

# SAVELAB

## Save and Restore ISPF Edit Labels

### Overview

ISPF Edit has supported labels on lines for many years but with many limitations. The most significant limitation is that the labels disappear when the Edit session ends. This greatly limits the usefulness of labels.

With SAVELAB that limitation is eliminated, along with many others that we didn't know were limitations until it became possible to retain the labels across Edit sessions.

SAVELAB is an ISPF Edit Macro that makes it possible to save all of the labels defined during an Edit session and then restore them the next time the data is opened in Edit. This makes the use of ISPF Edit Labels worthwhile and will improve the productivity of ISPF Edit users who learn to take advantage of this capability.

SAVELAB uses an ISPF Table so that the information is available to ISPF Edit regardless of which ISPF application ID invokes Edit. For those who like to share, it is possible to Export, and then Import the saved label information.

### Key Features of SAVELAB

SAVELAB supports the following features:

#### Save All Active Edit Labels

Using the command SAVELAB SAVE will save all active ISPF Edit Labels to an ISPF Table that is located in the partitioned dataset referenced by the ISPPROF DD. The saved information is stored using the dataset name and member name to keep the labels unique.

#### Restore ISPF Edit Labels

Using the command SAVELAB will restore all saved ISPF Edit Labels to the active Edit session.

To automate this, it is possible to make SAVELAB the initial Edit Macro. This can be done in various ways, but the easy way is to enter the command IMACRO SAVELAB on the Edit command line. Then every time a member in that dataset is edited SAVELAB will be invoked to restore any saved labels for that member.

When SAVELAB executes it performs other actions, regardless of whether a restore occurs:

1. Establishes a hook in ISPF Edit End and Save processing. This enables the automatic saving of any labels established during the Edit session without requiring the user to explicitly invoke the SAVELAB SAVE command.
2. Defines an alias of SL for SAVELAB so that the user only needs to enter SL instead of the full SAVELAB name.

## Alternate Restore Options

Some users have a generic Initial Edit command and, in that case, SAVELAB can be added to it easily.

In this situation SAVELAB offers two additional options:

### QUIET

Adding the QUIET parameter to the SAVELAB command will prevent any messages if there are no labels to restore. The hooks for Edit End and Save will be created along with the alias of SL for SAVELAB.

### IMAC

Adding the IMAC parameter to the SAVELAB command is similar to using QUIET with the exception that the hooks for Edit End and Save will not be created, nor will the SL alias for SAVELAB will be created.

## Viewing Labels

The SAVELAB SHOW command will display a popup, scrollable, table of all active Edit Labels. From this list any of the labels can be selected (S or cursor and Enter) to jump directly to.

Here is an example of the display for the SAVELAB exec:

```
Active Edit Labels
Command ==> Row 1 to 12 of 17
Command: Sort (Label) SortR (Record) Select: S or Enter
S  Label Record Data
- .VER      2 ver = "2.00"
- .HIST     100 * History:
- .DEFS     196 parse value '' with null free_labset quiet
- .CREAT    216 Address ISPExec
- .DOIT     228 option = translate(option)
- .REST     268 Do_Restore:
- .CLOSE    313 Close_out:
- .DEFA     322 Define_Alias:
- .SAVE     338 Do_Save:
- .SHOW     380 Do_Show:
- .SHOWA    389 do_showall:
- .CONV     500 Convert_Table:
```

## Managing Labels

The number of datasets and members will grow and there needs to be a way to manage them. Using the SAVELAB SHOW ALL command will display a table of all the saved label datasets.

```
----- Saved Label Dataset(Members) ----- Row 1 to 6 of 6
Command ==> |                               Scroll: CSR

Line: S Select D Delete U Use

S  Dataset(member)                               Date
-  SLBD.JCL.CNTL(BLDAPART)                       1 Feb 2020
-  SLBD.LIONEL.EXEC(PDSEGV)                       1 Feb 2020
-  SLBD.LIONEL.EXEC(SAVELAB)                     1 Feb 2020
-  SLBD.LIONEL.EXEC(TLDSI)                       1 Feb 2020
-  SLBD.SAVELAB.EXEC(SAVELAB)                   1 Feb 2020
-  SLBD.TEST.LIST                               1 Feb 2020
***** Bottom of data *****
```

From this list an entry can be:

1. Selected which will display the saved labels
2. Deleted
3. Use the label to import into the active data.

The selection display is basic and view only:

```
----- Saved Labels -----
Command ==> |                               Row 1 to 11 of 17

SLBD.LIONEL.EXEC(SAVELAB)

Label  Record #
.VER   2
.HIST  100
.DEFS  196
.CREAT 218
.DOIT  230
.REST  270
.CLOSE 315
.DEFA  324
.SAVE  340
.SHOW  382
.SHOWA 391
```

## Using Labels from another Dataset

The U option is very handy if the data with the labels is renamed or copied/moved to another dataset. All of the labels in the selected save set will be restored in the active data.

## Export and Import of Labels

Using the command `SAVELAB EXPORT dataset-name` will export all the active labels to a member in the specified dataset-name. The member name will be the same as the active member.

The command `SAVELAB IMPORT dataset-name` will import the labels from the dataset-name where the member name is the same as the active member name.

Note: Import and Export of a sequential dataset's labels are not supported at this time.