NAME
nsr_label – NetWorker resource type “NSR label”

SYNOPSIS
type: NSR label

DESCRIPTION
Each NSR label template is described by a single resource of type NSR label (see nsr_resource(5)). To edit the NSR label resources for a NetWorker server, type:
nsradmin -c "type:NSR label"
or use the nwadmin(8) GUI. See the nsradmin(8) manual page for more information on using the NetWorker administration program.

This resource describes the templates used to generate volume labels.

ATTRIBUTES
The following attributes are defined for resource type NSR label. The information in parentheses describes how the attribute values are accessed. Several additional attributes (e.g. administrator) are common to all resources, and are described in nsr_resource(5).

name  (create only, single string, static)
This attribute specifies the name of this label template. The label template is referred to by its name in the jukebox resource, see nsr_jukebox(5).
Example: name: Default;

fields  (read/write, list of strings)
This attribute specifies the constituent fields of a label template. When generating a volume name, the current value of each field is concatenated. The first field is considered the most significant, the last field the least. If there is a separator (see below) defined, then it will be placed between fields as they are concatenated to form a volume name. The fields are separated by commas.
There are four different types of fields: ‘numeric range’, ‘lower-case range’, ‘upper-case range’, and a ‘list of strings’. A ‘list of strings’ consists of space (‘ ’) separated strings. The other types are specified as starting and ending values separated by a dash (‘-’). The starting and ending values of a range must have the same number of characters.
The next attribute (see below) contains the current position or value of each field. After a volume name has been assigned to a volume, the next attribute is incremented. When the ending value is reached, the current value will wrap around to the starting value. A ‘list of strings’ field is incremented by selecting the next string in the list. A numeric range field is incremented by adding 1 to its current value. Lower-case and upper-case ranges are incremented by moving on to the next letter in the least significant position. In the example below, after aa.99, the next label would be ab.00.
Example: fields: aa-zz, 00-99;

separator  (read/write, single choice, null ok)
This attribute specifies the character to use to separate the label fields. It may be one of ‘.’, ‘_’, ‘:’, ‘-’ or NULL.
Example: separator: .;

next  (read/write, single string)
This attribute specifies the next volume name to use. After it is assigned to a volume, the next volume name will be generated and remembered here. The attribute consists of a component for each of the specified fields and the separator.
Example:
next: aa.00;

Using the separator and field attributes shown above, the next attribute would show: next: aa.01;
This would be followed by: next: aa.02;
EXAMPLES

A label resource named engineering is shown below. (Hidden options are not shown.) There are two range type fields defined, the first ranging from ‘aa’ to ‘zz’, the second from ‘00’ to ‘99’. The separator attribute has the value ‘.’ and it will be inserted in between the two fields. The next attribute holds the next name that will be generated by this template. After aa.00 is used, the 00 will be incremented. The new name will be aa.01. After 98 more names have been generated, the next attribute will hold the name aa.99. When this name is incremented, the next attribute will hold ab.00. After generating 67,500 more names, the next attribute will hold zz.99. This will be followed by aa.00.

    type: NSR label;
    name: engineering;
    fields: aa-zz, 00-99;
    separator: .;
    next: aa.00;

A label resource named accounting is shown below. The field attribute defines five component fields. The separator attribute has the value ‘.’. It will be inserted in between adjacent fields. The next attribute holds the next name that will be used with this template. After 0.23.aa.AA.first is used, the fifth field will be incremented. The new name will be 0.23.aa.AA.second. This will be followed by 0.23.aa.AB.first. After 1349 more volume names, the name will be 0.23.aa.ZZ.second. This will be followed by 0.23.ab.AA.first. After using 9.45.zz.ZZ.second, the name will wrap around to 0.23.aa.AA.first.

    type: NSR label;
    name: accounting;
    fields: 0-9, 23-45, aa-zz, AA-ZZ, first(second);
    separator: .;
    next: 0.23.aa.AA.first;

SEE ALSO

nwadmin(8), nsradmin(8), nsrjb(8), nsrmm(8), nsr(8), nsr_jukebox(5).