NAME

devlinks – adds /dev entries for miscellaneous devices and pseudo-devices

SYNOPSIS

/usr/sbin/devlinks [**–d**] [**–r** *rootdir*] [**–t** *table-file*]

AVAILABILITY

SUNWcsu

DESCRIPTION

devlinks creates symbolic links from the **/dev** directory tree to the actual block- and character-special device nodes under the **/devices** directory tree. The links are created according to specifications found in the *table-file* (by default **/etc/devlink.tab**).

devlinks is called each time the system is reconfiguration-booted, and can only be run after **drvconfig**(1M) is run, since **drvconfig**(1M) builds the kernel data structures and the **/devices** tree.

The *table-file* (normally **/etc/devlink.tab**) is an ASCII file, with one line per record. Comment lines, which must contain a hash character ('**#**') as their first character, are allowed. Each entry must contain at least two fields, but may contain three fields. Fields are separated by single TAB characters.

The fields are:

devfs-spec Specification of devinfo nodes that will have links created for them. This specification consists of one or more keyword-value pairs, where the keyword is separated from the value by an equal-sign ('**=**'), and keyword-value pairs are separated from one another by semicolons.

The possible keywords are:

type The devinfo device type. Possible values are specified in **ddi_create_minor_node**(9F).

name The name of the node. This is the portion of the **/devices** tree entry name that occurs before the first '**@**' or '**:**' character.

addr[*n*] The address portion of a node name. This is the portion of a node name that occurs between the '**@**' and the '**:**' characters. It is possible that a node may have a name without an address part, which is the case for many of the pseudo-device nodes. If a number is given after the *addr* it specifies a match of a particular comma-separated subfield of the address field: *addr1* matches the first subfield, *addr2* matches the second, and so on. *addr0* is the same as *addr* and matches the whole field.

minor[*n*] The minor portion of a node name – the portion of the name after the '**:**'. As with *addr* above, a number after the *minor* keyword specifies a subfield to match.

Of these four specifications, only the *type* specification must always be present.

name Specification of the **/dev** links that correspond to the devinfo nodes. This field allows **devlinks** to determine matching **/dev** names for the **/devices** nodes it has found. The specification of this field uses escape-sequences to allow portions of the **/devices** name to be included in the **/dev** name, *or* to allow a counter to be used in creating node names. If a counter is used to create a name, the portion of the name before the counter must be specified absolutely, and all names in the **/dev/**-subdirectory that match (up to and including the counter) are considered to be subdevices of the same device. This means that they should all point to the same directory, **name** and **address** under the **/devices/**-tree

The possible escape-sequences are:

\D Substitute the **device-name** (name) portion of the corresponding devinfo node-name.

\A*n* Substitute the *n*th component of the **address** component of the corresponding devinfo node name. Sub-components are separated by commas, and sub-component **0** is the whole **address** component.

\M*n* Substitute the *n*th sub-component of the **minor** component of the corresponding devinfo node name. Sub-components are separated by commas, and sub-component **0** is the whole **minor** component.

\N*n* Substitute the value of a 'counter' starting at *n*. There can be only one counter for each dev-spec, and counter-values will be selected so they are as low as possible while not colliding with already-existing link names.

In a dev-spec the **counter** sequence should not be followed by a digit, either explicitly or as a result of another escape-sequence expansion. If this occurs, it would not be possible to correctly match already-existing links to their counter entries, since it would not be possible to unambiguously parse the already-existing **/dev**-name.

extra-dev-link Optional specification of an extra **/dev** link that points to the initial **/dev** link (specified in field 2). This field may contain a **counter** escape-sequence (as described for the *dev-spec* field) but may not contain any of the other escape-sequences. It provides a way to specify an alias of a particular **/dev** name.

OPTIONS

–d Debugging mode – print out all **devinfo** nodes found, and indicate what links would be created, but do not do anything.

–r *rootdir* Use *rootdir* as the root of the **/dev** and **/devices** directories under which the device nodes and links are created. Changing the root directory does not change the location of the **/etc/devlink.tab** default table, nor is the root directory applied to the filename supplied to the **–t** option.

–t *table-file* Set the table file used by **devlinks** to specify the links that must be created. If this option is not given, **/etc/devlink.tab** is used. This option gives a way to instruct **devlinks** just to perform a particular piece of work, since just the links-types that **devlinks** is supposed to create can be specified in a command-file and fed to **devlinks**.

ERRORS

If **devlinks** finds an error in a line of the *table-file* it prints a warning message on its standard output and goes on to the next line in the *table-file* without performing any of the actions specified by the erroneous rule.

If it cannot create a link for some filesystem-related reason it prints an error-message and continues with the current rule.

If it cannot read necessary data it prints an error message and continues with the next *table-file* line.

EXAMPLES

Example **/etc/devlink.tab** fields are:

```
type=pseudo;name=win        win\M0
type=ddi_display        framebuffer/\M0        fb\N0
```

The first example states that all devices of type **pseudo** with a name component of **win** will be linked to **/dev/win***x*, where *x* is the minor-component of the *devinfo-name* (this is always a single-digit number for the **win** driver).

The second example states that all devinfo nodes of type **ddi_display** will be linked to entries under the **/dev/framebuffer** directory, with names identical to the entire minor component of the **/devices** name. In addition an extra link will be created pointing from **/dev/fb***n* to the entry under **/dev/framebuffer**. This entry will use a counter to end the name.

FILES

/dev	entries for the miscellaneous devices for general use
/devices	device nodes
/etc/devlink.tab	the default rule-file

SEE ALSO

ddi_create_minor_node(9F), **disks**(1M), **drvconfig**(1M), **ports**(1M), **tapes**(1M)

BUGS

It is very easy to construct mutually-contradictory link specifications, or specifications that can never be matched. The program does not check for these conditions.