Contents

Preface 5

JFP Reference Manual 1M : System Administration Commands 11
Intro_jfp(1M) 12
atok12mngtool(1m) 14
cssd(1M) 15
dpkeyserv(1M) 16
dpkeystat(1M) 17
jserver(1M) 18
wnnaccess(1M) 20
wnnds(1M) 22
wnnkill(1M) 23
wnnoffline(1M) 24
wnnudmerge(1M) 25
Preface

Both novice users and those familiar with the SunOS operating system can use online man pages to obtain information about the system and its features. A man page is intended to answer concisely the question “What does it do?” The man pages in general comprise a reference manual. They are not intended to be a tutorial.

Overview

The following contains a brief description of each man page section and the information it references:

- Section 1 describes, in alphabetical order, commands available with the operating system.
- Section 1M describes, in alphabetical order, commands that are used chiefly for system maintenance and administration purposes.
- Section 2 describes all of the system calls. Most of these calls have one or more error returns. An error condition is indicated by an otherwise impossible returned value.
- Section 3 describes functions found in various libraries, other than those functions that directly invoke UNIX system primitives, which are described in Section 2.
- Section 4 outlines the formats of various files. The C structure declarations for the file formats are given where applicable.
- Section 5 contains miscellaneous documentation such as character-set tables.
- Section 6 contains available games and demos.
- Section 7 describes various special files that refer to specific hardware peripherals and device drivers. STREAMS software drivers, modules and the STREAMS-generic set of system calls are also described.
Section 9 provides reference information needed to write device drivers in the kernel environment. It describes two device driver interface specifications: the Device Driver Interface (DDI) and the Driver/Kernel Interface (DKI).

Section 9E describes the DDI/DKI, DDI-only, and DKI-only entry-point routines a developer can include in a device driver.

Section 9F describes the kernel functions available for use by device drivers.

Section 9S describes the data structures used by drivers to share information between the driver and the kernel.

Below is a generic format for man pages. The man pages of each manual section generally follow this order, but include only needed headings. For example, if there are no bugs to report, there is no BUGS section. See the intro pages for more information and detail about each section, and man(1) for more information about man pages in general.

**NAME**
This section gives the names of the commands or functions documented, followed by a brief description of what they do.

**SYNOPSIS**
This section shows the syntax of commands or functions. When a command or file does not exist in the standard path, its full path name is shown. Options and arguments are alphabetized, with single letter arguments first, and options with arguments next, unless a different argument order is required.

The following special characters are used in this section:

- **[ ]** Brackets. The option or argument enclosed in these brackets is optional. If the brackets are omitted, the argument must be specified.

- **...** Ellipses. Several values can be provided for the previous argument, or the previous argument can be specified multiple times, for example, "filename ...".

- **|** Separator. Only one of the arguments separated by this character can be specified at a time.

- **{ }** Braces. The options and/or arguments enclosed within braces are interdependent, such that everything enclosed must be treated as a unit.
PROTOCOL
This section occurs only in subsection 3R to indicate the protocol description file.

DESCRIPTION
This section defines the functionality and behavior of the service. Thus it describes concisely what the command does. It does not discuss OPTIONS or cite EXAMPLES. Interactive commands, subcommands, requests, macros, and functions are described under USAGE.

IOCTL
This section appears on pages in Section 7 only. Only the device class that supplies appropriate parameters to the ioctl(2) system call is called ioctl and generates its own heading. ioctl calls for a specific device are listed alphabetically (on the man page for that specific device). ioctl calls are used for a particular class of devices all of which have an io ending, such as mtio(7I).

OPTIONS
This section lists the command options with a concise summary of what each option does. The options are listed literally and in the order they appear in the SYNOPSIS section. Possible arguments to options are discussed under the option, and where appropriate, default values are supplied.

OPERANDS
This section lists the command operands and describes how they affect the actions of the command.

OUTPUT
This section describes the output — standard output, standard error, or output files — generated by the command.

RETURN VALUES
If the man page documents functions that return values, this section lists these values and describes the conditions under which they are returned. If a function can return only constant values, such as 0 or -1, these values are listed in tagged paragraphs. Otherwise, a single paragraph describes the return values of each function. Functions declared void do not return values, so they are not discussed in RETURN VALUES.

ERRORS
On failure, most functions place an error code in the global variable errno indicating why they failed. This section lists alphabetically all error codes a function can generate and describes the conditions that cause each error. When more than
one condition can cause the same error, each
condition is described in a separate paragraph
under the error code.

USAGE
This section lists special rules, features, and
commands that require in-depth explanations. The
subsections listed here are used to explain built-in
functionality:

Commands
Modifiers
Variables
Expressions
Input Grammar

EXAMPLES
This section provides examples of usage or of how
to use a command or function. Wherever possible a
complete example including command-line entry
and machine response is shown. Whenever an
eexample is given, the prompt is shown as
example %, or if the user must be superuser,
example #. Examples are followed by explanations,
variable substitution rules, or returned values. Most
examples illustrate concepts from the SYNOPSIS,
DESCRIPTION, OPTIONS, and USAGE sections.

ENVIRONMENT VARIABLES
This section lists any environment variables that
the command or function affects, followed by a
brief description of the effect.

EXIT STATUS
This section lists the values the command returns to
the calling program or shell and the conditions that
cause these values to be returned. Usually, zero is
returned for successful completion, and values
other than zero for various error conditions.

FILES
This section lists all file names referred to by the
man page, files of interest, and files created or
required by commands. Each is followed by a
descriptive summary or explanation.

ATTRIBUTES
This section lists characteristics of commands,
utilities, and device drivers by defining the
attribute type and its corresponding value. See
attributes(5) for more information.

SEE ALSO
This section lists references to other man pages,
in-house documentation, and outside publications.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIAGNOSTICS</td>
<td>This section lists diagnostic messages with a brief explanation of the condition causing the error.</td>
</tr>
<tr>
<td>WARNINGS</td>
<td>This section lists warnings about special conditions which could seriously affect your working conditions. This is not a list of diagnostics.</td>
</tr>
<tr>
<td>NOTES</td>
<td>This section lists additional information that does not belong anywhere else on the page. It takes the form of an aside to the user, covering points of special interest. Critical information is never covered here.</td>
</tr>
<tr>
<td>BUGS</td>
<td>This section describes known bugs and, wherever possible, suggests workarounds.</td>
</tr>
</tbody>
</table>
JFP Reference Manual 1M : System Administration Commands
Intro_jfp(1M)

**NAME**
Intro_jfp, intro_jfp – introduction to JFP maintenance commands and application programs

**AVAILABILITY**
This section indicates which package contains the commands being described on this page. To be able to use the command, the indicated package must have been installed with the operating system. For information on how to add a package see pkgadd(1).

**DESCRIPTION**
This section describes, in alphabetical order, JFP commands that are used chiefly for system maintenance and administration purposes.

**COMMAND SYNTAX**
Unless otherwise noted, commands described in this section accept options and other arguments according to the following syntax:

```plaintext
name [option(s)] [cmdarg(s)]
```

where:
- **name** The name of an executable file
- **option** – noargletter(s) or, – argletter<option> where <> is optional white space
- **noargletter** A single letter representing an option without an argument
- **argletter** A single letter representing an option requiring an argument
- **optarg** Argument (character string) satisfying preceding argletter
- **cmdarg** Pathname (or other command argument) not beginning with – or, – by itself indicating the standard input

**ATTRIBUTES**
See attributes(5) for a discussion of the attributes listed in this section.

**SEE ALSO**
getopt(1), getopt(3C), attributes(5)

**DIAGNOSTICS**
Upon termination, each command returns 0 for normal termination and non-zero to indicate troubles such as erroneous parameters, bad or inaccessible data, or other inability to cope with the task at hand. It is called variously “exit code,” “exit status,” or “return code,” and is described only where special conventions are involved.

**NOTES**
Unfortunately, not all commands adhere to the standard syntax.

**LIST OF COMMANDS**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intro_jfp(1M)</td>
<td>introduction to JFP maintenance commands and application programs</td>
</tr>
<tr>
<td>atok12mngtool(1m)</td>
<td>Administrates ATOK12 user information</td>
</tr>
<tr>
<td>cssd(1M)</td>
<td>daemon which invokes and watches the CSs</td>
</tr>
<tr>
<td>dpkeyserv(1M)</td>
<td>Wnn6 license server</td>
</tr>
<tr>
<td>dpkeystat(1M)</td>
<td>displays and controls Wnn6 licensing information</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>jserver(1M)</td>
<td>Wnn6 Kana-Kanji conversion server</td>
</tr>
<tr>
<td>wnnaccess(1M)</td>
<td>Access control over Wnn6 Kana-Kanji conversion server/dictionary lookup server</td>
</tr>
<tr>
<td>wnnnds(1M)</td>
<td>Wnn6 dictionary lookup server</td>
</tr>
<tr>
<td>wnnkill(1M)</td>
<td>Terminates Wnn6 Kana-Kanji conversion server</td>
</tr>
<tr>
<td>wnnoffline(1M)</td>
<td>Updates Wnn6 dictionary database offline.</td>
</tr>
<tr>
<td>wnnudmerge(1M)</td>
<td>Wnn6 registered words automatic merge module</td>
</tr>
</tbody>
</table>
atok12mngtool(1m)

NAME  atok12mngtool – Administrates ATOK12 user information

SYNOPSIS  /usr/sbin/atok12mngtool

AVAILABILITY  JSatsvu

DESCRIPTION  The atok12mngtool command provides ATOK12 user information administration operations. This utility needs to be executed by superuser, and multiple processes cannot run simultaneously.

When invoked, atok12mngtool enters its command interpreter and awaits instructions from the user. When atok12mngtool is awaiting commands from the user, it displays the prompt Command>.

OPTIONS  The following options may be specified to the command interpreter:

a  Adds (registers) a user with a specified name. Usually you do not need to use this option because any user who attempts to use ATOK12 will be registered automatically.

d  Deletes (Unregisters) a user with a specified name.

l  Lists names of users who have been registered to use ATOK12. atok12mngtool starts printing user information after all user information has been processed. A plus symbol (+) will be printed every after 100 users are processed.

q  Quits this utility.

EXIT STATUS  The following exit values are returned:

0  Successful completion.

>0  An error occurred.

FILES  /usr/sbin/atok12mngtool

   ATOK12 user information administration utility

SEE ALSO  environ(5)

Japanese Input System Summary & Transition

ATOK12 User’s Guide
NAME  cssd – daemon which invokes and watches the CSs

SYNOPSIS  /usr/sbin/cssd [-f filename]

AVAILABILITY  SUNWjfpu

DESCRIPTION  cssd is the command which invokes and watches CS available in MLE (Multi Language Environment). After five seconds of the ending (including abnormal termination) of a CS, cssd re-invokes the CS.

CS STARTING INFORMATION FILE  In a CS starting information file, /etc/css.conf by default, some CS starting information directories are described. This file can be customized. cssd reads /etc/css.conf to get CS starting information directories, and then it invokes all executable files in the directories. Usually, each CS itself is not placed in the directories and is invoked indirectly from a script in one of the directories.

CS STARTING SCRIPT  A CS starting script is located in a CS starting information directory. cssd reads all CS starting information directories in the sequence specified in /etc/css.conf and finds to invoke all CS starting scripts. If two or more scripts which have the same filename are found, cssd invokes only the first one.

UPDATING CS STARTING INFORMATION  When cssd receives signal SIGHUP, it re-reads /etc/css.conf and re-finds all CS starting scripts, and then restarts, starts or terminates each script according to the following conditions.

A script whose modified time is changed (is to be restarted) A script which is newly added (is to be started) A script which is deleted (is to be terminated)

TO TERMINATE  When cssd receives signal SIGTERM, it sends SIGTERM to each CS under the cssd’s management and terminates itself (in general, each CS also terminates with the signal).

ERROR INFORMATION  As a facility of daemon, cssd sends error information at level ‘err’, and invoke/re-invoke information at level ‘notice’ to syslogd(1M).

OPTION  -f Specify a CS starting information file. Without this option, filename /etc/css.conf is used.

FILE  /etc/css.conf file for the information of CSs by default

SEE ALSO  css.conf(4), syslogd(1M)

BUGS  If a CS starting script is programmed so that itself is exec-ed by a CS, cssd understands that the process of the CS starting script is terminated in spite of the CS’s termination, and so puts syslog message out.
**NAME**

dpkeyserv - Wnn6 license server

**SYNOPSIS**


**DESCRIPTION**

dpkeyserv controls licensing information on the Wnn6 Kana–Kanji conversion server jserver. dpkeyserv normally starts by /etc/rc2.d/S94Wnn6 on system startup. The licensing information is stored in /etc/lib/locale/ja/wnn/dpkeylist.

**OPTIONS**

The following options can be specified.

- `-f file`
  Uses `file` as licensing information instead of the standard `kpkeylist`.

- `-a file`
  Uses `file` as access control file instead of the standard `dpkeyallow`.

- `-l`
  Stores the activity of dpkeyserv in the file specified with the environment variable `DPKEY_LOGFILE`. If this environment variable is not set, stores it in `/var/locale/ja/wnn/log/dpkeyserv.log`.

- `-L`
  Like the `-l` option, stores the activity of dpkeyserv. Also stores the dump of communication packets.

**FILES**

/etc/lib/locale/ja/wnn/dpkeylist
  Licensing information file

/etc/lib/locale/ja/wnn/dpkeyallow
  Access control file

**ATTRIBUTES**

See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

**SEE ALSO**

dpkeystat(1M)
dpkeystat - displays and controls Wnn6 licensing information

SYNOPSIS
/usr/lib/locale/ja/wnn/dpkeystat [-D host] [-h host] [-a application]
    [-u user] [-i interval] [ [-y] number]

DESCRIPTION
dpkeystat displays the status of license use and removes the licenses being used.

OPTIONS
The following options can be specified.

-D host
  Specifies the host on which dpkeyserv is running. If this option is omitted, the server access file
  /etc/lib/locale/ja/wnn/dpkeyservlist is referred. If the server access file is not found, the connection will be tried in the following order.
  1. localhost
  2. UNIX domain socket

-h host
  Specifies the host on which Kana-Kanji conversion server jserver is running. If this option is omitted, the connection is tried in the following order.
  1. localhost
  2. UNIX domain socket

-a application

-u user
  Displays user licensing information.

-i interval
  Repeats display of user licensing information at the interval specified until terminated by SIGINT and the like.

 [-y] number
  Specifies the license number to be removed. If the -y option is specified, the license is removed without confirmation.

FILES
/etc/lib/locale/ja/wnn/dpkeyservlist
  Server access file

ATTRIBUTES
See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

SEE ALSO
dpkeyserv(1M)
NAME
jserver – Wnn6 Kana-Kanji conversion server

SYNOPSIS
[-N no] [-pno port_no] [-ds default_wnnds_list] [+ ds]

DESCRIPTION
jserver provides one or more users with Kana–Kanji conversion features for
Japanese input. It is launched normally from /etc/rc2.d/S94Wnn6 upon the system
startup. jserver reads and processes the initialization file
/etc/lib/locale/ja/wnn/ja/jserverrc when launched.

OPTIONS
The following options are available.

- f file
  Reads as the initialization file instead of the default initialization
  file /etc/lib/locale/ja/wnn/ja/jserverrc.

- s file
  Stores log in file. When "-" is specified for file, the output is
directed to the standard output.

- h file
  Reads as the part of speech data file instead of the default part of
  speech data file /usr/lib/locale/ja/wnn/ja/hinsi.data.

- A file
  Reads as the access control file instead of the default access
  control file /etc/lib/locale/ja/wnn/wnnhosts.

- N no
  Uses the value obtained by adding no to the well-known port
  number (22273), as TCP port number.

- pno port_no
  Uses port_no as TCP port number instead of the well-known port
  number.

- ds
  Specifies the list of the default dictionary lookup servers. One of
  the following formats is used for dictionary lookup server.

  hostname
    Specifies the dictionary lookup server that uses
    the well-known port number (26208) on host
    hostname.

  hostname:no
    Specifies the dictionary lookup server that uses
    the well-known port number + no as port
    number on host hostname.

  hostname/port_no
    Specifies the dictionary lookup server that uses
    port_no as port number on host hostname.

Up to three dictionary lookup server can be specified by
separating them with a ",".

Example:

hostname1, hostname2:10, hostname/26228
When this option is used, `jserver` tries to search and connect to the dictionary lookup server in the order specified on the startup. The dictionary lookup server first connected is used as default server.

If this option is omitted and the default dictionary lookup server is defined in the initialization file `jserverrc`, the default server is tried.

If either this option is not specified or the default dictionary lookup server is not defined in the initialization file, `jserver` does not connect to the dictionary lookup server by default but uses relevant dictionary files directly.

**FILES**

<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/tmp/jd_sockV6</code></td>
<td>UNIX domain socket</td>
</tr>
<tr>
<td><code>/etc/lib/locale/ja/wnnja/jserverrc</code></td>
<td><code>jserver</code> initialization file</td>
</tr>
<tr>
<td><code>/etc/lib/locale/ja/wnn/wnnhosts</code></td>
<td>Wnn6 access control file</td>
</tr>
<tr>
<td><code>/usr/lib/locale/ja/wnn/ja/hinsi.data</code></td>
<td>Wnn6 part of speech data file</td>
</tr>
<tr>
<td><code>/usr/lib/locale/ja/wnn/ja/dic/iwanami/*</code></td>
<td>Wnn6 system dictionary</td>
</tr>
</tbody>
</table>

**ATTRIBUTES**

See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

**SEE ALSO**

`wnnstat(1)`, `wnnaccess(1M)`, `wnnds(1M)`, `wnnkill(1M)`, `jserverrc(4)`, `wnnhosts(4)`
NAME
wnnaccess – Access control over Wnn6 Kana-Kanji conversion server/dictionary lookup server

SYNOPSIS
/usr/sbin/wnnaccess [-D jserver_name] [-ds wnnds_name] [-L LANG_name]
[- | + [hostname | username]]

DESCRIPTION
wnnaccess adds or deletes hosts and users, obtains the current access control information, enables or disables access control for access permission retained by the Wnn6 Kana–Kanji conversion server/dictionary lookup server.

OPTIONS
The following options are available.

-D jserver_name Specifies the Wnn6 Kana–Kanji conversion server (jserver) to work on. If this option is omitted, wnnaccess refers to the serverdefs file and determines the Kana–Kanji conversion server to work on.

The Kana–Kanji conversion server must be specified in the following format.

hostname Kana–Kanji conversion server that uses the well-known port number (22273) on host hostname
hostname:no Kana–Kanji conversion server that uses port number of "the well-known port number plus no" on host hostname
hostname/port_no Kana–Kanji conversion server that uses port_no as port number on host hostname

-D ds wnnds_name Specifies the Wnn6 dictionary lookup server (wnnds) to work on.

The dictionary lookup server must be specified in the following format.

hostname Dictionary lookup server that uses the well-known port number (26208) on host hostname
hostname:no Dictionary lookup server that uses port number of "the well-known port number plus no" on host hostname
hostname/port_no Dictionary lookup server that uses port_no as port number on host hostname

-D LANG_name Specifies the language. The language is used to refer to the serverdefs file and determine the server to connect. Specify ja for the Solaris releases (Japanese version).
If no options are specified, `wnnaccess` writes the current access control information. The first line shows the current status of access control enabled or disabled. The second and following lines show hosts and users that are given access rights as in any of the following formats:

- `host–name` All users can access from this host.
- `host–name:` No users can access from this host.
- `host–name:user–name,user–name,…` Only users shown in the list can access from this host.
- `@user–name` This user can access from any host.

Specifying `-` option only enables access control. Specifying `+` option only disables access control. If access control is disabled, no restriction is imposed on connection to the server.

If `+` option and `hostname | username` are specified, the host or user will be added to the access control. If `-` option and `hostname | username` are specified, the host or user will be deleted from the access control. The format of `hostname | username` is the same as in displaying the current access control information. If neither `-` nor `+` option is specified, the behavior will be the same as in specifying `+` option.

**ATTRIBUTES**

See `attributes(5)` for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

**SEE ALSO** `jserver(1M), wnds(1M), wnnhosts(4), wnn_serverdefs(4)`
wnnds(1M)

NAME  wnnds – Wnn6 dictionary lookup server

SYNOPSIS  /usr/lib/locale/ja/wnn/wnnds [-s file] [-A file] [-N no] [-pno port_no]

DESCRIPTION  wnnds provides multiple Wnn6 Kana–Kanji conversion server (jserver) with
dictionary lookup-related processing for Wnn6 Kana–Kanji conversion. wnnds is
normally started by the /etc/rc2.d/S94Wnn6 command on the system startup.

OPTIONS  The following options are available.

- s file  Saves log in file. If "-" is specified for the file, the log file is directed to the
           standard error.

- A file  Reads file as access control file instead of the default access control file
           /etc/lib/locale/ja/wnn/wnnhosts.

- N number  Uses the value obtained by adding number to the well-known port number
           (26208) as TCP port number.

- pno port_no  Uses port_no as TCP port number instead of the well-known port number.

FILES  /etc/lib/locale/ja/wnn/wnnhosts
       Wnn6 access control file

       /usr/lib/locale/ja/wnn/ja/dic/iwanami/*
       Wnn6 system dictionary

ATTRIBUTES  See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

SEE ALSO  jserver(1M), wnnaccess(1M), wnnhosts(4)
**NAME**
wnnkill – Terminates Wnn6 Kana-Kanji conversion server

**SYNOPSIS**
/usr/sbin/wnnkill

**DESCRIPTION**
wnnkill terminates the Wnn6 Kana–Kanji conversion server (jserver) operating on the same host.

**ATTRIBUTES**
See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

**SEE ALSO**
jserver(1M)
NAME
wnnoffline – Updates Wnn6 dictionary database offline.

SYNOPSIS
/usr/lib/locale/ja/wnn/wnnoffline [-r file] [-l file]

DESCRIPTION
wnnoffline performs the following tasks to improve the Wnn6 conversion efficiency and save resource use.

- Updates frequency information for user dictionaries, frequency dictionaries, flexible intelligence-related user dictionaries, and flexible intelligence-related frequency dictionaries.
- Deletes unnecessary words.
- Starts the registered words automatic module.

wnnoffline is usually set to start through the cron command. The tasks that wnnoffline performs are defined in /etc/lib/locale/ja/wnn/offlinerc.

OPTIONS
The following options are available.

- `-r file` Reads file as initialization file instead of the default offlinerc file.
- `-l file` Saves the wnnoffline log in file.

FILES
/etc/lib/locale/ja/wnn/offlinerc

ATTRIBUTES
See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

SEE ALSO
jserver(1M), wnnudmerge(1M)
wnnudmerge merges words that are registered in multiple user dictionaries into a single dictionary. The merge method is defined in the /etc/lib/locale/ja/wnn/udmergerc file. Note that wnnudmerge should be run as part of the tasks performed by wnnoffline. Do not run wnnudmerge independently.

The following options are available.

- `n` wnnudmerge checks the udmergerc file. Does not merge dictionaries.
- `f file` Reads file as initialization file instead of the default udmergerc file.

ATTRIBUTES

See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Attribute Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWjwnsu</td>
</tr>
</tbody>
</table>

SEE ALSO jserver(1M), wnnoffline(1M)