during specified period of seconds (IDLE state). During this period, the configuration of members or of the mission can be modified. When this period expires, the wdx process comes back to the running status, taking account of the possible member or configuration modifications.

4 WitchDoctor-WDX Operator Interface

The following section contains a full description of the WitchDoctor-WDX commands and options.

4.1 Starting the wdx process

On each operating system, a command or procedure is defined to start the wdx process.

Syntax

- UNIX or GnuLinux : wdx [mission_name] -daemon [member_name]
- OpenVMS : @sys\$startup:wdxaut_startup.com [mission_name] [member_name]
- Windows NT : wdx_main [mission_name] [member_name])

The [mission_name] and [member_name] arguments are optional. If no [mission_name] is specified, the default mission name is wdx. The [mission_name] argument corresponds to the name of the mission defined in the configuration file. If no [member_name] is specified, the default member name is the TCP/IP hostname. The [member_name] argument corresponds to the member name specified in the mission file.

Only one wdx process at any time can be present per node for a mission.

4.2 wdx Command Format

The format of a command is:

- for UNIX/GnuLinux: wdx [mission] -option
- for OpenVMS : Define a (wdx) symbol that reports to the executable image of the wdx operator interface (see Section 3.1). wdx [mission] /option
- for Windows NT : wdx [mission] /option

where mission is the name of the mission, and options are a set of commands to manage the following functions:

- mission,
- process,
- mastership,
- resource,
- directory,
- information.

The chosen option can be abbreviated in commands. The software performs a check to avoid ambiguity between commands, and ambiguous options are rejected if they are too short.

4.2.1 Mission

This is an optional argument. If the argument mission is in the command line, then wdx refers to this mission defined by:

- the configuration files: mission.cfg, mission.loc,
- the log file,
- the UDP service used for communication: mission/udp

The mission name must not exceed 16 characters. If no mission argument is in the command line, then wdx command refers to the mission defined by:

- the configuration file: wdx.cfg, wdx.loc,
- the log file,
- the UDP service used for communication: wdx/udp

4.2.2 Options

The options of wdx must be used singly. If no option is entered, wdx displays a list of the available options. On success, each wdx command returns a success status as defined by the operating system. Otherwise an error status is returned.

4.3 Mission Management Commands

The Mission Management commands are distributed commands. When entered on a system, this command is processed on all the wdx processes.

4.3.1 Command: run

Syntax:

- UNIX/GnuLinux: wdx [mission] -run
- OpenVMS: wdx [mission] /run
- Windows NT: wdx [mission] /run

Set all wdx processes to the RUNNING state. Each member will become either Substitute or Satellite according to its role. If no Masterholder is in the mission the member which becomes the Masterholder for the mission, is the Substitute which has the highest priority.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.3.2 Command: kill

Syntax:

- UNIX/GnuLinux: wdx [mission] -kill
- OpenVMS: wdx [mission] /kill
- Windows NT: wdx [mission] /kill

Kill all wdx processes for the corresponding mission when the mission name is specified. It is useful for software maintenance of WitchDoctor-WDX for a running mission.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.3.3 Command: stop [snapshot]

Syntax:

- UNIX/GnuLinux: wdx [mission] -stop [snapshot]
- OpenVMS: wdx [mission] /stop [snapshot]
- Windows NT: wdx [mission] /stop [snapshot]

Set all wdx processes to IDLE state. Member activity is stopped after registration of a snapshot for the mission. When used with the start command, the stop command is useful to initiate a Warm Startup procedure. The name of the snapshot is an optional argument, if no snapshot argument is passed, then the mission name is used by default. The snapshot name length is limited to 8 characters. The snapshot file is located in the wdx directory, and has a .hot extension. This command also creates an ascii file snapshot.txt which contains a text copy of the displays of a list and of a version commands. This file includes a date and a time stamp.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X1000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error4-6

4.3.4 Command: start [snapshot]

Syntax:

- UNIX/GnuLinux: wdx [mission] -start [snapshot]
- OpenVMS: wdx [mission] /start [snapshot]
- Windows NT: wdx [mission] /start [snapshot]

Force all wdx processes to restart from the registered snapshot. This warm startup requires that all wdx processes of the mission be in IDLE state. The name of the snapshot is passed as an optional argument to the command line. If no snapshot argument is specified then the mission name is taken as snapshot. The snapshot name length is limited to 8 characters. When a start command is initiated, the wdx software restarts the different nodes according to the state and the locks on the resources that they had before the stop command. The information on the states and the locks are taken from the snapshot file. During a start procedure, the mission file are scanned; if a member no longer belongs to a mission, its resource locks are released and the wdx process of the member corresponding to this mission is cancelled. This feature is useful when modifying the wdx configuration (adding or removing a member) without disturbing the whole configuration.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X1000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.3.5 Command: pause number_of_seconds

Syntax:

- UNIX/GnuLinux: wdx [mission] -pause number_of_seconds
- OpenVMS: wdx [mission] /pause number_of_seconds
- Windows NT: wdx [mission] /pause number_of_seconds

Set the wdx processes to IDLE state. Member activity is stopped during specified period of seconds (number_of_second). During this period the configuration of members or of the mission can be modified. When this period expires, the wdx software restarts the different nodes taking account of the possible member or configuration modifications. If a member no longer belongs to a mission, its resource locks are released and the wdx process of the member corresponding to this mission is cancelled. This command is useful when some modifications must be done or tried without any consequence on the running applications. The wdx process is inactive during a specific period of time.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error4-8

4.3.6 Command: exception

Syntax:

- UNIX/GnuLinux: wdx [mission] -exception
- OpenVMS: wdx [mission] /exception
- Windows NT: wdx [mission] /exception

Initiate the triggering of the EXCEPTIONSCRIPT on all the members of the corresponding mission, without changing the member state. The OPERATOR argument is passed to the script in order to determine if the EXCEPTIONSCRIPT activation is due to a exception command or to a doublemaster detection.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.4 Process Management Commands

The Process Management commands are used to locally control the different wdx missions that can be started on a member.

4.4.1 Command: join

Syntax:

- UNIX/GnuLinux: wdx [mission] -join
- OpenVMS: wdx [mission] /join
- Windows NT: wdx [mission] /join

Set the local wdx process to the RUNNING state. The member will become either Substitute or Satellite according to its role. You do not need to issue a join command after a successful request for mastership.

Note

After a join command, if the member becomes Substitute when no Masterholder exists for the mission, that Substitute member will then become the Masterholder for this mission.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.4.2 Command: leave

Syntax:

- UNIX/GnuLinux: wdx [mission] -leave
- OpenVMS: wdx [mission] /leave
- Windows NT: wdx [mission] /leave

For a particular mission on a member, kill the corresponding wdx process.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.5 Masterholder/Substitute Management Commands

Each command is described in the following pages.

4.5.1 Command: master

Syntax:

- UNIX/GnuLinux: wdx [mission] -master
- OpenVMS: wdx [mission] /master
- Windows NT: wdx [mission] /master

The master command is used to declare the Masterholder member of the mission. When the member becomes Masterholder, the NEWMASTERSCRIPT is triggered. This command can be initiated only by the nodes that support the Masterholder/Substitute feature. You can have only one Masterholder for a mission (except when you have a network partitioning), so your request is rejected if a Masterholder member already exists for your mission or if the same request is currently processing on an other member.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

Restriction:

Before using the command master, ensure that all the members in the mission have an identical mission file.

4.5.2 Command: nomaster

Syntax:

- UNIX/GnuLinux: wdx [mission] -nomaster
- OpenVMS: wdx [mission] /nomaster
- Windows NT: wdx [mission] /nomaster

This option is used to release the lock on the mastertoken. The Masterholder member will return in the Substitute state with the lowest priority. The other Substitutes in the mission will compete for the mastertoken to become Masterholder.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.5.3 Command: getmaster

Syntax:

- UNIX/GnuLinux: wdx [mission] -getmaster
- OpenVMS: wdx [mission] /getmaster
- Windows NT: wdx [mission] /getmaster

When used by a Substitute member, the getmaster command forces a Substitute to become Masterholder. The getmaster command initiates a doublemaster state. The previous Masterholder returns to the Substitute state, after triggering the EXCEPTIONSCRIPT locally. In this case, the argument passed to the EXCEPTIONSCRIPT is DOUBLEMASTER.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6 Resource and Directory Management Commands

The following commands are used to manage the WitchDoctor-WDX resources lock mechanism.

The commands are:

- declare
- erase
- reserve
- mount
- free
- fault

- getresource
- nfs
- wait

These commands control virtual resources and directories. Some commands are for both resource and directory management, and others for directory management only. Until you enter a mount command, your directory declaration is considered as a resource declaration. The possible state for a resource or directory are the following:

- UNKNOWN
- FREE
- RESERVED
- MOUNTED
- FAULT

These argument are passed lowercase.

4.6.1 Command: declare res_name

Syntax:

- UNIX/GnuLinux: wdx [mission] -declare res_name
- OpenVMS: wdx [mission] /declare res_name
- Windows NT: wdx [mission] /declare res_name

This option is used to declare a virtual resource or mount point directory for the corresponding mission. The variable res_name is mandatory, and represents either a resource name or amount point directory. The variable res_name must not exceed 20 characters. When you declare a resource or a mount point, its state becomes FREE. The declare command is rejected when the resource is in one of the following states:

- RESERVED
- MOUNTED
- FAULT

The following example are for UNIX or GnuLinux systems.

```
# wdx mission_1 -declare resource_1
# wdx mission_1 -declare /mnt
```

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.2 Command: erase res_name

Syntax:

- UNIX/GnuLinux: wdx [mission] -erase res_name
- OpenVMS: wdx [mission] /erase res_name
- Windows NT: wdx [mission] /erase res_name

This option is used to release the declaration of a resource or a mount point directory. The variable res_name is

mandatory, and represents either a resource name or a mount point directory. The variable `res_name` must not exceed 20 characters. When you erase a resource or a mount point, its state becomes "UNKNOWN". The name of the resource or the name of the mount point directory is removed from the WitchDoctor-WDX tables. The erase command is rejected when the resource is in one of the following states:

- RESERVED
- MOUNTED
- FAULT

The following example are for UNIX or GnuLinux systems.

```
# wdx mission_1 -erase resource_1
# wdx mission_1 -erase /mnt
```

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.3 Command: reserve res_name

Syntax:

- UNIX/GnuLinux: wdx [mission] -reserve res_name
- OpenVMS: wdx [mission] /reserve res_name
- Windows NT: wdx [mission] /reserve res_name

This option is used to put an exclusive lock on a virtual resource. The variable `res_name` is mandatory, and represents in this case a resource. The variable `res_name` must not exceed 20 characters. When a resource is RESERVED, its state becomes RESERVED. The reserve command is accepted only if the resource is in the FREE state.

The following example are for UNIX or GnuLinux systems.

```
# wdx mission_1 -declare resource_1
# wdx mission_1 -reserve resource_1
```

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.4 Command: mount directory

This command is available only on UNIX/RISC or OSF/AXP systems.

Syntax:

- UNIX/GnuLinux: wdx [mission] -mount res_name

This option is used to put an exclusive lock on a mount point directory. The name of the directory is mandatory, and must not exceed 20 characters. The wdx process locks a mount point directory, only if this directory has been

locally MOUNTED with the mount command. After a successful mount command the new state for the directory is MOUNTED. If the directory is not locally MOUNTED, an error is returned:

```
directory not locally MOUNTED.
```

When the mount point is unMOUNTED (command umount) the lock on this directory is automatically released. The mount point returns in the FREE state. This option is useful to inform all the nodes in the mission that a file system is now available for NFS operations. The mount command is accepted only if the mount point is in the FREE state.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 = command failed , -1 syntax error

4.6.5 Command: free res_name

Syntax:

- UNIX/GnuLinux: wdx [mission] -free res_name
- OpenVMS: wdx [mission] /free res_name
- Windows NT: wdx [mission] /free res_name

This option is used to release the exclusive lock on a virtual resource or on the mount point directory. The variable res_name is mandatory, and represents either a resource name or a mount point directory. The variable res_name must not exceed 20 characters. The FREE command is RESERVED to the owner of the resource. You become the owner when you do the following on the resource or on the mount point directory:

- reserve the resource : reserve command
- mount the directory : mount command
- declare the resource or the mount point as FAULT: fault command.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 = command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.6 Command: fault res_name

Syntax:

- UNIX/GnuLinux: wdx [mission] -fault res_name
- OpenVMS: wdx [mission] /fault res_name
- Windows NT: wdx [mission] /fault res_name

This option is used to declare in FAULT state a resource or a mount point directory. The variable res_name is mandatory, and represents either a resource name or a mount point directory. The variable res_name must not exceed 20 characters. This option is useful to indicate that the resource or a mount point directory is unavailable. You can declare a resource or a mount point directory as FAULT when it is in one of the following states:

- UNKNOWN
- FREE
- RESERVED
- MOUNTED

When you enter the fault command for an unknown resource, the resource will be automatically declared in the wdx table with the "FAULT" state.

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.7 Command: getresource res_name

Syntax:

- UNIX/GnuLinux: wdx [mission] -getresource res_name
- OpenVMS: wdx [mission] /getresource res_name
- Windows NT: wdx [mission] /getresource res_name

This option is used to force the lock on a resource, even if another member holds a lock on this resource. The variable res_name is mandatory, and represents a resource name. The variable res_name must not exceed 20 characters. This command is not supported when it is issued for a mount point directory. When a member in the mission initiates this command, the resource owner releases the lock on the resource, and the initiator keeps the lock of this resource. This command is valid only when the resources are in one of the following states:

- FREE
- RESERVED
- FAULT

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.8 Command: nfs directory

Syntax:

- UNIX/GnuLinux: wdx [mission] -nfs directory
- OpenVMS: wdx [mission] /nfs directory
- Windows NT: wdx [mission] /nfs directory

When you initiate this command, you are blocked until a member in the mission initiates a mount command for the mount point directory that you specified in your nfs command. When the command completes, a string is returned with the following format:

```
membername:directory
```

where membername is the name of the member which has declared as MOUNTED the mount point directory, and directory is the mount point directory. If the mount point directory is already in the MOUNTED state before you initiate the nfs command, the string is returned immediately.

Note

If you initiate this command on an unknown mount point directory, you will be blocked until a member in the mission declares and mounts this directory.

Example on GnuLinux:

```
Three nodes in a mission MISSION_1:
```

```
Node NODE_1 is Masterholder
Node NODE_2 is Substitute
Node NODE_3 is Satellite.
```

```
The Masterholder node mounts the file system /mnt:
#mount /dev/rz3c /mnt
```

```
The Masterholder node locks the /mnt directory, in order to inform all the
members of the mission MISSION_1 that the /mnt directory is available:
```

```
#wdx MISSION_1 -mount /mnt
The /mnt directory is in MOUNTED state.
```

```
The Satellite node wants to mount the NFS directory /mnt, but does not know on
which node the directory has been mounted.
```

```
#wdx MISSION_1 -nfs /mnt
Wdx returns the following character strings
NODE_1:/mnt
```

```
The Satellite is now able to mount the NFS directory /mnt. If NODE_1 is a
hostname:
```

```
#mount NODE_1:/mnt /tmp
```

```
If member names are hostnames, then the operation can be directly performed by
entering:
```

```
#mount "wdx MISSION_1 -nfs /mnt" /tmp
```

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X1000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.6.9 Command: wait resource

Syntax:

- UNIX/GnuLinux: wdx [mission] -wait resource
- OpenVMS: wdx [mission] /wait resource
- Windows NT: wdx [mission] /wait resource

When you initiate this command, you are blocked until a member in the mission initiates a wdx RESERVE command for the resource that you specified in your wait command. When the command completes, a string is returned with the following format:

```
membername:resource
```

where membername is the name of the member which has declared the resource as RESERVED, and resource is the name of the resource. If the resource is in the RESERVED state before you initiate the wait command, the string is returned immediately.

Note

If you initiate this command on an unknown resource you will be blocked until a member in the mission declares and reserves this resource.

4.7 Information Commands

These commands are used to obtain local status information of your missions running on a member.

4.7.1 Command: version

Syntax:

- UNIX/GnuLinux: wdx [mission] -version
- OpenVMS: wdx [mission] /version
- Windows NT: wdx [mission] /version

Reports the values of important global variables of wdx. Some of the variables are adjustable values read from configuration file, others are compiled-in constants. The option is useful to see the limits of the current WitchDoctor-WDX implementation.

The following example is for GnuLinux:

```
Global variables for mission "wdx":
(Item marked with (*) are configurable in mission and local files)
Software version                = V1.0
Configuration file(*)           = V1.0 - 8-Jan-2003
Configuration checksum          = 1493838371
Scheduler period                = 1 second(s)
Process priority                = -20
Member name(*)                  = bounaberdix
Member type (*)                 = Masterholder - Substitute
MASTERTOKEN priority(*)        = 1
Broadcast time(*)              = 5
Validity time(*)               = 12
Redundancy(*)                  = 1
Interface 1(*)                  = ln0
Interface 2(*)                  = ln1
UDP service name                = wdx
NOSUBSTITUTESCRIPT(*)         = /var/shaman-x/wdx/wdx.nosubstitutescript
NEWMASTERSCRIPT(*)            = /var/shaman-x/wdx/wdx.newmasterscript
SATELLITESCRIPT(*)            = /var/shaman-x/wdx/wdx.satellitescript
EXCEPTIONSCRIPT(*)            = /var/shaman-x/wdx/wdx.exceptionscript
EVENTSCRIPT(*)                 = /var/shaman-x/wdx/wdx.eventsript
MEMBER_FAULT_TO_IDLE (*)       = YES
MEMBER_FAULT_TO_MASTER (*)     = YES
MEMBER_FAULT_TO_SUBSTITUTE (*) = YES
MEMBER_FAULT_TO_SATELLITE (*) = YES
MEMBER_IDLE_TO_FAULT (*)      = YES
MEMBER_IDLE_TO_MASTER (*)     = YES
MEMBER_IDLE_TO_SUBSTITUTE (*) = YES
MEMBER_IDLE_TO_SATELLITE (*)  = YES
MEMBER_MASTER_TO_FAULT (*)    = YES
MEMBER_MASTER_TO_IDLE (*)     = YES
MEMBER_MASTER_TO_SUBSTITUTE (*) = YES
MEMBER_MASTER_TO_SATELLITE (*) = YES
MEMBER_SUBSTITUTE_TO_FAULT (*) = YES
MEMBER_SUBSTITUTE_TO_IDLE (*) = YES
MEMBER_SUBSTITUTE_TO_MASTER (*) = YES
MEMBER_SUBSTITUTE_TO_SATELLITE (*) = YES
MEMBER_SATELLITE_TO_FAULT (*) = YES
```

```

MEMBER_SATELLITE_TO_IDLE (*)           = YES
MEMBER_SATELLITE_TO_MASTER (*)         = YES
MEMBER_SATELLITE_TO_SUBSTITUTE (*)     = YES
NETWORK_UNKNOWN_TO_ON (*)              = YES
NETWORK_UNKNOWN_TO_OFF (*)             = YES
NETWORK_ON_TO_UNKNOWN (*)              = YES
NETWORK_ON_TO_OFF (*)                  = YES
NETWORK_OFF_TO_UNKNOWN (*)             = YES
NETWORK_OFF_TO_ON (*)                  = YES
RESOURCE_UNKNOWN_TO_UNKNOWN (*)        = YES
RESOURCE_UNKNOWN_TO_FREE (*)           = YES
RESOURCE_UNKNOWN_TO_FAULT (*)          = YES
RESOURCE_UNKNOWN_TO_RESERVED (*)       = YES
RESOURCE_UNKNOWN_TO_MOUNTED (*)        = YES
RESOURCE_FREE_TO_UNKNOWN (*)           = YES
RESOURCE_FREE_TO_FREE (*)              = YES
RESOURCE_FREE_TO_FAULT (*)             = YES
RESOURCE_FREE_TO_RESERVED (*)          = YES
RESOURCE_FREE_TO_MOUNTED (*)          = YES
RESOURCE_FAULT_TO_UNKNOWN (*)          = YES
RESOURCE_FAULT_TO_FREE (*)             = YES
RESOURCE_FAULT_TO_FAULT (*)           = YES
RESOURCE_FAULT_TO_RESERVED (*)         = YES
RESOURCE_FAULT_TO_MOUNTED (*)         = YES
RESOURCE_RESERVED_TO_UNKNOWN (*)       = YES
RESOURCE_RESERVED_TO_FREE (*)          = YES
RESOURCE_RESERVED_TO_FAULT (*)         = YES
RESOURCE_RESERVED_TO_RESERVED (*)      = YES
RESOURCE_RESERVED_TO_MOUNTED (*)      = YES
RESOURCE_MOUNTED_TO_UNKNOWN (*)        = YES
RESOURCE_MOUNTED_TO_FREE (*)           = YES
RESOURCE_MOUNTED_TO_FAULT (*)          = YES
RESOURCE_MOUNTED_TO_RESERVED (*)       = YES
RESOURCE_MOUNTED_TO_MOUNTED (*)       = YES
MISMATCH (*)                           = YES

```

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.7.2 Command: list

Syntax:

- UNIX/GnuLinux: wdx [mission] -list
- OpenVMS: wdx [mission] /list
- Windows NT: wdx [mission] /list

List the current status of a mission.

The following example shows this command entered on "member1"

```

Status of mission "wdx" on member "bounaberdix"
MEMBER          STATUS          NETWORK-1      NETWORK-2
bounaberdix    Substitute    on             on
sinbadsailor   Substitute    on             on
alibaba        Masterholder on             n/a
tomthumb       Substitute    on             n/a

```

paganini	fault	unknown	n/a
RESOURCE/DIRECTORY	STATUS	BY	AT
MASTERTOKEN	reserved	alibaba	24-Jun-2003 13:41:49
disk_data_1	reserved	bounaberdux	24-Jun-2003 11:57:28
/data0	free	sinbadsailor	24-Jun-2003 11:57:55
X25_link	fault	alibaba	24-Jun-2003 11:58:25
appli_2	free	bounaberdux	24-Jun-2003 11:58:49

The columns have the following meaning:

1. MEMBER

The name of the member for the mission. It is the host name if no member name is specified when the mission is created.

2. STATUS

The member status can be:

- FAULT: The member is in the FAULT state. The mission on the member is not active.
- IDLE: The member is in the IDLE state. The mission on the member is started, but no actions can be performed by this member when an event appears.
- Masterholder: The member gets the MASTERTOKEN so it is in the Masterholder state.
- Substitute: The member is in the Substitute state.
- Satellite: The member is a satellite.

3. NETWORK-1 and NETWORK-2

Indicate the state of the network interface number 1 and 2, as they are seen by the local node for each member.

Possible values are:

- N/A: no interface is declared in the mission file.
- UNKNOWN: the remote interface is declared in the mission file, but the local member has never received a message through this interface.
- ON: the local member has received valid messages from the remote member through this interface.
- OFF: the validity period of messages, received from the remote member through this interface, has expired.

A local interface state is ON if at least one of the remote interface on the network is ON.

A local interface state is OFF if all the remote interfaces on the network are OFF.

4. RESOURCE/DIRECTORY

This is the name of a resource or a mount point directory.

5. STATE

The status depends on the type of the resource. If the resource is a virtual resource, the status may be one of:

- FREE
- RESERVED
- FAULT

If the resource is a mount point directory the status may be one of:

- FREE
- MOUNTED
- FAULT

6. BY

This is the name of the member which initiated the last state transition on the resource. For RESERVED, MOUNTED and FAULT states, this indicates also the name of the member which holds a lock on the resource.

7. AT

This is the time of the last state transition. This parameter is used to solve lock protocol owner conflicts.

Returned Status:

- UNIX/GnuLinux: 0 = success ,1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.7.3 Command: log

Syntax:

- UNIX/GnuLinux: wdx [mission] -log
- OpenVMS: wdx [mission] /log
- Windows NT: wdx [mission] /log

This option renames the current log file. This option is used to analyse the contents of the log file, or when the log file size increases too quickly. The name convention depends on the Operating System:

- the suffix .0 is added to the file name for UNIX systems.
- incrementation of the file version for OpenVMS systems.
- on Windows NT systems, a character 0 is appended at the end of the root name or replaces the last character of names which are 8 characters long (for example mission.log is renamed to mission0.log)

This option requires superuser privileges.

Returned Status:

- UNIX/GnuLinux: 0 = success ,1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.7.4 Command: namemaster

Syntax:

- UNIX/GnuLinux: wdx [mission] -namemaster
- OpenVMS: wdx [mission] /namemaster
- Windows NT: wdx [mission] /namemaster

Prints the member name of the current Masterholder node. If there is no Masterholder, the message:

```
masterholder.is.unknown
```

is printed.

Returned Status:

- UNIX/GnuLinux: 0 = success ,1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.7.5 Command: show

Syntax:

- UNIX/GnuLinux: wdx -show
- OpenVMS: wdx /show
- Windows NT: wdx /show

Prints the characteristics of all missions currently processing on the local computer.

Example:

```
Local processing missions:
MISSION NAME          LOCAL MEMBER NAME          Masterholder NAME
mission_1             sinbadsailor               sinbadsailor
wdx                   sinbadsailor               bounaberdux
```

In the above example, two missions are running on the system. This system for the first mission is Masterholder, and for the second mission, is Substitute or Satellite.

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.7.6 Command: status

Syntax:

- UNIX/GnuLinux: wdx [mission] -status
- OpenVMS: wdx [mission] /status
- Windows NT: wdx [mission] /status

Prints the status of the local member. The Status can be:

- Masterholder
- Substitute
- Satellite
- IDLE

Returned Status:

- UNIX/GnuLinux: 0 = success , 1 =command failed , -1 syntax error
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.7.7 Command: Help

Syntax

- UNIX/GnuLinux: Not Applicable. Use Man pages
- OpenVMS: wdx /help
- Windows NT: wdx /help

Display on-line help. This command is only implemented for OpenVMS andWindows NT versions.

Note:

The command can be issued without the presence of the wdx detached process, and it performs a check of the wdx environment variable on WindowsNT.

Returned Status:

- UNIX/GnuLinux: Not applicable
- OpenVMS: 1 = success, %X10000000 (hexa) = syntax or semantic error
- Windows NT: 0 = success , -1 = error

4.8 Option Restrictions

The use of each option is dependant on the member state.

<u>Feature</u>	<u>Keyword</u>	<u>IDLE</u>	<u>RUNNING</u>
Run mission	run	Yes	No
Kill mission	kill	Yes	Yes
Stop mission	stop	No	Yes
Start mission	start	Yes	No
Pause mission	pause	No	Yes
Exception	exception	No	Yes
Run local member	join	Yes	No
Kill local member	leave	Yes	Yes
Mastertoken lock request	master	Yes	Yes
Mastertoken lock release	nomaster	No	Yes
Get mastertoken	getmaster	No	Yes
Declare resource	declare	No	Yes
Erase resource	erase	No	Yes
Reserve resource	reserve	No	Yes
Free resource	free	No	Yes
Fault resource	fault	No	Yes
Get resource	getresource	No	Yes
Wait for lock	wait	No	Yes
Declare directory	declare	No	Yes
Mount directory	mount	No	Yes
Fault directory	fault	No	Yes
Free directory	free	No	Yes
Wait for NFS	mount nfs	No	Yes
List version	version	Yes	Yes
List data	list	Yes	Yes
Open log file	log	Yes	Yes
Get master name	mastername	Yes	Yes
Show running mission	show	Yes	Yes
Get local status	status	Yes	Yes
On line Help (OpenVMS, Windows NT only)	help	Yes	Yes

5 Using WitchDoctor-WDX in Windows NT

Note: This chapter is out of date and will be revisited at Windows XP new release date

Although the WitchDoctor-WDX functionalities are unchanged for the Windows NT version, the differences in the Operating System environment imply some changes in the use of the product. This chapter describes the specific operations required to use and to configure WitchDoctor-WDX on a Windows NT system. It describes:

- the post installation configuration of the system
- the declaration and configuration of the product as a service, to enable an automatic startup of the wdx process following a system reboot,
- the different methods to start the product and their purpose,
- the scripts and command procedures used in Windows NT,
- some restrictions imposed by Windows NT,
- time setting.

5.1 Post Installation Configuration

Once the installation of wdx is complete, the following actions must be performed before the WitchDoctor-WDX software can be run :

1. The environment variable wdxloc is defined during installation. This only takes effect, once we log out and log back in.
2. The path variable is only updated if the user chooses to do so. This may be defined as follows :
set path = %path%;c:\wdx
3. The files wdx.cfg and wdx.loc must be edited, and the wdx local requirements and configuration entered within. Note that if the name of the network interface is unknown, it may be retrieved from the registry as follows :

```
execute c:\WINNT35\system32\Regedt32
```

To obtain the TCP/IP line name select the following options :

```
HKEY_LOCAL-MACHINE
SOFTWARE
MICROSOFT
WINDOWS NT
CURRENT VERSION
NETWORK CARDS
```

Select the required card, and look at the SERVICENAME. This is the name by which the network line is known. The wdx.cfg file must be updated to contain this name for the line related to the Windows NT machine.

4. The WDX service must be registered in the services file. This file is located in c:\WINNT35\system32\drivers\etc\services. The following line should be added :
xxx rrr/udp #wdx mission
where xxx is the WDX mission name (by default WDX).
and rrr is the port number associated with the service (this number must be the same on all missions).

5.2 Start the wdx Process

There are 3 different modes to start the wdx process WDX_MAIN.EXE, each used for a different purpose:

1. as a service with an automatic startup,
2. as a service started manually,
3. with a standard run command from the command prompt window.

5.2.1 Service with an Automatic Startup

This is the mode used for normal operation, once the mission has been defined, configured and is ready for use. In this mode the process WDX_MAIN.EXE runs as a background task which does not require any separate command window. The wdx process runs with the privilege of the system administrator, and does not require any user logging to operate.

5.2.2 Service Started Manually

This mode is used only as interim operation, either to test the startup of WDX_MAIN.EXE with a mission as a background task, or to run a mission until the next system boot. This allows the system administrator to start a mission without restarting the system immediately, to enable the automatic startup of this mission as a service using the new list. The mission can be added and declared in the registry; in the list of missions for automatic startup, the declaration will be effective at the next reboot. The WDX_MAIN.EXE runs in the same way as the automatic startup mode, as a background task with no separate command prompt window.

5.2.3 From the Command Prompt Window

The WDX_MAIN.EXE process is started in a command prompt window by typing:

```
WDX_MAIN [mission] [member_name]
```

The process runs in the window. This mode is used only for testing a mission and associated scripts. The wdx must not be left operating unattended in this mode.

5.3 Service Declaration and Configuration of wdx Process

One instance of the wdx process WDX_MAIN.EXE must be started as a service process for each mission. For normal operation the service is declared with an automatic startup. This allows the wdx process to be started without any operator action or logging when the system starts. If you want to start the wdx process when you configure the service, make sure the mission and local files are correctly created on your system.

5.3.1 Service Declaration for Automatic Startup

This operation is a complementary part of the wdx configuration when you define a mission on your system. It is performed once you have decided the name of your mission. In this automatic startup service mode, a dedicated service WDX_AUTO is used to start the various instances of the WDX_MAIN.EXE with the different missions declared on the computer. The mission is declared in a list of missions for automatic startup. The wdx process is started for this mission by the WDX_AUTO service when the system is restarted. To declare the mission and to configure this WDX_AUTO service for automatic startup of wdx process, proceed as in the following example, which supposes a mission called TEST on a member called bandol.

Pre-required Information

- log in as System Administrator
- decide the name for the mission you want to create,
- decide the member_name used as argument with the mission,

Verification and Creation of Service WDX_AUTO and WDX_INI

The services WDX_AUTO and WDX_INI must be declared in the list of available services. WDX_AUTO and WDX_INI are reserved words for WitchDoctor-WDX. Verify these services are declared as follows:

- in the Control Panel window, open the Service dialog box with the Service icon,
- if created, these services will appear in the list of services displayed.

If the services are not already declared and do not appear in the list, create them as follows:

- open a command prompt window,

- set the default directory to \WDX,
- run the WDX_REG.EXE program to declare a service, using the commands:
WDX_REG WDX_AUTO WDX_REG WDX_INI
- this declares WDX_AUTO and WDX_INI as services to the Window NT registry, making them available in the existing list of services on the system.

Once the service WDX_AUTO has been created, it must be configured in the Service dialog box:

- in the Control Panel window, open the Service dialog box with the Service icon,
- WDX_AUTO appears in the list of services displayed,
- select WDX_AUTO by clicking on it,
- verify the Startup selected is Automatic,
- if the startup is not set to Automatic:
 - click on Startup button to configure the startup for the service,
 - specify Automatic as the startup type; this starts the wdx process WDX_MAIN.EXE every time the system starts,
 - click on OK to return in main service box,

Do not enter anything in the Startup Parameters box in this mode of operation.

- repeat the same check to verify WDX_INI is also declared as automatic.

Declaration of the mission (for service)

- select or open the command prompt window,
- set the default directory to \WDX,
- run the WDX_AMIS.EXE program to add the mission to the list of missions started automatically with the wdx process. Use the command:
WDX_AMIS mission [member_name]
where mission is the name of the mission and member_name is the host name or equivalent, as in:
WDX_AMIS TEST bandol
- this adds a subkey TEST with the data bandol in the service key WDX_AUTO of Window NT registry. All the subkeys in the key WDX_AUTO constitute a list of mission member_name entries used by the service WDX_AUTO.
To see the list of subkeys in the key WDX_AUTO, execute c:\WINNT35\system32\Regedt32
Select the following options :
HKEY_LOCAL_MACHINE
SYSTEM
CurrentControlSet
Services
WDX_AUTO

The configuration for automatic startup of the wdx process is now complete. The process can be started automatically with the mission declared each time the Windows NT system is started. By default the WDX process for each mission will be in the IDLE state. If a mission is required to be in a different state after system startup, then the startup information must be added into the WDX_AUTO.BAT file in the \WDX directory. For example if the mission is required to be in the the RUNNING state after system startup then the following line must be added :
c:\wdx\wdx mission /join

This startup will be effective at next system reboot unless you immediately start the service manually as follows:

- If several missions are needed on the computer, declare each mission as previously described
- in the Service dialog box, click the Start button.

5.3.2 Service declaration with Manual Start

In this mode, one or several new services can be created to start the wdx process WDX_MAIN.EXE as a background task without opening a separate command prompt window. However, one service must be created for each mission. The wdx process will not be restarted automatically when the system reboots. The main use of this mode is to test the start operation using the service environment as in normal operation. In this mode missions can be added to the list of missions declared for automatic startup while these missions are already running. It can also be used to start the wdx process with the new mission as a background task until the next system restart.

Pre-required and information

- log in as System Administrator,
- decide the mission name you want to create,
- decide a service name for this mission,
- decide of the member_name used as argument with the mission.

Declare a service for a mission

The following procedure uses a service name Autopilot with a mission TEST on a member called bandol.

- open a command prompt window,
- set the default directory to \WDX,
- run the WDX_REG.EXE program to declare a service, using the command below:
WDX_REG service
where service is the name of the service for the mission (in this case Autopilot).

This declares Autopilot as a service to the Window NT registry, and makes it available in the existing list of services on the system.

Configure and Start the New Service

- in the Control Panel window, open the Service dialog box with the Service icon,
- the service created (Autopilot) appears in the list of services displayed,
- click on the Autopilot service,
- enter the mission name TEST followed by the member_name bandol in the Startup Parameters box:
TEST bandol
- set the service to manual in the Startup column,
- click on Startup button to configure the startup for the service,
specify Manual as startup type,
click on OK to return in main service box.

The service can be started immediately with the Start button. The above sequence is equivalent to the Start command

```
WDX_MAIN.EXE TEST bandol
```

except that the process runs as a background task.

5.3.3 Disable or Stop an Existing Service

- log in as System Administrator.

Disable Automatic Startup for all Missions

If you want to disable the automatic startup of the wdx process for all missions during reboot:

- in the Control Panel window, open the Service dialog box with the Service icon,
- select the WDX_AUTO wdx service,
- click on Startup button to configure the startup for the service,

- Specify Disabled or Manual in the startup type box.

Disable Automatic Startup for one Mission

If you want to disable the automatic startup of the wdx process for one mission only during reboot:

- open a command prompt window,
- set the default directory to \WDX,
- delete the corresponding mission from the automatic startup list of missions. Use the command:
WDX_DMIS mission
where mission is the name of the mission. In our example this deletes the mission TEST.
- use the stop or kill command if you want to stop the corresponding wdx process immediately.

5.3.4 Suppress an Existing Service

- log in as System Administrator.

Service with Manual Start

To suppress a service created for one mission with a manual start:

- ensure no wdx process is running for the mission,
- ensure the corresponding service for the mission is Disabled or Stopped in the Service dialog box (as described above),
- open a command prompt window,
- set the default directory to \WDX,
- delete the service declaration in the Window NT registry. Use the command:
WDX_UREG service_name
where service_name is the name of the service, in our example Autopilot.
- the service Autopilot is deleted from the list of services on the system.

WDX_AUTO Service for Automatic Startup

- ensure all wdx processes are stopped for all the concerned missions,
- delete the declarations of all missions in the list from automatic startup using the WDX_DMIS program,
- ensure the WDX_AUTO service is Disabled in the Service dialog box (as described above),
- open a command prompt window,
- set the default directory to \WDX,
- delete the WDX_AUTO service declaration in the Window NT registry. Use the command:
WDX_UREG WDX_AUTO
- the service WDX_AUTO must be deleted from the list of services on the system.

5.4 Wdx Initialization service

During Windows NT system startup, a wdx service WDX_INI is used to ensure a cleanup of the WDX directory. This service is mandatory and must run prior the use of the Automatic Startup service. It starts the execution of the batch file WDX_ENV.BAT to delete the erroneous .dat files which might be incidently open in case of system crash. WDX_INI and WDX_ENV.BAT names are reserved for the wdx use, and must not be modified.

5.5 Startup errors

If an error occurs during WDX_AUTO service startup, an error message is written in the log file c:\WINNT35\system32\wdx_str.log.

5.6 Wdx Scripts

Scripts must be command batch files (.bat), which allows the use of arguments generated by the wdx process during script triggering. For script triggering WitchDoctor-WDX starts the process declared as script in the local file as a new process created in a new command prompt window. Consequently, this script process runs with its own environment and variables. If variables are required to be shared with other processes they must be set accordingly.

5.7 Time Setting

The local time is used as a time reference for communication exchanges between the computers of the mission. You can display and check the time used by the wdx on the different computers with the combination of the list and resource management commands, or by looking at the timestamps in the wdx log file. Use the Date/Time box in the Control Panel to verify or to set the local time used by WitchDoctor-WDX.

6 WitchDoctor-WDX behaviour in an OpenVMS cluster environment

6.1 Cluster, quorum and processes behaviour

Any OpenVMS cluster transition performs a quorum calculation. A quorum, in the cluster terminology, corresponds to a majority of participating systems to a cluster. It guarantees that systems do not suddenly work in separated clusters that would be disastrous for the clustered data. When the cluster members votes are less than the quorum value then all clustered activities are blocked and members wait for the votes that will allow to reach the quorum value, and then, continue the activities. Before OpenVMS 6.0, a quorum loss was involving a complete system blocking. Since the OpenVMS version 6.0, a process is jammed on a system during a "quorum lost" event, only if it requires a clustered resource (such as shared disks). That means that a process making only calculations or requiring only local resources continues to be processed.

6.2 WitchDoctor-WDX resiliency upon cluster transition with loss of quorum

Therefore, WitchDoctor-WDX gives the choice to the user to decide if application software will be jammed during a quorum loss, or not and if WitchDoctor-WDX is to notify it.

- If WitchDoctor-WDX gets jammed on purpose on Masterholder system, then the other mission members, receiving no more heartbeat messages from it, consider it has failed. The Substitute systems compete for MASTERTOKEN to become the new Masterholder for that mission. When former Masterholder system recovers its quorum, it releases its Mastertoken lock, according to DOUBLEMMASTER algorithm. This behaviour can meet application requirements.
- If WitchDoctor-WDX continues to run, then the other systems still receive its heartbeat messages, and cluster disruption is not detected. In case application that is supervised is not jammed either, this is a correct result.

6.3 The Cluster WDX resource

WitchDoctor-WDX implementation provides a solution, even in tricky cases, at installation time, by placing configuration files at the right place.

- If they are copied to a shareable device, in any case the WDX process will be jammed in case of "quorum lost" for any mission.
- If they are copied to a local disk : the WDX process will keep a chance to survive during cluster transition.

To handle safely this last situation, it is recommended to use a WitchDoctor-WDX resource. This resource represents the "local cluster quorum status".

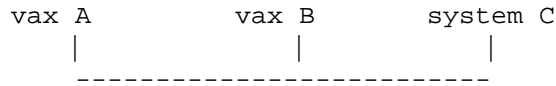
The user has the possibility to start, during the sys\$startup_vms.com file execution (in background, via a Run/detach/noswapping with the PSWAPM privilege) a DCL procedure to act on the WitchDoctor-WDX resources. This particular procedure is here called "Sensor".

The Sensor checks, first, if a cluster environment is set up, and in this case it tests the version of the clustering software (OpenVMS 6.0 or later). The Sensor waits for a few seconds, DECLARE and RESERVE a WitchDoctor-WDX resource, and then loops forever.

The resource name is chosen by the administrator. It is a parameter provided to the Sensor run time. This name must be different for each cluster member, because a resource can be reserved by only one member at a time. Thus the script is able to take the correct action depending on the node which experiments problems.

At regular intervals, the Sensor watches the quorum variation value. In case of quorum loss, the resource is set to FAULT. An eventscript starts on the local machine and on all the other reachable WitchDoctor-WDX members. For

each mission member, the fact that this resource is FAULT means that the cluster QUORUM is lost. When a system crash happens during this period, the resource passes from FAULT to FREE, and that will have a different meaning for the other members. When the quorum is regained, the Sensor sets to RESERVED the resource, starting again the eventscript on each member. When a system stop occurs (shutdown or crash), then the resource changes from RESERVED to FREE, and a correct interpretation is possible for the other mission members (system disappeared). In case of system "Freeze", whereas an other system has its cluster resource "FAULT", typically:



- 1 - Vax A down, so Sensor of Vax B sets its cluster resource to FAULT.
- 2 - System C "FREEZE" every alive systems. So nodes are in IDLE state.
- 3 - Vax A reboot, so its Sensor sets its cluster resource to RESERVED.
- 4 - The Vax B Sensor will try to set its cluster resource to RESERVED too.
But, as the node is in IDLE state, the command is denied.

To solve this state, the Sensor must retry, until success, the cluster resource setting BUT by reconsidering, in each loop, the current cluster votes, because, if the node A shutdowns again, Sensor of Vax B command would be useless. In the eventscript, the user has the possibility to take any decision concerning his application behaviour, following the general WitchDoctor-WDX rules. Of course, he has to write carefully his script files if he wants them to run even during this period.(because nodes go up and down). If the Sensor declares the resource whereas WDX is not ready, it must reissue the command. So, the Sensor must check that. When the user does not use this Sensor, then the cluster member still active will continue to be seen by the other mission member as if nothing was happening. The Sensor should be stopped before the last mission on a node, else, when a "quorum lost" occurs whereas WDX is stopped, the Sensor will receive an error because WDX commands will not work. However, the Sensor must be written to manage this kind of error.

Note:

A cluster transision with a quorum reached is not a problem for WDX process, WDX can just miss few heartbeat messages but the user can ajust the validation period.

7 WitchDoctor-WDX Application Programmable Interface

7.1 Architecture

To allow portable flexibility, the architecture between WDX process and WDX applications (such as wdx command interface) is defined, as follow :

- WDX applications send requests to the WDX process through the communication socket defined for the mission. (available in the WDX V2.0 version)
- WDX process uses dedicated channels to send back to the WDX application a status and a notification for the request. The dedicated channels are FIFO defined by the application when sending the request.

For future use, additional automatic event notification can be added to the above feature. The features supported by the application (request or/and event notification) are negotiated when connecting to the WDX process.

Notes:

C signals and shared memory are used for communication between WDX process and WDX commands. For some commands, the status is not the WDX process status but only a status indicating that the command has been successfully sent to the WDX process.

WDX process does not notify event to WDX commands.

WitchDoctor-WDX application interface is not designed as a general purpose library for HA-Aware applications. It provides the possibility to develop an alternative to the command interface.

Any application which connects to WitchDoctor-WDX API should conform to GPL terms and conditions.

7.2 Protocol definition

The communication protocol between the WDX application and the WDX process is defined below :

Messages exchanged by the processes

application_request.

request message send by the application to the WDX process. Several messages can be sent by the application :

- `command_request` : modify the status of the WDX process.
- `local_request` : gets local information for the mission.
- `lock_request` : resource(directory) lock request. A resource(directory) name is mandatory for these messages.
- `wait_request` : resource(directory) wait request. A resource(directory) name is mandatory for these messages.

process_acknowledge.

acknowledge message sent by the WDX process when a re-quest has been processed. (synchronous message)

process_notification.

asynchronous message sent by the WDX process when a `lock_request` has been completed.

process_event.

asynchronous message sent by the WDX process when a WDX event occurs on the system.

process_control.

asynchronous message sent by the WDX process to signal a control event. Control events are `first_message` , `last_message` and `messages_have_been_trashed`.

Communication channels :

- mission socket for application_request.
 - one FIFO for process_acknowledge. (synchronous communication)
 - one FIFO for process_notification, process_event and process_control. (asyn-chronous communication)
- FIFOs are created to avoid splitting messages into several parts.

Connection mode which selects the features supported by the application :

- request only (REQUEST_ONLY)
 - event notification only (EVENT_ONLY) (for future use)
 - request and event notification (REQUEST_AND_EVENT) (for future use)
- This connection mode is negotiated when the connection is established.

Limitations

- maximum number of WDX processes connected to a single application
MAX_PROCESS_FOR_APPLICATION.
- maximum number of application connected to a WDX process
MAX_APPLICATION_FOR_PROCESS.

Synchronous and asynchronous messages return :

- the status for the request : error, success.
- a formatted printable string.
- the date of the event or notification.

Note : the contents of the messages exchanged by the WDX commands and WDX process are defined in <REFERENCE>(COM).

7.3 API definition

From the WDX application, the communication with the WDX process is seen through 4 functions which are :

- `wdx_open` : initializes the communication with the WDX process. The connection mode is negotiated during this call.
- `wdx_close` : terminates the communication with the WDX process.
- `wdx_write` : formats and send a request to the WDX process. Waits for the asynchronous answer, then returns the status and printable answer.
- `wdx_read` : read a message on the synchronous channel, and returns all the availablefields (type, status, date and printable string).

7.4 Supported commands

The API supports the following commands :

command_request : no data required with these commands.

- WDX_GROUP_INIT : refers to <REFERENCE>(wdxinit) operator command
- WDX_GROUP_KILL : refers to <REFERENCE>(wdxkill) operator command
- WDX_GROUP_EXCEPTION : refers to <REFERENCE>(wdxexception) operator command
- WDX_MEMBER_IS_UP : refers to <REFERENCE>(wdxjoin) operator command
- WDX_WATCHDOG_STOP : refers to <REFERENCE>(wdxdaemonkill) operator command

- WDX_WATCHDOG_INIT : refers to <REFERENCE>(wdxdaemoninit) operator command

local_request : no data required with these commands.

- WDX_LOCALNAME : returns the local name of the WDX process
- WDX_LOG : refers to <REFERENCE>(wdxlog) operator command
- WDX_CONSTANT : refers to <REFERENCE>(wdxversion) operator command
- WDX_LIST : refers to <REFERENCE>(wdxlist) operator command
- WDX_MASTERNAME : returns the name of the masterholder
- WDX_STATUS : returns the status of the local WDX process.

lock_request : resource(directory) name is required.

- WDX_DECLARE : refers to <REFERENCE>(wdxdeclare) operator command
- WDX_ERASE : refers to <REFERENCE>(wdxerase) operator command
- WDX_RESERVE : refers to <REFERENCE>(wdxreserve) operator command
- WDX_MOUNT : refers to <REFERENCE>(wxmount) operator command
- WDX_FREE : refers to <REFERENCE>(wdxfree) operator command
- WDX_FAULT : refers to <REFERENCE>(wdxfault) operator command
- WDX_GETMASTER : refers to <REFERENCE>(wdxgetmaster) operator command. The mastertoken data is provided by the API.
- WDX_GET : refers to <REFERENCE>(wdxget) operator command
- WDX_MASTER : refers to <REFERENCE>(wdxmaster) operator command. The mastertoken data is provided by the API.
- WDX_NOMASTER : refers to <REFERENCE>(wdxnomaster) operator command. The mastertoken data is provided by the API.

wait_request : resource(directory) name is required

- WDX_WAIT_FOR_RESERVE : refers to <REFERENCE>(wdxwait) operator command
- WDX_WAIT_FOR_MOUNT : refers to <REFERENCE>(wdx nfs) operator command

APPENDIX

A Operating System Specific files

This appendix gives the various directory, path and files names used with WitchDoctor-WDX for each supported Operating System.

A.1 UNIX

- executable process image /etc/wdx
- WitchDoctor-WDX directory /var/shaman-x/wdx/mission-name/
- default mission file name is /var/shaman-x/wdx/wdx/wdx.cfg
- default local file name is /var/shaman-x/wdx/wdx/wdx.loc
- default mission log file name is /var/shaman-x/wdx/wdx/wdx.log
- default snapshot file name is /var/shaman-x/wdx/wdx/wdx.hot
- default file name created for user information by stop command is /var/shaman-x/wdx/wdx/wdx.txt
- System log file
- Installation Kit binary saveset name WDXBIN220
- Installation Kit MAN Pages saveset name WDXMAN220
- the UDP communication port name for UCX is the name of the mission, (by default wdx), defined in the file /etc/services

A.2 GnuLinux

- executable process image /sbin/wdx
- WitchDoctor-WDX startup file template /sbin/init/wdx
- WitchDoctor-WDX directory /var/shaman-x/wdx
- default mission file name is /var/shaman-x/wdx/wdx.cfg
- default local file name is /var/shaman-x/wdx/wdx.loc
- default mission log file name is /var/shaman-x/wdx/wdx.log
- default snapshot file name is /var/shaman-x/wdx/wdx.hot
- default file name created for user information by stop command is /var/shaman-x/wdx/wdx.txt
- System log file
- Installation Kit binary saveset name WDXBIN220
- Installation Kit MAN Pages saveset name WDXMAN220
- the UDP communication port name for UCX is the name of the mission, (by default wdx), defined in the file /etc/services

A.3 OpenVMS

- wdx detached process executable [WDXAUT]WDXAUT.EXE
- User Interface process executable [WDXAUT]WDXAUT_UI.EXE
- startup file SYS\$STARTUP:WDXAUT_STARTUP.COM
- WitchDoctor-WDX directory [WDXAUT]
- default mission file name is [WDXAUT]WDX.CFG
- default local file name is [WDXAUT]WDX.LOC
- default mission log file name is [WDXAUT]WDX.LOG
- default snapshot file name is [WDXAUT]WDX.HOT
- default file name created for user information by stop command is [WDXAUT]WDX.TXT
- Operator log file SYS\$MANAGER:OPERATOR.LOG
- help file SYS\$HELP:WDXAUT.HLB
- Installation Kit binary saveset name WDXAUT_AXP022.A for OpenVMS AXP
- Installation Kit binary saveset name WDXAUT_VAX022.A for OpenVMS VAX
- the UDP communication port name for UCX is the name of the mission, (by default wdx), defined

using the UCX command SET SERVICE.

A.4 Windows NT

- wdx main process executable \WDX\WDX_MAIN.EXE
- User Interface process executable \WDX\WDX.EXE
- Service startup program \WDX\WDX_SERV.EXE
- Service registration executable \WDX\WDX_REG.EXE
- Executable to delete a service registration \WDX\WDX_UREG.EXE
- Executable to Add a Mission and member registration for Automatic startup service \WDX\WDX_AMIS.EXE
- Executable to Delete a Mission and member registration \WDX\WDX_DMIS.EXE
- WitchDoctor-WDX directory \WDX
- default mission file name is \WDX\WDX.CFG
- default local file name is \WDX\WDX.LOC
- default mission log file name is \WDX\WDX.LOG
- default snapshot file name is \WDX\WDX.HOT
- default file name created for user information by stop command is \WDX\WDX.TXT
- System log file: Application Event File
- help file \WDX\WDX_HELP.HLP
- Service wdx Initialization program \WDX\WDX_INI.EXE
- Mandatory name for service wdx Initialization \WDX\WDX_INI
- wdx Initialization batch file \WDX\WDX_ENV.BAT
- Installation directory name \WDX
- Installation program executable \WDX\INSTALL.EXE
- Installation procedure file \WDX\INSTALL.INS
- the UDP communication port name for UCX is the name of the mission, (by default wdx), defined in the file \winnt\system32\drivers\etc\services

B Mission, Local and Support File Templates

B.1 Mission Configuration File

The following file is a template mission configuration file. This is a template available for any operating system.

```
#####
#                               WitchDoctor-WDX V1.0
#
#                               Mission file: mission.cfg
#####
# Mission file must be identical on all members in the mission.
#-----
# mission name :
# by:
#-----
#
# All fields are MANDATORY.
# Format:
# VERSION          1.0
# DATE             string
#
# VERSION ID is two digits. It is WDX version: 1.0
# DATE string is a 20 char string to identify this version of the mission file.
#
# <line> ::= MEMBER <member_name> <member_charact>
#
# <member_name>      ::= MEMBER_NAME
# <member_charact>   ::= <member_type> <member_priority>
#                   <member_broadcast> <member_validity>
#                   <member_redundancy> <network_1> <network_2>
#
# <member_type>      ::= <master_slave> | <satellite>
# <master_slave>     ::= MASTER/SUBSTITUTE
# <satellite>        ::= SATELLITE
# <member_priority>  ::= 0 is highest priority,
#                   substitute waits (priority x broadcast) seconds
#                   before competing for mastertoken
# <member_broadcast> ::= BROADCAST time in seconds.
#                   3 is minimum.
#                   sets period of watchdog messages over the networks
#                   long periods may affect some wdx commands response time
# <member_validity>  ::= VALIDITY period in seconds.
#                   BROADCAST time +1 is minimum.
#                   (3 x BROADCAST time) +1 is recommended
#                   sets WDX failure detection delay
# <member_redundancy> ::= number of sends for each watchdog message
#                   integer >= 1, 1 means no redundancy
# <network_1>        ::= <DEVICE_NAME> | "N/A"
# <network_2>        ::= <DEVICE_NAME> | "N/A"
#-----
# Example:
#-----
# VERSION 1.0
# DATE 1-Jan-2000-draft
# MEMBER name type priority broadcast validity redundancy network-1 network-2
# MEMBER member1 MASTER/SUBSTITUTE 0 3 10 2 eth0 eth1
# MEMBER member2 MASTER/SUBSTITUTE 0 3 10 2 eth0 eth1
# MEMBER member3 MASTER/SUBSTITUTE 1 3 10 1 eth0 N/A
# MEMBER member4 SATELLITE 0 3 30 1 eth0 N/A
```

B.2 Local File

The following file is a template local file for UNIX. For other operating systems the default script list changes.

```
#####
#                               WitchDoctor-WDX V1.0
#
#                               Local file: mission.loc
#####
# Local file can be different on every member in the mission.
#-----
# for mission:                  on member:
# by:
#-----
# Header format :
# VERSION          value       2 digits identify configuration release level
#                               value must be the same on all members
#                               used to check mission consistency
# DATE             string      this is a 20 char string to identify this file
#-----
VERSION              0.0
DATE                 1-Jan-2000-draft
#-----
# Script parameters are local parameters. They can be different on every
# member in the mission.
# Script parameters format :
# <script_line>      ::= <script_name> <script_path>
# <script_name>      ::= NEWMASTERSCRIPT | NOSUBSTITUTESCRIPT |
#                               SATELLITESCRIPT | EXCEPTIONSCRIPT | EVENTSCRIPT
# <script_path>      ::= any script file name
#                               an error in the path name is undetected until
#                               it is used.
#                               The content of script is not verified by WDX
#                               The end of script execution is not monitored by WDX
#                               All script activations must be tested
#                               after local file is modified
#-----
NEWMASTERSCRIPT      /bin/date +"Become MASTERHOLDER %H:%M:%S"
NOSUBSTITUTESCRIPT  /bin/date +"Ifeelalone at %H:%M:%S"
SATELLITESCRIPT      /bin/date +"MASTERHOLDER changed %H:%M:%S"
EXCEPTIONSCRIPT      /bin/date +"exception %H:%M:%S"
EVENTSCRIPT          /bin/echo
#-----
# Transition parameters are local parameters. They can be different on each
# member in the mission.
# Transition parameters format :
# <transition_line>  ::= <transition_name> <YES | NO>
#                               YES: the EVENTSCRIPT is started in this case
#                               NO:  the EVENTSCRIPT is NOT started in this case
# name value
#-----
NETWORK_UNKNOWN_TO_ON          YES
NETWORK_UNKNOWN_TO_OFF         YES
NETWORK_ON_TO_OFF              YES
NETWORK_ON_TO_UNKNOWN          YES
NETWORK_OFF_TO_UNKNOWN         YES
NETWORK_OFF_TO_ON              YES
MEMBER_FAULT_TO_IDLE           YES
MEMBER_FAULT_TO_MASTER         YES
MEMBER_FAULT_TO_SUBSTITUTE     YES
MEMBER_FAULT_TO_SATELLITE     YES
#-----
```

MEMBER_IDLE_TO_FAULT	YES
MEMBER_IDLE_TO_MASTER	YES
MEMBER_IDLE_TO_SUBSTITUTE	YES
MEMBER_IDLE_TO_SATELLITE	YES
MEMBER_MASTER_TO_FAULT	YES
MEMBER_MASTER_TO_IDLE	YES
MEMBER_MASTER_TO_SUBSTITUTE	YES
MEMBER_MASTER_TO_SATELLITE	YES
MEMBER_SUBSTITUTE_TO_FAULT	YES
MEMBER_SUBSTITUTE_TO_IDLE	YES
MEMBER_SUBSTITUTE_TO_MASTER	YES
MEMBER_SUBSTITUTE_TO_SATELLITE	YES
MEMBER_SATELLITE_TO_FAULT	YES
MEMBER_SATELLITE_TO_IDLE	YES
MEMBER_SATELLITE_TO_MASTER	YES
MEMBER_SATELLITE_TO_SUBSTITUTE	YES
RESOURCE_UNKNOWN_TO_UNKNOWN	YES
RESOURCE_UNKNOWN_TO_FREE	YES
RESOURCE_UNKNOWN_TO_RESERVED	YES
RESOURCE_UNKNOWN_TO_MOUNTED	YES
RESOURCE_UNKNOWN_TO_FAULT	YES
RESOURCE_FREE_TO_UNKNOWN	YES
RESOURCE_FREE_TO_FREE	YES
RESOURCE_FREE_TO_RESERVED	YES
RESOURCE_FREE_TO_MOUNTED	YES
RESOURCE_FREE_TO_FAULT	YES
RESOURCE_RESERVED_TO_UNKNOWN	YES
RESOURCE_RESERVED_TO_FREE	YES
RESOURCE_RESERVED_TO_RESERVED	YES
RESOURCE_RESERVED_TO_MOUNTED	YES
RESOURCE_RESERVED_TO_FAULT	YES
RESOURCE_MOUNTED_TO_UNKNOWN	YES
RESOURCE_MOUNTED_TO_FREE	YES
RESOURCE_MOUNTED_TO_RESERVED	YES
RESOURCE_MOUNTED_TO_MOUNTED	YES
RESOURCE_MOUNTED_TO_FAULT	YES
RESOURCE_FAULT_TO_UNKNOWN	YES
RESOURCE_FAULT_TO_FREE	YES
RESOURCE_FAULT_TO_RESERVED	YES
RESOURCE_FAULT_TO_MOUNTED	YES
RESOURCE_FAULT_TO_FAULT	YES
MISMATCH	NO

B.3 Support File

The following file is a template support file mission.sup.

```
#####
#                               WitchDoctor-WDX V1.0
#
#                               Support file: mission.sup
#####
# Support parameters are local parameters. They can be different on every
# member in the mission.
#-----
# for mission:                on member:
# by:
# VERSION 1.0
# DATE 1-Jan-2000-draft
#     I swear I read the manual before playing with this file :-((
#-----
# Format :
# Keyword                Possible values                Default (lines are optional)
#
# LOG_FLAG                DISABLE OR ENABLE                ENABLE
#                         DISABLE stops WDX logging
# LOG_FILE                log file name                mission_name.log
#                         Alternate log file or device for convenient testing
# PRIORITY                Process priority level                highest
#                         Low priority level is useful for testing
#                         Low priority can be used for monitoring system load
# PERIOD                  n (seconds)                1
#                         WDX activation period (badly tested if not 1)
# ACTIVITY                DISABLE OR ENABLE                DISABLE
#                         Self monitoring of WDX
# THRESHOLD                member_name nb_seconds                No threshold
#                         Alert if broadcast messages from member_name
#                         are not received on time
# STATISTICS                nb_seconds                No statistics
#                         Additionnal diagnostic tool
#-----
# Example:
#
# LOG_FLAG                ENABLE
# LOG_FILE                /var/shaman-x/wdx/wdx.log
# PRIORITY                +20
# PERIOD                  1
# ACTIVITY                DISABLE
# THRESHOLD                member 6
# STATISTICS                60
```

C Mission, Local and Support File Examples

C.1 Example Mission File

```
#####
#
#                               WitchDoctor-WDX V1.0
#
#                               Mission file: db-server.cfg
#####
# Mission file must be identical on all members in the mission
#-----
# for mission:          db-server
# by:                  dominique
#-----
#
# All fields are MANDATORY.
# Format:
# VERSION              1.0
# DATE                 string
#
# VERSION ID is two digits. It is WDX version: 1.0
# DATE string is a 20 char string to identify this version of the mission file.
#
# <line> ::= MEMBER <member_name> <member_charact>
#
# <member_name>      ::= MEMBER_NAME
# <member_charact>   ::= <member_type> <member_priority>
#                   <member_broadcast> <member_validity>
#                   <member_redundancy> <network_1> <network_2>
#
# <member_type>      ::= <master_slave> | <satellite>
# <master_slave>     ::= MASTER/SUBSTITUTE
# <satellite>        ::= SATELLITE
# <member_priority> ::= 0 is highest priority,
#                   substitute waits (priority x broadcast) seconds
#                   before competing for mastertoken
# <member_broadcast> ::= BROADCAST time in seconds.
#                   3 is minimum.
#                   sets period of watchdog messages over the networks
#                   long periods may affect some wdx commands response time
# <member_validity> ::= VALIDITY period in seconds.
#                   BROADCAST time +1 is minimum.
#                   (3 x BROADCAST time) +1 is recommended
#                   sets WDX failure detection delay
# <member_redundancy> ::= number of sends for each watchdog message
#                   integer >= 1, 1 means no redundancy
# <network_1>        ::= <DEVICE_NAME> | "N/A"
# <network_2>        ::= <DEVICE_NAME> | "N/A"
#-----
# Example:
#-----
VERSION 1.0
DATE 8-Aug-2003
# MEMBER name type priority broadcast validity redundancy network-1 network-2
MEMBER green MASTER/SUBSTITUTE 1 3 10 1 eth0 eth1
MEMBER yellow MASTER/SUBSTITUTE 1 3 10 1 eth0 eth1
MEMBER red MASTER/SUBSTITUTE 0 3 10 1 eth0 eth1
MEMBER blue SATELLITE 1 3 30 1 SE0 N/A
MEMBER orange SATELLITE 0 3 30 1 lance01 N/A
```

C.2 Example Local File (UNIX)

```
#####
#                               WitchDoctor-WDX V1.0
#
#                               Local file: db-server.loc
#####
# Local file can be different on every member in the mission.
#-----
# for mission: db-server      on member: green
# by: dominique
#-----
# Header format :
# VERSION          value      2 digits identify  configuration release level
#                               value must be the same on all members
#                               used to check mission consistency
# DATE             string     This is a 20 char string to identify this file
#-----
VERSION    2.1
DATE      8-Aug-2003
#-----
# Script parameters are local parameters. They can be different on every
# member in the mission.
# Script parameters format :
# <script_line>          ::= <script_name> <script_path>
# <script_name>          ::= NEWMASTERSCRIPT | NOSUBSTITUTESCRIPT |
#                               SATELLITESCRIPT | EXCEPTIONSCRIPT | EVENTSCRIPT
# <script_path>          ::= any script file name
#                               an error in the path name is undetected until
#                               it is used.
#                               The content of script is not verified by WDX
#                               The end of script execution is not monitored by WDX
#                               All script activations must be tested
#                               after local file is modified
#-----
NEWMASTERSCRIPT          /var/shaman-x/wdx/db-server/db-server.newmasterscript
NOSUBSTITUTESCRIPT      /var/shaman-x/wdx/db-server/db-server.nosubstitutescript
SATELLITESCRIPT         /var/shaman-x/wdx/db-server/db-server.satellitescript
EXCEPTIONSCRIPT         /var/shaman-x/wdx/db-server/db-server.exceptionscript
EVENTSCRIPT             /var/shaman-x/wdx/db-server/db-server.eventsript
#-----
# Transition parameters are local parameters. They can be different on each
# member in the mission.
# Transition parameters format :
# <transition_line>      ::= <transition_name> <YES | NO>
#                               YES: the EVENTSCRIPT is started in this case
#                               NO:  the EVENTSCRIPT is NOT started in this case
# name value
#-----
NETWORK_UNKNOWN_TO_ON          NO
NETWORK_UNKNOWN_TO_OFF        NO
NETWORK_ON_TO_OFF             NO
NETWORK_ON_TO_UNKNOWN         NO
NETWORK_OFF_TO_UNKNOWN        NO
NETWORK_OFF_TO_ON            NO
MEMBER_FAULT_TO_IDLE          YES
MEMBER_FAULT_TO_MASTER        YES
MEMBER_FAULT_TO_SUBSTITUTE    YES
MEMBER_FAULT_TO_SATELLITE     NO
MEMBER_IDLE_TO_FAULT          YES
MEMBER_IDLE_TO_MASTER         YES
```

MEMBER_IDLE_TO_SUBSTITUTE	YES
MEMBER_IDLE_TO_SATELLITE	NO
MEMBER_MASTER_TO_FAULT	YES
MEMBER_MASTER_TO_IDLE	YES
MEMBER_MASTER_TO_SUBSTITUTE	YES
MEMBER_MASTER_TO_SATELLITE	NO
MEMBER_SUBSTITUTE_TO_FAULT	YES
MEMBER_SUBSTITUTE_TO_IDLE	YES
MEMBER_SUBSTITUTE_TO_MASTER	YES
MEMBER_SUBSTITUTE_TO_SATELLITE	NO
MEMBER_SATELLITE_TO_FAULT	NO
MEMBER_SATELLITE_TO_IDLE	NO
MEMBER_SATELLITE_TO_MASTER	NO
MEMBER_SATELLITE_TO_SUBSTITUTE	NO
RESOURCE_UNKNOWN_TO_UNKNOWN	YES
RESOURCE_UNKNOWN_TO_FREE	YES
RESOURCE_UNKNOWN_TO_RESERVED	YES
RESOURCE_UNKNOWN_TO_MOUNTED	YES
RESOURCE_UNKNOWN_TO_FAULT	YES
RESOURCE_FREE_TO_UNKNOWN	YES
RESOURCE_FREE_TO_FREE	YES
RESOURCE_FREE_TO_RESERVED	YES
RESOURCE_FREE_TO_MOUNTED	YES
RESOURCE_FREE_TO_FAULT	YES
RESOURCE_RESERVED_TO_UNKNOWN	YES
RESOURCE_RESERVED_TO_FREE	YES
RESOURCE_RESERVED_TO_RESERVED	YES
RESOURCE_RESERVED_TO_MOUNTED	YES
RESOURCE_RESERVED_TO_FAULT	YES
RESOURCE_MOUNTED_TO_UNKNOWN	YES
RESOURCE_MOUNTED_TO_FREE	YES
RESOURCE_MOUNTED_TO_RESERVED	YES
RESOURCE_MOUNTED_TO_MOUNTED	YES
RESOURCE_MOUNTED_TO_FAULT	YES
RESOURCE_FAULT_TO_UNKNOWN	YES
RESOURCE_FAULT_TO_FREE	YES
RESOURCE_FAULT_TO_RESERVED	YES
RESOURCE_FAULT_TO_MOUNTED	YES
RESOURCE_FAULT_TO_FAULT	YES
MISMATCH	NO

C.3 Example Support File

The following file is a template support file mission.sup.

```
-----
#####
#                               WitchDoctor-WDX V1.0
#
#                               Support file: db-server.sup
#####
# Support parameters are local parameters. They can be different on every
# member in the mission.
#-----
# for mission: db-server      on member: green
# by: dominique
# VERSION 1.0
# DATE 13-Aug-2003
#       I swear I read the manual before playing with this file :-((
#-----
# Format :
# Keyword           Possible values           Default (lines are optional)
#
# LOG_FLAG          DISABLE OR ENABLE           ENABLE
#                  DISABLE stops WDX logging
# LOG_FILE          log file name           mission_name.log
#                  Alternate log file or device for convenient testing
# PRIORITY          Process priority level   highest
#                  Low priority level is useful for testing
#                  Low priority can be used for monitoring system load
# PERIOD            n (seconds)             1
#                  WDX activation period (badly tested if not 1)
# ACTIVITY          DISABLE OR ENABLE       DISABLE
#                  Self monitoring of WDX
# THRESHOLD        member_name nb_seconds   No threshold
#                  Alert if broadcast messages from member_name
#                  are not received on time
# STATISTICS        nb_seconds              No statistics
#                  Additionnal diagnostic tool
#-----
# Example:
#
LOG_FLAG    ENABLE
LOG_FILE    /var/shaman-x/wdx/wdx.log
PRIORITY    +20
PERIOD      1
ACTIVITY    DISABLE
THRESHOLD   yellow 4
STATISTICS  60
```

D Sample General Management Files

D.1 Overview

The following UNIX examples show the scripts written for the "wdx" mission, which include the following characteristics:

- computers wdxm1 and wdxm2 compete for mastership,
- computers wdxm3 and wdxm4 are satellites,
- each computer has two Ethernet interfaces configured "ln0" and "ln1",
- Masterholder runs as a server for the SERVER application,
- Masterholder mounts two common file systems comdisk1 and comdisk2,
- Substitute is the backup for the SERVER application,
- Substitute mounts NFS the common disks comdisk1 and comdisk2,
- satellites connect to the Masterholder using Telnet to access the SERVER application.

Note

Telnet connections to the Masterholder member are transparent using IP floating addresses, for example server_address. In this situation, the Masterholder is always seen as server_address from any computer.

D.2 Sample NOSUBSTITUTESCRIPT

```
wdx.nosubstitutescript
-----
#!/bin/sh
echo "WDX: Substitute no longer seen by 'hostname' at '/bin/date'" >/dev/console
cat >/tmp/foo <<EOF
Hello

This mail is from node 'hostname' which is '/etc/wdx -status' in the "wdx"
mission.
You are informed that Substitute is no longer available for the mission.
Please check.
Regards
NOSUBSTITUTESCRIPT
.
EOF
#
# DECnet mail to administrator
#
mail -s 'WDX message' admin::admin </tmp/foo
```

D.3 Sample NEWMASTERSCRIPT

```
wdx.newmasterscript
-----
/bin/date +'hostname' becomes MASTERHOLDER %H:%M:%S" >/dev/console
/bin/echo "Failover starting by process number $$" >/dev/console
wall <<eof
Failover by 'hostname' is starting
eof
#
# Unmount NFS disks, then start as a MASTERHOLDER
#
umount /comdisk1
umount /comdisk2
```

```

/var/shaman-x/wdx/wdx.rc.masterholder
wall <<eof
Failover terminated
eof

```

D.4 Sample SATELLITESCRIPT

```

#!/bin/sh
/bin/date +"MASTERHOLDER CHANGED at %H:%M:%S " >/dev/console
wall <<eof
New Masterholder available for mission wdx
Use "telnet SERVER_ADDRESS" to re-start your client application
eof

```

D.5 Sample EXCEPTIONSCRIPT

```

#!/bin/sh
/bin/date +"exceptionscript triggered by WDX %H:%M:%S" >/dev/console
case $1 in
DOUBLEMASTER)
#
# Double master is detected :
# unmount the common disks
# stop SERVER application
# mount NFS the masterholder common disks
#
umount /comdisk1
umount /comdisk2
/bin/SERVER_STOP
/etc/mount -o rw `/etc/wdx -nfs /comdisk2' /comdisk2 >/dev/console 2>&1
/etc/mount -o rw `/etc/wdx -nfs /comdisk1' /comdisk1 >/dev/console 2>&1
;;
EXCEPTION)
#
# The operator asked for a general shutdown of all nodes
# as a predefined action
#
shutdown -r now > /dev/console
;;
esac

```

D.6 Sample EVENTSCRIPT

```

#!/bin/sh
#
# Print the script arguments
#/
bin/date +"WDX event $1 $2 $3 $4 $5 $6 %H:%M:%S" >/dev/console

```

D.7 Sample UNIX rc.local Script

```

rc.local
-----
# @(#)rc.local 4.3 (UNIX) 1/25/91
/bin/hostname wdxm2
/etc/ifconfig qe0 16.188.192.58 broadcast 16.255.255.255 netmask 255.0.0.0

```

```

/etc/ifconfig ne0 17.188.192.58 broadcast 17.255.255.255 netmask 255.0.0.0
/etc/ifconfig lo0 localhost
#
# Site specific stuff here
#
#
# Start WDX
#
/etc/wdx -process >/dev/console 2>&1
#
# Determine who will be masterholder and execute appropriate rc.script
/etc/wdx -list >/dev/console
/etc/wdx -master >/dev/console 2>&1
case $? in
0) [ -f /var/shaman-x/wdx/wdx/wdx.rc.masterholder ] && /var/shaman-
  x/wdx/wdx/wdx.rc.masterholder ;;
1) [ -f /var/shaman-x/wdx/wdx/wdx.rc.substitute ] && /var/shaman-
  x/wdx/wdx/wdx.rc.substitute ;;
2) [ -f /var/shaman-x/wdx/wdx/wdx.rc.satellite ] && /var/shaman-
  x/wdx/wdx/wdx.rc.satellite ;;
esac
#!/bin/sh
#!This is wdx.rc.masterholder
#!This file is executed at boot time when the computer detects
#!that it is masterholder.
#!
echo "MASTERHOLDER starts CRITICALSERVER application ..." >/dev/console
#
# Mount the common disks then starts the server application
#
/etc/wdx -declare /comdisk1
/etc/wdx -declare /comdisk2
if ( /etc/chpt /dev/rrz2e > /dev/null ) then
/etc/fsck /dev/rz2g > /dev/console
/etc/mount /dev/rz2g /comdisk1 > /dev/console
/etc/wdx -mount /dev/comdisk1
else
echo Could not access /dev/rz2
fi
#
if ( /etc/chpt /dev/rrz3e > /dev/null ) then
/etc/fsck /dev/rz3g > /dev/console
/etc/mount /dev/rz3g /comdisk2 > /dev/console
/etc/wdx -mount /dev/comdisk2
else
echo Could not access /dev/rz3
fi
#
/bin/SERVER_START > /dev/console
#!/bin/sh
#!This is wdx.rc.substitute
#!This file is executed at boot time when the computer detects
#!that it is Substitute.
#!
echo "SUBSTITUTE is mounting/NFS common disks ..." >/dev/console
/etc/wdx -join
/etc/mount -o rw '/etc/wdx -nfs /comdisk2' /comdisk2 >/dev/console 2>&1
/etc/mount -o rw '/etc/wdx -nfs /comdisk1' /comdisk1 >/dev/console 2>&1

```


E Sample Resource Management Files

E.1 Overview

This example explains how to expedite the daily transfer of a file through an X25 communication line, using a specific application called TRANSFER. The file is located on the FILE server (the FILE server is secured using two computers which share data disks, and manage by another mission). The "transfer" mission includes the following characteristics:

- computers wdxm1 and wdxm2 and wdxm3 compete for mastership,
- each computer has two Ethernet interfaces configured "In0" and "In1",
- computer wdxm2 has an X25 communication device attached to it and the line is called L2,
- an X25 gateway is connected to the Ethernet network. It has an X25 communication line called L1,
- wdxm1, wdxm2 and wdxm3 have a copy of an application called /bin/TRANSFER.
- /bin/TRANSFER is a specific application able to copy a file through X25.
- /bin/TRANSFER returns the following values:
 - 0 : success,
 - -1 : remote computer unreachable,
 - -2 : line unavailability.

Each day at 10 pm, the Masterholder starts the copy of the file using TRANSFER. wdxm1 and wdxm3 transfers using L1, and wdxm2 transfers using either L1 or L2. In any case of failure, before or during file transfer, recovery will automatically happen and file transfer will be achieved by backup hardware.

E.2 Sample UNIX rc.local Script for TRANSFER Mission

```
rc.local
-----
# @(#)rc.local 4.3 (UNIX) 6/23/93
/bin/hostname wdxm2
/etc/ifconfig qe0 16.188.192.58 broadcast 16.255.255.255 netmask 255.0.0.0
/etc/ifconfig ne0 17.188.192.58 broadcast 17.255.255.255 netmask 255.0.0.0
/etc/ifconfig lo0 localhost
#
# Site specific stuff here
#
#
# Start WDX and try to become Masterholder
#
echo "Start transfer mission"
/etc/wdx transfer -process >/dev/console 2>&1
/etc/wdx transfer -master >/dev/console 2>&1
#
# If resources L1 and L2 are unknown then declare them
#
if ( /etc/wdx transfer -list | egrep '^L1 > /dev/null ) then
echo "L1 already declared" > /dev/console
else
/etc/wdx transfer -declare L1 > /dev/console
fi
if ( /etc/wdx transfer -list | egrep '^L2 > /dev/null ) then
echo "L2 already declared" > /dev/console
else
/etc/wdx transfer -declare L2 > /dev/console
fi
```

E.3 Sample /etc/crontab File

```
# Start wdx.crontab script at 10pm daily
0 22 * * * /var/shaman-x/wdx/transfer/transfer.crontab
```

E.4 Sample /var/shaman-x/wdx/transfer.crontab

```
#!/bin/sh
# /var/shaman-x/wdx/transfer/transfer.crontab
#
# It is time to transfer : Masterholder declares the "TRANSFER" resource
#
wdx_status=`/etc/wdx transfer -status`
case $wdx_status in
masterholder)
/etc/wdx/transfer -free L1
/etc/wdx/transfer -free L2
/etc/wdx transfer -declare TRANSFER
;;
*)
;;
esac
```

E.5 Sample /var/shaman-x/wdx/transfer.eventscript

```
#!/bin/sh
# /var/shaman-x/wdx/transfer/transfer.eventscript
#
# In this file, manage only L1, L2, TRANSFER and TRANSFER_DONE resources.
#
case $1 in
RESOURCE)
case $2 in
L1)
echo WDX resource L1 from $3 to $4 by $5 > /dev/console
;;
L2)
echo WDX resource L2 from $3 to $4 by $5 > /dev/console
;;
TRANSFER)
/var/shaman-x/wdx/transfer/transfer.eventscript.TRANSFER $3 $4 $5
;;
TRANSFER_DONE)
echo WDX resource TRANSFER_DONE from $3 to $4 by $5 > /dev/console
;;
esac
;;
*)
;;
esac
#!/bin/sh
# /var/shaman-x/wdx/transfer/transfer.eventscript.TRANSFER
#
case $2 in
reserved)
line_status=`/etc/wdx transfer -list | egrep '^L1 | awk '{print $2}'`
if test ${L1} = "reserved" then
```

```

line_name=L1
fi
line_status=`/etc/wdx transfer -list | egrep '^L2 | awk '{print $2}'`
if test ${L2} = "reserved" then
line_name=L2
fi
if test -z $line_name
then
echo X25 lines are not available. Can't transfer tonight > /dev/console
echo Try again in 10 minutes
sleep 10
/var/shaman-x/wdx/transfer/transfer.eventscript.TRANSFER $1 $2 $3
fi
transfer_status=/bin/TRANSFER $line_name
case $transfer_status in
0)
echo transfer terminated > /dev/console
wdx transfer -declare TRANSFER_DONE
wdx transfer -free TRANSFER
wdx transfer -erase TRANSFER
wdx transfer -erase TRANSFER_DONE
wdx transfer -free ${line_name}
;;
-1)
echo Remote computer unreachable > /dev/console
echo Try again in 10 minutes > /dev/console
sleep 10
/var/shaman-x/wdx/transfer/transfer.eventscript.TRANSFER $1 $2 $3
;;
-2)
echo X25 line ${line_name} unavailable > /dev/console
wdx transfer -fault ${line_name}
wdx transfer -free TRANSFER
;;
esac
free)
if test "${1}" = "unknown"
then
/etc/wdx transfer -reserve TRANSFER
else
if ( /etc/wdx transfer -list | egrep '^TRANSFER_DONE ) then
echo transfer terminated > /dev/console
else
echo Transfer not terminated > /dev/console
echo Line or computer crashed > /dev/console
host=`/bin/hostname`
case $host in
wdxm2)
line_status=`/etc/wdx transfer -list | egrep '^L2 | awk '{print $2}'`
line_name=L1
if test "${line_status}" = "fault"
line_status=`/etc/wdx transfer -list | egrep '^L1 | awk '{print $2}'`
line_name=L2
then
fi
;;
*)
line_status=`/etc/wdx transfer -list | egrep '^L1 | awk '{print $2}'`
line_name=L2
;;

```

```
esac
if test "${line_status}" = "free"
then
/etc/wdx transfer -reserve ${line_name}
/etc/wdx transfer -reserve TRANSFER
fi
fi
fi
;;
esac
```

F WitchDoctor-WDX messages

Message format

The format of messages in the mission log file is:

```
severity: date, message
```

where:

- severity can have one of the values:
 - INFO: information messages used to log all the state transition events.
 - WARNING: warning messages used to log non-fatal errors.
 - ERROR: fatal error messages.
- date is the date and time at which the message was logged. date is expressed in Greenwich time,
- message is the text message reported by wdx.

INFO Messages

Message text

Description

Process supports Masterholder/Substitute features	When wdx process starts
Process supports Satellite features only	When wdx starts
Wdx started	When wdx process starts
Local member is running	Setup command
All members are running	Run command
Mission started	Start command
Mission stopped	Stop command
Mission paused	Pause command
Exception script triggered	Exception command
Local wdx process killed	Prockill command
All wdx processes in mission killed	Kill command
Log file closed by operator	Log command
Log file opened by operator	Log command
Lock request "request" for resource from member	Lock protocol request message
Lock answer "answer" for resource from member	Lock protocol answer message
Lock transition "transition" for resource from member	Lock protocol transition message

Transition message text

Condition

Resource "resource" status by member	On the member where the transition is detected
Resource MASTERTOKEN status by member	On the member where the transition is detected
For member, network number (interface) old_status to new_status	On the member where the transition is detected
member member old_status to new_status	On the member where the transition is detected
Satellite detects Masterholder disappearance	On each satellite when Masterholder failure is detected
Satellite detects Masterholder change	On each satellite when Masterholder change is detected
Masterholder detects Substitutes disappearance	On Masterholder when Substitutes are no longer available
Double Masterholder detected	On Masterholder which triggers the EXCEPTIONSCRIPT

WARNING messages

<u>Message text</u>	<u>Description</u>
Too many version lines listed in mission file	Only 1 VERSION line is required
Too many date lines listed in mission file	Only 1 DATE line is required
Too many nodes listed in mission file	Only 16 member lines are supported
Message received with invalid checksum	Invalid checksum message
Message received with invalid software version version (X.Y)	Processes of different versions are running on the LAN
Message received with invalid mission key (K)	Invalid mission header key. Another mission has been started on the LAN using the same UDP port number
Message received with invalid configuration file version (C)	Invalid mission file. An other member is started with a different mission file
mount_point not a local mount point directory	When user typed "wdx [mission] mount resource"
Write socket failed on interface	Wdx process failed to send a message. Have a look at your network configuration
Read socket failed	Wdx process failed to send a message. Have a look at your network configuration
Write FIFO failed	Wdx process failed to write a message to Wdx command

ERROR messages

<u>Message text</u>	<u>Description and/or action</u>
No license registered for wdx	To start WitchDoctor-WDX processes a license is required
wdx process already started	A wdx process has already been started
wdx process not started	wdx process must be started before wdx command process
Wdx path not defined	The wdx directory or environment variable does not exist
Can't get ifaddr for interface interface	Check your network configuration
Can't get netmask for interface interface	Check your network configuration
Can't get broadcast for interface interface	Check your network configuration
Communication error	Communication failed with wdx process
Can't connect to wdx process	Connection rejected by lack of resource
Can't open mission file file	The mission file file can't be found
Mission file syntax	Refer to lines below then correct mission file
Invalid keyword (keyword)	Refer to line below and correct mission file
Missing date line in mission file	Add a DATE line in mission file
Missing version line in mission file	Add a VERSION line in mission file
Missing member line in mission file	Add MEMBERs lines in mission file
Invalid mission line format for line	Refer to line below then correct mission file
Invalid keyword (keyword)	Refer to line below and correct local file
Type (type) must be either Master/Substitute or Satellite	Refer to line below and correct mission file
Priority value (priority) must be a positive integer value	Refer to line below and correct mission file
Broadcast value (broadcast) must be a non null positive integer value	Refer to line above then correct mission file
Validity value (validity) must be a positive integer value	Refer to line below then correct mission file
Validity value (validity) must be a positive integer, greater than broadcast value (broadcast)	Refer to line below then correct mission file
Redundancy value (redundancy) must be a positive integer value	Refer to line below then correct mission file
Invalid version format (version)	Correct the VERSION field
Invalid date format (date)	Correct the DATE field
Can't open local file file	The local file file can't be found
Local file syntax	Refer to lines below then correct local file
Invalid local line format for line	Refer to line below then correct local file
Mask (mask) must be either YES or NO	Refer to line below and correct local file
Can't find member name "member" in mission file	Correct mission file or startup command
Can't get hostname	Check if a hostname is defined on your system
localhost is an unknown host	Check the contents of your TCP/IP hosts database
service/UDP is an unknown service	Update your TCP/IP services database
Snapshot and mission not match	The specified snapshot doesn't refer to the current mission
Recvfrom failed	The wdx process failed to receive a message
Mission name too long	Mission names are 16 characters long
Member name too long	Member names are 32 characters long
Name of resource too long	Resource names are 20 characters long
Name of snapshot too long	Snapshot names are 8 characters long
Command restricted to superuser	Log in as superuser
Ambiguous qualifier	Enter more characters for a command qualifier
Multiple qualifier not allowed	Qualifier must be used singly
Can't open help file file	Help file missing

Wdx Command Messages

The format of Wdx Command messages is :

message

Wdx Command Success Messages

<u>Message text</u>	<u>Description</u>
Local member is running	Join command
All members are running	Run command
Mission started	Start command
Mission stopped	Stop command
Mission paused	Pause command
Exception script triggered	Exception command
Local wdx process killed	Leave command
All wdx processes in mission killed	Kill command
Log file initialised	Log command
No mission currently processing	No mission running for show command
masterholder.is.unknown	No Masterholder defined for namemaster command
resource declared	Declare command
resource erased	Erase command
resource reserved	Reserve, master, getmaster or getresource commands
resource mounted	Mount command
resource freed	Free or nomaster commands
resource faulted	Fault command

Wdx Command Error Messages

<u>Message text</u>	<u>Description</u>
Can't declare resource	Declare command
Can't erase resource	Erase command
Can't reserve resource	Reserve or master command
Can't mount resource	Mount command
Can't free resource	Free or nomaster command
Can't fault resource	Fault command
Can't get resource	Getresource or getmaster command
Command rejected by mission	Some processes for the mission rejected the command
Command not allowed : bad process status	Some commands depend on wdx process state
Config file error	Invalid start command due to configuration error
Command not allowed for satellite	Satellite doesn't support MASTERTOKEN command
wdx process terminated	Another command has terminated the process to become Masterholder
Too many resources managed	24 resources are already defined
Not a local mount point	Resource is not a local UNIX mount point
Not allowed for mount point	command not allowed for mount point
Command rejected by lock protocol	The lock protocol has rejected the command

GNU GENERAL PUBLIC LICENSE

Table of Contents
 GNU GENERAL PUBLIC LICENSE
 Preamble
 TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND
 MODIFICATION
 How to Apply These Terms to Your New Programs

GNU GENERAL PUBLIC LICENSE Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed

as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it. Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.) The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable. If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the

author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the program's name and an idea of what it does.
Copyright (C) yyyy name of author

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school,

if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Copyright notice above.
Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111, USA

Updated: Last modified: Sun Jul 15 13:13:30 CEST 2001

H GNU Free Documentation License

Version 1.2, November 2002
 Copyright (C) 2000,2001,2002 Free Software Foundation, Inc.
 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- State on the Title page the name of the publisher of the Modified Version, as the publisher.
- Preserve all the copyright notices of the Document.
- Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- Include an unaltered copy of this License.
- Preserve the section Entitled "History". Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers. If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so

long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Copyright notice above.

Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111, USA

Updated: \$Date: 2003/03/03 17:05:55 \$ \$Author: brett \$